





# Wave Spectrum of Flowing Drops

Maria Guskova<sup>1,2</sup>  and Lev Shchur<sup>1,2</sup> 

<sup>1</sup> Landau Institute for Theoretical Physics, Chernogolovka, Russia  
{mguskova, lshchur}@hse.ru, lev@landau.ac.ru

<sup>2</sup> HSE University, Moscow, Russia

**Abstract.** We simulate the drop-chain movement in the two-dimensional channel using the lattice Boltzmann method coupled with the Immersed Boundary approach. We choose the asymmetric initial state of the drop chain to generate drop oscillations and calculate the wave spectrum. The numerical results coincide qualitatively with the experimental results obtained in the quasi-two-dimensional microfluidic device.

**Keywords:** Lattice Boltzmann method · Immersed boundary method · Supercomputing · Computer simulations · Drop oscillations

## 1 Introduction

Microfluidic devices are subject to interdisciplinary studies [5], and investigation of the fluid dynamics under microfluidic conditions is a challenging problem. The restricted geometry and inhomogeneity of the complex flow compounds need careful analysis of the parameter influence on the flow details.

In the paper, we investigate the movement of the chain drop inside the channel of finite width filled with the liquid. We consider a Poiseuille flow of liquid in the channel. We use lattice Boltzmann method (LBM) coupled with immersed boundary (IB) method in simulations. We compare the results of our simulations with the results of the physics experiment in which the behavior of water droplets in oil in a quasi-two-dimensional flow was studied [2]. In the experiment, the speed of water droplets due to friction against the upper and lower walls of the channel is approximately five times less than the average flow speed. In the experiment with a chain of drops, drops oscillate, and longitudinal and transverse waves appeared. The emergence of waves was explained by the long-range hydrodynamic interaction of drops with each other. We have to note that the Reynolds number is very small and equal approximately  $10^{-4}$ .

In some works [8, 14, 16, 21], computer simulation of the one-dimensional chain of particles or drops motion was carried out.

The work [14] presents the results of numerical integration of the equations of droplet motion obtained in [2]. They found some difference with the experimental results. Unlike the oscillation spectra of a physical experiment, there is no straight line in the spectrum since it arises from defects in the channel.

In the work [8], a computer simulation of the motion of a chain of three-dimensional drops between two parallel planes is performed. The authors used the HYDROMULTIPOLE algorithm to expand the flow field into the lateral Fourier modes in the planes parallel to the walls [1]. They investigated the influence of deformability of bodies. In contrast with the physical experiment, drops are not confined but sphere-shaped.

The authors of [21] have found new patterns of collective behavior of particles in a quasi-two-dimensional problem. The work did not consider the motion of a one-dimensional chain but used the method of the lattice Boltzmann equation. The D2Q9 model was used, moving hard disks were simulated via transfer of momentum in the “bounce-back” boundary condition, using a first-order boundary interpolation method.

In [16] a computer simulation of a chain of drops in a two-dimensional narrow channel was performed. The droplets are represented as hard disks, and the liquid is represented by idealized point particles with masses of  $m$ . At short distances, the multi-particle collision dynamics method does not resolve hydrodynamics, and the WCA potential was used to simulate the interaction of particles with each other and with the walls. A friction force was applied to the disks in order to change the two-dimensional formulation to a quasi-two-dimensional one. The initial coordinates in the chain were set as follows: alternating the  $y$ -coordinate (above the axis/below the axis) and alternating the distances between the centers of the particles along the  $X$  axis. The right particle in a pair moves faster than the left one; it catches up with the next pair and becomes left. The different distances along the  $X$  axis cause the pairs to re-form, which leads to longitudinal oscillations. The spectra of the obtained oscillations along the  $X$  and  $Y$  axes are similar to the results of [2].

This paper investigates the applicability of the immersed boundary method combined with the lattice Boltzmann method [6] for simulation of the longitudinal and transverse oscillations arising in a one-dimensional chain of drops. Unlike other discussed above works [8, 14, 16], we avoid the introduction of the effective interaction potentials between particles. The reason is that we would like to check the natural appearance of the wave spectrum reported in [2] without any assumptions. The second difference from the above-mentioned works is in the more realistic boundary conditions along the channel, and the Poiseuille profile is specified along the channel width. We model drops as solid disks, and the interaction between different drops and between drops and walls is wholly resolved through the liquid, which is simulated by the lattice Boltzmann method. Another significant difference is a purely two-dimensional case and the absence of a friction force acting on the discs from the side of the upper and lower walls. Because of this, the particles have a speed close to the flow. The spectrum of the simulated oscillations is compared with the experimental [2] spectrum.

