10-21-2005

Least-squares finite-element scheme for the lattice Boltzmann method on an unstructured mesh

Yusong Li
University of Nebraska - Lincoln, yli7@unl.edu

Eugene J. LeBoeuf
Vanderbilt University, Nashville, TN

P. K. Basu
Vanderbilt University, Nashville, TN

Least-squares finite-element scheme for the lattice Boltzmann method on an unstructured mesh

Yusong Li, Eugene J. LeBoeuf,* and P. K. Basu
Department of Civil and Environmental Engineering, Vanderbilt University, Nashville, TN 37325, USA
(Received 17 January 2005; published 21 October 2005)

A numerical model of the lattice Boltzmann method (LBM) utilizing least-squares finite-element method in space and the Crank–Nicolson method in time is developed. This method is able to solve fluid flow in domains that contain complex or irregular geometric boundaries by using the flexibility and numerical stability of a finite-element method, while employing accurate least-squares optimization. Fourth-order accuracy in space and second-order accuracy in time are derived for a pure advection equation on a uniform mesh; while high stability is implied from a von Neumann linearized stability analysis. Implemented on unstructured mesh through an innovative element-by-element approach, the proposed method requires fewer grid points and less memory compared to traditional LBM. Accurate numerical results are presented through two-dimensional incompressible Poiseuille flow, Couette flow, and flow past a circular cylinder. Finally, the proposed method is applied to estimate the permeability of a randomly generated porous media, which further demonstrates its inherent geometric flexibility.

DOI: 10.1103/PhysRevE.72.046711 PACS number(s): 47.11.+j

I. INTRODUCTION

In the last decade, the lattice Boltzmann method (LBM) has been developed as an effective tool to simulate complex fluid flow problems [1,2]. Historically, LBM originated from lattice gas automata (LGA), which views fluids as arrays of particles residing on a discrete lattice, evolving with specific interactive propagation and collision rules. Improvements to the LBM relative to the LGA include extending single particle occupation variables to particle distribution functions [3], developing of a linearly stable collision operator [4,5], and utilizing a single time relaxation approximation [6], which provides LBM an improved capability to eliminate statistical noise and enhanced computational efficiency. LBM, similar to LGA, however, is restricted to uniform lattice structures, which severely limits its potential application to many practical problems, e.g., flow in porous media, where representations of complex pore geometry require a very fine uniform lattice, thus necessitating additional computing resources [7]. More recently, it was determined that, although the coupling between discretization of velocity spaces and physical spaces is an essential part of LGA dynamics, it is not critical for the LBM [8]. It is in this light that many efforts were forwarded to improve the LBM such that it is able to more flexibly apply to nonuniform grids. Those improvements can be classified as: (i) interpolation techniques; (ii) grid refinement techniques; and (iii) numerical lattice Boltzmann methods.

Interpolation techniques, first proposed by He, Luo, and Dembo [9], extend LBM to nonuniform rectangular meshes by interpolating the density distribution at the grid sites from the square lattices. An extension of this technique is a Taylor-series expansion and least-squares-based LBM proposed by Shu, Niu, and Chew [10]. Instead of direct interpolation, a Taylor series expansion is implemented to estimate the density function at the grid sites, and a least-squares scheme is implemented to minimize errors. Although this approach removes the rectangular shape restriction and possesses a meshless feature, collisions still take place on the grid points. More recently, a local time step technique [11] was applied to this interpolation supplemented LBM, which greatly reduces CPU time required to obtain steady-state solutions.

Grid refinement techniques refine the lattice locally where more precision is required or the geometry is more complex, passing the data between fine and coarse lattices via a particular algorithm. Filippova and Hänel [12] coupled the LBM with a local second-order hierarchical grid refinement and boundary fitting scheme. The approach not only possesses an improved ability to treat curved boundaries, but also provides higher computational accuracy, especially in thin boundary layers where solutions possess highly anisotropic features. Utilizing a multigrid architecture, Lin and Lai [13] proposed a composite block-structured LBM, which allows one-way interaction at the post-streaming stage without rescaling the discrete distribution function. Pointing out that Lin's algorithm is inaccurate, and that Filippova's approach presents singularity for $\tau=1$, Dupuis and Chopard [14] proposed an alternative grid refinement algorithm, which can accelerate the flow settlement process a thousand times faster than a single grid resolution. Grid refinement techniques present a promising direction for the development of LBM; however, its limited application to regular rectangular grid structures restricts the flexibility of these methods.

Numerical lattice Boltzmann methods combine LBM with traditional numerical methods such as finite difference (FD), finite volume (FV), and finite element (FE) methods to increase computational efficiency and accuracy, while adapting LBM to irregular mesh. Based on Runge–Kutta time discretization and various spatial discretization schemes, Chen and co-workers [15,16] combined FD and LBM in a number of ways. The central difference scheme was first proposed by Cao *et al.* [16] in Cartesian coordinates, and was later extended to curvilinear coordinates with nonuniform grids [17]. Nannelli and Succi [18] proposed the first finite volume formulation of LBM. Later, Amati, Succi, and Benzi [19] pre-

*All correspondence should be addressed to: Eugene J. LeBoeuf, FAX: (615) 322-3365; Email: eugene.j.leboeuf@vanderbilt.edu

sented a finite volume formulation of the LBM, where a piece-wise linear interpolation scheme was used to estimate the volume-averaged particle distribution in a nonuniform coarse lattice. Another volumetric formulation of LBM was developed by Chen [15], which can be applied to arbitrary mesh while achieving exact adherence to conservation laws and equilibrium conditions. Peng and co-workers [20–22] proposed additional versions of the finite volume LBM (FV–LBM) using both triangular and rectangular elements, which appears to be flexible for both internal and external boundaries. More recently, this method was further developed from both theoretical and practical aspects by Ubertini and co-workers [23,24], who demonstrated that the method does not present significant numerical viscosity effects (at the second order) in the mesh size. As an early effort to combine FE methods with LBM, Lee and Lin [25,26] presented a characteristic Galerkin discrete Boltzmann equation (CGDBE), which implements a Taylor–Galerkin procedure.

Traditionally, FE methods [27] have allowed simulation of more complex, and hence, realistic geometries relative to FD and FV methods. In standard computational fluid dynamics (CFD), FV methods, however, are more widespread. A significant reason lies in the nature of the convection operators of fluid flow, which are first order, and thus non-self-adjoint. For equations with non-self-adjoint operators, the classical Galerkin method is often corrupted by spurious oscillations or wiggles [28]. Least-squares finite element (LSFE) method, on the other hand, was recently shown to be a robust and efficient way to solve non-self-adjoint equations. It always leads to symmetric, positive definite, linear systems of equations without using techniques such as upwinding, staggered grids and operator splitting techniques [29]. Compared with Taylor–Galerkin-based FE methods, LSFE method possesses improved stability. Furthermore, for more complex systems, Taylor–Galerkin-based FE methods may promote oscillations at discontinuities [28] or at solid-liquid interfaces with boundary layers or high velocity gradients. Those oscillations may be suppressed by adding dissipation terms like those in "upwind" and "artificial viscosity" schemes, which, however, are dependent on the specific parameters of the problem. For non-self-adjoint systems, such as the lattice Boltzmann equation, it is thus reasonable to apply LSFE, which represents a promising approach to extend LBM to more practical and complex domains while simultaneously benefiting from finite element methods' superior stability and flexibility. It is in this light that we implement a new FE–LBM, which utilizes LSFE in space and a Crank–Nicolson scheme in time.

