Article

Accelerated Computational Fluid Dynamics Simulations of Microfluidic Devices by Exploiting Higher Levels of Abstraction

Michel Takken [1,*] and Robert Wille [1,2,*]

1   School of Computation, Information and Technology, Technical University of Munich, Arcisstraße 21, 80333 München, Germany
2   Software Competence Center Hagenberg GmbH (SCCH), Softwarepark 32a, 4232 Hagenberg, Austria
*   Correspondence: michel.takken@tum.de (M.T.); robert.wille@tum.de (R.W.)

Abstract: The design of microfluidic devices is a cumbersome and tedious process that can be significantly improved by simulation. Methods based on *Computational Fluid Dynamics* (CFD) are considered state-of-the-art, but require extensive compute time—oftentimes limiting the size of microfluidic devices that can be simulated. Simulation methods that abstract the underlying physics on a higher level generally provide results instantly, but the fidelity of these methods is usually worse. In this work, a simulation method that accelerates CFD simulations by exploiting simulation methods on higher levels of abstraction is proposed. Case studies confirm that the proposed method accelerates CFD simulations by multiple factors (often several orders of magnitude) while maintaining the fidelity of CFD simulations.

Keywords: lab-on-a-chip; microfluidics; simulation; CFD

1. Introduction

A microfluidic device, or a *Lab-on-a-Chip* (LoC), is a device that performs lab operations on the microscale through a set of fluid manipulations [1]. Such devices are commonly used for, e.g., personalized medical care [2], point-of-care diagnostics [3] (well-known examples of such devices are pregnancy tests [4] or the SARS-CoV-2 tests [5]) and the food industry [6]. They have recently been proven to be more widely applicable in, e.g., geosciences [7] or fuel cell technology [8]. In that regard, much potential lies in microfluidic devices.

The development of those microfluidic devices is a cumbersome and tedious process that often requires multiple expensive and time-consuming design cycles [9]. To advance the design of microfluidic devices, reliable and quick simulation methods are necessary to predict whether a design works as intended. The simulation of Newtonian flow through a single microchannel can be performed using the Hagen–Poiseuille Law [10] and the flow profile across the channel can be assumed to be parabolic [11]. However, channel-based microfluidic devices consist of multiple interconnected microchannels, which require more extensive simulation methods.

To this end, most designers utilize methods from *Computational Fluid Dynamics* (CFD), such as the *Finite Volume Method* (FVM, [12]), *Finite Element Method* (FEM, [13]), or the *Lattice Boltzmann Method* (LBM, [14]), to obtain results of good fidelity. In recent overviews for microfluidics modeling, the FVM [15], FEM [15], and LBM [16] are listed as numerical approaches to solving the *Navier–Stokes Equations* (NSE) for microfluidics. Another recent work describes the workflow of setting up simulations for microfluidic devices [17] with OpenFOAM v9.0 [18], which uses the FVM. Hence, these simulation methods can be considered state-of-the-art for modeling microfluidic devices.

However, CFD simulations can take up to days or even weeks, even on dedicated workstations [19]. In practice, this obviously limits the use of CFD simulations for microfluidic devices to single components of the device [20]. To fully understand and design

a harmonious device, it is critical to simulate the behavior of *all* components *and* their interaction with each other.

Alternatively, high abstraction simulation methods, i.e., simulation methods that abstract the underlying physics on a higher level (also known as reduced-order modeling), generally simulate microfluidic devices almost instantly (i.e., in less than a second) [20]. An example of such a method is to draw an analogy between the Hagen–Poiseuille law and Ohm's law and apply analogous methods from electrical circuit engineering to channel-based microfluidic devices [21]. This approach is not limited to basic fluid flow, but can also be applied to problems with, e.g., droplets [22], or capillary-driven flows for paper-based microfluidics [23]. Such methods may not provide results of comparable fidelity, but can still simulate some parts of microfluidic devices, e.g., channels, with relatively good accuracy [20].

In this work, the nature of these high abstraction simulation methods is exploited to substantially accelerate CFD simulations of microfluidic devices with hardly any loss of fidelity. To this end, a two-stage approach is proposed: First, regions of the microfluidic device are identified that can sufficiently be simulated at a high level of abstraction. Afterward, the corresponding simulations are conducted, and the respectively obtained results are communicated between the simulation engines. Case studies (using continuous channel-based microfluidic devices as a representative) confirm the potential of this approach: The proposed approach does not only generate simulation results faster than the original CFD method, but constantly does so by several factors or even several orders of magnitude—while, at the same time, maintaining the fidelity of the results.

The remainder of this work is structured as follows: First, we review the simulation methods for microfluidic devices in more detail, focusing on methods on low and high abstraction levels. Afterward, the accelerated CFD simulation method is described in a general fashion in Section 3. Implementation details of this method are then provided in Section 4. Finally, we demonstrate the resulting solution and compare it against solutions of CFD simulations for a set of test cases in Section 5—confirming that the accelerated method is faster than the CFD method for all test cases while maintaining the fidelity. Finally, the paper is concluded in Section 6.

2. Simulation Methods for Microfluidic Devices

Simulation methods for microfluidic devices can be categorized into methods based on *Computational Fluid Dynamics* (CFD, [12–14]) on the low abstraction level, and methods on high abstraction levels [21–23], sometimes referred to as 1D methods. CFD methods can be considered state-of-the-art in the design of microfluidic devices [15–17], whereas the high abstraction level methods are often used to derive initial estimates during the design process [9]. In this section, we review the CFD and high abstraction level simulation methods.

2.1. Review of CFD Methods

A low abstraction simulation of microfluidic devices can be obtained through methods from *Computational Fluid Dynamics* (CFD, [12–14]). In CFD, the fluid behavior is modeled by the *Navier–Stokes Equations* (NSE), which are the fundamental governing equations for fluid dynamics. We restrict ourselves to the incompressible NSE [12–14], which are given by the mass equation

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1}$$

where $\boldsymbol{u}$ is the flow velocity vector, and the momentum equation

$$\frac{\partial}{\partial t}(\boldsymbol{u}) + \nabla \cdot \left(\boldsymbol{u}\boldsymbol{u}^T\right) = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u}, \tag{2}$$

where ρ is the fluid density, p is the pressure, and ν is the kinematic viscosity.

The analytical solution of the incompressible NSE exists only for a few simple fluid dynamics problems. For microfluidic devices, the equations generally have to be solved numerically. To this end, a wide variety of numerical methods has been developed. For example:

- The *Finite Volume Method* (FVM, [12]). The FVM splits the computational domain into grid cells, and the NSE are solved numerically on each grid cell. By this, averaged values for the flow velocity and pressure are obtained in each cell.
- The *Finite Element Method* (FEM, [13]). Similar to the FVM, the domain is split into cells. However, the solution is represented by a set of elements, e.g., polynomials.
- The *Lattice Boltzmann Method* (LBM, [14]). Rather than solving the NSE, this method solves the Boltzmann equation—which can be proven, through the Chapman–Enskog theory, to macroscopically solve the NSE [14].

CFD methods have in common that they, in a general sense, acquire results of good fidelity and are, therefore, helpful for the design of delicate microfluidic devices or components. However, CFD methods require significant computational resources—in terms of computational memory and time. A CFD simulation of entire microfluidic devices can be hard to get right, and compute times can get up to days or weeks, even on dedicated workstations [19].

