Engineering Analysis with Boundary Elements 132 (2021) 221–230

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

A multi-domain direct boundary element formulation for particulate flow in microchannels

Alper Topuz [a], Besim Baranoğlu [b,c], Barbaros Çetin [a,d,*]

[a] Microfluidics and Lab-on-a-chip Research Group, Mechanical Eng. Dept., İ.D. Bilkent University, 06800 Ankara, Turkey
[b] Dept. Aeronautical Engineering, Atılım University, 06836 Ankara, Turkey
[c] Metal Forming Center of Excellence, Atılım University, 06836 Ankara, Turkey
[d] UNAM-National Nanotechnology Research Center and Institute of Materials Science and Nanotechnology, Bilkent University, 06800 Ankara, Turkey

ARTICLE INFO

Keywords:
BEM
Multi-domain
Particulate flow
Particle tracking

ABSTRACT

In the present study, a multi-domain boundary element formulation is developed for high surface-area-to-volume ratio problems (i.e. particulate flow in high aspect ratio microfluidic channels, in a porous medium or in microfluidic devices with repetitive structures). The solution domain is decomposed into subdomains and the variable condensation technique is implemented. The solution matrices are built for each subdomain, and the matrices are updated at each time step only for the subdomains in which the particles move at each time step. Ghost domains, which are fictitious domains encapsulating the interfaces between the subdomains, are also introduced in the formulation to treat the particles crossing the interfaces between the subdomains. The formulation reveals that the computation of the subdomain matrices is further simplified for solution domains composed of periodic structures. The results of our study revealed that speed-up values as high as 50 is achievable with the current formulation.

1. Introduction

The boundary element method (BEM) is a well-established numerical tool for the modeling of Stoke's flow. The main idea of the method is to convert the partial differential equations to boundary integral equations, and obtain a system of linear equations which will be solved for unknown boundary values after the discretization of the boundary of the solution domain with proper elements, [1,2]. Alternative to the numerical models based on volume discretization, the boundary only discretization nature of the BEM provides a significant reduction in the number of unknowns for the problems with low surface-area-to-volume ratio or for the problems with unbounded domain. The boundary element representation of creeping flow was first developed by Youngren and Acrivos [3] for the analysis of flow over passing 3D and axisymmetric rigid particles. Following this study, Youngren and Acrivos extended their work for similar flow over deformable bubbles [4], and later a similar study involving viscous drops was introduced by Rallison and Acrivos [5]. Researchers realized that BEM is a viable option for the simulation of many particles immersed in fluids in the Stoke's flow regime which was vital to predict the macroscopic and rheological properties of suspensions (i.e. predicting the sedimentation rate and relative viscosities of suspensions) [6–8]. Since BEM requires only discretization of the boundary, the mesh of the problem can be translated and rotated according to the instantaneous motion of the particles. Moreover, since the geometric representation of the particles are achieved accurately by boundary elements, there is no restriction on the shape of the particles.

With development of advanced micro- and nano-fabrication techniques, microfluidics become an active research area in 1990s. Although gaseous flow in microscale may include some additional effects [9–11] (e.g. slip boundary condition at the wall, compressibility effects even at low Mach number flows, thermal creep etc.) which are not present in macroscale, liquid flow in microchannels is governed by Stoke's flow and no-slip boundary condition unless electrokinetic effects in the case of electric field driven flows or electroviscous effects in the case of pressure driven flows (which may come into picture in microchannels with dimensions less than 5 μm for deionized water) are present [12]. Microfluidics technology offers unique advantages for the next generation diagnostic and therapeutic devices. The manipulation of bio-particles in buffer solutions is the key ingredient for many microfluidic operations such as bio-particle separation, bio-particle sorting, lysis of bio-particles, etc. [13], thus the simulation of particle motion is a crucial step for an efficient design of microfluidic platforms. In microfluidic applications, typical flow speed inside a microchannel is low which results in very low Reynolds number, and

* Corresponding author at: Microfluidics and Lab-on-a-chip Research Group, Mechanical Eng. Dept., İ.D. Bilkent University, 06800 Ankara, Turkey.
E-mail address: barbaros.cetin@bilkent.edu.tr (B. Çetin).

https://doi.org/10.1016/j.enganabound.2021.07.012
Received 14 March 2021; Received in revised form 23 June 2021; Accepted 25 July 2021
Available online 7 August 2021

hence Stoke's flow. In this sense, the simulation of particle trajectory inside microchannels is a moving boundary problem which is similar to the simulation of many particles immersed in Stoke's flow, an exception being the inclusion of strong confinement effect of the microchannels that may be in the form of a complex microfluidic channel network. In addition, particles may be under the action of electrical, magnetic and/or acoustic fields as well as the flow field. Therefore, particle–wall interactions are as important as particle–particle interactions which may not be only due to hydrodynamic interactions for microfluidic applications. Actually, these particle–wall interactions are responsible for the successful manipulation of the bio-particles especially for particle separation and trapping applications [13].

To simulate the particle motion for microfluidics, there are basically two approaches: (i) Lagrangian Particle Tracking Method (LTM) and (ii) stress tensor approach [14]. In LTM, particles are assumed to be point particles and the effects of the particle's presence on the field variables are ignored. In this case, only the effect of the field variables on the particle is considered. One major advantage of LTM is that it does not require high computational cost. Once the field variables are obtained without the presence of particles, particle trajectories can be evaluated at the post-processing step using the correlations to obtain the forces acting on the particles as a function of field variables. LTM is valid if the particle concentration is low (*i.e.* dilute solution) to justify the neglection of particle–particle interactions, and if the particle size is small compared to the channel dimensions to justify the neglection of particle–wall interactions. Therefore, LTM can simulate many particles by ignoring any particle–particle interactions, but allowing a statistical analysis [15–19]. Despite its limitations, LTM has been implemented for the simulation of particle motion in the literature for electrokinetic [18,20–22], acoustophoretic [15–17] as well as inertial microfluidic applications [19]. In the stress tensor approach, the size effect of particles are incorporated and the field variables are solved with the presence of finite-sized particles. The resultant force on the particles can be obtained by integrating the appropriate stress tensor on the particles' surface. Owing to the Stokes flow nature of the flow field, and modeling of electric, acoustic and/or magnetic field with linear differential equations for microfluidics, BEM is a perfect match for the simulation of particle trajectories inside microchannels.

For particle tracking with BEM, both direct [6–8,23–26] and indirect [27–33] boundary element formulations were developed in the literature. The direct boundary element formulation is in terms of physical quantities, and the resulting equations strongly resembles Fredholm integral equations of first kind. Due to the fact that the kernel functions have singular nature, the resulting system of equations may be poorly conditioned, hence direct solution methods such as Gauss elimination needs to be implemented. The situation may worsen for the case of particles with large aspect ratio and/or particles are in close proximity of each other or the confinement. However, poor conditioning can be circumvented by the application of higher-order elements and accurate calculation of the singular integrals [34]. Alternatively, the indirect boundary element formulation is in terms of distribution of doublets over the boundary with fictitious double layer density, and the resulting equations are Fredholm integral equations of second kind. The resulting system of equations are well-posed, hence the solution is stable and iterative solvers with pre-conditioners may be implemented. A major difficulty with the indirect methods is the problem specific formulation and problem specific iterative solution of the problem [33]. Moreover, the indirect formulation may require problem specific additional mathematical derivations for the determination of an external force (e.g. electric, magnetic and/or acoustic) acting on the particle whereas the implementation of the direct formulation is straightforward for any other physics. A direct boundary element formulation, which included a condensation process for the particle nodes imposing rigid body motion to obtain only the particles' trajectories and avoid the calculation of the element interactions on non-moving boundaries at each time-step, was developed to simulate the particle motion inside microfluidic channels [35]. Additionally, direct boundary element formulations were also developed to simulate the motion of particle under the flow and electric fields in a microfluidic device [14,36,37].

The advantages offered by BEM for particulate flows are hindered as the surface-area-to-volume ratio of the problem increases. Some examples of these kind of problems are the particle flow in high aspect ratio microchannels and/or particle flow in a porous medium, more specifically flow in direct lateral displacement (DLD) microfluidic devices. DLD devices utilizes the hydrodynamic interactions of particles with a series of periodic or non-periodic posts within a microchannel to achieve size based separation [13,38,39]. For these large scaled problems, direct boundary element formulation becomes unfeasible since the resulting system of equations are fully populated and have high condition numbers which makes the solution process computationally costly. In addition, a huge main memory allocation is required for the computation on large matrices [40–43]. One option to overcome these drawbacks is to renounce BEM and employ Stokesian dynamics – a semi-analytical method – which relies on constructing a resistance matrix for particle-obstacle systems to capture the hydrodynamic interactions. Stokesian dynamics become a powerful tool when the particles and obstacles possess spherical geometry and distance between the particles and obstacles are either large or small. Non-spherical geometries are not well suited for the method and hydrodynamic interactions of bodies can be resolved with an interpolation when the distance is mediocre and thus such a technique is not well-suited for DLD chip simulations [44].