As a follow-up to a Rapid Communication [30], this paper details the derivation and example applications of LSFE–LBM. Subsequent to this Introduction, a numerical formulation section, including a numerical derivation of the LSFE–LBM and a discussion of important implementation issues, is presented, followed by a thorough theoretical analysis of the accuracy and stability of the method. The implementation of LSFE–LBM is exemplified through two-dimensional incompressible Poiseuille flow, Couette flow, flow past a circular cylinder, and flow in porous media. The paper concludes by summarizing the advantages of the LSFE–LBM, and a discussion of its future potential.

II. NUMERICAL FORMULATIONS

A. Numerical derivation

The starting point of LSFE–LBM is the discrete lattice Boltzmann equation

$$\frac{\partial f_i}{\partial t} + \vec{c}_i \cdot \vec{\nabla} f_i = \Omega_i \quad (i = 1,2,\ldots,N) \tag{1}$$

where f_i represents the particle velocity distribution function, c_i is the velocity along the ith direction, N is the number of different velocities in the model, and Ω_i denotes the collision operator which is commonly approximated by the Bhatnagar–Gross–Krook model [31]

$$\Omega_i = -\frac{1}{\tau}(f_i - f_i^{eq}) \tag{2}$$

where τ is the relaxation time, f_i^{eq} is the local equilibrium given by

$$f_i^{eq} = \rho\omega_i\left(1 + \frac{\vec{u}\cdot\vec{c}_i}{c_s^2} + \frac{(\vec{u}\cdot\vec{c}_i)^2 - c_s^2\vec{u}^2}{2c_s^4}\right) \tag{3}$$

in which ω_i is the weighting parameter for each velocity direction. The nodal density ρ and the macroscopic velocity $\vec{u}$ is defined by

$$\rho = \sum_i f_i, \quad \rho\vec{u} = \sum_i f_i c_i. \tag{4}$$

Nine possible directional velocities are used in this study, where ω_i in Eq. (3) equals $\frac{4}{9}$ for $i=0$, $\frac{1}{9}$ for $i=1,2,3,4$, and $\frac{1}{36}$ for $i=5,6,7,8$. The nine velocities are defined as

$$\vec{c}_i = \begin{cases} (0,0) & (i=0) \\ c_s\left(\cos\left[(i-1)\frac{\pi}{2}\right], \sin\left[(i-1)\frac{\pi}{2}\right]\right) & (1 \leqslant i \leqslant 4) \\ c_s\sqrt{2}\left(\cos\left[(i-5)\frac{\pi}{2}+\frac{\pi}{4}\right], \sin\left[(i-5)\frac{\pi}{2}+\frac{\pi}{4}\right]\right) & (5 \leqslant i \leqslant 8). \end{cases} \tag{5}$$

Construction of LSFE–LBM first considers application of the θ-method to treat time-space approximations. Setting the time step $\Delta t = t^{n+1} - t^n$, and given f_i^n for the previous time step, the solution f_i^{n+1} for the current time step is determined from

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} + \theta \vec{c} \cdot \nabla f_i^{n+1} + (1-\theta)\vec{c} \cdot \nabla f_i^n = \theta \Omega_i^{n+1} + (1-\theta)\Omega_i^n. \tag{6}$$

In this work, $\theta = \frac{1}{2}$ is implemented, which corresponds to the Crank–Nicolson scheme, providing for second-order accuracy in time. Under this condition, a standard form of LSFE can be obtained by rearranging Eq. (6):

$$c_x \frac{\partial f_i^{n+1}}{\partial x} + c_y \frac{\partial f_i^{n+1}}{\partial y} + A f_i^{n+1} = p_i,$$

$$A = \frac{2}{\Delta t} + \frac{1}{\tau},$$

$$p_i = \left(\frac{2}{\Delta t} - \frac{1}{\tau}\right) f_i^n + \frac{1}{\tau}(f_i^{eq,n+1} + f_i^{eq,n}) - \left(c_x \frac{\partial f_i^{n+1}}{\partial x} + c_y \frac{\partial f_i^{n+1}}{\partial y}\right). \tag{7}$$

For brevity, operator $\boldsymbol{L}$ is used, and Eq. (7) can be written in the following form:

$$L f^{n+1} = p. \tag{8}$$

For finite element analysis, the problem domain can first be subdivided into a set of finite elements, and then approximated by the solution $f_h^{e,n+1}$ in a finite element subspace as:

$$f_h^{e,n+1} = \sum_{j=1}^{\alpha} N_j f_j^{n+1} \tag{9}$$

where N_j denotes the element shape function, α represents the number of nodes in an element, and f_j is the nodal value at the jth node. Introducing Eq. (9) into Eq. (8) for an element, we get

$$E = L f_h^{e,n+1} - p_h^e \tag{10}$$

where E is the residual due to elemental approximation. The LSFE is based on the minimization of the squares of the residual for the subspace

$$\varphi(f^{n+1}) = \int_{\Omega_e} E^2 d\Omega_e = \int_{\Omega_e} (L f_h^{e,n+1} - p_h^e)^2 d\Omega_e, \tag{11}$$

$$\frac{\partial \varphi(f^{n+1})}{\partial f_\alpha^{n+1}} = \int_{\Omega_e} (L N_\alpha)^T (L f_h^{e,n+1} - p_h^e) d\Omega_e = 0, \quad \alpha = 1,2,\ldots,n \tag{12}$$

where Ω_e is the domain of the eth element,and the exponent T denotes the transpose. For each element, the following set of linear algebraic equations can be derived from Eq. (12):

$$K_e F_e^{n+1} = P_e \tag{13}$$

where F_e^{n+1} is the vector of nodal values at the current time step. K_e is the elemental matrix given by

$$K_e = \int_{\Omega_e} Q^T Q d\Omega_e \tag{14}$$

where Q is a $(1 \times \alpha)$ vector defined by:

$$Q = CB + AN = \lfloor c_x \; c_y \rfloor \begin{bmatrix} \frac{\partial N_1}{\partial x} & \cdots & \frac{\partial N_\alpha}{\partial x} \\ \frac{\partial N_1}{\partial y} & \cdots & \frac{\partial N_\alpha}{\partial y} \end{bmatrix} + A \lfloor N_1 \cdots N_\alpha \rfloor. \tag{15}$$

The element vector P_e in Eq. (13) is

$$P_e = \int_{\Omega_e} Q^T p_h^e d\Omega_e \tag{16}$$

As presented in Eq. (7), p_h^e is related to the previous time step f_i^n and $f_i^{eq,n}$ values, and the current time step $f_i^{eq,n+1}$ value. An extrapolation is applied to express $f_i^{eq,n+1}$ as proposed by Mei and Shyy [17],

$$f_i^{eq,n+1} = 2 f_i^{eq,n} - f_i^{eq,n-1} \tag{17}$$

where f_i^n, $f_i^{eq,n}$ and $f_i^{eq,n-1}$ can be approximated in the subspace similar to f_i^{n+1}:

$$f_h^n = \sum_{j=1}^{\alpha} N_j f_j^n, \quad f_h^{eq,n} = \sum_{j=1}^{\alpha} N_j f_j^{eq,n}, \quad f_h^{eq,n-1} = \sum_{j=1}^{\alpha} N_j f_j^{eq,n-1} \tag{18}$$

B. Boundary conditions

A typical boundary condition for first-order differentiation equations can be expressed as:

$$f_i = g \quad \text{on } \Gamma \tag{19}$$

where, Γ denotes a homogeneous boundary condition when g equals 0, and a heterogeneous boundary condition when g is not 0.

