Computer Physics Communications 185 (2014) 1739–1746

Contents lists available at ScienceDirect

Computer Physics Communications

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

BIOTC: An open-source CFD code for simulating biomass fast pyrolysis☆

Qingang Xiong, Soroush Aramideh, Alberto Passalacqua, Song-Charng Kong*

Department of Mechanical Engineering, Iowa State University, Ames, IA 50011, USA

ARTICLE INFO

Article history:
Received 27 November 2013
Received in revised form
23 January 2014
Accepted 4 February 2014
Available online 19 February 2014

Keywords:
Biomass fast pyrolysis
Multi-fluid model
Computer simulation
Chemical reactions
Fluidized-bed reactor
OpenFOAM

ABSTRACT

The BIOTC code is a computer program that combines a multi-fluid model for multiphase hydrodynamics and global chemical kinetics for chemical reactions to simulate fast pyrolysis of biomass at reactor scale. The object-oriented characteristic of BIOTC makes it easy for researchers to insert their own sub-models, while the user-friendly interface provides users a friendly environment as in commercial software. A laboratory-scale bubbling fluidized bed reactor for biomass fast pyrolysis was simulated using BIOTC to demonstrate its capability.

Program summary

Program title: BIOTC-2.1.x

Catalogue identifier: AESJ_v1_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/AESJ_v1_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: Standard CPC licence, http://cpc.cs.qub.ac.uk/licence/licence.html

No. of lines in distributed program, including test data, etc.: 82295

No. of bytes in distributed program, including test data, etc.: 266197

Distribution format: tar.gz

Programming language: C++.

Computer: All capable of running Linux.

Operating system: Linux.

Has the code been vectorized or parallelized?: Parallelized with MPI.

Classification: 22.

External routines: MPI, OpenFOAM (http://www.openfoam.org)

Nature of problem:
Computational fluid dynamics (CFD) simulation of biomass fast pyrolysis at reactor scale can help reveal the details of the process and develop an understanding of the underlying mechanisms for reactor operation and optimization. However, the existing CFD codes, commercial or open-source, still pose difficulties for users to carry out an efficient simulation. Therefore, an open-source CFD code that integrates the merits of mainstream commercial and open-source codes, i.e., user-friendly interface and convenient sub-model implementation, is by all means necessary for reactor-scale simulation of biomass fast pyrolysis.

Solution method:
A multi-fluid model (MFM) is used to solve the multiphase fluid dynamics, while a global reaction mechanism is employed to solve the chemical reactions in the fluidized-bed reactors. Partial differential equation solvers and ordinary differential equation solvers provided by OpenFOAM are used to solve the

☆ This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (http://www.sciencedirect.com/science/journal/00104655).

* Corresponding author. Tel.: +1 515 294 3244.
E-mail addresses: qgxiong@126.com, xiong@iastate.edu (Q. Xiong), kong@iastate.edu (S. Kong).

http://dx.doi.org/10.1016/j.cpc.2014.02.012

MFM conservation equations and the chemical reaction equations, respectively. The coupling of MFM and chemical reactions is realized by a time-split numerical scheme.

Running time:
It depends on the dimension of the reactor and the complexity of the in-reactor process. Typically, for the tutorial case provided, the run time ranges from several hours to tens of hours.
The small test case provided takes approximately 30 min on a serial machine.

1. Introduction

Computer simulation has been increasingly used as an effective tool to study biomass fast pyrolysis at all scales, spanning from single biomass particle [1–7] to the integrated engineering process [8–10]. Among the wide spectra of biomass fast pyrolysis, understanding the physiochemical phenomena taking place at the reactor scale using computational fluid dynamics (CFD) has received considerable interest because of its importance in bridging the fundamental particle-scale study and the practical plant-scale operations [11,12]. Thus, CFD of biomass fast pyrolysis at the reactor scale has become a critical subject of research in recent years [13–24].