In order to circumvent the disadvantages of single domain BEM, multi-domain boundary element method (MDBEM) has been proposed in the literature. Unlike single domain BEM, MDBEM decomposes the solution domain into number of subdomains, discretizes the boundary of each subdomain and finally either solves each local system of linear equations iteratively or assembles a global matrix by employing compatibility conditions at the interfaces. Since the resulting matrices are smaller than that of the single domain case, memory requirement, and hence the CPU time can be drastically reduced [45,46]. Furthermore, the final assembly matrix is diagonally dominant, therefore the condition numbers of the resulting matrices significantly improve [42,47,48]. MDBEM was successfully employed for heat transfer [40,41,43,45,46,49–54], elasticity [42,55–57] and Stoke's flow [47,48,58–61] problems, and the studies reported significant improvements both in memory requirements and CPU time. In MDBEM, two common approaches were implemented in the solution of boundary element equations. Some studies employed iterative schemes [40,41,45–47,50,52–54,58–61], and the others employed variable condensation technique [42,43,49,51,55–57] in which some variables are eliminated and the size of the final system of equations are reduced. The selection of iterative schemes or variable condensation technique is problem specific. For large scale problems, iterative techniques may suffer from convergence issues, and the fine tuning of convergence parameters as well as initial guesses may be required [41,46]. Condensation techniques, on the other hand, are not easy to implement on computer when the solution domain is partitioned with unstructured subdomains. Microfluidic networks are slender structures with large aspect ratio (*i.e.* large length over width ratio). This slender geometry can be easily decomposed into consecutive structured subdomains. However, the convergence of the iterative methods may be problematic for the case of large numbers of consecutive structured subdomains since in most of the typical particulate flow problems boundaries are subjected to homogeneous Dirichlet and Neumann boundary conditions (e.g. no slip and pressure outlet), and so the convergence of the iterative schemes may be severely affected [41]. Therefore, the topology with consecutive structured subdomains is more suitable for the condensation techniques rather than the iterative methods. Despite the aforementioned benefits of MDBEM, the implementation is not straightforward for particulate flow problems. In multi-domain formulation, subdomain boundaries

are present within the solution domain. Since particles are modeled as finite sized objects, they cannot cross the interfaces between the subdomains. Furthermore as a particle approaches a subdomain interface near-singularities may occur. In the case of poor treatment of these integrals, the physics of the problem might be disrupted.

1.1. Present study

Our group developed a direct boundary element formulation based on the stress tensor approach for tracking multiple particles through a condensation process for the particle nodes imposing rigid body motion for microfluidic applications [35], and recently, extended the formulation for multi-physics problems [14]. In the present study, a multi-domain boundary element formulation is developed for the problems with a high surface-area-to-volume ratio (*i.e.* particulate flow in high aspect ratio microfluidic channels, in a porous media or in microfluidic devices with repetitive post structures). The solution domain is decomposed into subdomains and the variable condensation technique is implemented. The solution matrices are built for each subdomain, and the matrices are updated only for the subdomains in which the particles move at each time step. Ghost domains, which are the fictitious domains encapsulating the interfaces between the subdomains, are also introduced in the formulation to treat the particles crossing the interfaces between the subdomains. The formulation is demonstrated in 2D for constant elements without loss of generality, since the extension to higher-order element and/or 3D is straight forward, and the results for higher-order elements and/or 3D would be similar to the cases considered in this study as long as the number of degree of freedom of the resulting equations are similar. The formulation reveals that the computation of the subdomain matrices is further simplified for the solution domains composed of periodic structures. This paper is organized as follows: in Section 2, the MDBEM formulation is presented in a general form. Section 3 gives the details of the computation and the comparison of multi-domain formulation with single domain is presented. In addition, the computational performance of the formulation is also presented for a DLD microchannel architecture for non-periodic and periodic geometries in 2D. The effect of number subdomains and number of particles tracked are also demonstrated. Finally, the summary and conclusions are presented in Section 4.

2. Boundary element formulation

2.1. Boundary integral formulation

For an incompressible Newtonian fluid flowing at low Reynolds numbers can be modeled by Stokes equation as:

$$\nabla \cdot \mathbf{u} = 0 \tag{2.1}$$

$$-\nabla P + \mu \nabla^2 \mathbf{u} + \rho \mathbf{g} = 0 \tag{2.2}$$

where P is pressure, $\mathbf{u}$ is the velocity vector, μ is the dynamic viscosity and $\mathbf{g}$ is the gravitational acceleration. The boundary element equation for Stokes flow reads as:

$$\mathbf{C}(\mathbf{x}_o)\mathbf{u}(\mathbf{x}_o) + \int_{\partial D} \mathbf{u}(\mathbf{x}) \left[\mathbf{n} \cdot \nabla \mathcal{G}(\mathbf{x}_o, \mathbf{x}) \right] d\mathrm{l}(\mathbf{x}) = \int_{\partial D} \mathbf{t}(\mathbf{x}) \mathcal{G}(\mathbf{x}_o, \mathbf{x}) d\mathrm{l}(\mathbf{x}) \tag{2.3}$$

where $\mathbf{u}$ and $\mathbf{t}$ are the velocity and the traction vectors, respectively. The field (source) point is denoted by $\mathbf{x}_o$, and the varied (integration) point is denoted by $\mathbf{x}$. The 2D Stokeslet, $\mathcal{G}(\mathbf{x}_o, \mathbf{x})$, is given by

$$\mathcal{G}(\mathbf{x}_o, \mathbf{x}) = \frac{1}{2\pi\mu} \left[-\mathbf{I} \ln |\mathbf{r}| + \frac{\mathbf{r} \otimes \mathbf{r}}{\mathbf{r} \cdot \mathbf{r}} \right] \tag{2.4}$$

where $\mathbf{r} = \mathbf{x} - \mathbf{x}_o$. Constants $\mathbf{C}(\mathbf{x}_o)$ take the values depending upon the position of the point $\mathbf{x}_o$: If $\mathbf{x}_o$ is on a smooth boundary then $\mathbf{C}(\mathbf{x}_o) = \mathbf{I}/2$, if it is inside the solution domain then $\mathbf{C}(\mathbf{x}_o) = \mathbf{I}$ and if it is outside the solution domain $\mathbf{C}(\mathbf{x}_o) = \mathbf{0}$. Discretization of Eq. (2.3) leads to a system of linear equations which can be expressed in matrix form as:

$$\mathbf{H} \cdot \mathbf{u} = \mathbf{G} \cdot \mathbf{t} \tag{2.5}$$

In this study, the field variables are approximated by constant elements and the geometric variables are approximated by linear elements (in the text, this element type will be referred to as the *constant element*). This way, the singular integrals can be evaluated analytically.

Velocity vector, $\mathbf{u}$, includes the components of the velocity at each boundary node and $\mathbf{t}$ is a vector that includes the components of the corresponding traction vector at the same boundary nodes. The resulting system matrices $\mathbf{H}$ and $\mathbf{G}$ are $2N \times 2N$ in 2D and $3N \times 3N$ in 3D where N is the number of elements in the computational domain. In this study, it is assumed that either the component of the velocity or the traction or a linear combination in a given direction at each node is assigned as a boundary condition on the external boundaries.

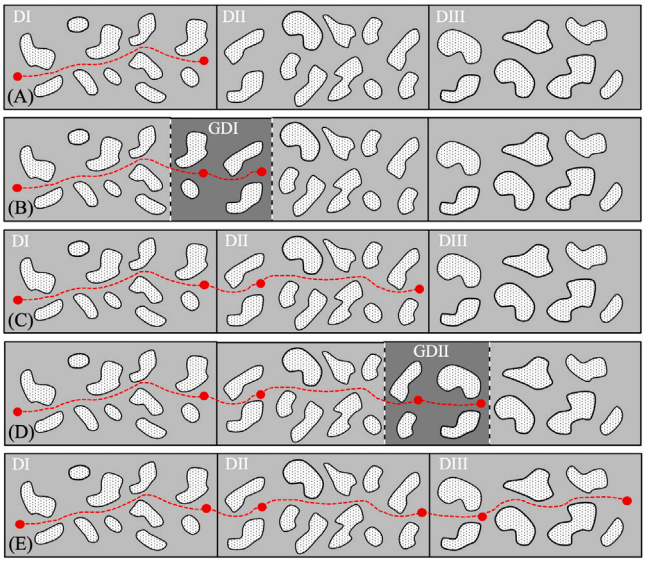

Fig. 1. Schematic representation of the particle flow in the solution domain.

2.2. Multi-domain boundary element formulation

The schematic representation of how the formulation works is demonstrated in Fig. 1. Our goal is to obtain the particle trajectories within a microfluidic channel as shown in Fig. 1-(E) with the red curve. As a general case, a 2D microchannel with number of random inclusions is presented. For the multi-domain formulation, the solution domain is divided into smaller subdomains (in the schematic figure the domain is divided into three pieces denoted by DI, DII and DIII). Once the particle comes close to the interface between the different subdomains, a ghost domain (denoted by GDI and GDII on the figure) is introduced. When the particle passes the interface and is sufficiently away from the interface, the ghost domain is removed from the solution domain. This process continues as the particle reaches the exit of the solution domain.