This essential type of boundary condition is of great convenience to LSFE–LBM. Well-established LBM boundary methods [32–34], e.g., bounce back conditions, constant velocity conditions, and pressure gradient conditions, can be readily applied. At each time step, the boundary values of f_h^{n+1} can be calculated in a similar manner to traditional LBMs, which are then applied to the LSFE–LBM scheme as essential boundary conditions. Meanwhile, macroscopic boundary conditions are imposed through the equilibrium function, f_h^{eq}. For unstructured mesh, special attention should be noted when implementing periodic boundary conditions, where corresponding nodes for inlet and outlet boundary are required. A more detailed discussion on periodic boundary

conditions for the treatment of unstructured mesh is included in Section IV D.

C. Implementation issues

The LSFE method leads to a linear system of equations, as described by Eq. (13), which requires solving at each time step. Since K_e in Eq. (13) is always symmetric and positive definite, a preconditioned conjugate gradient (CG) method can be well applied as a tool for efficient solution. The CG iterative updating formula can be expressed as:

$$F^{n+1} = F^n - M^{-1}(KF^n - P), \tag{20}$$

where, M is a nonsingular preconditioning matrix serving to accelerate convergence of the iteration. In this study, Jacobi preconditioned CG (JPCG) [28] is applied, where the diagonal matrix of K is utilized as the preconditioner matrix M.

As presented in Eq. (20), matrix multiplication, $K*F$, is involved in the JPCG algorithm. Traditionally, a sparse and large global matrix system will require assemblage prior to the multiplication operation, necessitating a large amount of computer memory usage and significant computing time, thus restricting the size of the problem. To overcome this issue, an element-by-element approach [35], which stores information only at the element level, was implemented. Following is a brief description of this element-by-element approach.

A global matrix may be expressed as:

$$K = \sum_{e=1}^{N_e} K_g^e, \tag{21}$$

where, N_e is the number of elements in the system, and K_g^e is a matrix with global size. The components of K_g^e are all zero except those corresponding to the nodes in element e. Meanwhile, the global vector P can be expressed as:

$$P = KF = \left(\sum_{e=1}^{N_e} K_g^e\right)F = \sum_{e=1}^{N_e}(K_g^e F_g^e) = \sum_{e=1}^{N_e} P_g^e, \tag{21a}$$

where F_g^e is a modified global vector, whose components are all zero, except those corresponding to the nodes of element e whose values remain the same as in global vector F. Thus, individual matrix-vector products P_g^e may be obtained by computing an element matrix vector product

$$P^{de} = K^e F^{de}, \tag{22}$$

and expanding the vector P^{de} into appropriate position of P_g^e. In Eq. (22), F^{de} is an element level vector which extracts values from corresponding components of F_g^e. In this way, matrix-vector multiplication can be conducted at the element level, independently and concurrently without storing the global matrix.

Several benefits are observed by employing this element-by-element approach. For a system with 2500 nodes, the memory usage of the element-by-element approach is about 130 times less than a typical assembled global matrix approach, and the computing speed is approximately 4 times faster. These advantages will likely become even more obvious for larger systems, where memory leakage may occur while storing global matrices.

III. ANALYSIS OF LSFE–LBM

A. Accuracy analysis

Since the collision term has no effect on numerical accuracy, for simplicity, a one-dimensional pure advection equation is utilized to analyze the accuracy of LSFE–LBM

$$\frac{\partial f_i}{\partial t} + c_i\frac{\partial f_i}{\partial x} = 0 \quad (i = 1,2,\ldots,N). \tag{23}$$

Applying the θ-method to treat time-space approximations, and implementing LSFE scheme as presented above with uniform linear element, Eq. (23) will lead to a discretized format for a typical node j:

$$\left[1 + \left(\tfrac{1}{6} - \beta^2\theta^2\right)\delta^2\right](F_j^{n+1} - F_j^n) = -\frac{\beta}{2}(F_{j+1}^n - F_{j-1}^n) + \theta\beta^2(F_{j+1}^n - 2F_j^n + F_{j-1}^n), \tag{24}$$

where, δ^2 denotes the second-order variation operator, and $\beta = c\Delta t/\Delta x$.

In order to determine the accuracy of Eq. (24), a Taylor-series expansion of f around time t and the node j is considered:

$$\begin{aligned} f(x \pm \Delta x, t + \Delta t) &= f(x,t) \pm \Delta x f_x(x,t) + \Delta t f_t(x,t) \\ &\quad + \frac{\Delta x^2}{2} f_{xx}(x,t) \pm \Delta x \Delta t f_{xt}(x,t) \\ &\quad + \frac{\Delta t^2}{2} f_{tt}(x,t) \pm \frac{\Delta x^3}{6} f_{xxx}(x,t) + \frac{\Delta x^2 \Delta t}{2} f_{xxt}(x,t) \\ &\quad \pm \frac{\Delta x \Delta t^2}{2} f_{xtt}(x,t) + \frac{\Delta t^3}{6} f_{ttt}(x,t) + \ldots. \end{aligned} \tag{25}$$

Implementing Taylor series expansion on F_j^{n+1}, F_j^n, F_{j+1}^n, F_{j-1}^n, F_{j+1}^{n+1}, and F_{j-1}^{n+1} in Eq. (24), and utilizing the recursive application relationship of the advection equation. [36], i.e.,

$$f_{tt} = c^2 f_{xx}, \quad f_{xxt} = -cf_{xxx}, \quad f_{xtt} = c^2 f_{xxx}, \quad f_{ttt} = -c^3 f_{xxx}, \tag{26}$$

the transient truncation error can be derived as:

$$\varepsilon_t = f_t + cf_x = \Delta t\left(\theta c^2 - \frac{c^2}{2}\right)f_{xx} + \Delta t^2\left(\frac{c^3}{6} - c^3\theta^2\right)f_{xxx} + O(\Delta t^3, \Delta x^4). \tag{27}$$

When $\theta = \frac{1}{2}$, corresponding to the Crank–Nicolson scheme, the transient truncation error is:

$$\varepsilon_t = -\frac{\Delta t^2}{12} f_{xxx} + O(\Delta t^3, \Delta x^4) = O(\Delta t^2, \Delta x^4). \tag{28}$$

Thus, for uniform linear elements, LSFE–LBM enjoys similar accuracy as the CGDBE method presented by Lee and

TABLE I. A comparison of accuracy and stability characteristics of FD–LBM, CGDBE, and LSFE–LBM for the pure advection equation on a uniform mesh.

		FD–LBM	CGDBE	LSFE–LBM
Accuracy	Space	Second order	Fourth order	Fourth order
	Time	Second order	Second order	Second order
Stability[a]		Conditional/ Unconditional	$\Delta t \leqslant \Delta x/3\lvert e\rvert$	Unconditional

[a]Stability of FD-LBM is based upon the specific time discretization scheme used. Δx=element size Δt=time step. e=discrete velocity in the characteristic direction [25].

Lin [25], i.e., fourth-order accuracy in space and second-order accuracy in time. Compared to the second-order accuracy in space for FD-based LBM, it is clear that FE-based LBM increases numerical accuracy. Further, the spatial accuracy is dependent on the order of the shape functions. If higher order shape functions are employed, higher order accuracy will be expected. In this work, linear shape functions are utilized for all test examples. It is important to note that the temporal and spatial accuracy discussed here is for LSFE, not for the incompressible Navier–Stokes equation. When applying LSFE to recover the incompressible Navier–Stokes equation, there exists an additional error in the order of $O(M_a^2)$, where M_a is the Mach number of the flow.