2.2. Review of Methods on High Abstraction Levels

Methods with a high level of abstraction simulate microfluidic devices by abstracting the underlying physics on a high level. Generally, these methods are based on solutions that can be obtained analytically for fairly simple problems. They map the results to similar problems under a set of simplifying assumptions, or models are obtained empirically through fitted data from experiments or pre-simulations [24]. These methods are generally of poor fidelity, but results for entire microfluidic devices are usually acquired instantly (i.e., in less than a second) [20]. More precisely, high abstraction simulation methods have been proposed for the following microfluidic platforms [25]:

- *Continuous channel-based microfluidics*. This platform consists of a network of rectangular channels with width and height in the order of micrometers. Liquid flow through these channels can practically always be assumed to be laminar, and the flow profile in a channel can be accurately solved using the Hagen–Poiseuille law, i.e.,

$$\Delta p = Q \cdot R_H, \tag{3}$$

  where Q is the flow rate and R_H is the hydraulic resistance of a channel. Using the *Modified Nodal Analysis* (MNA, [21]), the pressure and flow rates of all channels in a connected network can be calculated.
- *Droplet-based microfluidics*. This platform has a network of channels, similar to the continuous channel-based microfluidics platform, with additional droplets of a fluid that is immiscible with the carrier fluid (continuous phase). The hydraulic resistance in Equation (3) can be split into the resistance of the channel R_H^{channel} and the resistance of a droplet R_H^{droplet} present in that channel, i.e.,

$$R_H = R_H^{\text{channel}} + R_H^{\text{droplet}}. \tag{4}$$

  Based on this and the MNA, this platform can be simulated on a high abstraction level [22].
- *Paper-based microfluidics*. In paper-based microfluidics, a liquid is transported through a two-dimensional sheet of paper using capillary force. The one-dimensional transport of a liquid front through a porous medium is given by the Washburn equation, i.e.,

$$L^2 = \frac{\gamma D t}{4\mu}, \tag{5}$$

where L is the traversed distance of the fluid front, γ is the effective surface tension, D is the diffusivity coefficient, t is time, and μ is the dynamic viscosity. Based on Equation (5), the capillary transportation of fluid can be simulated for porous channels with arbitrary cross-sectional shapes [23].

3. Accelerating CFD Simulations

Motivated by the fast computation time of methods with a high level of abstraction, the possibility to accelerate CFD simulations by exploiting said methods is investigated. In this work, we aim to accelerate CFD simulations for steady-state flow of continuous channel-based microfluidic devices as a representative. Firstly, the semantics of continuous channel-based microfluidics are covered in this section, and the potential for a faster simulation is highlighted. Afterward, a method will be proposed that explicitly exploits that potential and accelerates CFD simulations. Based on that, implementation details for this method and a summary of corresponding case studies, including evaluation results, are provided in the following sections.

3.1. Continuous Channel-Based Microfluidics

In continuous channel-based microfluidics, we consider a network as sketched in the middle of Figure 1. This example network has three inlets and one outlet and contains a homogeneous fluid (no mixture). It is depicted here as a two-dimensional network of channels with width w, and the extension to the third dimension for real-world microfluidic devices can be performed by adding a height parameter h. Furthermore, without loss of generality, we assume an adiabatic system and ignore gravity effects, such that the only relevant fields are the pressure p and velocity u of the fluid.
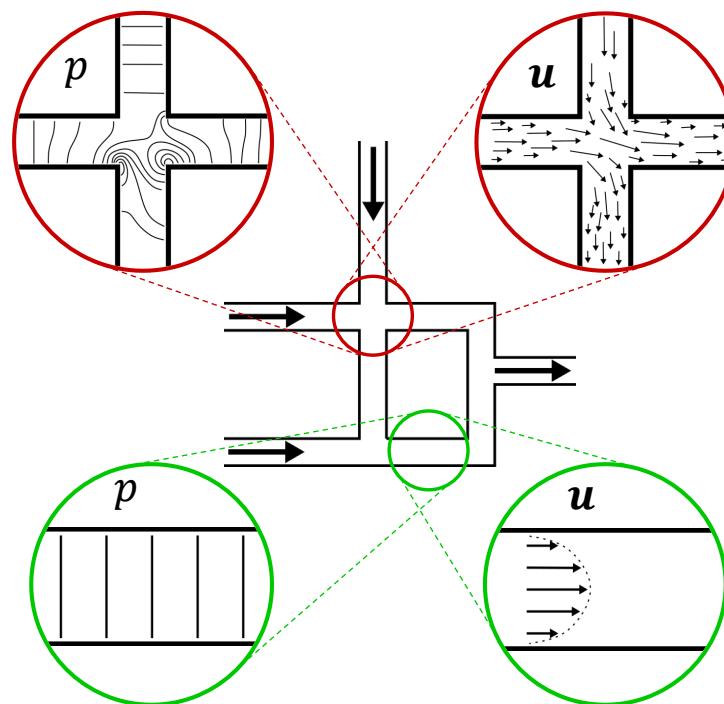

Figure 1. Example network for continuous channel-based microfluidics. The pressure p and velocity fields u are shown in detail for a crossing (red) and a straight channel section (green).

However, as in this setup, the pressure and velocity fields of a fluid in a microfluidic network are generally complex and require dedicated methods to solve Equations (1) and (2). This is sketched in Figure 1 for the pressure and velocity field of the steady-state flow at the location where channels cross (the red circles at the top). The contour lines of the pressure field and vectors of the velocity field are chaotic and not easy to predict. On the other hand,

if we look at the pressure and velocity fields of a straight channel section (the green circles at the bottom), we notice a more organized and streamlined flow. The contour lines of the pressure field are straight, evenly spaced, and perpendicular to the channel. We can see here that, for straight channels, the pressure is a point-value along the channel and equal over the channel's cross-section. The velocity vectors are also organized and show a parabolic (for Newtonian fluids [20]) flow profile, as can be found, for a two-dimensional channel, by using

$$\boldsymbol{u}(x,y) = \left(\begin{matrix} \dfrac{y(h-y)}{2\mu} \dfrac{\partial p}{\partial x} & 0 \end{matrix} \right)^{T}, \tag{6}$$

where x is the direction parallel to the channel, and y is perpendicular to the channel. This is the cornerstone, on which Equation (3) and, therefore, the high abstraction simulation method for continuous channel-based microfluidics is based.

3.2. The Proposed Accelerated Method

Based on the observation above, we can exploit the high abstraction level simulation method in regions where the flow is highly organized (green circles in Figure 1) and use CFD simulations for regions where the flow is chaotic and hard to predict (red circles in Figure 1). Exploiting the high-speed simulation feature of methods on a high abstraction level not only significantly reduces the compute time for large microfluidic devices (resulting in more favorable scaling of simulations) but also reduces the required memory. To this end, two steps must be taken to apply the accelerated method:

1. The required fidelity for the complete network Ω must be defined and Ω must be split into Ω_{low}- and Ω_{high}-regions, such that $\Omega_{\text{high}} \cup \Omega_{\text{low}} = \Omega$ and Ω_{high} is as large as possible. Here, Ω_{low} is the set of regions that require a good fidelity method and should be simulated on a low abstraction level, whereas Ω_{high} is the set of regions that can be simulated relatively accurately using methods on a high abstraction level.
2. The resulting pressure and velocity fields of the corresponding simulation methods must be equal (or at least in close vicinity) on the interface $\Gamma = \Omega_{\text{high}} \cap \Omega_{\text{low}}$ to ensure continuity of the complete solution φ in the complete network Ω. This means that the pressure and velocity values must be communicated between the simulation methods and subsequently updated in Ω_{low} and Ω_{high}.

In the next section, the implementation details on both steps are illustrated.

4. Implementation Details

To properly describe the implementation details for the method proposed above, the continuous channel-based microfluidic device sketched in Figure 1 is used as a running example. Furthermore, the LBM and MNA (as reviewed in Section 2) are used as simulation methods for Ω_{low} and Ω_{high}, respectively. The idea proposed above can be realized as follows.