2.2.1. Subdomain level equations

Depending upon the problem geometry, various partitioning schemes can be utilized to construct subdomains. For unstructured subdomain networks, when used with variable condensation techniques, a proper enumeration of the subdomains and subdomain locations significantly reduces the condition numbers by increasing the sparsity of the matrices [42,47]. In addition, overlapping subdomains can be used to avoid singular and near-singular integrations [51]. In the present

work, we consider a 2D microchannel with periodic and non-periodic inclusions. Schematic representation of our computational domain is shown in Fig. 2 (for the clarity of the figures the inclusions are presented on the figure). Since the channels considered are rectangular, (n) subdomains are placed in a consecutive and structured manner, which results in $(n-1)$ interfaces. Each subdomain $(\Omega^{(\alpha)})$ has two interfaces ($I_{\alpha-1}$ and I_α) except the first and last subdomains as shown in the figure. The subdomains are numbered from left to right. Depending on the position of the particles, some subdomains may include many particles or a single particle, and some subdomains may be not include any particle at all. The boundary of each subdomain is discretized with constant elements. In order to implement multi-domain formulation, the nodes on the boundaries are classified as non-interface, interface and particle nodes following the notation in [1,43]. Interface nodes are shared by two subdomains and non-interface nodes belong to only one subdomain. Particle nodes on which boundary conditions are not specified belong to the particles. Note that problem under consideration is a Stoke's problem, and no external source exists, therefore no need for any volume integral. In case of presence of volume integrals, there would be nodes inside the solution domain. Then, these nodes would be classified as internal nodes [43].

The condensation procedure takes place in two steps: (i) we first condense the non-interface nodes keeping the particle and interface nodes, and then (ii) we condensate the particle nodes imposing rigid body motion and compatibility conditions. We first construct the conventional BEM matrices for each subdomain. For subdomain-α, the subdomain level BEM matrices can be written as:

$$\begin{bmatrix} \mathbf{G}^{(\alpha)}_{AA} & \mathbf{G}^{(\alpha)}_{AK} \\ \mathbf{G}^{(\alpha)}_{KA} & \mathbf{G}^{(\alpha)}_{KK} \end{bmatrix} \begin{Bmatrix} \mathbf{t}^{(\alpha)}_{A} \\ \mathbf{t}^{(\alpha)}_{K} \end{Bmatrix} = \begin{bmatrix} \mathbf{H}^{(\alpha)}_{AA} & \mathbf{H}^{(\alpha)}_{AK} \\ \mathbf{H}^{(\alpha)}_{KA} & \mathbf{H}^{(\alpha)}_{KK} \end{bmatrix} \begin{Bmatrix} \mathbf{u}^{(\alpha)}_{A} \\ \mathbf{u}^{(\alpha)}_{K} \end{Bmatrix} \tag{2.6}$$

where subscript A stands for non-interface nodes, and K for interface and particle nodes. Note that either the velocity or traction components of the non-interface nodes are known whereas neither the velocity nor traction components on the interface and particle nodes are known. Assuming all velocities are specified on the non-interface nodes, and all traction values are unknown without loss of generality, Eq. (2.6) can be recast into:

$$\begin{aligned} \mathbf{G}^{(\alpha)}_{AA}\cdot\mathbf{t}^{(\alpha)}_{A} + \mathbf{G}^{(\alpha)}_{AK}\cdot\mathbf{t}^{(\alpha)}_{K} &= \mathbf{H}^{(\alpha)}_{AA}\cdot\mathbf{u}^{(\alpha)}_{A} + \mathbf{H}^{(\alpha)}_{AK}\cdot\mathbf{u}^{(\alpha)}_{K} \\ \mathbf{G}^{(\alpha)}_{KA}\cdot\mathbf{t}^{(\alpha)}_{A} + \mathbf{G}^{(\alpha)}_{KK}\cdot\mathbf{t}^{(\alpha)}_{K} &= \mathbf{H}^{(\alpha)}_{KA}\cdot\mathbf{u}^{(\alpha)}_{A} + \mathbf{H}^{(\alpha)}_{KK}\cdot\mathbf{u}^{(\alpha)}_{K} \end{aligned} \tag{2.7}$$

To proceed with the first step of the condensation process, $\mathbf{t}^{(\alpha)}_{A}$ can be solved using Eq. (2.7) to obtain the interface equation (i.e. $\mathbf{t}^{(\alpha)}_{K}$) as:

$$\mathbf{t}^{(\alpha)}_{K} = \left(\mathbf{\Lambda}^{(\alpha)}\right)^{-1}\left(\mathbf{\Gamma}^{(\alpha)}\cdot\mathbf{u}^{(\alpha)}_{K} + \boldsymbol{\phi}^{(\alpha)}\right) \tag{2.8}$$

where

$$\begin{aligned} \mathbf{\Lambda}^{(\alpha)} &= \mathbf{H}^{(\alpha)}_{KK} - \mathbf{G}^{(\alpha)}_{KA}\cdot\left(\mathbf{G}^{(\alpha)}_{AA}\right)^{-1}\cdot\mathbf{G}^{(\alpha)}_{AK} \\ \mathbf{\Gamma}^{(\alpha)} &= \mathbf{G}^{(\alpha)}_{KK} - \mathbf{G}^{(\alpha)}_{KA}\cdot\left(\mathbf{G}^{(\alpha)}_{AA}\right)^{-1}\cdot\mathbf{H}^{(\alpha)}_{AK} \\ \boldsymbol{\phi}^{(\alpha)} &= \left[\mathbf{H}^{(\alpha)}_{AK} - \mathbf{G}^{(\alpha)}_{KA}\cdot\left(\mathbf{G}^{(\alpha)}_{AA}\right)^{-1}\cdot\mathbf{H}^{(\alpha)}_{AA}\right]\cdot\mathbf{u}^{(\alpha)}_{A} \end{aligned} \tag{2.9}$$

Eq. (2.8) can be re-written as:

$$\mathbf{t}^{(\alpha)}_{K} = \mathbf{A}^{(\alpha)}\cdot\mathbf{u}^{(\alpha)}_{K} + \mathbf{b}^{(\alpha)} \tag{2.10}$$

where $\mathbf{A}^{(\alpha)} = \left(\mathbf{\Lambda}^{(\alpha)}\right)^{-1}\cdot\mathbf{\Gamma}^{(\alpha)}$ and $\mathbf{b}^{(\alpha)} = \left(\mathbf{\Lambda}^{(\alpha)}\right)^{-1}\cdot\boldsymbol{\phi}^{(\alpha)}$.

2.2.2. Compatibility conditions at the interface

For a subdomain, $\Omega^{(\alpha)}$, without any particle, $\mathbf{t}^{(\alpha)}_{K}$ consists of the traction values on the left interface ($I_{\alpha-1}$) and right interfaces (I_α) of the $\Omega^{(\alpha)}$, respectively. In this case, Eq. (2.10) can be written as:

$$\begin{Bmatrix} -\mathbf{t}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{t}^{(\alpha)}_{I_\alpha} \end{Bmatrix} = \begin{bmatrix} \mathbf{A}^{(\alpha)}_{(\alpha-1)(\alpha-1)} & \mathbf{A}^{(\alpha)}_{(\alpha-1)\alpha} \\ \mathbf{A}^{(\alpha)}_{\alpha(\alpha-1)} & \mathbf{A}^{(\alpha)}_{\alpha\alpha} \end{bmatrix} \begin{Bmatrix} \mathbf{u}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{u}^{(\alpha)}_{I_\alpha} \end{Bmatrix} + \begin{Bmatrix} \mathbf{b}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{b}^{(\alpha)}_{I_\alpha} \end{Bmatrix} \tag{2.11}$$

For the first ($\alpha = 1$) and last domain ($\alpha = n$), since there is only one interface, Eq. (2.10) can be written as:

$$\mathbf{t}^{(1)}_{I_1} = \mathbf{A}^{(1)}_{11}\cdot\mathbf{u}^{(1)}_{I_1} + \mathbf{b}^{(1)}_{I_1} \tag{2.12}$$

$$\mathbf{t}^{(n)}_{I_{n-1}} = \mathbf{A}^{(n)}_{(n-1)(n-1)}\cdot\mathbf{u}^{(n)}_{I_{n-1}} + \mathbf{b}^{(n)}_{I_{n-1}} \tag{2.13}$$