B. Stability analysis

Stability analysis is applied to the pure advection equation in similar fashion to the accuracy analysis. Application of von Neumann stability analysis to the discretized format of LSFE–LBM, i.e., Eq. (23), reveals unconditional stability with any Courant–Friedrichs–Lewy (CFL) number for the pure advection equation on a uniform mesh if θ in Eq. (6) is in the range of $[\frac{1}{2}, 1]$. This unconditional stability, derived from the implicit nature of LSFE, provides a significant advantage over CGDBE, which is only conditionally stable due to its explicit treatment of the advection term. A comparison of accuracy and stability among LSFE–LBM, CGDBE, and FD–LBM is given in Table I.

Stability analysis based upon the pure advection equation simplifies the analysis procedure by neglecting the nonlinear collision term. Although this simplified analysis may represent some stability property of LSFE–LBM, it is not sufficient to reflect the true stability feature of the method. Thus, the numerical stability of the LSFE–LBM is further studied via the linearization approach proposed by Sterling and Chen. [8], as detailed below.

Rearranging Eq. (13) into a more desirable format for stability analysis:

$$U_e\left(f_i^{n+1} + \frac{1-\theta}{\theta} f_i^n\right) + \frac{W_e}{\theta \Delta t}(f_i^{n+1} - f_i^n) = W_e\left(\Omega_i^{n+1} + \frac{1-\theta}{\theta}\Omega_i^n\right), \tag{29}$$

where, $U_e = \int_{\Omega_e} Q^T C B d\Omega_e$, $W_e = \int_{\Omega_e} Q^T N d\Omega_e$, and C, B, N are as defined in Eq. (15). In order to implement von Neumann stability analysis, it is necessary to transform Eq. (29) into a discrete form. For the purpose of simplicity, a uniform triangular mesh is utilized here, as illustrated in Fig. 1.

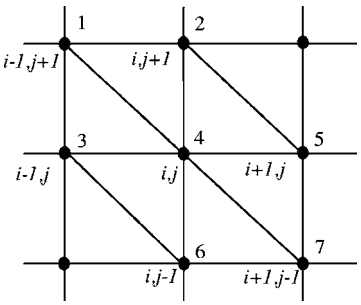

FIG. 1. The schematic plot of neighboring point distribution around the point (i,j) in a uniform triangular mesh for LSFE–LBM.

Assembling global matrixes based on this mesh, a discrete formula for point (i,j) can be expressed as:

$$\sum_{p=1}^{7} U_p\left(f_i^{n+1} + \frac{1-\theta}{\theta} f_i^n\right) + \frac{1}{\theta \Delta t}\sum_{p=1}^{7} W_p(f_i^{n+1} - f_i^n) = \sum_{p=1}^{7} W_p\left(\Omega_i^{n+1} + \frac{1-\theta}{\theta}\Omega_i^n\right), \tag{30}$$

where, p is the node number as denoted in Fig. 1. As proposed by Sterling and Chen. [8], f_i can be expanded as:

$$f_i = \overline{f^{(0)}} + f_i', \tag{31}$$

where, f_i' is the fluctuating quantity, and $\overline{f^{(0)}}$ is the global equilibrium population, not varying in space or time. It can be shown that the constant property of $\overline{f^{(0)}}$ results in the relationship:

$$U_p \overline{f^{(0)}} = 0. \tag{32}$$

Applying a Taylor-series expansion to the collision operator around $\overline{f^{(0)}}$ gives:

$$\Omega_i(f_j) = \left(\frac{\partial \Omega_i}{\partial f_j}\right)_{\overline{f^{(0)}}} f_j' + \Omega_i(\overline{f^{(0)}}) + O(f_j'^2) \approx \left(\frac{\partial \Omega_i}{\partial f_j}\right)_{\overline{f^{(0)}}} f_j' = G_{ij} f_j'. \tag{33}$$

Substituting this first-order approximation of the collision operator after the expansion of f_i by Eq. (30) in Eq. (31), and utilizing the relationship in Eq. (32), Eq. (30) is reduced to the following form:

$$\sum_{p=1}^{7} U_p\left(f_i'^{n+1} + \frac{1-\theta}{\theta} f_i'^{n}\right) + \frac{1}{\theta \Delta t}\sum_{p=1}^{7} W_p(f_i'^{n+1} - f_i'^{n}) = \sum_{p=1}^{7} W_p G\left[f_j'^{n+1} + \frac{1-\theta}{\theta} f_j'^{n}\right]. \tag{34}$$

Performing a Fourier transform of Eq. (34), it follows

$$F_i(k,t+\Delta t) = Z_{ij}F_j(k,t), \tag{35}$$

where, $F_j(k,t)=\int f_j'(x,t)\exp(-k\cdot x)dx$ and $k=(k_x,k_y)$ is the wave number. Matrix Z is given by:

$$Z=\left[\sum_{p=1}^{7}(A_p-W_pG)E_p\right]^{-1}\left[\sum_{p=1}^{7}\left(B_p+\frac{1-\theta}{\theta}W_pG\right)E_p\right], \tag{36}$$

where,

$$A_p=\mathrm{diag}\left\{\frac{W_p}{\theta\Delta t}+U_p\right\}_{9\times 9},\quad B_p=\mathrm{diag}\left\{\frac{W_p}{\theta\Delta t}-\frac{1-\theta}{\theta}U_p\right\}_{9\times 9},$$

and $E_p=\exp(ik\,dl_p)I_{9\times 9}$.

When the spectral radius of matrix Z, i.e., the largest value of eigenvalues, is not larger than unity, the system approaches stability. When the wave number is zero, E_p matrices become identity matrices, resulting in eigenvalues of matrix Z:

$$\left\{1,\quad\frac{(\theta-1)\Delta t+\tau}{\theta\Delta t+\tau}\right\},$$

with three and six multiplicity respectively, independent of macroscopic velocity. In this special case, the stability of the system is guaranteed when $\tau>\max\{0,(\frac{1}{2}-\theta)\Delta t\}$.

When the wave number is nonzero, the stability of LSFE–LBM is dependent on a number of parameters, including time step, element size, wave number, and relaxation time, similar to that reported in other studies for LBM on irregular mesh [8,37]. It is therefore not feasible to evaluate the full effects of these parameters on stability; rather, simplifications and restrictions may be imposed. In this study, the influences of collision frequency $\omega=\Delta t/\tau$, ratio dt/dl, and mean velocity u are evaluated with several simplifications. These include, (i) fix $\theta=0.5$; (ii) evaluate wave number vector only in the range $[0,\pi]$; (iii) use uniform mesh, i.e., $dx=dy=dl$; (iv) keeping the mean velocity and the wave number vector horizontal corresponding to the most likely unstable condition identified by Sterling and Chen [8].

A program written in MATLAB (version 6.5.1, The MathWorks, Inc., Natick, Massachusetts) was employed to numerically calculate eigenvalues of the matrix Z, and thus to determine the stability boundary. An iterative scheme is used in which values of ω, dt/dl, and τ were selected, and the mean flow velocity u is incrementally increased until the maximum eigenvalue exceeds unity. The resulting mean flow velocity is coined in terms of a maximum stable velocity, which is utilized to construct the stability boundary by varying the value of ω for several different dt/dl and τ values. Although it is very difficult to present a complete illustration of the dependence of the LSFE–LBM stability on physical parameters and numerical discretization, Figs. 2 and 3 shed light on the stability feature of LSFE–LBM with selected τ and dt/dl values.