4.1. Step 1: Identifying the Required Fidelity

Recall that the first step aims at identifying (ideally many) Ω_{high}-regions where a high abstraction simulation method is sufficient and, hence, can be utilized to accelerate the required computations. As sketched before in Figure 1, those regions can usually be identified easily. For example, straight channels belong to Ω_{high}, while, e.g., crossings and T-junctions better remain in Ω_{low}. Γ should be located inside a straight channel, sufficiently far from, e.g., a crossing or junction, where the flow is organized and the pressure can be regarded as a point-value along the channel. The resulting separation of the network is illustrated in Figure 2a, where the two-dimensional regions represent Ω_{low}, and the lines and nodes constitute Ω_{high}.
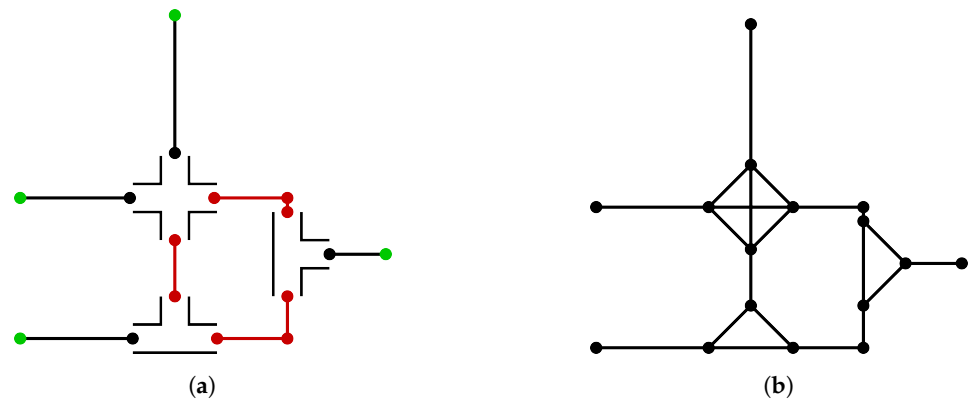
(**a**)	(**b**)

Figure 2. The example network in the proposed method. (**a**) The separation of the example network into Ω_{low} and Ω_{high}. (**b**) The network during the initial iteration; Ω_{low} is replaced by fully connected graphs, and the resulting network is used to find q^0.

Please note that the identification step is not restricted to straight channels, crossings and T-junctions. Also, arbitrary channel shapes or components on the microfluidic device (e.g., heaters, mixers, droplet generators, etc.) could be identified as Ω_{low} as long as corresponding high abstraction level simulation methods are available.

4.2. Step 2: Defining the Communication

The second step aims at ensuring that the complete solution φ is continuous in Ω. Hence, the correspondingly obtained values (here, pressure and velocity fields) from both simulation methods need to be adequately communicated from/to Ω_{high} and Ω_{low}, and they must be subsequently updated in the respective regions. This is similar to simulation methods of multiphysics problems [26,27]. Eventually, once these values align, we obtain a converged complete solution φ. In the proposed method, this is accomplished using an iterative method, i.e., a method that tries to find a local solution (fixed point) iteratively [28,29]. More precisely,

$$q^n = f(q^{n-1}), \tag{7}$$

where, in this case, q^n is the quantity on Γ that is communicated between Ω_{low} and Ω_{high} at timestep n, and $f(q)$ is the update function, given by the iterative method. The stability and convergence characteristics of the iterative method highly depend on the quality of the initial approximate solution, i.e., the initial condition q^0.

To find the initial condition q^0, the complete network is first solved completely using the MNA, i.e., the high abstraction method. In Figure 1, Ω_{low} consists of crossings and T-junctions but could, in reality, contain *any* arbitrary shape that can be simulated using CFD. Therefore, the regions in Ω_{low} are replaced by fully connected graphs, where each in-/outlet is treated as a node. This replacement is depicted in Figure 2b. The resulting network can be solved using only the MNA. This is *not* the solution φ to the actual problem, since we abstracted Ω_{low} by fully connected graphs. However, the solution to this abstracted problem can be used as an initial condition q^0 for the iterative method.

Having that, the communication can be performed in either of the two ways sketched in Figure 3. A low abstraction solver (such as the LBM solver) calculates the pressure and velocity fields directly, whereas a high abstraction solver (such as the MNA solver) calculates the flow rate in a channel rather than the velocity field. This means that if we map the flow rate from the MNA solver to the LBM solver (Figure 3a), we need to extrapolate the flow profile based on the flow rate, whereas the reverse mapping (Figure 3b) can be done directly (provided that Γ is located sufficiently far in a straight channel section, such that the pressure is sufficiently uniform on the channel cross-section and can be treated as a point value). Regions that belong to Ω_{high} but are not connected to ground nodes (the green nodes in Figure 2a) need to communicate according to Figure 3a in at least one node.

If this is not the case, the absolute pressure, i.e., the pressure difference with respect to the reference pressure p^0 at the ground nodes, is not propagated correctly. These regions are highlighted in red in Figure 2a).
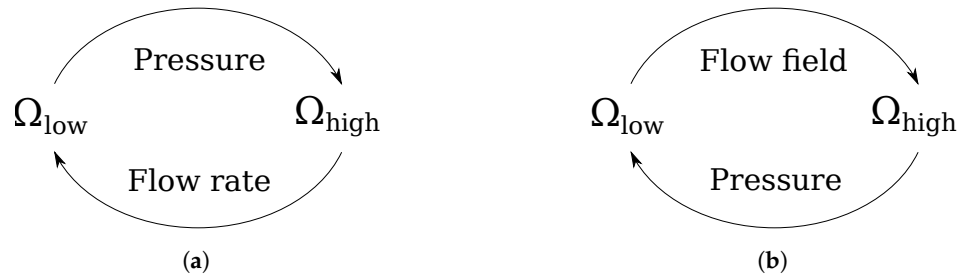

(**a**)   (**b**)

Figure 3. Communication schemes for the pressure and flow fields. (**a**) Communicate the flow rate from Ω_{high} to Ω_{low} and the pressure vice versa. Here, the flow field information must be extrapolated from the communicated flow rate. (**b**) Communicate the flow field from Ω_{low} to Ω_{high} and the pressure vice versa.

To ensure a converged complete solution, we used an iterative method based on *Successive Over-Relaxation* (SOR, [28]) to find the values for pressure and velocity on Γ. From the initial condition q^0, the boundary conditions of the LBM solver are updated in every iteration according to the newly found values from the MNA, corrected by a relaxation factor α, i.e.,

$$q_{\text{low}}^n = (1 - \alpha)\, q_{\text{low}}^{n-1} + \alpha\, q_{\text{high}}^{n-1}. \tag{8}$$

Here, q_{high}^n is evaluated using the MNA with the most recent pressure and velocity information obtained from the LBM solver. The LBM is in itself also an iterative solver, and the frequency at which the boundary conditions are updated influences the stability of the LBM. To ensure stability in the LBM, the values of the boundary conditions are only updated every θ "LBM iteration steps". The iterative approach in Equation (8) is solved until the convergence criterion

$$|q^n - q^{n-1}| \leq \epsilon \tag{9}$$

is met, where ϵ can be chosen arbitrarily small (until machine precision is reached).

5. Case Studies and Evaluation Results

The approach to accelerate CFD simulations, as proposed above, has been implemented for the considered continuous channel-based microfluidics. As corresponding simulation tools, we used OpenLB v1.5 [30] for the LBM and an in-house implementation of the MNA (which, as discussed above, are used as the low and high abstraction simulation methods, respectively). The source code of the proposed method, that has been developed for this work, is available online (available at: https://github.com/cda-tum/mmft-hybrid-simulator; accessed on 6 October 2023) .