For a subdomain $\Omega^{(\alpha)}$ with particle(s), $\mathbf{t}^{(\alpha)}_{K}$ consists of traction values on the left interface, on the particle and on the right interface, and Eq. (2.10) reads as

$$\begin{Bmatrix} -\mathbf{t}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{t}^{(\alpha)}_{P} \\ \mathbf{t}^{(\alpha)}_{I_\alpha} \end{Bmatrix} = \begin{bmatrix} \mathbf{A}^{(\alpha)}_{(\alpha-1)(\alpha-1)} & \mathbf{A}^{(\alpha)}_{(\alpha-1)P} & \mathbf{A}^{(\alpha)}_{(\alpha-1)\alpha} \\ \mathbf{A}^{(\alpha)}_{P(\alpha-1)} & \mathbf{A}^{(\alpha)}_{PP} & \mathbf{A}^{(\alpha)}_{P\alpha} \\ \mathbf{A}^{(\alpha)}_{\alpha(\alpha-1)} & \mathbf{A}^{(\alpha)}_{\alpha P} & \mathbf{A}^{(\alpha)}_{\alpha\alpha} \end{bmatrix} \begin{Bmatrix} \mathbf{u}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{u}^{(\alpha)}_{P} \\ \mathbf{u}^{(\alpha)}_{I_\alpha} \end{Bmatrix} + \begin{Bmatrix} \mathbf{b}^{(\alpha)}_{I_{\alpha-1}} \\ \mathbf{b}^{(\alpha)}_{P} \\ \mathbf{b}^{(\alpha)}_{I_\alpha} \end{Bmatrix} \tag{2.14}$$

where subscript P stands for particle. In here, the size of the $\mathbf{u}^{(\alpha)}_{I_\alpha}$ is $3N^{(\alpha)}_{I_\alpha}\times 1$ where $N^{(\alpha)}_{I_\alpha}$ is the number of nodes on I_α, the size of the $\mathbf{u}^{(\alpha)}_{P}$ is $3N^{(\alpha)}_{P}\times 1$ where $N^{(\alpha)}_{P}$ is the total number of nodes on the particles' surface in $\Omega^{(\alpha)}$. Once the subdomain level equations are written for each subdomain, the compatibility condition at each interface $\mathbf{u}^{(\alpha)}_{I_\alpha} = \mathbf{u}^{(\alpha+1)}_{I_\alpha}$ and $\mathbf{t}^{(\alpha)}_{I_\alpha} = -\mathbf{t}^{(\alpha+1)}_{I_\alpha}$ can be imposed by the appropriate matrix operation to obtain the global assembly matrix as:

$$\begin{Bmatrix} 0 \\ \mathbf{t}^{(2)}_{P} \\ 0 \\ \vdots \\ 0 \\ \vdots \\ \mathbf{t}^{(n-1)}_{P} \\ 0 \end{Bmatrix} = \tilde{\mathbf{A}}\cdot\begin{Bmatrix} \mathbf{u}^{(1)}_{I_1} \\ \mathbf{u}^{(2)}_{P} \\ \mathbf{u}^{(2)}_{I_2} \\ \vdots \\ \mathbf{u}^{(\alpha)}_{I_\alpha} \\ \vdots \\ \mathbf{u}^{(n-1)}_{P} \\ \mathbf{u}^{(n-1)}_{I_{n-1}} \end{Bmatrix} + \tilde{\mathbf{b}} \tag{2.15}$$

where $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ are the modified $\mathbf{A}$ and $\mathbf{b}$ matrices after the necessary matrix operations, respectively. For each interface (I_α), there is a row whose left-hand side value is equal to zero, and for particle(s) in each subdomain, there is a row whose left-hand side is equal to $\mathbf{t}^{(\alpha)}_{P}$ which is the traction vector on the particles' surfaces in $\Omega^{(\alpha)}$. At this point, the first step of the condensation process is completed. For the second step of the condensation process, rigid body motion constraint will be introduced.

2.2.3. Rigid body motion

In the case of a particulate flow, the problem at hand is a moving boundary problem. There exist contours inside the solution domain which represents the inclusions and/or the particles inside the solution domain. Although no-slip boundary condition is imposed on the inclusions, neither traction nor velocity values are known on the particle boundary. However, a standard BEM formulation requires imposition of a boundary condition on the boundary of a solution domain. As a closure for the problem, we impose the constraint associated with the rigid body motion of the particles (assuming rigid particles), which means each boundary point has a velocity given by

$$\mathbf{u}_P = \mathbf{u}_B + \boldsymbol{\omega}\times\mathbf{r}_P \tag{2.16}$$

where $\mathbf{u}_P$ is the velocity at a node on the boundary of the particle, $\mathbf{u}_B$ is the velocity of the selected center of the particle, $\boldsymbol{\omega}$ is the angular velocity vector and $\mathbf{r}_P$ is the relative position vector of the particle point to the selected center of the particle. Imposing Eq. (2.16) for all the boundary points on the particle can be related to the linear and angular velocities at the centroid of the particles through a kinematic matrix $\mathbf{M}$ as:

$$\mathbf{u}_P = \mathbf{M}\cdot\mathbf{u}_B \tag{2.17}$$

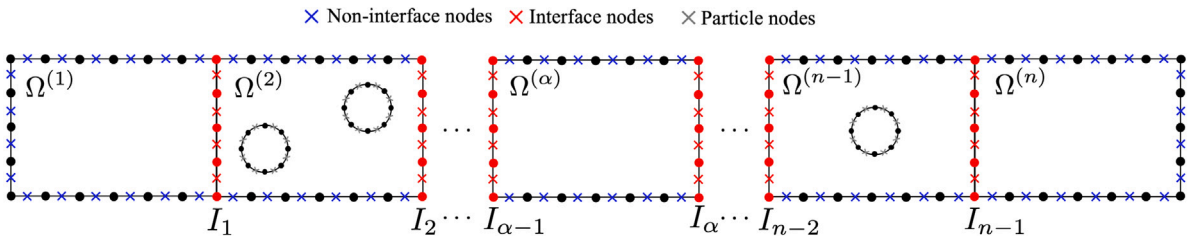

Fig. 2. Schematics of the solution domain.

$$\mathbf{u}_P = \left\{ \{u_1 \quad u_2 \quad u_3\}^1 \quad \{u_1 \quad u_2 \quad u_3\}^2 \quad \cdots \quad \{u_1 \quad u_2 \quad u_3\}^{N_P^{(\alpha)}} \right\}^{\mathbf{T}}$$
$$\mathbf{u}_B = \{u_1^B \quad u_2^B \quad u_3^B \quad \omega_1^B \quad \omega_2^B \quad \omega_3^B\}^{\mathbf{T}}$$

where $\mathbf{u}_q$ is the velocity vector of each particle node, $\mathbf{u}_B$ contains the linear and angular velocity at the center of gravity of the particle. Similarly, the force and moment on each particle can be obtained in a matrix form as [35]:

$$\mathbf{f}_B = \mathbf{F} \cdot \mathbf{t}_P \tag{2.18}$$
$$\mathbf{f}_B = \{f_1^B \quad f_2^B \quad f_3^B \quad m_1 \quad m_2 \quad m_3\}^{\mathbf{T}}$$
$$\mathbf{t}_P = \left\{ \{t_1 \quad t_2 \quad t_3\}^1 \quad \{t_1 \quad t_2 \quad t_3\}^2 \quad \cdots \quad \{t_1 \quad t_2 \quad t_3\}^{N_P^{(\alpha)}} \right\}^{\mathbf{T}}$$

where $\mathbf{f}_B$ contains the net force and moments with respect to center of gravity, $\mathbf{t}_P$ is the corresponding traction vector on the particles' surface and $N_P^{(\alpha)}$ is the number of particles in each subdomain. Neglecting the particle acceleration which can be justified with the low Re number nature of the flow field, and as a result, imposing the force-free and moment-free condition for the particle (i.e. the summation of all the forces and the moments on the particle are equal to zero), $\mathbf{f}_B = \mathbf{f}_{\text{ext}}$ for each particle (note that if there is no external force other than drag, $\mathbf{f}_{\text{ext}} = 0$) [35]. To condensate the particle nodes, in Eq. (2.17), $\mathbf{u}_P = \mathbf{M} \cdot \mathbf{u}_B$ can be substituted, and equating $\mathbf{f}_B^{(\alpha)} = \mathbf{F}^{(\alpha)} \cdot \mathbf{t}_P^{(\alpha)} = \mathbf{f}_{\text{ext}}^{(\alpha)}$ for each particle as:

$$\begin{Bmatrix} 0 \\ \mathbf{f}_{\text{ext}}^{(2)} \\ 0 \\ \vdots \\ 0 \\ \vdots \\ \mathbf{f}_{\text{ext}}^{(n-1)} \\ 0 \end{Bmatrix} = \tilde{\mathbf{F}} \cdot \tilde{\mathbf{A}} \cdot \tilde{\mathbf{M}} \cdot \begin{Bmatrix} \mathbf{u}_{I_1}^{(1)} \\ \mathbf{u}_B^{(2)} \\ \mathbf{u}_{I_2}^{(2)} \\ \vdots \\ \mathbf{u}_{I_\alpha}^{(\alpha)} \\ \vdots \\ \mathbf{u}_B^{(n-1)} \\ \mathbf{u}_{I_{n-1}}^{(n-1)} \end{Bmatrix} + \tilde{\mathbf{F}} \cdot \tilde{\mathbf{b}} \tag{2.19}$$

where $\tilde{\mathbf{F}}$ and $\tilde{\mathbf{M}}$ are

$$\tilde{\mathbf{F}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & & & \cdots & & \mathbf{0} \\ \mathbf{0} & \mathbf{F}^{(2)} & \mathbf{0} & & & & \\ & \ddots & \ddots & \ddots & & & \vdots \\ & & \mathbf{0} & \mathbf{I} & \mathbf{0} & & \\ \vdots & & & \ddots & \ddots & \ddots & \\ & & & & \mathbf{0} & \mathbf{F}^{(n-1)} & \mathbf{0} \\ \mathbf{0} & & \cdots & & & \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{2.20}$$

$$\tilde{\mathbf{M}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & & & \cdots & & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{(2)} & \mathbf{0} & & & & \\ & \ddots & \ddots & \ddots & & & \vdots \\ & & \mathbf{0} & \mathbf{I} & \mathbf{0} & & \\ \vdots & & & \ddots & \ddots & \ddots & \\ & & & & \mathbf{0} & \mathbf{M}^{(n-1)} & \mathbf{0} \\ \mathbf{0} & & \cdots & & & \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{2.21}$$