## 2 Model and Methods

In this section, we give essentials of the lattice Boltzmann method and immersed boundary method and the derivation of the forces acting in two-dimension on the immersed boundary.

### 2.1 Lattice Boltzmann Method

Discretizing the Boltzmann equation in the space of velocities, coordinates and time, we obtain the lattice Boltzmann equation [11]:

$$f_i(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) = f_i(\vec{x}, t) + \Omega_i(\vec{x}, t).$$

According to this expression, (pseudo-) particles  $f_i(\vec{x}, t)$  move with the speed  $\vec{c}_i$  to the neighboring point  $\vec{x} + \vec{c}_i \Delta t$  in time  $\Delta t$ . At the same time, the particles are affected by the collision operator  $\Omega_i$ . In this work the collision operator BGK (Bhatnagar-Gross-Krook) is used:

$$\Omega_i(f) = -\frac{f_i - f_i^{eq}}{\tau} \Delta t.$$

The set of velocities and the dimension of the problem determine the model [11], for example, D2Q9 (Fig. 1a) corresponds to a two-dimensional problem, where the velocity can be directed from a node to 4 corners of a square, to 4 midpoints of the sides square, and one does not go out of the node. Model D3Q19 (Fig. 1b) corresponds to a three-dimensional problem, and the velocities are set in 19 directions: to 12 midpoints of the edges of the cube, to 6 midpoints of faces, and one does not leave the node.

The density functions relax to the equilibrium state  $f_i^{eq}$  at a rate determined by the relaxation time  $\tau$ . The equilibrium state is defined as:

$$f_i^{eq}(\vec{x}, t) = w_i \rho \left( 1 + \frac{\vec{u} \cdot \vec{c}_i}{c_s^2} + \frac{(\vec{u} \cdot \vec{c}_i)^2}{c_s^2} - \frac{\vec{u} \cdot \vec{u}}{2c_s^2} \right), \quad (1)$$

where the weights  $w_i$  correspond to a set of velocities. The equilibrium state is such that its moments are equal to the moments  $f_i$ , i.e.  $\sum_i f_i^{eq} = \sum_i f_i = \rho$  and  $\sum_i \vec{c}_i f_i^{eq} = \rho \vec{u}$ . This state  $f_i^{eq}$  depends only on the local density  $\rho$  and the flow rate  $\vec{u}$ . Calculation formulas:  $\rho(\vec{x}, t) = \sum_i f_i(\vec{x}, t)$ ,  $\rho \vec{u}(\vec{x}, t) = \sum_i \vec{c}_i f_i(\vec{x}, t)$ .

The BGK lattice equation (fully discretized Boltzmann equation with BGK collision operator) can be written as

$$f_i(x + c_i \Delta t, t + \Delta t) = f_i(x, t) - \frac{\Delta t}{\tau} (f_i(x, t) - f_i^{eq}(x, t)).$$

This equation consists of two separate steps:

#### 1. Collision step

$$f_i^*(x, t) = f_i(x, t) - \frac{\Delta t}{\tau} (f_i(x, t) - f_i^{eq}(x, t)),$$

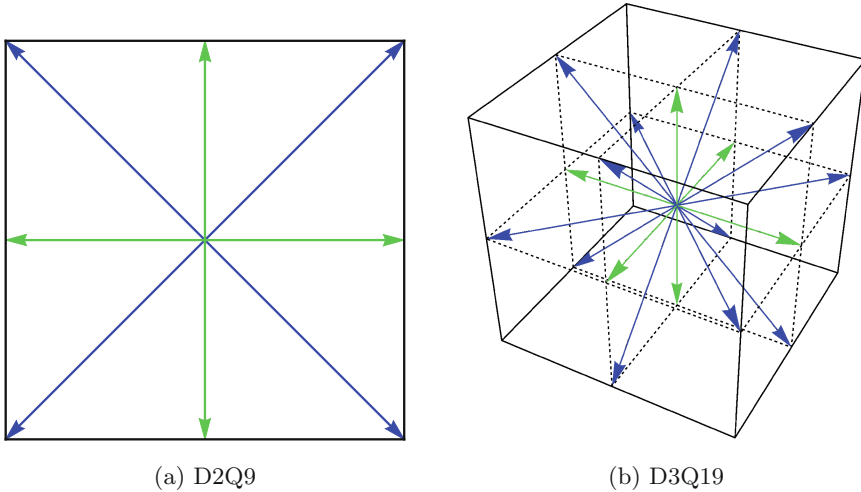


Fig. 1. Models

Where  $f_i^*$  represents the density function after collision and  $f_i^{eq}$  can be found from  $f_i$  by the formula (1). A collision can be effectively implemented in the form:

$$f_i^*(x, t) = f_i(x, t) \left(1 - \frac{\Delta t}{\tau}\right) + f_i^{eq}(x, t) \frac{\Delta t}{\tau}.$$

2. Streaming step

$$f_i(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) = f_i^*(\vec{x}, t).$$

Collision is just local algebraic operations. First, the density  $\rho$  and the macroscopic velocity  $\vec{u}$  are calculated in order to find the equilibrium density functions  $f_i^{eq}$  by the formula (1) and the post-collision density function  $f_i^*$ . After the collision, the resulting density function  $f_i^{eq}$  propagates to neighboring nodes. After both stages are done, the one-time step has passed, the operations are repeated.

2.2 Immersed Boundary Method

The immersed boundary method is used to simulate moving boundaries, such as blood cells, bubble dynamics, and many others. Peskin first proposed the method to simulate blood flow in 1972. Since that time, the method has been utilized many times to simulate suspensions: in conjunction with differential schemes of the Navier-Stokes equation [12, 17, 20] and with the lattice Boltzmann method [3, 7, 10, 15, 19, 22]. We use IB-LBM as it allows to simulate moving solid bodies and IB-LBM for 3D case is already implemented in Palabos library, in this work 3D case was adapted to 2D problem.

The main idea of the immersed boundary method is to represent the boundary of a body as a set of Lagrangian points, which in the general case do not

coincide with the Euler lattice sites. The combined IB-LBM approach is an iterative method. At each iteration, the body force value for the Lagrangian points is calculated – the velocities from the nodes of the Euler grid are interpolated to points on the boundary. Then the body force acting from the drop on the liquid is calculated at the lattice nodes. The calculated force is used to compute the velocity at the points of the lattice. Finally, the velocities are interpolated to the droplet boundary point, and the force acting on the droplet is corrected.

If there is an external body force  $\vec{g}(\vec{x}, t)$ , then the lattice equation for the particle distribution functions  $f_i(\vec{x}, t)$  can be solved in two steps [7]

1.  $f_i(\vec{x}, t)$  update, without taking into account the body force

$$f_i^*(\vec{x} + \vec{c}_i \Delta x, t + \Delta t) = f_i(\vec{x}, t) - \frac{1}{\tau} [f_i(\vec{x}, t) - f_i^{eq}(\vec{x}, t)]. \quad (2)$$

2.  $f_i^*$  corrected by body force

$$f_i(\vec{x}, t) = f_i^*(\vec{x}, t + \Delta t) + 3\Delta x E_i \vec{c}_i \cdot \vec{g}(\vec{x}, t + \Delta t). \quad (3)$$

The IB method assumes that an incompressible viscous fluid is inside and outside the boundary. Then the body force is applied to the lattice nodes near the boundary to satisfy the non-slip condition. The methods for determining the body force differ between different IBM versions. In paper [7] authors use the multi-direct forcing method proposed in [22]. The advantage of this method over the others is that the non-slip condition can be satisfied accurately [18].

Assuming  $f_i(\vec{x}, t)$ ,  $\vec{u}(\vec{x}, t)$  and  $p(\vec{x}, t)$  are known, intermediate values of  $f_i^*(\vec{x}, t + \Delta t)$  and  $\vec{u}_i^*(\vec{x}, t)$  can be computed from

$$\vec{u} = \sum_{i=1}^b f_i \vec{c}_i, \text{ where } b = 9 \text{ for D2Q9 and } b = 19 \text{ for D3Q19} \quad (4)$$

and

$$p = \frac{1}{3} \sum_{i=1}^b f_i, \text{ where } b = 9 \text{ for D2Q9 and } b = 19 \text{ for D3Q19}. \quad (5)$$