Using the resulting implementation, several case studies were conducted to evaluate whether the idea proposed in this work indeed yields an improvement in CFD simulations. This section summarizes the respectively obtained findings. To this end, first the setup of the case studies is reviewed. Afterward, the obtained results are presented and discussed.

5.1. Setup: Considered Cases and Parameters

In our case studies, four different continuous channel-based microfluidic networks, as shown in Figure 4 (denoted Network 1–4 in the following), were considered. All networks are two-dimensional. For each of these networks, we considered different amounts of disconnected regions in Ω_{low}, as well as different lengths of the channels that constitute Ω_{high}—providing a proper variety of test cases with different coverages of Ω_{low}- and

Ω_{high}-regions. More precisely, following the discussion from Section 4, all junctions and crossings of channels are identified as Ω_{low}-region (indicated by the red dotted squares in Figure 4), and the connecting channels are identified as Ω_{high}. Γ was always placed inside a straight channel at a distance of two channel thicknesses from the corresponding junction or crossing. The length l of all channels is subsequently set to 1, 2, 3, and 4 mm for all four networks, whereas the Ω_{low}-regions remain constant. Overall, this leads to a total of 16 separate cases.

(**a**) Network 1

(**b**) Network 2

(**c**) Network 3
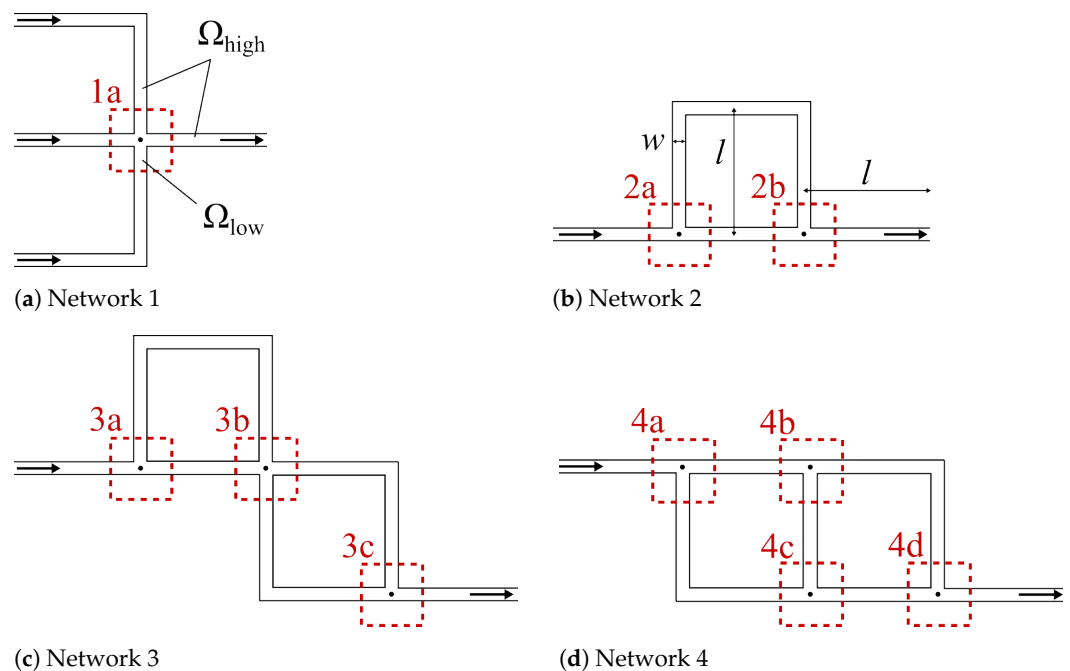
(**d**) Network 4

Figure 4. The networks of the considered case studies. The number of separate regions in Ω_{low} increments with the case studies, starting at one region (**a**) for Network 1 and ending at four regions (**a**–**d**) for Network 4.

The channels of all networks are rectangular with a width of 100 μm, and all networks have the inlets located on the left-hand side, with pressure boundary conditions of 1000 Pa, and outlets on the right-hand side, with pressure boundary conditions of 0 Pa. For all test cases, the fluid inside the network is an incompressible homogeneous fluid with a density of 1000 kg/m^3 and a kinematic viscosity of $1 \cdot 10^{-6}$ m^2/s.

The test cases are solved on a regular grid with a resolution of 20 grid cells across the width of each channel, where applicable (i.e., the resolution of the CFD simulation and Ω_{low}). Additionally, for the proposed method, we set the values $\theta = 10$, $\epsilon = 0.01$, and $\alpha = 0.01$ for Networks 1 and 2, and 0.003 for Networks 3 and 4. All simulations were performed without compiler optimization on a single CPU core (no parallelism) of an AMD Ryzen Threadripper PRO 5955WX CPU [31].

5.2. Obtained Results

In order to evaluate the performance of the proposed method, two aspects are essential: The runtime required for the respective CFD simulations (as we are aiming to accelerate them), as well as the accuracy (as potential accelerations ideally should yield the same results).

Concerning the former, Table 1 lists the respectively obtained results. Here, for all networks from Figure 4 (listed from left to right), as well as for all channel lengths (listed in the rows), the correspondingly required runtimes of the original as well as the proposed method are provided. Additionally, the resulting speed-ups obtained by the proposed method are listed. As mentioned previously, the LBM was used in this work as a representative for CFD simulation methods. However, the computational complexity of the LBM is

similar to that of, e.g., the FVM [20] and speed-ups of similar order of magnitude can be expected for other simulation methods.

Table 1. Required runtimes of the original CFD simulations and the proposed method with corresponding speed-ups.

	Network 1			Network 2			Network 3			Network 4		
	Runtime [hh:mm:ss]		Speed-Up	Runtime [hh:mm:ss]		Speed-Up	Runtime [hh:mm:ss]		Speed-Up	Runtime [hh:mm:ss]		Speed-Up
l	CFD	Proposed		CFD	Proposed		CFD	Proposed		CFD	Proposed	
1	03:15:51	00:02:04	94.7	01:34:08	00:09:07	10.3	04:11:49	00:20:05	12.5	02:21:41	00:44:28	3.2
2	13:23:38	00:02:12	365.2	06:16:43	00:08:55	42.2	11:55:38	00:15:41	55.0	07:56:19	00:59:12	8.0
3	19:48:30	00:02:26	487.0	14:10:54	00:08:44	81.9	28:32:50	00:15:57	107.3	16:16:49	01:07:49	14.4
4	52:27:35	00:02:58	1055.4	21:48:48	00:08:13	159.3	49:55:21	00:15:39	191.2	29:03:13	01:19:53	21.8

Concerning accuracy, direct comparisons of the pressure and velocity fields obtained with CFD simulations and the proposed method are given in Figures 5–8. Figure 5 shows all the obtained results for Region 1a of Network 1 (Figure 4a) for all four channel lengths l, i.e., the complete Network 1-column in Table 1. Figures 6–8, respectively, show the obtained results for all Ω_{low}-regions in Networks 2, 3, and 4 Figure 4b–d, at channel length $l = 1$, i.e., the top row in Table 1. Finally, we list the pressure values and velocity magnitudes obtained by both approaches for all test cases in Tables 2 and 3. These values were taken in the Ω_{low}-regions (labeled 1a, 2a, ..., 4c, 4d in Figure 4) at the measuring points indicated by the black dots in Figure 4. Since the networks were simulated for four different channel lengths l, each measuring point has four pressure values.