The global system above can be solved for the interface and particle velocities at each time step. Strictly speaking, as the particle is moving, $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ matrices need to be updated. However, the particle–wall and particle–particle interactions come into play when the separation distance is less than 10×diameter of the particle (d), therefore the effect of one particle in one subdomain would not affect the solution for other subdomains unless the particle is close to the interfaces of the subdomain. In this sense, if a particle is within the $\pm 5d$ of the interface, $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ matrices are not updated during the computations. In the implementation, the ghost domains are generated before the particle reaches the proximity of $5d$ to an interface, and the size of the ghost domains is also chosen so that the distance between the interfaces of the ghost domain is as at least $\pm 10d$ away from the interfaces of the subdomains to expedite the computations.

3. Results and discussion

The computations were performed both for non-periodic and periodic geometries with different number of particles. The geometry was discretized with linear elements whereas the field variables were approximated by constant elements. 20-point Gauss quadrature was employed for the regular integrals, the number of integration points were increased for near singular integrals according to degree of near singularity. Weakly and strongly singular integrals were evaluated analytically. To validate our BE formulation, the comparison of our results for a benchmark problem from the literature [62] is given in Appendix. The formulation developed by our group [35] together with Euler time integration with a time step of 10^{-3} was implemented to obtain the particle trajectories. DELL R720 workstation with 32 Core CPU with 384 GB of RAM is used for the simulations, and code development was performed in MATLAB programming environment. The speed-ups were obtained relative to the single-domain computation.

For the sake of simplicity, we assume particles are buoyant in the buffer solution, and hence the only force acting on the particles is the hydrodynamic force which implies $\mathbf{f}_b = 0$. Two geometries both of which are 2D are considered to investigate the speed-up factors as a function of the number of subdomains. The first one is a microfluidic channel for DLD separation with inclusions but without any repeating structure which can be seen in Fig. 3 (top row). Strictly speaking, this geometry has also a repeating structure, however we call this geometry as non-periodic due to the fact that when the domain is divided into a number of subdomains, the subdomains will not be the same. The second one is a microchannel with staggered and periodic inclusions. The major difference between two geometries is the layout of the non-moving inclusions. Inclusions inside DLD have non-regular topology, and hence not easy to partition to symmetric subdomains; whereas the staggered one has inclusions with regular topology, and therefore, easy to partition to various symmetric subdomain configurations. A uniform inlet velocity of 250 μm/s and zero pressure boundary conditions are assigned at the inlet and outlet of the microchannel, respectively. No-slip boundary conditions are assigned for the channel walls and non-moving inclusions (*i.e.* circular posts). 3 μm and 6 μm particles (in diameter) are released from the same initial position close to the lower channel wall. As seen in the figure, with the appropriate geometric configuration, two particles have clear separation distance between each other since the hydrodynamic interaction of the particles with the inclusions differs

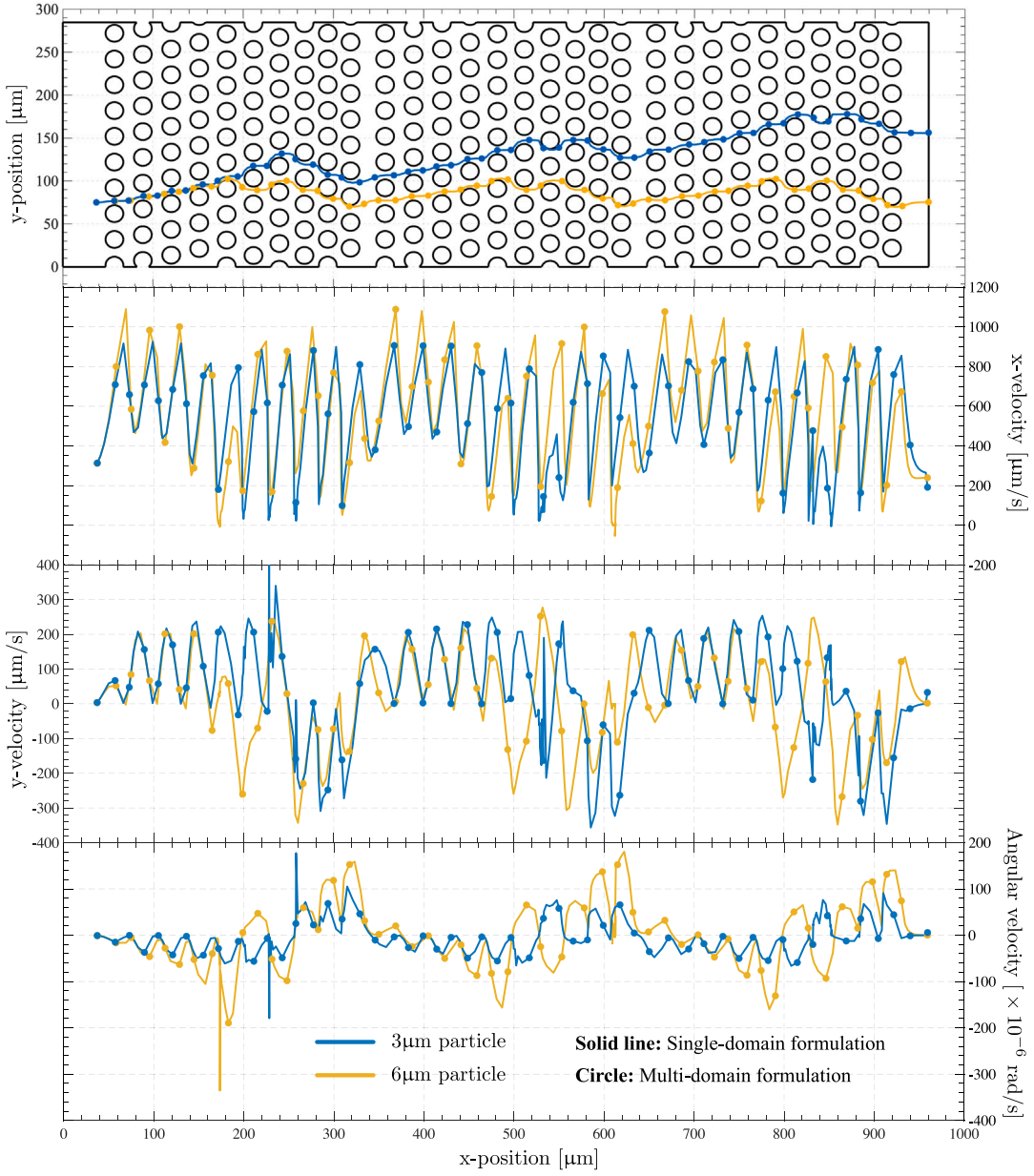

Fig. 3. Comparison of MDBEM formulation with single domain formulation.

depending on the size of the particle. Our computational domain has about one millimeter length which is smaller than the length of typical DLD chips [38,39]. One can expect that the distance between the two particles would definitely amplify with the increasing length of the microchannel. The trajectories of the both particles are presented in the figure. The trajectories are quite close to the inclusions where strong hydrodynamic interactions occur. The computational model can capture these interactions quiet effectively and as a result separation by size is realized as observed in the literature [13,38,39]. The x- and y-velocities as well as the angular velocities of the both particles are illustrated in Fig. 3. The comparison of the single-domain and multi-domain formulations in terms of the particle trajectory and particle velocities are also given in Fig. 3, and quite a well agreement is observed between the two formulations which verifies our multi-domain formulation.