Limited by the relatively large dimension of a typical reactor and the current computational capability, numerical approaches that track each individual particle, e.g., direct numerical simulation [25–29] and discrete particle simulation [30–33], are still not applicable since millions to billions of particles need be resolved. By modeling all phases as interpenetrating continua, the so-called multi-fluid model (MFM) [34–36] has been widely adopted in the CFD simulation of chemical reactors from the laboratory to industrial scales. This is also the case for the computer simulation of reactor-scale biomass fast pyrolysis [37–46]. In MFM, all solid phases are formulated as continua on discretized grids whose sizes are one to two orders of magnitude larger than the particle diameter, and conservation laws and reaction kinetics are used to describe the hydrodynamics and chemical reactions, respectively. Partially differential equation (PDE) solvers are applied to solve MFM for transport phenomena and ordinarily differential equation (ODE) solvers are used to compute chemical reaction rates.

Several commercial and open-source codes use MFM to simulate biomass fast pyrolysis at the reactor scale, such as Fluent and MFIX. Although commercial codes offer many convenient features, e.g., user friendly interface and mesh generators for complex geometries, it can be difficult for the users to implement new numerical algorithms and physical/chemical models. On the other hand, an open-source code allows the users a greater freedom to develop and incorporate new models; however, this freedom is bounded with the fact that users need to be experts in numerical modeling and programming. Additionally, open-source codes usually do not have a good user interface or capabilities to handle complex geometries. Therefore, an advanced open-source code that can integrate the merits of commercial codes and conventional open-source codes is favored.

OpenFOAM (Open-source Field Operation And Manipulation) [47], which was developed very recently and is updated annually, provides a new approach in developing open-source codes for CFD applications. The finite-volume-based discretization scheme allows OpenFOAM to treat complex boundaries easily. Its object-oriented development strategy can allow users to implement their own sub-models into the baseline codes with minimal effort. In addition, the user interface of OpenFOAM, which is supported by hundreds of utilities, is very close to those of commercial codes. Because of these advantages, OpenFOAM has found a great number of applications in all fields of computational sciences and engineering [48,49], even financial simulation. Thus, developing an open-source CFD code based on the framework of OpenFOAM to simulate biomass fast pyrolysis at reactor scale appears to be a promising approach that combines the two important features of a CFD code, i.e., user-friendly interface as in commercial codes and transparent structure as in conventional open-source codes.

In this paper, our recently developed open-source code [50–52] for simulating biomass fast pyrolysis at reactor scale based on OpenFOAM, BIOTC (BIOmass Thermochemical Conversion), is described. The background mathematical formulations, i.e., MFM and chemical reaction kinetics, are described first. Then the code structure and development strategy are presented in detail. Finally, a sample simulation was conducted to demonstrate the capability of BIOTC.

2. Mathematical formulations

2.1. Multi-fluid model

In BIOTC, the multiphase fluid dynamics is described by MFM. In MFM, one gas phase and an arbitrary number of solid phases, in which each phase can contain an arbitrary number of species, are considered. This is a preferred representation of the phase composition in a typical biomass fast pyrolysis reactor, in which biomass particles are mixed with other inert solids such as sand or limestone, and all solid particles are fluidized by gases. The continuity, momentum, energy, and species conservation equations are employed to describe the spatio-temporal evolutions of all phases, in which local instantaneous balance is established. The conservation equations are presented in the following based on M solid phases.

2.1.1. Gas phase

The continuity equation to describe the gas phase volume fraction is

$$\frac{\partial \alpha_g \rho_g}{\partial t} + \nabla \cdot (\alpha_g \rho_g \boldsymbol{U}_g) = \sum_{m=1}^{M} R_{gsm} \tag{1}$$

where α_g, ρ_g and $\boldsymbol{U}_g$ are the volume fraction, material density, and velocity vector, respectively. R_{gsm} is the source term to account for the interphase chemical reactions between the gas phase and solid phase m. In BIOTC, the ideal gas law is used.

The momentum conservative equation is

$$\begin{aligned}\frac{\partial (\alpha_g \rho_g \boldsymbol{U}_g)}{\partial t} &+ \nabla \cdot (\alpha_g \rho_g \boldsymbol{U}_g \boldsymbol{U}_g) \\ &= \nabla \cdot \boldsymbol{\tau}_g - \alpha_g \nabla p + \sum_{m=1}^{M} \beta_{gsm} (\boldsymbol{U}_{sm} - \boldsymbol{U}_g) \\ &+ \sum_{m=1}^{M} \boldsymbol{\psi}_{gsm} + \alpha_g \rho_g \boldsymbol{g}\end{aligned} \tag{2}$$