Table 2. Obtained pressure values of all test-cases at the measuring points (as denoted in Figure 4a–d) obtained from the original CFD simulation and the proposed method.

		Pressure [Pa]				Pressure [Pa]	
Ω_{low}	l	CFD	Proposed	Ω_{low}	l	CFD	Proposed
1a	1	710.1	712.6	3c	1	296.5	294.4
	2	680.8	681.8		2	275.0	272.9
	3	552.7	553.9		3	270.7	268.7
	4	670.6	671.0		4	269.1	267.2
2a	1	661.8	662.3	4a	1	729.1	728.9
	2	643.6	643.6		2	712.0	711.4
	3	639.6	639.5		3	708.5	707.7
	4	638.2	637.9		4	707.3	706.3
2b	1	398.9	397.8	4b	1	568.9	567.0
	2	373.4	372.2		2	539.6	537.8
	3	368.2	367.0		3	533.9	532.0
	4	366.3	365.1		4	532.0	529.9
3a	1	748.1	748.0	4c	1	508.5	506.7
	2	736.8	736.5		2	480.7	478.6
	3	734.7	734.1		3	475.3	473.1
	4	734.0	733.3		4	473.3	471.0
3b	1	565.3	563.0	4d	1	325.9	323.8
	2	541.7	539.8		2	302.7	300.4
	3	537.0	535.3		3	298.2	296.0
	4	535.5	533.7		4	296.5	294.4

Table 3. Obtained velocity magnitudes of all test-cases at the measuring points (as denoted in Figure 4a–d) obtained from the original CFD simulation and the proposed method.

| Ω_{low} | l | Velocity [mm/s] | | Ω_{low} | l | Velocity [mm/s] | |
		CFD	Proposed			CFD	Proposed
1a	1	363.3	369.0	3c	1	192.3	192.4
	2	203.6	207.8		2	102.2	99.3
	3	152.7	155.3		3	69.9	70.2
	4	106.8	109.0		4	53.2	53.5
2a	1	378.6	384.9	4a	1	280.52	285.8
	2	183.0	186.8		2	136.2	139.5
	3	119.6	122.3		3	89.3	91.6
	4	88.6	90.6		4	66.3	68.1
2b	1	344.0	346.7	4b	1	178.2	180.3
	2	169.3	171.1		2	84.0	85.3
	3	112.4	113.6		3	54.9	55.8
	4	84.2	85.2		4	40.8	41.5
3a	1	274.8	279.8	4c	1	143.6	144.7
	2	132.5	130.8		2	76.3	77.2
	3	86.7	88.1		3	51.6	52.3
	4	64.3	66.0		4	38.9	39.5
3b	1	181.5	183.2	4d	1	231.0	231.2
	2	93.0	91.1		2	120.9	121.4
	3	62.3	63.3		3	81.8	82.3
	4	46.8	47.5		4	62.0	62.3

(**a**) Pressure field of Region *1a* with *l* is 1.

(**b**) Velocity field of Region *1a* with *l* is 1.

(**c**) Pressure field of Region *1a* with *l* is 2.

(**d**) Velocity field of Region *1a* with *l* is 2.

Figure 5. *Cont.*

(**e**) Pressure field of Region *1a* with *l* is 3.

(**f**) Velocity field of Region *1a* with *l* is 3.

(**g**) Pressure field of Region *1a* with *l* is 4.
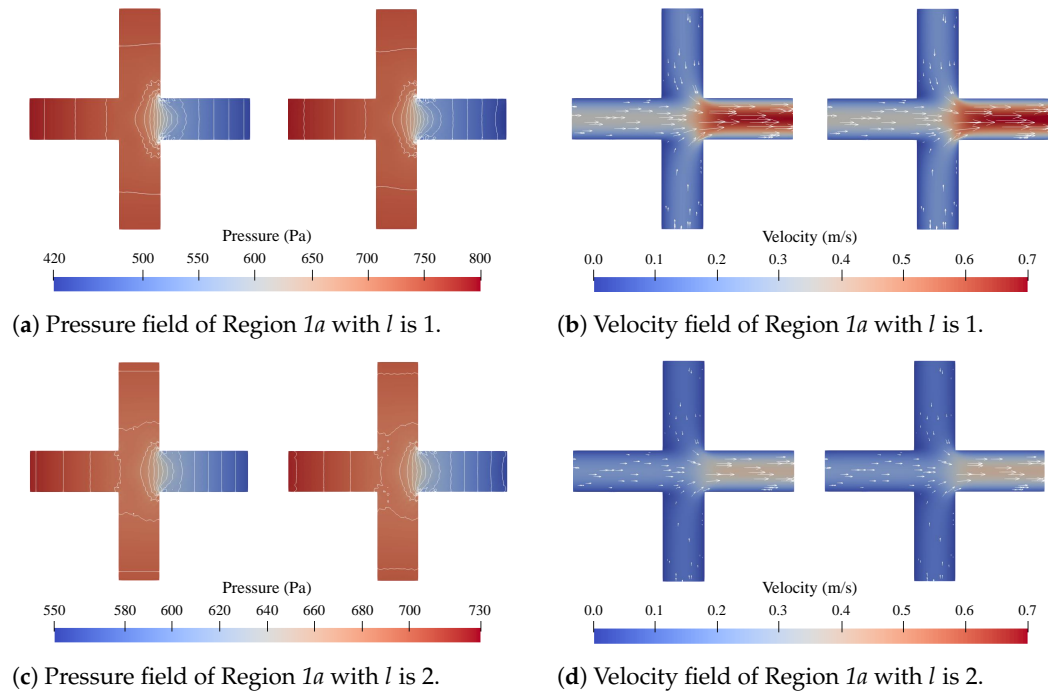