Let  $\vec{X}_k(t + \Delta t)$  are Lagrangian points on the moving boundary and  $\vec{U}_k(t + \Delta t)$ , ( $k = 1, \dots, N$ ) are velocities at those points. Then the temporal velocities  $\vec{u}^*(\vec{X}_k, t + \Delta t)$  at the Lagrangian boundary points  $\vec{X}_k$  can be interpolate

$$\vec{u}^*(\vec{X}_k, t + \Delta t) = \sum_x \vec{u}^*(\vec{x}, t + \Delta t) W(\vec{x} - \vec{X}_k) (\Delta x)^d, \quad (6)$$

where  $\sum_x$  is the sum over all lattice nodes  $\vec{x}$ ,  $W$  is the weight function proposed by Peskin [12] and  $d$  is the dimension. The weighting function  $W$  is given by

$$W(x, y, z) = \frac{1}{\Delta x} w\left(\frac{x}{\Delta x}\right) \cdot \left(\frac{1}{\Delta x}\right) w\left(\frac{y}{\Delta x}\right) \cdot \left(\frac{1}{\Delta x}\right) w\left(\frac{z}{\Delta x}\right), \quad (7)$$

$$w(r) = \begin{cases} 1/8 \left( 3 - 2|r| + \sqrt{1 + 4|r| - 4r^2} \right), & |r| \leq 1, \\ 1/8 \left( 5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2} \right), & 1 \leq |r| \leq 2, \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

In the three-dimensional case, the weight function is the product of three one-dimensional weight functions; in the two-dimensional case, of two.

The body force  $\vec{g}(\vec{x}, t + \Delta t)$  is determined by the following iterative procedure:

0. Calculate the initial value of the body force at the Lagrangian boundary points

$$\vec{g}_0(\vec{X}_k, t + \Delta t) = Sh \frac{\vec{U}_k - \vec{u}^*(\vec{X}_k, t + \Delta t)}{\Delta t}, \quad (9)$$

where  $Sh/\Delta t = 1/\Delta x$ .

1. Calculate the body force at the nodes of the Euler mesh at the  $l$ -th iteration

$$\vec{g}_l(\vec{x}, t + \Delta t) = \sum_{k=1}^N \vec{g}_l(\vec{X}_k, t + \Delta t) W(\vec{x} - \vec{X}_k) \Delta V, \quad (10)$$

where the body force is applied not to one Lagrangian boundary point, but to a small volume element  $\Delta V$ . In this method,  $\Delta V$  is selected as  $S/N \times \Delta x$ , where  $S$  is the surface area of the body and  $S/N$  must be of the order  $(\Delta x)^{d-1}$ .

2. Adjust the velocities at the nodes of the Euler grid

$$\vec{u}_l(\vec{x}, t + \Delta t) = \vec{u}^*(\vec{x}, t + \Delta t) + \frac{\Delta t}{Sh} \vec{g}_l(\vec{x}, t + \Delta t). \quad (11)$$

3. Interpolate velocity at Lagrangian boundary points

$$\vec{u}_l(\vec{X}_k, t + \Delta t) = \sum_x \vec{u}_l(\vec{x}, t + \Delta t) W(\vec{x} - \vec{X}_k) (\Delta x)^d. \quad (12)$$

4. Update the body force

$$\vec{g}_{l+1}(\vec{X}_k, t + \Delta t) = \vec{g}_l(\vec{X}_k, t + \Delta t) + Sh \frac{\vec{U}_k - \vec{u}_l(\vec{X}_k, t + \Delta t)}{\Delta t}. \quad (13)$$

It is known that the choice  $l = 5$ ,  $\vec{g}_{l=5}(\vec{x}, t + \Delta t)$  is sufficient to satisfy the non-slip condition [18].

### 2.3 Combined IB-LBM Algorithm

Therefore, the short protocol of the combined IB-LBM algorithm consists of the following significant steps

0. Let initial values are  $f_i(\vec{x}, 0)$ , calculate  $\vec{u}(\vec{x}, 0)$  and  $p(\vec{x}, 0)$  from Eqs. (4) and (5),

1. Calculate  $\vec{X}_k(t + \Delta t)$  and  $\vec{U}_k(t + \Delta t)$  from equations of body motion,
2. Compute  $f_i^*(\vec{x}, t + \Delta t)$  from Eq. (2) and  $u^*(\vec{x}, t + \Delta t)$  from Eq. (4), then evaluate  $u^*(\vec{X}_k, t + \Delta t)$  from Eq. (6),
3. Compute  $\vec{g}(\vec{x}, t + \Delta t)$  Eqs. (9–13),
4. Calculate  $f_i(\vec{x}, t + \Delta t)$ ,  $\vec{u}(\vec{x}, t + \Delta t)$  and  $p(\vec{x}, t + \Delta t)$  by Eq. (5), Eq. (4),
5. Go to the next time step and return to 1.