where $\boldsymbol{\tau}_g$ is the stress tensor, β_{gsm} is the interphase momentum exchange coefficient between gas and solid phase m, $\boldsymbol{\psi}_{gsm}$ is the gas–solid momentum exchange due to chemical reactions, and $\boldsymbol{g}$ is the gravity vector. $\boldsymbol{\tau}_g$ is expressed in a Newtonian form,

$$\boldsymbol{\tau}_g = 2\alpha_g\mu_g\boldsymbol{D}_g + \alpha_g\lambda_g tr(\boldsymbol{D}_g)\boldsymbol{I} \tag{3}$$

where $\boldsymbol{I}$ is the identity tensor and $\boldsymbol{D}_g$ is the strain tensor. μ_g and λ_g are dynamic and bulk viscosity, respectively. In biomass fast pyrolysis, as the gas–solid drag force plays the dominant role over other effects, such as virtual mass force and lift force, only the gas–solid drag force is included into β_{gsm}, though other effects can be added easily. Several sub-models to calculate β_{gsm}, such as the Gidaspow drag model [53], Syamlal–O'Brien drag model [54], and EMMS drag model [55], are available in BIOTC. For example, the Gidaspow drag model calculates β_{gsm} as

$$\beta_{gsm} = \begin{cases} \dfrac{3}{4}C_{Dm}\dfrac{\rho_g\alpha_g\alpha_{sm}|\boldsymbol{U}_g - \boldsymbol{U}_{sm}|}{d_{sm}}\alpha_g^{-2.65} \\ \quad \alpha_g \geq 0.8 \\ 150\dfrac{\alpha_{sm}(1-\alpha_g)\mu_g}{\alpha_g d_{sm}^2} + \dfrac{7}{4}\dfrac{\rho_g\alpha_{sm}|\boldsymbol{U}_g - \boldsymbol{U}_{sm}|}{d_{sm}} \\ \quad \alpha_g < 0.8 \end{cases} \tag{4}$$

$$C_{Dm} = \begin{cases} \dfrac{24}{\mathrm{Re}_m}(1 + 0.15\mathrm{Re}_m^{0.687}) & \mathrm{Re}_m < 1000 \\ 0.44 & \mathrm{Re}_m \geq 1000 \end{cases}$$

$$\mathrm{Re}_m = \frac{\rho_g d_{sm}|\boldsymbol{U}_g - \boldsymbol{U}_{sm}|}{\mu_g}$$

$\boldsymbol{\psi}_{gsm}$ is described as follows.

$$\boldsymbol{\psi}_{gsm} = R_{gsm}[\zeta\boldsymbol{U}_{sm} + (1-\zeta)\boldsymbol{U}_g],$$
$$\zeta = \begin{cases} 0 & R_{gsm} < 0 \\ 1 & R_{gsm} \geq 0. \end{cases} \tag{5}$$

The energy equation is

$$\frac{\partial(\alpha_g\rho_g C_{pg}T_g)}{\partial t} + \nabla\cdot(\alpha_g\rho_g C_{pg}T_g\boldsymbol{U}_g) = \nabla\cdot\boldsymbol{q}_g + \sum_{m=1}^{M} h_{gsm}(T_{sm} - T_g) + \sum_{m=1}^{M}\chi_{gsm} + \Delta H_g \tag{6}$$

where T_g, C_{pg} and $\boldsymbol{q}_g$ are temperature, heat capacity, and conductive heat flux, respectively. Fourier's law is used to calculate the conductive heat flux,

$$\boldsymbol{q}_g = \alpha_g\kappa_g\nabla T_g \tag{7}$$

where κ_g is the thermal conductivity. h_{gsm} is the gas–solid heat transfer coefficient, similar to gas–solid momentum transfer coefficient, and ΔH_g is the heat resulting from the internal gas phase reactions. χ_{gsm} is the gas–solid energy transfer arising from chemical reactions between gas and solid phases. Similar to β_{gsm}, several sub-models to calculate h_{gsm} are available in BIOTC for users to select, such as the Ranz–Marshall heat transfer model [56], Gunn heat transfer model [57], and Li–Mason heat transfer model [58]. For example, the Ranz–Marshall heat transfer model calculates h_{gsm} as

$$h_{gsm} = \frac{6\alpha_{sm}\kappa_g Nu_m}{d_{sm}^2}$$
$$Nu_m = 2 + 0.6\mathrm{Re}_m^{1/2}\,\mathrm{Pr}^{1/3} \tag{8}$$
$$\mathrm{Pr} = \frac{C_{pg}\mu_g}{\kappa_g}.$$