Figure 2 presents the correlation between maximum stable velocity u and collision frequency ω for different dt/dl under a fixed τ value. As the value of ω decreases, the maximum stable velocity first increases, and then is held constant near 0.58. This existence of a limiting stable velocity was also observed by Sterling and Chen [8] for traditional LBM, implying the underlying inability of using a finite set of particle velocities to represent high flow velocity. The observed limiting stable velocity for the LSFE–LBM of 0.58, corresponding to a Mach number $M_a=u/c_s=1.0$, however, is higher than 0.42 for a traditional D2Q9 model derived by Sterling and Chen [8], indicating that the LSFE scheme actually enhanced the stability of LBM. Meanwhile, Fig. 2 shows that the stability region is increased with increased dt/dl values.

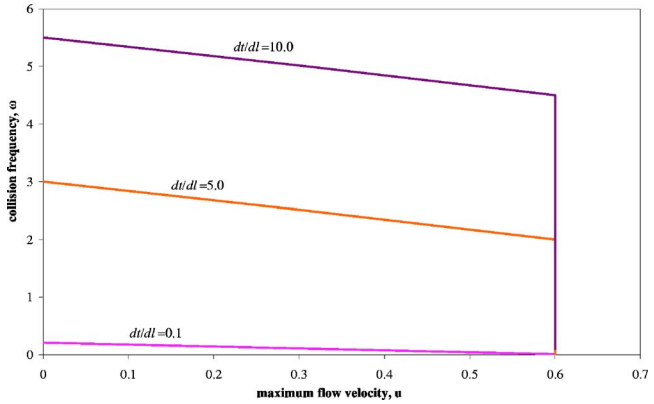

FIG. 2. (Color online) The stability boundaries as function of mean velocity u and collision frequencies ω for selected dt/dl when $\tau=1.0$.

Similar to Fig. 2, Fig. 3 presents the relationship between ω and the maximum stable velocity, but for different τ values. Again, with decreased ω, the maximum stable velocity increases until the largest possible value of 0.58 is reached. The smaller curve slopes for larger τ values agree well with other simulation results that suggest that LBM tends toward stability at higher values of velocity for a larger τ value. Moreover, it is observed in Fig. 3 that smaller τ values possess larger stability regions relative to larger τ values, which implicates trends in the dependency of stability on the numerical discretization. For a certain ω, a smaller τ value corresponds to a smaller dt value, and thus a smaller dl value

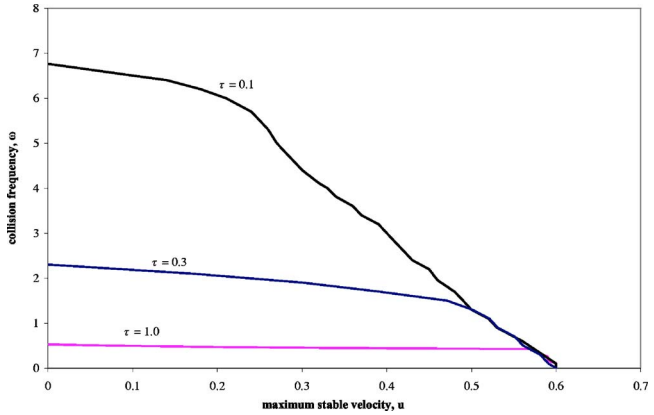

FIG. 3. (Color online) The stability boundaries as function of mean velocity u and collision frequencies ω for selected τ when $dt/dl=1.0$.

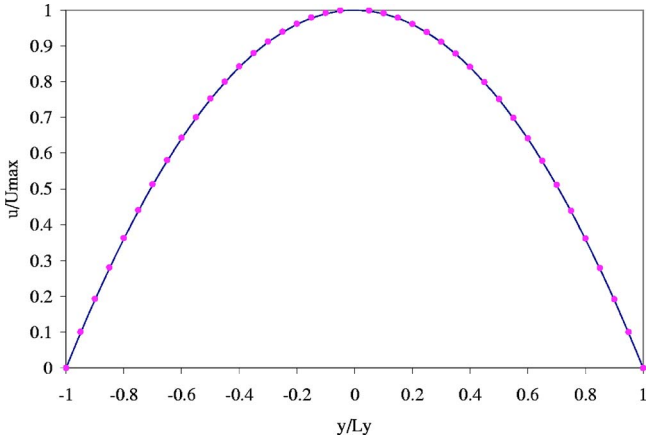

FIG. 4. (Color online) The comparison of LSFE–LBM solution (points) and analytical solution (line) for normalized velocity profile for Poiseuille flow. In LSFE–LBM, the relaxation time, τ, is 0.05, the particle density, ρ, is 1.0, the maximum velocity, u_{max}, is 0.1, and the half width of the channel, H, is $\frac{5}{6}$.

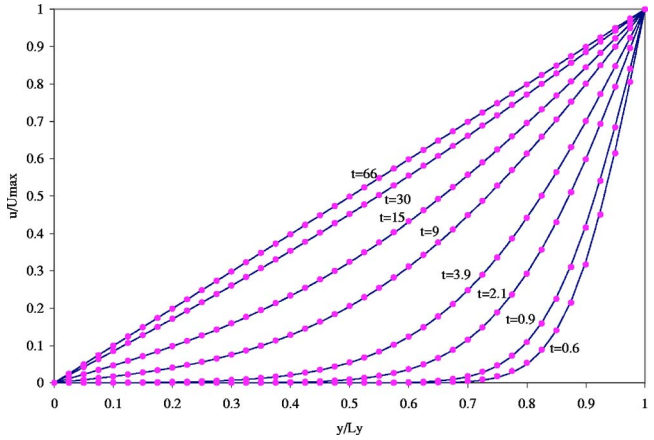

FIG. 5. (Color online) The comparison of LSFE–LBM solution and analytical solution for Couette flow. The points represent the LSFE–LBM solution, while the lines denote the analytical solution. The time step is 0.03, the relaxation time, τ, is 0.05, the maximum velocity, u_{max}, is 0.1, and the width of the channel, D, is 5/3.

due to the fixed dt/dl in Fig. 3. Therefore, the smaller dt and dl values may lead to larger stability regions, although, on the other hand, a higher dt/dl ratio tends to be more stable, based on Fig. 2.

IV. NUMERICAL RESULTS

Demonstration of the validity and power of LSFE–LBM is illustrated in the following test problems in this section, including: Poiseuille flow, Couette flow, flow past a circular cylinder, and flow in porous media.

A. Poiseuille flow

Poiseuille flow, i.e., channel flow driven by a constant pressure gradient, is first simulated to validate LSFE–LBM. An analytical solution to plane Poiseuille flow in a channel is provided by Eq. (37) [38]:

$$v_x(y) = u_{\max}\left[1 - \left(\frac{y}{H}\right)^2\right], \quad -H \leqslant y \leqslant H \tag{37}$$

where x is the spatial longitudinal dimension, y is the spatial transverse dimension, $u_{\max}$ is the maximum velocity at the parabolic velocity profile, and H is the half width of the channel. In our LSFE–LBM implementation, the initial flow velocity is zero, relaxation time, τ, is 0.05, particle density, ρ, is 1.0, $u_{\max}$ is 0.1, and H is $\frac{5}{6}$. A periodic boundary condition is applied in the x-direction, and a body force $G = 2\nu u_{\max}/H^2$ is applied in the x-direction to initiate the flow, where ν is the viscosity. This system possesses a Reynolds number ($R_e = u_{\max} 2H/\nu$) of 10, and a Mach number ($Ma = u_{\max}/c_s$) of 0.173. Results presented in Fig. 4 illustrate that LSFE–LBM achieves close agreement with the analytical solution.