In Fig. 4A, the speed-up factors for different number of subdomains are presented for the DLD chip with non-periodic inclusions. Two test runs were conducted with two particles in order to compare the effect of number of elements that were used to discretize the solution domain (3560 and 30 220 elements for the same geometry). For both test runs the problem is divided into several subdomains ranging between two and twenty-two. The size of the ghost domains are at least $20d$ long. Depending on the size of the ghost domain, the number of elements on the ghost domain varies but in the simulations it is between 25 and 40 for the case of 3560 channel elements, and between 250 and 300 for the case of 30 220 channel elements. As depicted in the figure, higher the number of elements, higher the speed-up. Speed-up factors as high as 30 are achieved with the proposed multi-domain formulation. In order to generate BEM matrices, one has to solve an N-body problem (one load point interacts with all elements). Multi-domain formulation has the same main feature of the N-body problem, so the computational time increases with increasing N. Multi-domain formulation becomes more efficient as the number of elements to discretize the single domain increases. This has mainly two reasons: First, the construction of the BEM matrices of slender geometries includes unnecessary calculations for the far-field interactions. As the distance between the load point and element grows, the interaction level decreases with $\mathcal{O}(1/r)$, therefore

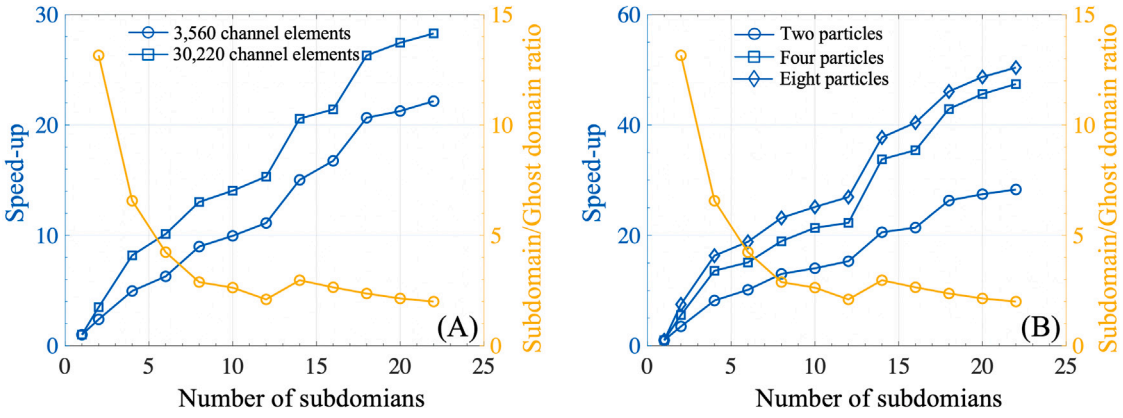

Fig. 4. Speed-up as a function of number of subdomains for a non-periodic geometry.

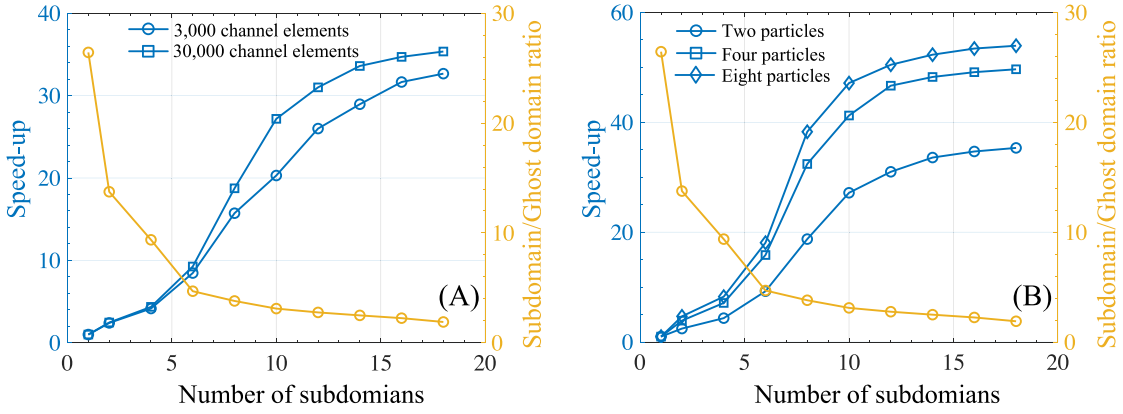

Fig. 5. Speed-up as a function of number of subdomains for a periodic geometry.

the effect of far-field elements diminishes. Actually this is the underlying idea of fast-multipole methods for BEM formulations [63]. As the number of elements becomes larger, the number of unnecessary calculations tremendously increases due to N-body problem nature of BEM. In addition, for a particulate flow problems (*i.e.* moving boundary problems), this N-body problem needs to be solved again and again through the course of motion which leads to heavy computational load. Such a drawback cannot be avoided with a single domain formulation for slender body moving boundary problems. Second, keeping in mind the BEM matrices are fully populated, as the size of BEM matrices grows condition numbers increases which may lead to ill-conditioned system that requires special care for the solution of the system.

In Fig. 4A, one can observe two subsequent first-order system like convergence (from single domain to 13 subdomains, and 13 subdomains to 22) which is inherent to multi-domain formulations. The speed-up factors are saturated at a certain number of subdomains since each and every subdomain introduces an additional interface to the solution domain which results in extra elements. In 2D, a single domain with N-elements leads to $2N\times2N$ BEM matrices (this would be $3N\times3N$ in 3D). On the other hand, a domain with (n) subdomains results in $N/n+(n-1)\varepsilon$ elements with $[2N/n+2(n-1)\varepsilon]\times[2N/n+2(n-1)\varepsilon]$ BEM matrices where ε is the number of elements on an interface which is considerable smaller than N for slender bodies. As (n) grows so does $[(n-1)\varepsilon]$, and the size of the matrices becomes comparable with (N/n) and the improvement stops at a certain point. As an extension to conventional multi-domain formulations, we have also introduced a ghost domain that encapsulates the subdomain boundaries inside the solution domain to ensure seamless transition of particles from one subdomain to another. The presence of the ghost domain also necessitates extra computation time (note that the velocity on ghost domain boundaries are computed during the post-process and this calculation is out of discussion at this point). Therefore, the key factor here is the ratio of the ghost domain and subdomain ratio which severely determines the behavior of the convergence curves (note the subdomain/ghost domain ratio is also given on the second axis). Inherently, the size of a ghost domain needs to be smaller than the size of a subdomain. We keep the ghost domain size about the same as we further subdivide the solution domain until 13 subdomains. Then, the size of ghost domains is halved, and the size of the subdomains further reduced. At this point, the ratio of the size of the subdomains to ghost domains experience an increase, however the convergence behavior re-initiates after this point on.

In Fig. 4B, the effect of the number of particles is represented. Speed-factor is the indication of the performance of the multi-domain formulation against the single domain formulation. To be able to compare the computational performance of different cases, CPU time for each case is also summarized in Table 1. As the number of particles increases, CPU time also rises due to the increase in the size of the system matrices. However, as seen in the table, when the number particles increases from two to four, the CPU time also almost doubles which is actually quite unexpected since the majority of the elements on the non-moving boundaries, and the elements of the moving boundaries (*i.e.* particles) do not contribute to the overall size of the system matrices. The reason for this behavior is the formulation we implemented for the tracking of particles—the details of which is given in [35]. In this formulation, the partition of the system matrices which corresponds to non-moving boundaries are computed once and then stored for further calculations. In this way, the number of particles is the determining parameter rather than the number of channel elements for the CPU time. The cases with four and eight particles have similar behavior to that of the case with two particles, but the speed-up factors are better, and reaches about 50 which is a significant improvement achieved by just

Table 1
CPU time and speed-up values for non-periodic geometry.

# of channel elements	Two particles		Two particles		Four particles		Eight particles	
	3560		**30,220**		**30,220**		**30,220**	
# of subdomain	CPU time [s]	Speed-up	CPU time [s]	Speed-up	CPU time [s]	Speed-up	CPU time [s]	Speed-up
1	7828	1.00	88 474	1.00	156 652	1.00	221 155	1.00
2	3275	2.39	25 309	3.49	28 138	5.56	29 466	7.50
4	1578	4.96	10 799	8.19	11 530	13.6	13 542	16.3
6	1245	6.28	8732	10.1	10 371	15.1	11 736	18.8
8	871.1	8.98	6796	13.0	8264	19.0	9548	23.2
10	785.9	9.95	6301	14.0	7328	21.4	8799	25.1
12	703.6	11.1	5776	15.3	7035	22.3	8211	26.9
14	521.0	15.0	4299	20.6	4637	33.8	5859	37.7
16	467.3	16.8	4131	21.4	4422	35.4	5471	40.4
18	378.9	20.7	3363	26.3	3650	42.9	4800	46.1
20	368.1	21.3	3222	27.5	3434	45.6	4540	48.7
22	353.1	22.2	3126	28.3	3302	47.4	4388	50.4

Table 2
CPU time and speed-up values for a periodic geometry.