Similar to $\boldsymbol{\psi}_{gsm}$, χ_{gsm} reads

$$\chi_{gsm} = R_{gsm}[\zeta C_{psm}T_{sm} + (1-\zeta)C_g T_g],$$
$$\zeta = \begin{cases} 0 & R_{gsm} < 0 \\ 1 & R_{gsm} \geq 0. \end{cases} \tag{9}$$

The species mass equation is

$$\frac{\partial\alpha_g\rho_g Y_{gk}}{\partial t} + \nabla\cdot(\alpha_g\rho_g Y_{gk}\boldsymbol{U}_g) = \nabla\cdot\boldsymbol{j}_{gk} + R_{gk} \tag{10}$$

where Y_{gk} and $\boldsymbol{j}_{gk}$ denote the mass fraction and diffusive flux of kth species, respectively. Fick's law is used to calculate the species diffusive flux,

$$\boldsymbol{j}_{gk} = \alpha_g\rho_g D_{gk}\nabla Y_{gk}. \tag{11}$$

It is worth noting that the source term R_{gk} includes contributions from all chemical reactions between species k with other species in gas and solid phases.

2.1.2. Solid phases

The conservation equations of solid phase m are described in the following. The continuity equation of solid phase m is

$$\frac{\partial\alpha_{sm}\rho_{sm}}{\partial t} + \nabla\cdot(\alpha_{sm}\rho_{sm}\boldsymbol{U}_{sm}) = R_{sm} \tag{12}$$

where R_{sm} accounts for all of the chemical reactions between solid phase m with other phases, including gas.

The momentum equation of solid phase m is

$$\frac{\partial(\alpha_{sm}\rho_{sm}\boldsymbol{U}_{sm})}{\partial t} + \nabla\cdot(\alpha_{sm}\rho_{sm}\boldsymbol{U}_{sm}\boldsymbol{U}_{sm}) = \nabla\cdot\boldsymbol{\tau}_{sm} - \alpha_{sm}\nabla p + \beta_{gsm}(\boldsymbol{U}_g - \boldsymbol{U}_{sm}) + \sum_{l=1,l\neq m}^{M}\beta_{slm}(\boldsymbol{U}_{sl} - \boldsymbol{U}_{sm}) + \boldsymbol{\psi}_{sm} + \alpha_{sm}\rho_{sm}\boldsymbol{g} \tag{13}$$

β_{slm} accounts for the momentum exchange between solid phase l and m. $\boldsymbol{\psi}_{sm}$ accounts for the momentum transfer by chemical reactions between solid phase m with other phases. In the present model, τ_{sm} is expressed as

$$\tau_{sm} = -p_{sm}\boldsymbol{I} + 2\alpha_{sm}\mu_{sm}\boldsymbol{D}_{sm} + \alpha_{sm}\lambda_{sm}tr(\boldsymbol{D}_{sm})\boldsymbol{I} \tag{14}$$

where p_{sm}, μ_{sm} and λ_{sm} are the so-called granular pressure, dynamic viscosity, and bulk viscosity, respectively. These three granular properties are determined by the kinetic theory of granular flows (KTGF) [59], a function of the granular energy Θ_{sm}.

The energy equation of solid phase m is

$$\frac{\partial(\alpha_{sm}\rho_{sm}C_{psm}T_{sm})}{\partial t} + \nabla\cdot(\alpha_{sm}\rho_{sm}C_{psm}T_{sm}U_{sm}) = \nabla\cdot q_{sm} + h_{gsm}(T_g - T_{sm}) + \chi_{sm} + \Delta H_{sm} \tag{15}$$

where ΔH_{sm} considers all reactions that solid phase m is involved in.

The mass species equation of solid phase m is

$$\frac{\partial\alpha_{sm}\rho_{sm}Y_{smk}}{\partial t} + \nabla\cdot(\alpha_{sm}\rho_{sm}Y_{smk}\boldsymbol{U}_{sm}) = R_{smk}. \tag{16}$$

Because of the lack of accurate data on solid phase diffusive coefficients in the literature and the dominant role of convection, in Eq. (16), the diffusion term is not included.