## 2.4 Drop Motion

The drop is represented by a set of evenly distributed points on the boundary of a rigid disk. The drop motion is defined by classical mechanics laws

$$M \frac{d\vec{U}(t)}{dt} = \vec{F}(t), \quad (14)$$

$$\vec{I} \frac{d\vec{\Omega}(t)}{dt} + \vec{\Omega}(t) \times [\vec{I} \vec{\Omega}(t)] = \vec{T}(t), \quad (15)$$

where  $M = \pi R^2 \rho$  is the droplet mass,  $\vec{U}(t)$  is the linear velocity,  $\vec{F}(t) = -\sum \vec{g}(t)$  the net force acting on the drop, calculated by the immersed boundary method,  $\vec{I}$  is the tensor of inertia, and for a hard disk it is described by the formulas:  $I_z = \frac{1}{2}MR^2$ ,  $I_x = I_y = \frac{1}{4}MR^2$ ,  $\vec{\Omega}$  angular velocity,  $\vec{T} = -\sum \vec{r} \times \vec{g}$  moment of forces, acting on a drop.

The velocity at the boundary points  $\vec{u}_b$  can be calculated as

$$\vec{u}_b = \vec{U} + \vec{\Omega} \times (\vec{r}_b - \vec{X}), \quad (16)$$

Where  $\vec{r}_b$  are coordinates of the boundary points and  $\vec{X}$  is the center of mass.

In the two-dimensional case, the moment of forces has one nonzero component and  $\vec{\Omega} = \{0, 0, \omega\}^T$ , thus the Eq. (15) is simplified:

$$\frac{1}{2}MR^2 \frac{d\omega}{dt} = T_z. \quad (17)$$

We integrate Eqs. (14–17) using the explicit Euler method.

## 3 Simulation

### 3.1 Simulation Setup

Boundary conditions above and below are walls (bounce-back), on the left and on the right, the Poiseuille profile is set with an average velocity  $u_{av} = 0.015$ . The channel width is  $80\Delta x$ , the channel length is 250 times longer. Particle radius  $R = 5\Delta x$ . A particle in the IB-LBM method is represented by 100 points uniformly distributed around a circle. The number of IBM iterations is 4. Reynolds number  $Re = 0.25$ , which is three orders of magnitude higher than the Reynolds number in the [2] experiment, but it allows to run computations in a reasonable

time. After the flow is fully developed, the particles are inserted on the central axis of the channel with the distance between the centers of the particles  $a = 4R$ . The middle drop is shifted  $R/2$  above the axis in order to excite oscillations. The scheme for 12 particles is shown in Fig. 2. The density of the drop is greater than the density of the surrounding liquid  $\rho_{drop} = 1000/840 \approx 1.19$ , which corresponds to the ratio of the density of water to the density of oil. The density of the fluid is  $\rho_{fluid} = 1$ . The model is a two-dimensional D2Q9 [11].

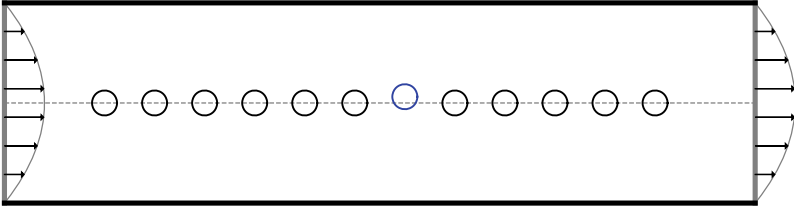


Fig. 2. Computational setup.

### 3.2 Details of Simulation

For the IB-LBM simulation, the open-source software Palabos [13] was used, and some modifications described in the preceding section incorporated in the code. Palabos is a library written in C++ that uses MPI for parallelization. Palabos partitions lattice into smaller regions that are distributed over the nodes or cores of a parallel machine. Other data types which require a small amount of storage space are duplicated on every node. Chain of particles processing is done as a data processor (already implemented in Palabos), which is parallelized. The current problem has been computed on one node, but it will be transformed into 3d problem, and more threads and more nodes are proposed to use in the future.