(**h**) Velocity field of Region *1a* with *l* is 4.
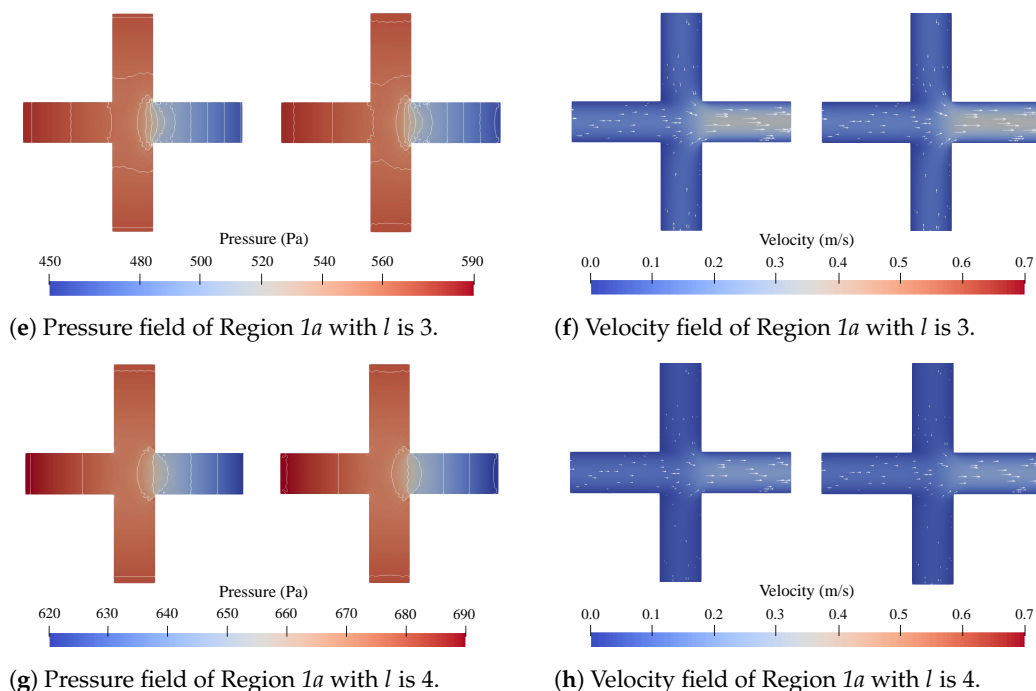
Figure 5. The pressure and velocity fields obtained from the CFD simulations (**left**) and the proposed method (**right**) for Region *1a* in Network 1 with *l* is 1, 2, 3 and 4.
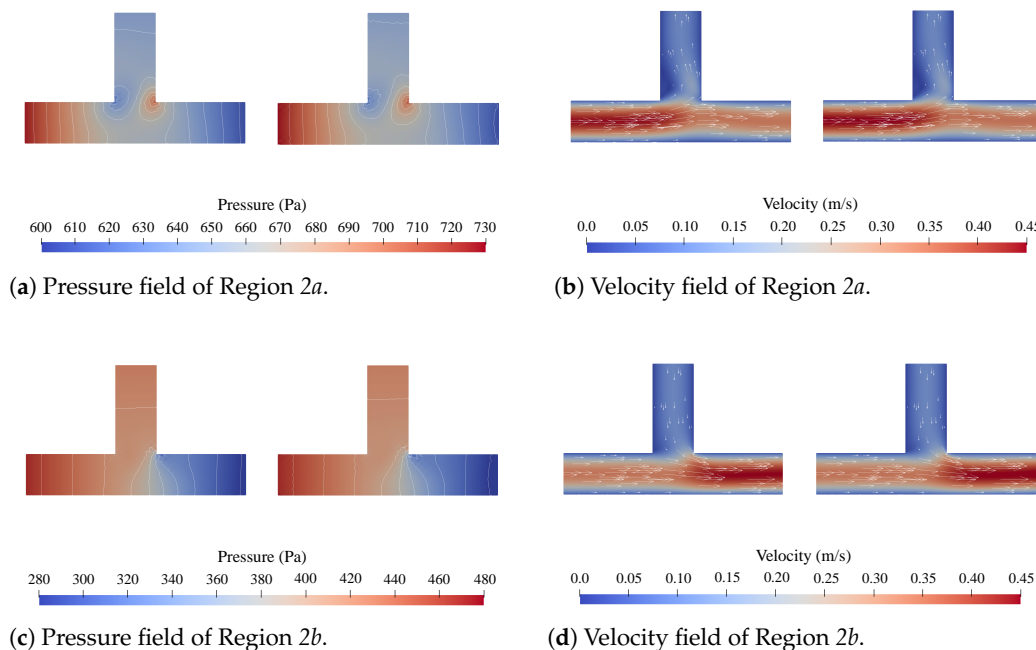

(**a**) Pressure field of Region *2a*.

(**b**) Velocity field of Region *2a*.

(**c**) Pressure field of Region *2b*.

(**d**) Velocity field of Region *2b*.

Figure 6. The pressure and velocity fields obtained from the CFD simulations (**left**) and the proposed method (**right**) for Regions *2a* and *2b* in Network 2 at *l* is 1.

The results clearly confirm the improvement and benefit of the proposed acceleration method. First, we can see from the numbers summarized in Tables 2 and 3 that both the original CFD simulation, as well as the proposed method, more or less provide the same results. That is, using higher levels of abstractions for Ω_{high}-regions does not significantly affect the simulation results. At the same time, this enables impressive speed-ups. In fact, the numbers summarized in Table 1 do not only show that the proposed method *always* generates the results faster than the original CFD method, but also that it constantly does so by several factors—in many cases even by up to three orders of magnitudes.
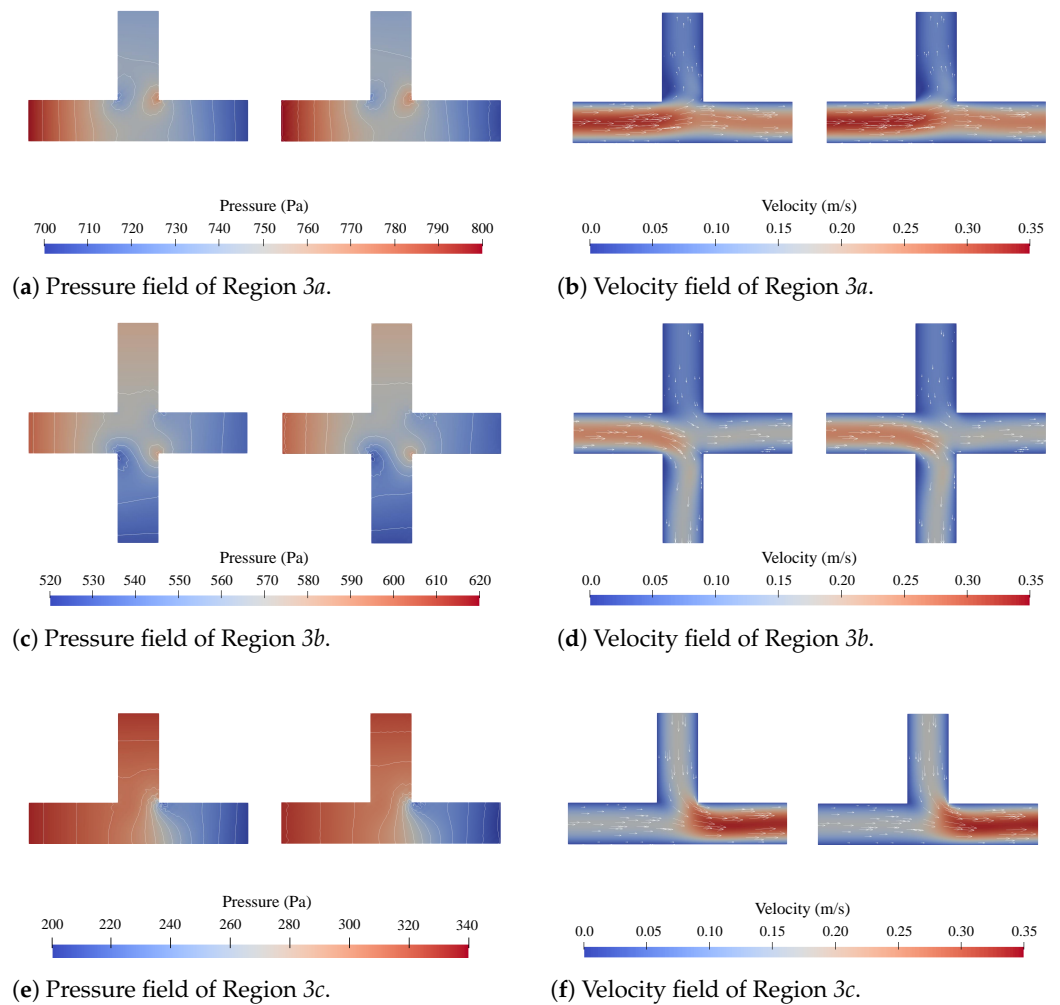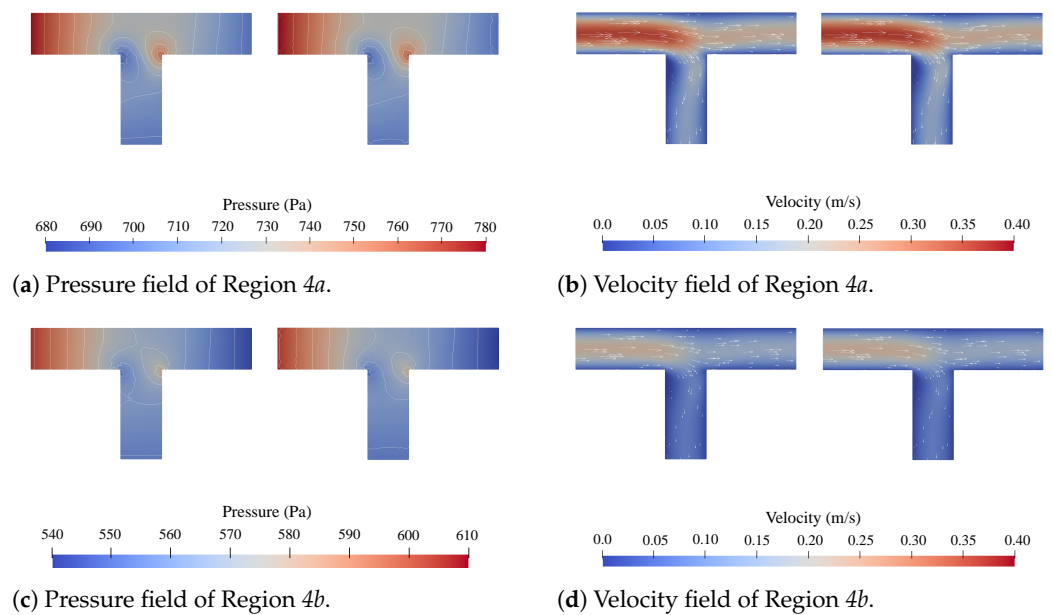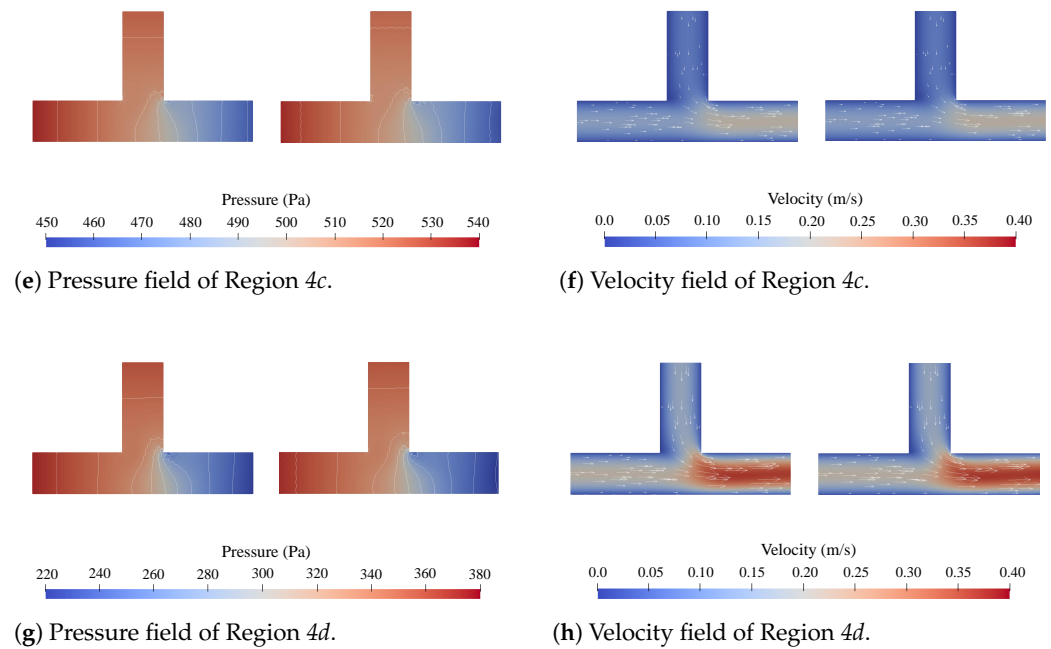
(**a**) Pressure field of Region *3a*.