2.2. Chemical reaction kinetics

In fast pyrolysis, because of the lack of oxidizing agents, biomass is rapidly decomposed to gas, tar, and biochar through primary decompositions. When temperature promises, tar will crack into gas and/or biochar. The primary reactions of biomass fast pyrolysis are featured to be endothermic while the secondary tar cracking reactions are endothermic or exothermic, depending on the operating conditions and initial biomass compositions. Because the actual processes associated with biomass fast pyrolysis are very complex and the accurate description of the fundamental mechanisms is still under development, BIOTC employs global biomass

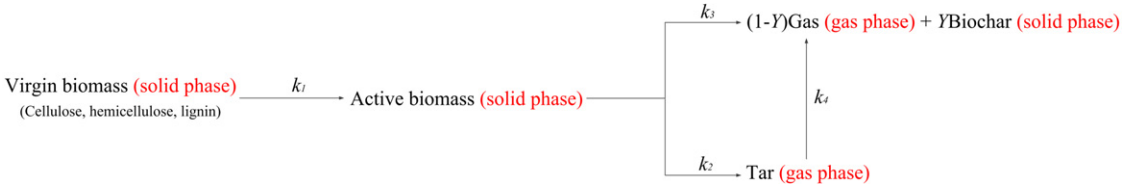

Fig. 1. Reaction steps in the modified Broido–Shafizadeh mechanism [60].

decomposition reaction mechanisms to simulate the chemical kinetics. It is worth noting that alike the sub-models for gas–solid momentum and energy transfer in MFM, BIOTC is free for users to incorporate their own sophisticated reaction kinetics through the CHEMKIN chemical reaction software which has already been integrated into OpenFOAM. However, because our understanding of the actual mechanisms of biomass fast pyrolysis is far from satisfactory so far, in the following, we just use a global reaction mechanism as an example to demonstrate the modeling of biomass fast pyrolysis in BIOTC.

A multi-component multi-step parallel decomposition mechanism, i.e., the modified Broido–Shafizadeh scheme [60], was chosen for simulating the chemical kinetics of biomass fast pyrolysis in reactors. In this mechanism, biomass that is initially composed of cellulose, hemicellulose, and lignin, is first decomposed into tar, gas, and biochar through two competitive reactions. In the second step, tar is cracked into gas. All reactions are modeled using first-order irreversible reaction rates based on the Arrhenius rate equation.

$$k = A\exp(-E/RT). \tag{17}$$

The schematic of the reaction paths is shown in Fig. 1. The reaction constants, i.e., pre-exponential factor (A), activation energy (E), and heat of reaction (ΔH), are listed in Table 1.

2.3. Solution procedure for multi-fluid model

The solution strategy for MFM in BIOTC is developed based on the standard compressible, two-phase flow solver in OpenFOAM, i.e., "compressibleTwoPhaseEulerFoam". Because there are more than one solid phases, the numerical solution procedure in BIOTC is more complicated than that in the original "compressibleTwoPhaseEulerFoam". Thus, the numerical treatments in BIOTC for MFM are presented in detail. In addition, because of the chemical reaction terms in all of the conservation equations, a time-split scheme [61] is used to solve the multiphase fluid dynamics and chemical reaction rates separately. In this scheme, the conservation equations without chemical reaction terms are first solved. Then some of the solved field variables, such as the density ρ, volume fraction α, species mass fraction Y, and temperature T, are used to calculated chemical reaction rates. Finally, the computed reaction terms are used to update relevant field variables.

Assuming that there are two solid phases in BIOTC, the numerical solution for the conservation equations in MFM are derived as follows. The numerical treatments for a system with more than two solid phases can be derived in a similar way. In each finite volume, as the sum of the volume fractions of all phases should be unity, only the solid phase continuity equations need to be solved. Thus, the solid phase continuity equations are discretized fully implicitly as

$$\begin{aligned}\frac{\partial \alpha_{s1}^t \rho_{s1}}{\partial t} + \nabla\cdot(\alpha_{s1}^t \rho_{s1}\boldsymbol{U}_{s1}) = 0\\ \frac{\partial \alpha_{s2}^t \rho_{s2}}{\partial t} + \nabla\cdot(\alpha_{s2}^t \rho_{s2}\boldsymbol{U}_{s2}) = 0\end{aligned} \tag{18}$$