The experiments were carried out on a high-performance computing cluster “cHARISMa”. The parallel implementation uses MPI with the library OpenMPI version 3.1.4. The simulation was run with 22 threads on a node with an Intel Xeon Gold 6152 2.1 GHz processor with DDR4 2.666 GHz 768 GB memory.

Computational domain is  $80\Delta x \times 20000\Delta x$ ,  $\Delta t = 0.0015$ ,  $1.5 \cdot 10^6$  iterations were done until the flow is fully developed. 50,000 LBM iterations were performed on average in 4 min in 22 threads. 50,000 IB-LBM iterations for a chain of length 12 were performed on average in 16 min in 22 threads (with drop coordinates savings in every iteration).

### 3.3 Spectrum Calculation

The drop chain oscillations’ spectrum was calculated similarly to Ref. [2]. The crystal oscillation spectrum was calculated from the coordinates  $[x(n, t), y(n, t)]$  of all particles  $n = 1 \dots N_d$  at the time moments  $t = 1 \dots N_f$ . To apply the discrete Fourier transform, it is necessary to calculate the deviations of the particles



from their positions in the crystal. In the direction of the  $Y$  axis, the deviation will be the difference between the  $y$ - coordinates of the center of the particle and the center of the channel. To calculate longitudinal oscillations, a new coordinate  $\xi$  is introduced, which is the difference between neighboring particles along the  $X$  axis:

$$\xi(n, t) = \begin{cases} 0, & n = 1 \\ x(n, t) - x(n - 1, t), & n > 1. \end{cases} \quad (18)$$

Next, the discrete Fourier transform in space and time was calculated using the NumPy library (Python [4]):

$$X(k, \omega) = \sum_{n=1}^{N_d} \sum_{t=1}^{N_f} \xi(n, t) e^{-(2\pi i/N_d)(k-1)(n-1)} e^{-(2\pi i/N_f)(\omega-1)(t-1)} \quad (19)$$

and

$$Y(k, \omega) = \sum_{n=1}^{N_d} \sum_{t=1}^{N_f} y(n, t) e^{-(2\pi i/N_d)(k-1)(n-1)} e^{-(2\pi i/N_f)(\omega-1)(t-1)} \quad (20)$$

where  $k, \omega, n$  and  $t$  are indices. The wave numbers and frequencies can be calculated as  $2\pi k/L$  and  $2\pi\omega/T$ , respectively, where  $L$  is the length of the particle chain and  $T$  is the simulation period. Further, for each  $k$ , the maximum of the spectrum was determined modulo for  $|X(k, \omega)|^2$  and  $|Y(k, \omega)|^2$ , disregarding the values on the line  $\omega = 0$ . These maxima are marked in the figures with blue dots. For symmetry, values for negative  $k$  indices have been shown, although they do not provide new information.

## 4 Results

In order to determine the required number of particles in a chain to obtain oscillations similar to a physical experiment, chains of different lengths were simulated using the IB-LBM. For chains with 3, 4, 5, and 6 drops, the spectrum of longitudinal and transverse oscillations is not similar to the results of the experiment shown in the Figs. 1e,f of Ref. [2]. Beginning with a chain length of 10 drops, the spectrum of transverse oscillations approaches the experimental one. We do not obtain similarity of the longitudinal spectrum with one reported in Ref. [2].

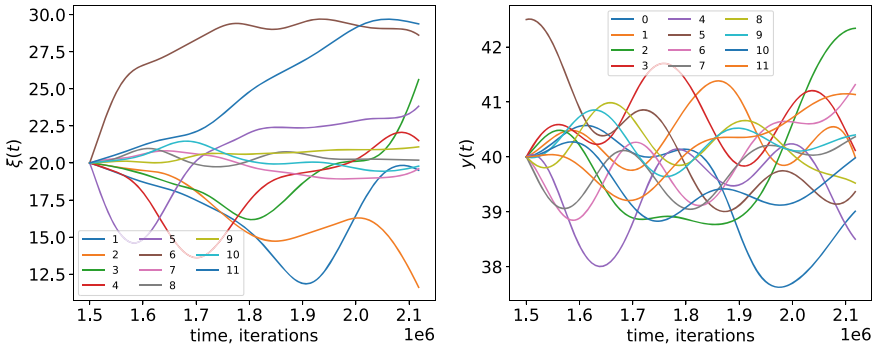
Figure 3 shows the coordinates  $\xi(t)$  and  $y(t)$  for a chain of 12 particles - we will use this coordinates for the calculation of the spectrum. One can see from the figure that there are no longitudinal oscillations for the chain of 12 particles. The transverse spectrum coincide well with those reported in [2]. Figures (4a, 5b) show colormap for the logarithms of  $|X(k, \omega)|^2$  and  $|Y(k, \omega)|^2$ .