(**b**) Velocity field of Region *3a*.

(**c**) Pressure field of Region *3b*.

(**d**) Velocity field of Region *3b*.

(**e**) Pressure field of Region *3c*.

(**f**) Velocity field of Region *3c*.

Figure 7. The pressure and velocity fields obtained from the CFD simulations (**left**) and the proposed method (**right**) for Regions *3a*, *3b*, and *3c* in Network 3 at l is 1.

(**a**) Pressure field of Region *4a*.

(**b**) Velocity field of Region *4a*.

(**c**) Pressure field of Region *4b*.

(**d**) Velocity field of Region *4b*.

Figure 8. *Cont*.

(**e**) Pressure field of Region *4c*.

(**f**) Velocity field of Region *4c*.

(**g**) Pressure field of Region *4d*.

(**h**) Velocity field of Region *4d*.

Figure 8. The pressure and velocity fields obtained from the CFD simulations (**left**) and the proposed method (**right**) for Regions *4a*, *4b*, *4c*, and *4d* in Network 4 at *l* is 1.

5.3. Discussion

The obtained results, as summarized above, clearly show the benefits of the proposed method. On top of that, they also provide further, more detailed insights, as well as implications and ideas for further extensions. These are discussed in the following.

Firstly, with respect to the accuracy, the results presented in Figures 5–8 and Tables 2 and 3 show a strong alignment between the proposed method and the corresponding CFD simulations. From Figure 5, it can be concluded that there seems to be a slight improvement in the pressure field as the channel length *l* increases, but this is negligible. The results in Figures 6–8 show that the method is applicable for problems with multiple disconnected Ω_{high}-regions, with a negligible increase in inaccuracy for Ω_{high}-regions that are located between others, such as 3b, 4b and 4c. Following from the pressure contour lines obtained from the CFD approach, which generally appear straight towards Γ, the location of Γ can be said to be sufficiently far away from the crossings and junctions and did not strongly influence the performance of the proposed method. The pressure contour lines obtained from the proposed approach, however, consistently show a curvature near Γ, which can be attributed to numerical intricacies of the underlying implementation of the boundary conditions. This slight inaccuracy can also explain the propagated error to Regions 3b, 4b, and 4c.

The exploitation of higher levels of abstraction is possible due to the availability of reduced order modeling methods for microfluidic flow. In this work, the Hagen–Poiseuille law was used to represent the fluid flow through straight two-dimensional channels. Using a different method of high abstraction, that models flow through three-dimensional channels, such as presented in [32], would allow the proposed method to be extended to three dimensional problems. With this added dimension and, therefore, even worse computational complexity of CFD simulations, the speed-up of the proposed method can be expected to be of even higher orders of magnitude. Provided that methods of high abstraction exist, the proposed method can even be extended to include other physical phenomena, such as diffusion, heat dissipation, or droplets. For straight channels, diffusion and heat dissipation can simply be modeled as a time-dependent transport across the channel width, and high abstraction simulation approaches for droplets are performed by

adding the droplet hydraulic resistance to the channel [33]. Further work is required to include additional physical phenomena in the proposed method.

6. Conclusions

In this work, we proposed an accelerated CFD simulation method for microfluidic devices. The core idea was to utilize higher levels of abstraction whenever possible to improve the simulation runtime while maintaining the fidelity. We developed and implemented a prototype of the resulting simulation approach, using continuous channel-based microfluidic devices as a representative platform. Results obtained from corresponding case studies confirmed the promises of the proposed approach: using higher levels of abstractions for the simulation did not significantly affect the fidelity of the simulation results, but allowed for substantial speed-ups of up to three orders of magnitude. Based on this premise, and with the inclusion of other physical phenomena such as diffusion, heat dissipation, or multiphase flow (e.g., droplets), similar accelerations can be expected for further microfluidic platforms, which is left for future work. Overall, this provides the foundation for more research toward exploiting higher levels of abstraction for simulating microfluidic devices in future work.