It is observed in modeling Poiseuille flow, and later in Couette flow, that the viscosity of the fluid follows the relationship $\nu = \tau/3$, which is different from the viscosity relationship of the traditional D2Q9 lattice Boltzmann method $\nu = (\tau - dt/2)/3$. Although the negative component of the D2Q9 viscosity expression is derived from numerical errors, the equation enables very small viscosities by properly adjusting τ and dt values. The absence of the negative component in the viscosity expression of LSFE–LBM, however, enables the use of very small τ values to achieve small viscosity values, which may lead to high Reynolds numbers. A more thorough exploration of the efficiency of LSFE–LBM with high Reynolds numbers, however, necessitates additional theoretical and numerical tests.

B. Couette flow

The second application of unsteady Couette flow is used to evaluate the temporal accuracy of LSFE–LBM. Different from Poiseuille flow, here the top plate is moving along the x-direction at a constant velocity, $u_{\max}$, while the bottom plate remains stationary. The analytical solution for Couette flow is [38]:

$$u(y,t) = u_{\max}\frac{y}{D} + \sum_{i=1}^{\infty} \frac{2u_{\max}(-1)^i}{\lambda_i D} e^{-1/\lambda_i^2 t} \sin \lambda_i y, \quad 0 \leqslant y \leqslant D, \tag{38}$$

where $\lambda_i = i\pi/D$, $m = 1,2,3\ldots$.

A periodic boundary condition is applied in the x-direction, and the Reynolds number $R_e = u_{\max} D/\nu$ is again set equal to 10, where D represents the width of the channel. The time step is 0.03, relaxation time, τ, is 0.05, particle density is 1.0, $u_{\max}$ is 0.1, and D is $\frac{5}{3}$. A comparison of numerical results and the analytical solution is shown in Fig. 5.

C. Flow past a circular cylinder

LSFE–LBM was also applied to simulate steady-state flow past a circular cylinder, since this problem has been widely employed [25,37,39] as a benchmark problem to vali-

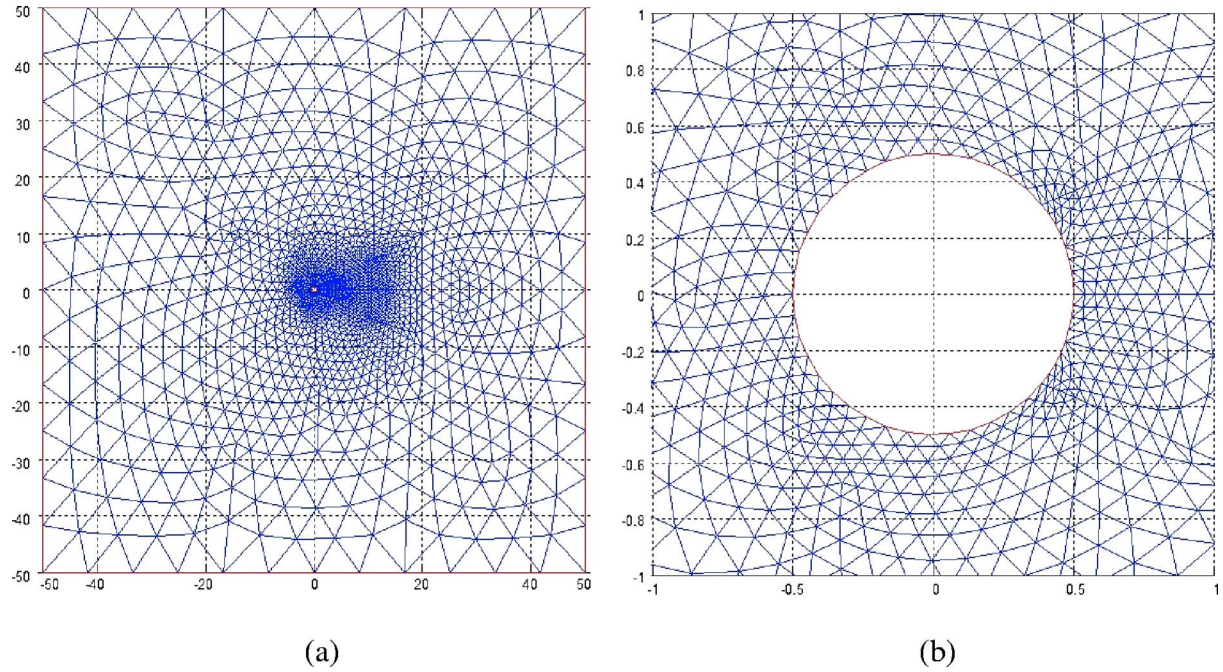

FIG. 6. (Color online) The unstructured mesh for flow past a circular cylinder in (a) the entire computational domain and (b) the vicinity of the cylinder.

date different numerical methods. Here, the results are compared with previous numerical and experimental studies. Simulation is carried out in a square domain with width $W=100d, M_a=0.1$, and $R_e=u_\infty d/\nu=20$, where u_∞ is the free stream velocity and d is the diameter of the circular cylinder. Unstructured triangular mesh is applied, as shown in Fig. 6, which includes 2544 nodes and 4992 elements.

Simulation starts from an irrotational potential flow. Free stream velocity u_∞ is enforced on the domain boundaries, while keeping the distribution function in its equilibrium state. Periodic boundary conditions are implemented for the inlet and outlet of the simulation domain, while a bounce back rule is imposed to ensure the non-slip condition at the cylinder surface.

LSFE–LBM simulation results show a pair of stationary recirculation eddies appearing behind the cylinder, as reported in many previous studies [25,37,39]. Geometric parameters of the flow are measured and listed in Table II, including the separation angle Θ_s and the ratio of wake length to cylinder radius L/r_0, where wake length L is defined as the distance from the rearmost point of the cylinder to the end of the wake. Dynamic parameters, including the drag coefficient (C_D) and the stagnation pressure coefficients at the front, $C_p(\pi)$, and at the end, $C_p(0)$, of the cylinder,

TABLE II. The comparison of geometric and dynamic parameters of flow past a circular cylinder with previous studies.

Authors	L/r_0	Θ_s	C_D	$-C_p(0)$	$C_p(\pi)$
Tritton[a] Ref. [40]	1.86	41.6			
Coutanceau and Bouard[a] Ref. [41]	1.86	44.8			
Nieuwstadt and Keller[b] Ref. [42]	1.786	43.37	2.053	0.582	1.274
Dennis and Chang[b] Ref. [43]	1.88	43.7	2.045	0.589	1.269
Fornberg[b] Ref. [54]			2.000	0.54	1.28
He and Doolen[c] Ref. [39]	1.843	42.96	2.152	0.567	1.233
Guo and Zhao[d] Refs. [25,37]	1.824	43.59	2.048	0.512	1.289
Lee and Lin[e] Ref. [25]	1.85	44.08	1.998	0.530	1.248
Present[f]	1.835	44.64	2.011	0.551	1.262

[a]Experiment.
[b]Numerical simulation of Navier–Stokes equations.
[c]Interpolation-supplemented LBM on structured mesh with 181×241 grid points.
[d]Explicit finite-difference LBM on structured mesh with 129×64 grid points.
[e]Characteristic Galerkin discrete LBM on unstructured mesh with 2568 grid points
[f]Least squares finite-element LBM on unstructured mesh with 2544 grid points.