After the initial Poiseuille flow in the channel is fully developed, the drops were placed on the channel axis, while one drop in the middle of the chain shifted from the axis for the quarter drops radius  $R$ . Because of the Poiseuille profile,

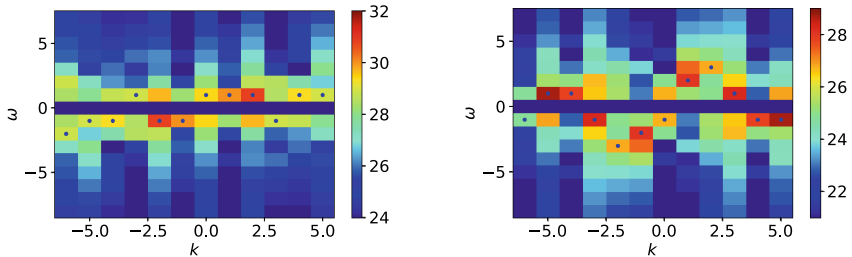
the velocity vanishes on the walls, and it takes maximum value at the channel center. This imposes the force acting on the shifted drop towards the channel axis. The transverse drop movement induces waves in the liquids, which act on the other drops directly as well as indirectly through the reflection from the channel walls. This is our explanation of the drop chain collective oscillations leading to the spectrum shown in the Fig. 3.

We found that the narrower the channel, the more pronounced the transverse oscillations are.

The peculiarity of the IB-LBM can explain the absence of longitudinal oscillations – the drops in this work have a velocity very close to the flow velocity.

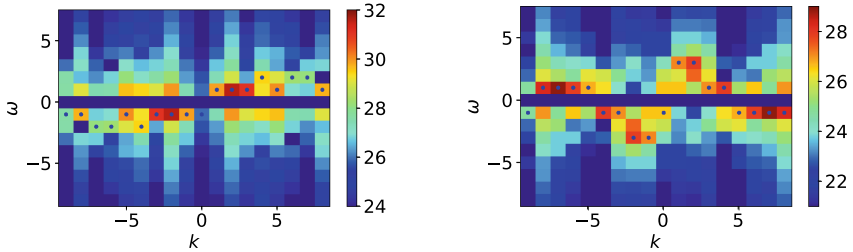


**Fig. 3.** Coordinates  $\xi(t)$  and  $y(t)$  for the chain of 12 drops.



(a) Colormap for logarithm of  $|X(k, \omega)|^2$  (b) Colormap for logarithm of  $|Y(k, \omega)|^2$

**Fig. 4.** Spectrum for the chain of 12 drops



(a) Colormap for logarithm of  $|X(k, \omega)|^2$  (b) Colormap for logarithm of  $|Y(k, \omega)|^2$

**Fig. 5.** Spectrum for the chain of 18 drops

## 5 Conclusion

The paper presents some results of studying the possibility of the drops' long-range hydrodynamic interaction simulation in a flow by the immersed boundary method coupled with the lattice Boltzmann method. A series of computational experiments were carried out for chains of particles of different lengths. The spectra of longitudinal and transverse oscillations in the crystal were constructed. We found excellent agreement of the transverse drop oscillation spectrum with the experimental one for the drop chain length starting from 10. We propose the main effect of the particular spectrum is due to the cross-influence of the drop oscillations via the generated waves in liquid, which acts on other drops directly and after reflection from the channel walls.

**Acknowledgment.** This research was supported in part through computational resources of HPC facilities at HSE University [9].

Research supported by Russian Science Foundation grant 19-11-00286. MG is partially supported by the Russian Foundation for Basic Research grant 20-37-90086.