# of channel elements	Two particles		Two particles		Four particles		Eight particles	
	3000		**30,000**		**30,000**		**30,000**	
# of subdomain	CPU time [s]	Speed-up	CPU time [s]	Speed-up	CPU time [s]	Speed-up	CPU time [s]	Speed-up
1	5096	1.00	19 064	1.00	32 452	1.0	42 541	1.00
2	2116	2.40	7750	2.46	8308	3.90	9075	4.68
4	1227	4.15	4372	4.36	4553	7.12	5143	8.27
6	602.2	8.46	2059	9.26	2047	15.9	2351	18.1
8	324.4	15.7	1017	18.8	1001	32.4	1111	38.3
10	251.4	20.3	701.3	27.2	786.7	41.3	902.2	47.1
12	196.2	26.0	614.9	31.0	695.2	46.7	842.5	50.4
14	176.2	28.9	567.4	33.6	672.3	48.3	813.0	52.3
16	161.6	31.7	549.3	34.7	660.4	49.1	796.0	53.4
18	156.1	32.7	539.2	35.4	653.4	49.7	788.4	53.4

Table A.1
Comparison between present study and results from literature for a stationary cylinder in a Poiseuille flow ($Re = 2 \times 10^{-4}$)

k(= **a**/**b**)	Present Study (PS)	Results from [62]	Faxen results from [62] (FR)	Rel. % error between PS and FR	Rel. % error between [62] and FR
0.0100	3.4012	3.5393	3.4057	0.13	3.77
0.0250	4.5218	4.7145	4.5282	0.14	3.96
0.0500	6.0123	5.9999	6.0214	0.15	0.36
0.1000	8.8907	9.1630	8.9060	0.17	2.80
0.2000	16.1724	16.3585	16.2072	0.21	0.92
0.4000	48.4175	48.8511	48.6229	0.42	0.47
0.8000	1231.8493	1256.1650	–	–	–
0.9000	7681.9006	7361.3287	–	–	–
0.9500	44,488.9807	44,093.1632	–	–	–

modifying formulation without any manipulation on the computation side.

In microfluidic applications, repeating sections exists in many cases, therefore the performance of the proposed multi-domain formulation is also evaluated for a periodic geometry. Fig. 5 illustrates the performance of the multi-domain formulation with a periodic geometry, and the corresponding CPU time for each case is also tabulated in Table 2. Since the computational domain has repeated subdomains, the BEM matrices are computed only once for the subdomains without any particles. As a result, CPU time significantly decreases compared to the case with non-periodic geometry. This trend also holds as the number of particles increases. As the number of particles increases from two to four, this time CPU time is not doubled this is due to fact that the overhead of the calculation of the partition of the system matrices corresponding to the non-moving boundaries.

Another important observations is that the decrease in the ratio of the size of subdomains to ghost domains is gradual, and so does the increase in the speed-up. This is due to the fact that the size of the subdomains is less restricted and an interface can be located between any two repeating structures regarding the periodic structure of the solution domain. However, owing to the fact that the limitations on the selection of the size of the subdomains and the ghost domains, the ratio reaches a limit where the speed-up values also saturate. Like the case with non-periodic geometry, the speed-up factors of 50 is achieved with a significant reduction in CPU time.

4. Conclusions

A multi-domain boundary element formulation is developed for particulate flow in microchannels. Extension to the conventional MDBEM, ghost domains encapsulating the interfaces between the subdomains are introduced in the formulation to treat the particles crossing the interfaces between the subdomains. The formulation is demonstrated in 2D without loss of generality. Although the formulation presented in this work only includes hydrodynamic interactions, it may easily be extended for multi-physics problems for wider range of microfluidic applications. Our group recently implemented BEM formulation for AC/DC electrokinetic particle manipulation [14] and acoustophoretic bio-particle manipulation [64]. The current multi-domain formulation can be also extended to handle multi-physic problems for microfluidic applications.

Just the results for 2D geometry are demonstrated in this study, however the performance of the multi-domain formulation in 3D would be similar if not better. Especially the results for the cases with periodic

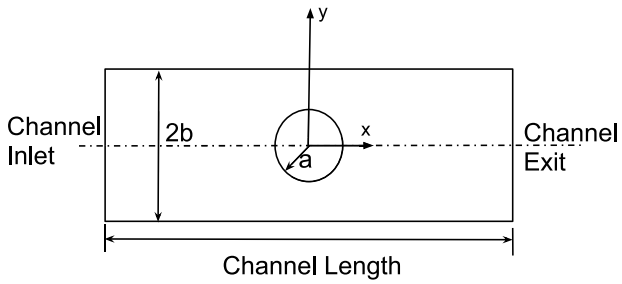

Fig. A.1. Schematic drawing of the benchmark problem.

geometry is valuable to give an idea about the 3D cases. The benefit of the formulation is emphasized for higher number of elements. Although the cases with many inclusions are considered in 2D, in 3D even for a straight microchannel would have similar number of elements (if not well above), and hence a straight microfluidic channel can be treated as a periodic geometry. Since the number of elements introduced with each interface would be relatively small compared to the total number of elements in the solution domain in 3D, one can expect that the performance of the proposed formulation would be even better for 3D geometries with a comparable number of degree of freedom. The computation is further simplified for the solution domains composed of periodic structures, and as illustrated in Fig. 5, our multi-domain formulation exhibits a very promising performance for the domains with periodic structures. Therefore, we believe our multi-domain formulation together with the particle tracking formulation will be very beneficial for the microfluidics community. Moreover, the results of our study revealed that speed-up values above 50 is achieved with the current formulation. Keeping in mind the calculation of elements of the BEM matrices is inherently parallel process, the computational performance of the formulation can be further enhanced with parallel programing [35] which will be one of our future research directions.

CRediT authorship contribution statement

Alper Topuz: Conceptualization, Software, Validation, Writing. **Besim Baranoğlu:** Conceptualization, Writing. **Barbaros Çetin:** Supervision, Conceptualization, Visualization, Formal analysis, Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

B.Ç. would like to acknowledge fundings from the Turkish Academy of Sciences through Outstanding Young Scientist Program (TÜBA-GEBİP) and The Science Academy, Turkey Distinguished Young Scientist Award (BAGEP).

Appendix. Validation of the BE formulation

A benchmark problem from the literature [62] is employed to validate the BE formulation. The drag force generated on a stationary particle placed in a Poiseuille flow as shown in Fig. A.1 is considered. The non-dimensional drag force is calculated for varying k–parameter which is the ratio of the particle radius to channel height. The results given in [62] are based on finite element (FE) formulation. For comparison, the results are also compared against the Faxén formula, which is an analytical solution for the same problem and valid for $k \leq 0.5$. The results are summarized in Table A.1. The relative error of the current formulation is less than 1% against the Faxén formula. Note that the relative error of the results in [62] is higher than that of the present study. The relative error is less than 5% between the BE and FE formulations.