References

1. Connery, N.; Gadegaard, N. 30 years of microfluidics. *Micro Nano Eng.* **2019**, *2*, 76–91. [CrossRef]
2. Mathur, L.; Ballinger, M.; Utharala, R.; Merten, C.A. Microfluidics as an enabling technology for personalized cancer therapy. *Small* **2020**, *16*, 1904321. [CrossRef] [PubMed]
3. Pandey, C.M.; Augustine, S.; Kumar, S.; Kumar, S.; Nara, S.; Srivastava, S.; Malhotra, B.D. Microfluidics based point-of-care diagnostics. *Biotechnol. J.* **2018**, *13*, 1700047. [CrossRef] [PubMed]
4. Schonhorn, J.E.; Fernandes, S.C.; Rajaratnam, A.; Deraney, R.N.; Rolland, J.P.; Mace, C.R. A device architecture for three-dimensional, patterned paper immunoassays. *Lab Chip* **2014**, *14*, 4653–4658. [CrossRef]
5. Yang, M.; Tang, Y.; Qi, L.; Zhang, S.; Liu, Y.; Lu, B.; Yu, J.; Zhu, K.; Li, B.; Du, Y. SARS-CoV-2 point-of-care (POC) diagnosis based on commercial pregnancy test strips and a palm-size microfluidic device. *Anal. Chem.* **2021**, *93*, 11956–11964. [CrossRef]
6. He, S.; Joseph, N.; Feng, S.; Jellicoe, M.; Raston, C.L. Application of microfluidic technology in food processing. *Food Funct.* **2020**, *11*, 5726–5737. [CrossRef]
7. Zhu, X.; Wang, K.; Yan, H.; Liu, C.; Zhu, X.; Chen, B. Microfluidics as an emerging platform for exploring soil environmental processes: A critical review. *Environ. Sci. Technol.* **2022**, *56*, 711–731. [CrossRef] [PubMed]
8. Wang, Y.; Luo, S.; Kwok, H.Y.; Pan, W.; Zhang, Y.; Zhao, X.; Leung, D.Y. Microfluidic fuel cells with different types of fuels: A prospective review. *Renew. Sustain. Energy Rev.* **2021**, *141*, 110806. [CrossRef]
9. Grimmer, A.; Chen, X.; Hamidović, M.; Haselmayr, W.; Ren, C.L.; Wille, R. Simulation before fabrication: A case study on the utilization of simulators for the design of droplet microfluidic networks. *RSC Adv.* **2018**, *8*, 34733–34742. [CrossRef]
10. Kirby, B.J. *Micro-and Nanoscale Fluid Mechanics: Transport in Microfluidic Devices*; Cambridge University Press: Cambridge, UK , 2010.
11. Bruus, H. *Theoretical Microfluidics*; Oxford University Press: Oxford, UK, 2007; Volume 18.
12. Ferziger, J.H.; Perić, M.; Street, R.L. *Computational Methods for Fluid Dynamics*; Springer: Cham, Switzerland, 2019.
13. Blazek, J. *Computational Fluid Dynamics: Principles and Applications*; Butterworth-Heinemann: Oxford, UK, 2015.
14. Krüger, T.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O.; Silva, G.; Viggen, E.M. *The Lattice Boltzmann Method*; Springer International Publishing: Cham, Switzerland, 2017; Volume 10, pp. 4–15.
15. Rapp, B.E. *Microfluidics: Modeling, Mechanics and Mathematics*; Elsevier: Amsterdam, The Netherlands, 2022.

16. Bazaz, S.R.; Mashhadian, A.; Ehsani, A.; Saha, S.C.; Krüger, T.; Warkiani, M.E. Computational inertial microfluidics: A review. *Lab Chip* **2020**, *20*, 1023–1048. [CrossRef]

17. Ebner, P.; Wille, R. CFD for Microfluidics: A Workflow for Setting Up the Simulation of Microfluidic Devices. In Proceedings of the Euromicro Conference on Digital System Design (DSD), Durres, Albania, 6–8 September 2023.

18. Jasak, H. OpenFOAM: Open source CFD in research and industry. *Int. J. Nav. Archit. Ocean. Eng.* **2009**, *1*, 89–94

19. Glatzel, T.; Litterst, C.; Cupelli, C.; Lindemann, T.; Moosmann, C.; Niekrawietz, R.; Streule, W.; Zengerle, R.; Koltay, P. Computational fluid dynamics (CFD) software tools for microfluidic applications—A case study. *Comput. Fluids* **2008**, *37*, 218–235. [CrossRef]

20. Takken, M.; Wille, R. Simulation of Pressure-Driven and Channel-Based Microfluidics on Different Abstract Levels: A Case Study. *Sensors* **2022**, *22*, 5392. [CrossRef] [PubMed]

21. Oh, K.W.; Lee, K.; Ahn, B.; Furlani, E.P. Design of pressure-driven microfluidic networks using electric circuit analogy. *Lab Chip* **2012**, *12*, 515–545. [CrossRef] [PubMed]

22. Grimmer, A.; Hamidović, M.; Haselmayr, W.; Wille, R. Advanced simulation of droplet microfluidics. *ACM J. Emerg. Technol. Comput. Syst.* **2019**, *15*, 1–16. [CrossRef]

23. Elizalde, E.; Urteaga, R.; Berli, C.L.A. Rational design of capillary-driven flows for paper-based microfluidics. *Lab Chip* **2015**, *15*, 2173–2180. [CrossRef] [PubMed]

24. Fink, G.; Ebner, P.; Hamidović, M.; Haselmayr, W.; Wille, R. Accurate and efficient simulation of microfluidic networks. In Proceedings of the 26th Asia and South Pacific Design Automation Conference, Tokyo, Japan, 18–21 January 2021; pp. 85–90.

25. Tsur, E.E. Computer-aided design of microfluidic circuits. *Annu. Rev. Biomed. Eng.* **2020**, *22*, 285–307. [CrossRef]

26. Bungartz, H.J.; Schäfer, M. *Fluid-Structure Interaction: Modelling, Simulation, Optimisation*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006; Volume 53.

27. Perelman, T. On conjugated problems of heat transfer. *Int. J. Heat Mass Transf.* **1961**, *3*, 293–303. [CrossRef]

28. Hadjidimos, A. Successive overrelaxation (SOR) and related methods. *J. Comput. Appl. Math.* **2000**, *123*, 177–199. [CrossRef]

29. Kelley, C.T. *Solving Nonlinear Equations with Newton's Method*; SIAM: Philadelphia, PA, USA, 2003.

30. Krause, M.; Kummerländer, A.; Avis, S.; Kusumaatmaja, H.; Dapelo, D.; Klemens, F.; Gaedtke, M.; Hafen, N.; Mink, A.; Trunk, R.; et al. OpenLB–Open source lattice Boltzmann code. *Comput. Math. Appl.* **2021**, *81*, 258–288. [CrossRef]

31. AMD Ryzen™ Threadripper™ PRO 5955WX. Available online: https://www.amd.com/en/product/11801 (accessed on 13 July 2023).

32. Cornish, R.J. Flow in a pipe of rectangular cross-section. *Proc. R. Soc. Lond. Ser. Contain. Pap. Math. Phys. Character* **1928**, *120*, 691–700.

33. Vanapalli, S.A.; Banpurkar, A.G.; Van Den Ende, D.; Duits, M.H.; Mugele, F. Hydrodynamic resistance of single confined moving drops in rectangular microchannels. *Lab Chip* **2009**, *9*, 982–990. [CrossRef] [PubMed]