## References

1. Baron, M., Bławdziewicz, J., Wajnryb, E.: Hydrodynamic crystals: collective dynamics of regular arrays of spherical particles in a parallel-wall channel. *Phys. Rev. Lett.* **100**(17), 174502 (2008)
2. Beatus, T., Tlusty, T., Bar-Ziv, R.: Phonons in a one-dimensional microfluidic crystal. *Nat. Phys.* **2**(11), 743–748 (2006)
3. Feng, Z.G., Michaelides, E.E.: The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems. *J. Comput. Phys.* **195**(2), 602–628 (2004)
4. Harris, C.R., et al.: Array programming with NumPy. *Nature* **585**(7825), 357–362 (2020). <https://doi.org/10.1038/s41586-020-2649-2>
5. Ibrahim, M., Chakrabarty, K., Zeng, J.: Biocybig: a cyberphysical system for integrative microfluidics-driven analysis of genomic association studies. *IEEE Trans. Big Data* **6**(4), 609–623 (2020). <https://doi.org/10.1109/TBDATA.2016.2643683>

6. Inamuro, T.: Lattice Boltzmann methods for moving boundary flows. *Fluid Dyn. Res.* **44**(2) (2012). <https://doi.org/10.1088/0169-5983/44/2/024001>
7. Inamuro, T.: Lattice Boltzmann methods for moving boundary flows. *Fluid Dyn. Res.* **44**(2), 024001 (2012)
8. Janssen, P.J., Baron, M.D., Anderson, P.D., Blawdziewicz, J., Loewenberg, M., Wajnryb, E.: Collective dynamics of confined rigid spheres and deformable drops. *Soft Matter* **8**(28), 7495–7506 (2012). <https://doi.org/10.1039/c2sm25812a>
9. Kostenetskiy, P., Chulkevich, R., Kozyrev, V.: HPC resources of the higher school of economics. *J. Phys. Conf. Ser.* **1740**(1), 012050 (2021)
10. Krüger, T., Gross, M., Raabe, D., Varnik, F.: Crossover from tumbling to tank-treading-like motion in dense simulated suspensions of red blood cells. *Soft Matter* **9**(37), 9008–9015 (2013)
11. Krüger, T., Kusumaatmaja, H., Kuzmin, A., Shardt, O., Silva, G., Viggen, E.M.: *The Lattice Boltzmann Method*, vol. 10, pp. 978–973, 4–15. Springer, Cham (2017). <https://doi.org/10.1007/978-3-319-44649-3>
12. Lai, M.C., Peskin, C.S.: An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *J. Comput. Phys.* **160**(2), 705–719 (2000)
13. Latt, J., et al.: Palabos: parallel lattice Boltzmann solver. *Comput. Math. Appl.* **81**, 334–350 (2020)
14. Liu, B., Goree, J., Feng, Y.: Waves and instability in a one-dimensional microfluidic array. *Phys. Rev. E* **86**(4), 046309 (2012)
15. Mountrakis, L., Lorenz, E., Malaspinas, O., Alowayyed, S., Chopard, B., Hoekstra, A.G.: Parallel performance of an IB-LBM suspension simulation framework. *J. Comput. Sci.* **9**, 45–50 (2015)
16. Schiller, U.D., Fleury, J.B., Seemann, R., Gompper, G.: Collective waves in dense and confined microfluidic droplet arrays. *Soft Matter* **11**(29), 5850–5861 (2015)
17. Silva, A.L.E., Silveira-Neto, A., Damasceno, J.: Numerical simulation of two-dimensional flows over a circular cylinder using the immersed boundary method. *J. Comput. Phys.* **189**(2), 351–370 (2003)
18. Suzuki, K., Inamuro, T.: Effect of internal mass in the simulation of a moving body by the immersed boundary method. *Comput. Fluids* **49**(1), 173–187 (2011)
19. Thorimbert, Y., Marson, F., Parmigiani, A., Chopard, B., Lätt, J.: Lattice Boltzmann simulation of dense rigid spherical particle suspensions using immersed boundary method. *Comput. Fluids* **166**, 286–294 (2018)
20. Uhlmann, M.: An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.* **209**(2), 448–476 (2005)
21. Uspal, W.E., Doyle, P.S.: Collective dynamics of small clusters of particles flowing in a quasi-two-dimensional microchannel. *Soft Matter* **8**(41), 10676–10686 (2012)
22. Wang, Z., Fan, J., Luo, K.: Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles. *Int. J. Multiph. Flow* **34**(3), 283–302 (2008)