where the superscript t means that the variable to be solved is discretized using implicit schemes. In the following, if not mentioned in specific, superscript $t-1$ and no superscript mean using the value of the previous time step and the newest obtained value, respectively. Solution of the resulted matrices can obtain the spatial distributions of α_{s1} and α_{s2} at the current time step. It is worth noting that in order to accelerate the numerical convergence while maintaining the boundness of the volume faction of each phase ($0<\alpha<1$) when high density ratio is encountered, the decomposed scheme of $\boldsymbol{U}_{sm}$ proposed by Weller [62] to solve the convection term is employed to solve Eq. (18). Detailed information about this scheme can be found in Weller [62].

For the momentum conservation equations, due to the coupling of pressure and velocity, the PIMPLE algorithm, a combination of the PISO [63] and SIMPLE [64] algorithms, is employed to decouple the correlation between pressure and velocity efficiently while maintaining numerical stability and accuracy. This algorithm requires a velocity predictor and a correction loop in which the velocity is corrected after the solution of pressure. Along with the PIMPLE algorithm, a partial implicit scheme is used to treat the interphase coupling terms. The key steps to solve momentum conservation equations are as follows. Eqs. (2) and (13) without the terms of pressure and chemical reaction for gas phase and solid phase 1 and 2 are written as

$$\begin{aligned}&\frac{\partial(\alpha_g\rho_g\boldsymbol{U}_g^t)}{\partial t} + \nabla\cdot(\alpha_g\rho_g\boldsymbol{U}_g^t\boldsymbol{U}_g^t)\\ &= \nabla\cdot\boldsymbol{\tau}_g^t + \sum_{m=1}^{M}\beta_{gsm}^{t-1}(\boldsymbol{U}_{sm}-\boldsymbol{U}_g^t) + \alpha_g\rho_g\boldsymbol{g}\\ &\frac{\partial(\alpha_{s1}\rho_{s1}\boldsymbol{U}_{s1}^t)}{\partial t} + \nabla\cdot(\alpha_{s1}\rho_{s1}\boldsymbol{U}_{s1}^t\boldsymbol{U}_{s1}^t)\\ &= \nabla\cdot\boldsymbol{\tau}_{s1}^t + \beta_{gs1}^{t-1}(\boldsymbol{U}_g-\boldsymbol{U}_{s1}^t) + \beta_{s21}^{t-1}(\boldsymbol{U}_{s2}-\boldsymbol{U}_{s1}^t) + \alpha_{s1}\rho_{s1}\boldsymbol{g}\\ &\frac{\partial(\alpha_{s2}\rho_{s12}\boldsymbol{U}_{s2}^t)}{\partial t} + \nabla\cdot(\alpha_{s2}\rho_{s2}\boldsymbol{U}_{s2}^t\boldsymbol{U}_{s2}^t)\\ &= \nabla\cdot\boldsymbol{\tau}_{s2}^t + \beta_{gs2}^{t-1}(\boldsymbol{U}_g-\boldsymbol{U}_{s2}^t) + \beta_{s21}^{t-1}(\boldsymbol{U}_{s1}-\boldsymbol{U}_{s2}^t) + \alpha_{s2}\rho_{s2}\boldsymbol{g}.\end{aligned} \tag{19}$$

The interphase momentum coefficients, β_{gsm} and β_{s21}, are precalculated using the velocities in the previous time step. Then, Eq. (19) is discretized and results in linear matrices as the following form

$$\begin{aligned}[\boldsymbol{M}_g]\,\boldsymbol{U}_g^t &= \boldsymbol{B}_g\\ [\boldsymbol{M}_{s1}]\,\boldsymbol{U}_{s1}^t &= \boldsymbol{B}_{s1}\\ [\boldsymbol{M}_{s2}]\,\boldsymbol{U}_{s2}^t &= \boldsymbol{B}_{s2}\end{aligned} \tag{20}$$

where $[\boldsymbol{M}]$ and $\boldsymbol{B}$ are the coefficient matrix and source vector, respectively. By decomposing the left hand side into diagonal and non-diagonal part, Eq. (20) is reformulated as