were also measured and listed in Table II. The stagnation pressure coefficient C_p is defined as:

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho u_\infty^2} \tag{39}$$

where, p is the pressure which can be evaluated directly using:

$$p = c_s^2 \rho. \tag{40}$$

The drag coefficient, C_D, is defined as

$$C_D = \frac{1}{\rho u_\infty^2 r_0} \int S\, ndl \tag{41}$$

where, n is the normal direction of the cylinder wall and S is the stress tensor given by:

$$S = pI + \rho\nu(\nabla u + [\nabla u]^T). \tag{42}$$

As shown in Table II, the results of LSFE–LBM are in good agreement with previous experimental studies [40,41] and finite-difference based CFD methods by Nieuwstadt and Keller [42] and Dennis and Chang [43]. Compared to LBM on structured mesh [37,39], LSFE–LBM achieves good agreement with the simulation results while using a much smaller number of grid points. Approximately the same number of grid points as CGDBE [25] on the unstructured mesh is utilized by LSFE–LBM in this study. However, due to the implicit feature of the LSFE scheme, LSFE–LBM can be implemented with a larger time step.

D. Flow in porous media

Traditional LBM has been successfully applied to study fluid flow in porous media by numerous studies [44–53]. To demonstrate its ability to address complex geometries and compare its performance with traditional LBM, we here apply LSFE–LBM to simulate flow in a porous medium and estimate permeability of the simulation domain. At low Reynolds number for single-phase flow, specific permeability k of porous media, in units of L^2 or Darcy, can be described within the context of Darcy's law

$$q = -\frac{k}{\mu} \nabla p \tag{43}$$

where, q $[L/T]$ is the specific flow rate, μ $[M/LT]$ is the viscosity of the fluid, and ∇p $[M/L^2T^2]$ represents the pressure gradient.

In this study, porous media are envisioned as a statistical distribution of nonoverlapping circular disks representing soil particles distributed in a rectangular two-dimensional uniform continuum representing the pore space through which a fluid flows. Simulation is conducted on a 1 mm $\times$1 mm domain with porosity 0.5, and randomly generated particle diameters obeying a lognormal distribution with geometric mean 100 μm and coefficient of variance (COV) 0.3. For comparison, both traditional LBM with uniform mesh and LSFE–LBM with unstructured mesh are considered. Figure 7 illustrates an example irregular triangular mesh for LSFE–LBM.

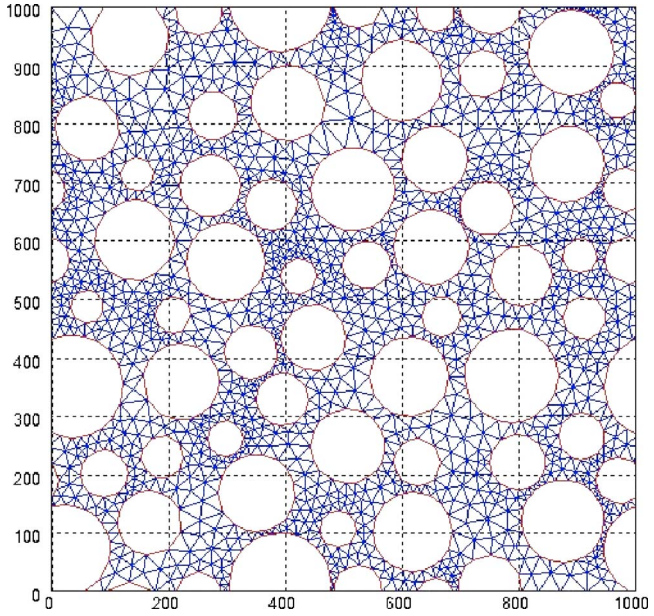

FIG. 7. (Color online) An example of unstructured mesh for flow in the porous media.

No-flow boundary conditions were applied on upper and lower edges, which are parallel to the main flow direction. Periodic boundary conditions were applied at the inlet and outlet of the domain, which require that the last column of nodes face nodes of the same y-axis values at the first column. For porous media with complex geometry, an unstructured mesh commonly will not provide such a symmetric node structure for domain inlet and outlet. To overcome this problem, an additional buffer area without any soil particles was added at the domain inlet and outlet. A similar technique was also utilized by Ubertini and co-workers [23,24] to treat zero-gradient boundary conditions for unstructured mesh, where acceptable results were found. Bounce back boundary conditions are applied to guarantee the non-slip condition at the surface of particles. To mimic the effects of a pressure gradient along the horizontal direction, an external body force was enforced on the fluid in the porous media, which generates fluid flow at low Reynolds number of approximately 0.05. Darcy's law [Eq. (43)] may thus be applied to calculate the permeability of the porous media.

Traditional LBM was carried out with increasing numerical resolution until the effects of spatial discretiztion were negligible. As presented in Fig. 8, at grid point number 2001×2001, the permeability estimated by traditional LBM approaches a stable value of 35.56 Darcy, which is utilized as a standard value to compare with LSFE–LBM simulation results. Grid points of 8866 are utilized in LSFE–LBM simulation, leading to a permeability value of 33.59 Darcy. While the relative error between the two methods is only approximately 5.5%, the number of grid points utilized by traditional LBM is about 452 times that used by LSFE–LBM. Meanwhile, the memory usage of LSFE–LBM is only about $\frac{1}{25}$ of traditional LBM.

It is worthy to mention that, since LSFE–LBM inherits the computational complexity from the LSFE method, the computing demand is higher. For simple geometric systems,

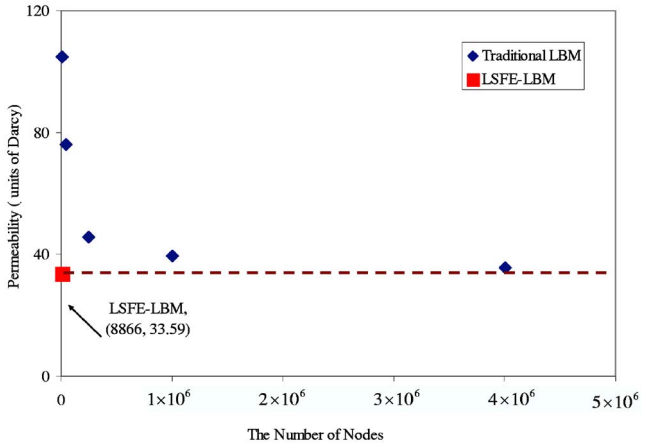

FIG. 8. (Color online) A comparison of the performance of traditional LBM and LSFE–LBM on simulating flow in the porous media.

such as Poiseuille flow and Couette flow, LSFE–LBM will require greater computational effort relative to traditional LBM. However, the reduced grid point requirement for unstructured mesh will partially offset the negative influence on computational efficiency. In this specific example, LSFE–LBM with 8866 nodes and traditional LBM with 2001 $\times$2001 nodes require approximately the same amount of time to achieve equilibrium.

V. SUMMARY AND FUTURE DIRECTIONS

In this paper, we present a new numerical model for LBM by implementing a least-squares finite-element scheme on unstructured mesh. Following a theoretical accuracy and stability analysis, LSFE–LBM was applied to a variety of test problems, including Poiseuille flow, Couette flow, flow past a cylinder, and flow in porous media.