References

[1] Gaul L, Kögl M, Wagner M. Boundary element methods for engineers and scientists: An introductory course with advanced topics. Springer Science & Business Media; 2013.
[2] Brebbia CA, Telles JCF, Wrobel LC. Boundary element techniques: Theory and applications in engineering. Springer Science & Business Media; 2012.
[3] Youngreen GK, Acrivos A. Stokes flow past a particle of arbitrary shape: a numerical method of solutions. J Fluid Mech 1975;69:377–403.
[4] Youngreen GK, Acrivos A. On the shape of a gas bubble in a viscous extensional flow. J Fluid Mech 1976;76:433–42.
[5] Rallison JM, Acrivos A. A numerical study of the deformation and burst of a viscous drop in an extensional flow. J Fluid Mech 1978;89:191–200.
[6] Tran-Cong T, Phan-Thien N, Graham AL. Stokes problems of multiparticle systems: Periodic arrays. Phys Fluids A 1990;2(5):666–73.
[7] Phan-Thien N, Tran-Cong T, Graham AL. Shear flow of periodic arrays of particle clusters: a boundary-element method. J Fluid Mech 1991;228:275–93.
[8] Mondy LA, Ingber MS, Dingman SE. Boundary element method simulations of a ball falling through quiescent suspensions. J Rheol 1991;35(5):825–48.
[9] Cetin B, Bayer O. Evaluation of Nusselt number for a flow in a microtube using second-order slip model. Therm Sci 2011;15 Suppl. 1:103–9.
[10] Colin S. Gas microflows in the slip flow regime: A critical review on convective heat transfer. J Heat Transfer 2012;134(2):020908.
[11] Cetin B. Effect of thermal creep on heat transfer for a two-dimensional microchannel flow: An analytical approach. J Heat Transfer 2013;135(10):101007.
[12] Cetin B, Travis BE, Li D. Analysis of the electro-viscous effects on pressure-driven liquid flow in a two-section heterogeneous microchannel. Electrochim Acta 2008;54(2):660–4.
[13] Cetin B, Ozer MB, Solmaz ME. Microfluidic bio-particle manipulation for biotechnology. Biochem Eng J 2014;92:63–82.
[14] Cetin B, Oner SD, Baranoglu B. Modeling of dielectrophoretic particle motion: Point particle vs finite-sized particle. Electrophoresis 2017;38:1407–18.
[15] Buyukkocak S, Ozer MB, Cetin B. Numerical modeling of ultrasonic particle manipulation for microfluidic applications. Microfluid Nanofluid 2014;17(6):1025–37.
[16] Cetin B, Ozer MB, Cagatay E, Buyukkocak S. An integrated acoustic and dielectrophoretic particle manipulation in a microfluidic device for particle wash and separation fabricated by mechanical machining. Biomicrofluidics 2016;10(014112).
[17] Şahin MA, Cetin B, Özer MB. Investigation of effect of design and operating parameters on acoustophoretic particle separation via 3D device-level simulations. Microfluid Nanofluid 2020;24(1):8.
[18] Zeinali S, Cetin B, Oliaei S, Karpat Y. Fabrication of continuous flow microfluidic device with 3D electrode structures for high throughput DEP applications using mechanical machining. Electrophoresis 2015;36:1432–42.
[19] Rasooli R, Cetin B. Assessment of Lagrangian modeling of particle motion in a spiral microchannel for inertial microfluidics. Micromachines 2018;9(9):433.
[20] Kang KH, Xuan X, Kang Y, Li D. Effects of DC-dielectrophoretic force on particle trajectories in microchannels. J Appl Phys 2006;99(064702):1–8.
[21] Cetin B, Li D. Continuous particle separation based on electrical properties using AC-dielectrophoresis. Electrophoresis 2009;30:3124–33.
[22] Cetin B, Li D. Lab-on-a-chip device for continuous particle and cell separation based on electrical properties via AC-dielectrophoresis. Electrophoresis 2010;31:3035–43.
[23] Tran-Cong T, Phan-Thien N. Stokes problems of multiparticle systems: A numerical method for arbitrary flows. Phys Fluids A 1989;1(3):453—-61.
[24] Ingber MS. Numerical simulation of the hydrodynamic interaction between a sedimenting particle and a neutrally buoyant particle. Int J Numer Methods Fluids 1989;9(3):263–73.
[25] Ingber MS. Dynamic simulation of the hydrodynamic interaction among immersed particles in Stokes flow. Internat J Numer Methods Fluids 1990;10(7):791–809.
[26] Dingman SE, Ingber MS, Mondy LA, Abbott JR, Brenner H. Particle tracking in three-dimensional Stokes flow. J Rheol 1992;36(3):413–40.
[27] Power H, Miranda G. SIAM J Appl Math 1987;47(4):689–98.
[28] Karrila SJ, Kim S. Integral equations of the second kind for stokes flow: DIRECT solution for physical variables and removal of inherent accuracy limitations. Chem Eng Commun 1989;82(1):123–61.
[29] Xijun F, Yeow YL. A boundary integral equation method for the Stokes problem of multiparticle systems. Phys Fluids A 1992;4(5):1074–6.
[30] Power H, de Power BF. Second-kind integral equation formulation for the slow motion of a particle of arbitrary shape near a plane wall in a viscous fluid. SIAM J Appl Math 1993;53(1):60–70.

[31] Power H. A second kind integral equation formulation for the low Reynolds number interaction between a solid particle and a viscous drop. J Eng Math 1996;30(1):225–37.
[32] AA Mammoli, MS Ingber. Parallel multipole BEM simulation of two-dimensional suspension flows. Eng Anal Bound Elem 2000;24(1):65–73.
[33] Power H, Gomez JE. The completed second kind integral equation formulation for Stokes flow with mixed boundary conditions. Commun Numer Methods Eng 2001;17(4):215–27.
[34] Chan CY, Beris AN, Advani Suresh G. Second-order boundary element method calculations of hydrodynamic interactions between particles in close proximity. Internat J Numer Methods Fluids 1992;14(9):1063–86.
[35] Karakaya Z, Baranoglu B, Cetin B, Yazici A. A parallel boundary element formulation for tracking multiple particle trajectories in Stoke's flow for microfluidic applications. CMES-Comput Model Eng Sci 2015;104:227–49.
[36] House DL, Luo H. Electrophoretic mobility of a colloidal cylinder between two parallel walls. Eng Anal Bound Elem 2010;34(5):471–6.
[37] House DL, Luo H. Effect of direct current dielectrophoresis onthe trajectory of a non-conducting colloidalsphere in a bent pore. Electrophoresis 2011;32:3277–85.
[38] Huang LR, Cox EC, Austin RH, Sturm JC. Continuous particle separation through deterministic lateral displacement. Science 2004;304(5673):987–90.
[39] Li N, Kamei DT, Ho C-M. On-chip continuous blood cell subtype separation by deterministic lateral displacement. In: 2007 2nd IEEE international conference on nano/micro engineered and molecular systems. IEEE; 2007, p. 932–6.
[40] Divo E, Kassab AJ, Rodriguez F. Parallel domain decomposition approach for large-scale three-dimensional boundary-element models in linear and nonlinear heat conduction. Numer Heat Transfer B 2003;44(5):417–37.
[41] Kamiya N, Iwase H, Kita E. Parallel implementation of boundary element method with domain decomposition. Eng Anal Bound Elem 1996;18(3):209–16.
[42] Kane JH, Kumar BL Kashava, Saigal S. An arbitrary condensing, noncondensing solution strategy for large scale, multi-zone boundary element analysis. Comput Methods Appl Mech Engrg 1990;79(2):219–44.
[43] Gao X-W, Guo L, Zhang C. Three-step multi-domain BEM solver for nonhomogeneous material problems. Eng Anal Bound Elem 2007;31(12):965–73.
[44] Frechette J, Drazer G. Directional locking and deterministic separation in periodic arrays. J Fluid Mech 2009;627:379.
[45] Ramšak M, Škerget L. A highly efficient multidomain BEM for multimillion subdomains. Eng Anal Bound Elem 2014;43:76–85.
[46] Wang B, Feng Y, Pieraccini S, Scialò S, Fidelibus C. Iterative coupling algorithms for large multidomain problems with the boundary element method. Internat J Numer Methods Engrg 2019;117(1):1–14.
[47] Liu G, Thompson KE. A domain decomposition method for modelling Stokes flow in porous materials. Internat J Numer Methods Fluids 2002;38(11):1009–25.
[48] Bush MB. Stratified Newtonian flow calculations by the boundary element method. Comput Mech 1991;7(3):195–204.
[49] Gao X-W, He M-C. A new inverse analysis approach for multi-region heat conduction BEM using complex-variable-differentiation method. Eng Anal Bound Elem 2005;29(8):788–95.
[50] Ramšak M, Škerget L. 3D multidomain BEM for solving the Laplace equation. Eng Anal Bound Elem 2007;31(6):528–38.
[51] Bui TT, Popov V. Domain decomposition boundary element method with overlapping sub-domains. Eng Anal Bound Elem 2009;33(4):456–66.
[52] Ingber MS, Tanski JA, Alsing P. A domain decomposition tool for boundary element methods. Eng Anal Bound Elem 2007;31(11):890–6.
[53] Erhart K, Divo E, Kassab AJ. A parallel domain decomposition boundary element method approach for the solution of large-scale transient heat conduction problems. Eng Anal Bound Elem 2006;30(7):553–63.
[54] Iljaž J, Wrobel LC, Hriberšek M, Marn J. Subdomain BEM formulations for the solution of bio-heat problems in biological tissue with melanoma lesions. Eng Anal Bound Elem 2017;83:25–42.
[55] Ahmad S, Banerjee PK. Multi-domain BEM for two-dimensional problems of elastodynamics. Internat J Numer Methods Engrg 1988;26(4):891–911.
[56] Lu X, Wu W-L. A new subregion boundary element technique based on the domain decomposition method. Eng Anal Bound Elem 2005;29(10):944–52.
[57] Layton JB, Ganguly S, Balakrishna C, Kane JH. A symmetric Galerkin multi-zone boundary element formulation. Internat J Numer Methods Engrg 1997;40(16):2913–31.
[58] Larson RE, Higdon Jonathan JL. Microscopic flow near the surface of two-dimensional porous media. Part 1. Axial flow. J Fluid Mech 1986;166:449–72.
[59] Larson RE, Higdon Jonathan JL. Microscopic flow near the surface of two-dimensional porous media. Part 2. Transverse flow. J Fluid Mech 1987;178:119–36.
[60] Ravnik J, Škerget L, Žunič Z. Combined single domain and subdomain BEM for 3D laminar viscous flow. Eng Anal Bound Elem 2009;33(3):420–4.
[61] Ravnik J, Škerget L, Žunič Z. Velocity–vorticity formulation for 3D natural convection in an inclined enclosure by BEM. Int J Heat Mass Transfer 2008;51(17–18):4517–27.
[62] Richou AB, Ambaria A, Lebeyc M, Nacirid JK. Drag force on a circular cylinder midway between two parallel plates at very low Reynolds numbers–Part 1: Poiseuille flow (numerical). Chem Eng Sci 2004;59(15):3215–22.
[63] Liu Y. Fast multipole boundary element method: Theory and applications in engineering. Cambridge University Press; 2009.
[64] Açıkgöz HN. Numerical investigation of living cells under ultrasonic excitation in microfluidic devices (Master's thesis), Turkey: Middle East Technical University; 2020.