$$\begin{aligned}[\boldsymbol{M}_g]_{diag}\,\boldsymbol{U}_g^t &= \boldsymbol{B}_g - [\boldsymbol{M}_g]_{non\text{-}diag}\,\boldsymbol{U}_g^t\\ [\boldsymbol{M}_{s1}]_{diag}\,\boldsymbol{U}_{s1}^t &= \boldsymbol{B}_{s1} - [\boldsymbol{M}_{s1}]_{non\text{-}diag}\,\boldsymbol{U}_{s1}^t\\ [\boldsymbol{M}_{s2}]_{diag}\,\boldsymbol{U}_{s2}^t &= \boldsymbol{B}_{s2} - [\boldsymbol{M}_{s2}]_{non\text{-}diag}\,\boldsymbol{U}_{s2}^t.\end{aligned} \tag{21}$$

Table 1
Reaction rate constants in the modified Broido–Shafizadeh scheme [60].

Components	Reaction	Y	$A(s^{-1})$	E(J/mol)	ΔH (J/kg)
Cellulose	k_{1c}		2.8×10^{19}	2.424×10^5	0
	k_{2c}		3.28×10^{14}	1.965×10^5	2.55×10^5
	k_{3c}	0.35	1.3×10^{10}	1.505×10^5	2.55×10^5
Hemicellulose	k_{1h}		2.1×10^{16}	1.867×10^5	0
	k_{2h}		8.75×10^{15}	2.024×10^5	2.55×10^5
	k_{3h}	0.6	2.6×10^{11}	1.457×10^5	2.55×10^5
Lignin	k_{1l}		9.6×10^8	1.076×10^5	0
	k_{2l}		1.5×10^9	1.438×10^5	2.55×10^5
	k_{3l}	0.75	7.7×10^6	1.114×10^5	2.55×10^5
Tar	k_4		4.28×10^6	1.08×10^5	-4.2×10^4

Setting the velocity in the right hand side of Eq. (21) by its value of previous time step, we can obtain

$$\begin{aligned}
\boldsymbol{H}_g &= \boldsymbol{B}_g - \left[\boldsymbol{M}_g\right]_{non\text{-}diag} \boldsymbol{U}_g^{t-1} \\
\boldsymbol{H}_{s1} &= \boldsymbol{B}_{s1} - \left[\boldsymbol{M}_{s1}\right]_{non\text{-}diag} \boldsymbol{U}_{s1}^{t-1} \\
\boldsymbol{H}_{s2} &= \boldsymbol{B}_{s2} - \left[\boldsymbol{M}_{s2}\right]_{non\text{-}diag} \boldsymbol{U}_{s2}^{t-1}.
\end{aligned} \tag{22}$$

Thus, the predictor of velocity is obtained as

$$\begin{aligned}
\tilde{\boldsymbol{U}}_g^t &= \frac{\boldsymbol{H}_g}{\left[\boldsymbol{M}_g\right]_{diag}} \\
\tilde{\boldsymbol{U}}_{s1}^t &= \frac{\boldsymbol{H}_{s1}}{\left[\boldsymbol{M}_{s1}\right]_{diag}} \\
\tilde{\boldsymbol{U}}_{s2}^t &= \frac{\boldsymbol{H}_{s2}}{\left[\boldsymbol{M}_{s2}\right]_{diag}}.
\end{aligned} \tag{23}$$

Then, the phase velocity to be solved at the current time step can be expressed as

$$\begin{aligned}
\boldsymbol{U}_g^t &= \tilde{\boldsymbol{U}}_g^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_g\right]_{diag}} \\
\boldsymbol{U}_{s1}^t &= \tilde{\boldsymbol{U}}_{s1}^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_{s1}\right]_{diag}} \\
\boldsymbol{U}_{s2}^t &= \tilde{\boldsymbol{U}}_{s2}^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_{s2}\right]_{diag}}.
\end{aligned} \tag{24}$$

Because of the constraint that the sum of the volume fractions of all phases is unity, the following is obtained

$$\begin{aligned}
\nabla \cdot \left(\alpha_g \boldsymbol{U}_g^t + \alpha_{s1} \boldsymbol{U}_{s1}^t + \alpha_{s2} \boldsymbol{U}_{s2}^t\right) = &-\frac{\alpha_g}{\rho_g}\frac{d\rho_g}{dt} - \frac{\alpha_{s1}}{\rho_{s1}}\frac{d\rho_{s1}}{dt} \\
&- \frac{\alpha_{s2}}{\rho_{s2}}\frac{d\rho_{s2}}{dt}
\end{aligned} \tag{25}$$