Accuracy analysis results suggest that LSFE–LBM enjoys fourth-order accuracy in space and second-order accuracy in time. The von Neumann linearized stability analysis indicates that the stability of LSFE–LBM is dependent on physical and numerical discretization parameters. Its improved stability property is confirmed by a higher limiting stable velocity. Good agreement was achieved between LSFE–LBM and analytical solutions, experimental results, and previous numerical results. The flexibility of LSFE–LBM for complex geometric systems was demonstrated by a successful simulation of a fluid flow in porous media problem with reduced memory requirements.

Encouraging results from this work suggest that LSFE–LBM will be a promising addition to the family of LBM, especially for geometrically complex domains. Further improvements of LSFE–LBM, however, are suggested. First, LSFE–LBM numerical tests were primarily applied to complex geometries with low Reynolds numbers, corresponding to our research focus area. Although it is predicted that LSFE–LBM will provide potential for employment in high Reynolds number conditions, additional efforts are required to validate this point. Second, LBM is well-suited for distributed computing. Since element contributions are computed independently, element-by-element based LSFE–LBM can also be easily implemented in parallel. It is thus worthwhile to provide a more thorough study on the performance of LSFE–LBM following parallelization. Finally, since finite volume based CFDs are more commonly employed relative to finite element based CFDs, it will be meaningful to further assess and compare the performance of LSFE–LBM, FV–LBM, and FV based CFDs.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 0088912. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

[1] S. Chen and G. D. Doolen, Annu. Rev. Fluid Mech. **30**, 329 (1998).
[2] R. Benzi, S. Succi, and M. Vergassola, Phys. Rep., Phys. Lett. **222**, 145 (1992).
[3] G. R. McNamara and G. Zanetti, Phys. Rev. Lett. **61**, 2332 (1988).
[4] F. Higuera, S. Succi, and R. Benzi, Europhys. Lett. **9**, 345 (1989).
[5] F. J. Higuera and J. Jiménez, Europhys. Lett. **9**, 663 (1989).
[6] S. Y. Chen, H. D.Chen, D. Martinez, and W. Matthaeus, Phys. Rev. Lett. **67**, 3776 (1991).
[7] S. Succi, G. Amati, and R. Benzi, J. Stat. Phys. **81**, 5 (1995).
[8] J. D. Sterling and S. Chen, J. Comput. Phys. **123**, 196 (1996).
[9] X. He, L.-S. Luo, and M. Dembo, J. Comput. Phys. **129**, 357 (1996).
[10] C. Shu, X. D. Niu, and Y. T. Chew, Phys. Rev. E **65**, 036708 (2002).
[11] T. Imamura, K. Suzuki, T. Nakamura, and M. Yoshida, J. Comput. Phys. **202**, 645 (2005).
[12] O. Filippova and D. Hanel, J. Comput. Phys. **147**, 219 (1998).
[13] C.-L. Lin and Y. G. Lai, Phys. Rev. E **62**, 2219 (2000).
[14] A. Dupuis and B. Chopard, Phys. Rev. E **67**, 066707 (2003).
[15] H. Chen, Phys. Rev. E **58**, 3955 (1998).
[16] N. Z. Cao, S. Y. Chen, S. Jin, and D. Martinez, Phys. Rev. E **55**, R21 (1997).
[17] R. Mei and W. Shyy, J. Comput. Phys. **143**, 426 (1998).
[18] F. Nannelli and S. Succi, J. Stat. Phys. **68**, 401 (1992).
[19] G. Amati, S. Succi, and R. Benzi, Fluid Dyn. Res. **19**, 289 (1997).
[20] G. Peng, H. Xi, and C. Duncan, Phys. Rev. E **58**, R4124 (1998).
[21] G. Peng, H. Xi, and C. Duncan, Phys. Rev. E **59**, 4675 (1999).
[22] H. Xi, G. Peng, and S.-H. Chou, Phys. Rev. E **59**, 6202 (1999).

[23] S. Ubertini, G. Bella, and S. Succi, Phys. Rev. E **68**, 016701 (2003).
[24] S. Ubertini and S. Succi, Comput. Fluid Dyn. J. **5**, 85 (2005).
[25] T. Lee and C.-L. Lin, J. Comput. Phys. **171**, 336 (2001).
[26] T. Lee and C.-L. Lin, J. Comput. Phys. **185**, 445 (2003).
[27] G. Strang and G. J. Fix, in *An Analysis of the Finite Element Method* (Prentice-Hall, Englewood Cliffs, 1973).
[28] B.-n. Jiang, in *The Least-Squares Finite Element Method: Theory and Applications in Computational Fluid Dynamics and Electromagnetics* (Springer, New York, 1998).
[29] X. Ding (University of Kentucky, Lexington, Kentucky, 1999).
[30] Y. Li, E. J. LeBoeuf, and P. K. Basu, Phys. Rev. E **69**, 065701 (2004).
[31] P. Bhatnagar, E. P. Gross, and M. K. Krook, Phys. Rev. **94**, 511 (1954).
[32] Q. Zou and X. He, Phys. Fluids **9**, 1591 (1997).
[33] S. Chen, D. Martinez, and R. Mei, Phys. Fluids **8**, 2527 (1996).
[34] R. S. Maier, R. S. Bernard, and D. W. Grunau, Phys. Fluids **8**, 1788 (1996).
[35] A. J. Wathen, Comput. Methods Appl. Mech. Eng. **74**, 271 (1989).
[36] G. Comini, M. Manzan, and C. Nonino, Int. J. Numer. Methods Fluids **20**, 443 (1995).
[37] Z. Guo and T. S. Zhao, Phys. Rev. E **67**, 066709 (2003).
[38] W. M. Deen, in *Analysis of Transport Phenomena* (Oxford University Press, New York, 1998).
[39] X. He and G. D. Doolen, J. Comput. Phys. **134**, 306 (1997).
[40] J. D. Tritton, J. Fluid Mech. **6**, 231 (1959).
[41] M. Coutanceau and R. Bouard, J. Fluid Mech. **79**, 231 (1977).
[42] F. Nieuwstadt and H. B. Keller, Comput. Fluids **1**, 59 (1973).
[43] S. C. R. Dennis and G. Z. Chang, J. Fluid Mech. **42**, 471 (1970).
[44] S. Chen, K. Diemer, G. D. Doolen, K. Eggert, C. Fu, S. Gutman, and B. J. Travis, Physica D **47**, 72 (1991).
[45] A. Cancelliere, C. Chang, E. Foti, D. H. Rothman, and S. Succi, Phys. Fluids A **12**, 2085 (1990).
[46] D. H. Rothman, Geophysics **53**, 509 (1988).
[47] L. Talon, J. Martin, N. Rakotomalala, D. Salin, and Y. C. Yortsos,Water Resour. Res. **39**, 1135 (2003).
[48] D. Zhang and Q. Kang, Geophys. Res. Lett. **31**, L12504 (2004).
[49] D. Zhang, R. Y. Zhang, S. Y. Chen, and V. E. Soll, Geophys. Res. Lett. **27**, 1195 (2000).
[50] M. C. Sukop and D. Or, Water Resour. Res. **40**, W01509 (2004).
[51] M. A. A. Spaid and F. R. Phelan,Jr.,Phys. Fluids **9**, 2468 (2004).
[52] C. Pan, M. Hilpert, and C. T. Miller, Water Resour. Res. **40**, W01501 (2004).
[53] Y. Li, E. J. LeBoeuf, P. K. Basu, and S. Madadevan, Adv. Water Resour. **28**, 835 (2005)
[54] B. Fornberg, J. Fluid Mech. **98**, 819 (1980).