Combining Eqs. (24) and (25) gives

$$\begin{aligned}
&\nabla \cdot \left[\alpha_g \left(\tilde{\boldsymbol{U}}_g^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_g\right]_{diag}}\right) + \alpha_{s1}\left(\tilde{\boldsymbol{U}}_{s1}^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_{s1}\right]_{diag}}\right)\right. \\
&\quad \left. + \alpha_{s2}\left(\tilde{\boldsymbol{U}}_{s2}^t - \frac{\nabla p^t}{\left[\boldsymbol{M}_{s2}\right]_{diag}}\right)\right] \\
&= -\frac{\alpha_g}{\rho_g}\frac{d\rho_g}{dt} - \frac{\alpha_{s1}}{\rho_{s1}}\frac{d\rho_{s1}}{dt} - \frac{\alpha_{s2}}{\rho_{s2}}\frac{d\rho_{s2}}{dt}.
\end{aligned} \tag{26}$$

Thus, by discretizing p^t implicitly and solving the resulted Laplacian equation of Eq. (26), the updated pressure field is obtained. Finally the ultimate velocity of each phase is corrected using Eq. (24).

Similarly, the energy equations are solved partially implicitly while the species equations are solved fully implicitly, e.g., for solid phase **1**

$$\begin{aligned}
&\frac{\partial(\alpha_{s1}\rho_{s1}C_{ps1}T_{s1}^t)}{\partial t} + \nabla \cdot (\alpha_{s1}\rho_{s1}C_{ps1}T_{s1}^t U_{s1}) \\
&\quad = \nabla \cdot \boldsymbol{q}_{s1}^t + h_{gs1}^{t-1}(T_g - T_{s1}^t)
\end{aligned} \tag{27}$$

$$\frac{\partial \alpha_{s1}\rho_{s1}Y_{s1k}^t}{\partial t} + \nabla \cdot (\alpha_{s1}\rho_{s1}Y_{s1k}^t \boldsymbol{U}_{s1}) = 0. \tag{28}$$

2.4. Solution procedure for reaction kinetics

The solution procedure for chemical reaction terms is more straightforward than that for the multiphase fluid dynamics. In the time-split scheme, after the first stage in obtaining the intermediate field variables from the solution of MFM, each finite volume is viewed as independent where the reaction rate equations are established. For example, the reaction rate equations for the mechanism shown in Fig. 1 are

$$\begin{aligned}
\frac{dc_{biomass}}{dt} &= -k_1 c_{biomass} \\
\frac{dc_{active-biomass}}{dt} &= k_1 c_{biomass} - (k_2 + k_3) c_{active-biomass} \\
\frac{dc_{tar}}{dt} &= k_2 c_{active-biomass} - k_4 c_{tar} \\
\frac{dc_{gas}}{dt} &= (1 - Y) k_3 c_{active-biomass} + k_4 c_{tar} \\
\frac{dc_{biochar}}{dt} &= Y k_3 c_{active-biomass}
\end{aligned} \tag{29}$$

where c is the mass concentration of species. The fully implicitly discretized equations of Eq. (29) become

$$\begin{aligned}
\frac{c_{biomass}^t - c_{biomass}^{t-1}}{\Delta t} &= -k_1 c_{biomass}^t \\
\frac{c_{active-biomass}^t - c_{active-biomass}^{t-1}}{\Delta t} &= k_1 c_{biomass}^t - (k_2 + k_3) c_{active-biomass}^t \\
\frac{c_{tar}^t - c_{tar}^{t-1}}{\Delta t} &= k_2 c_{active-biomass}^t - k_4 c_{tar}^t \\
\frac{c_{gas}^t - c_{gas}^{t-1}}{\Delta t} &= (1 - Y) k_3 c_{active-biomass}^t + k_4 c_{tar}^t \\
\frac{c_{biochar}^t - c_{biochar}^{t-1}}{\Delta t} &= Y k_3 c_{active-biomass}^t
\end{aligned} \tag{30}$$

and the resulting linear matrix equations are solved using the stiff ODE solver. A variety of ODE solvers are available in original OpenFOAM for users to select.

After the solution of the chemical reaction terms, the resulting reaction rates and heat of reactions are used to update all the

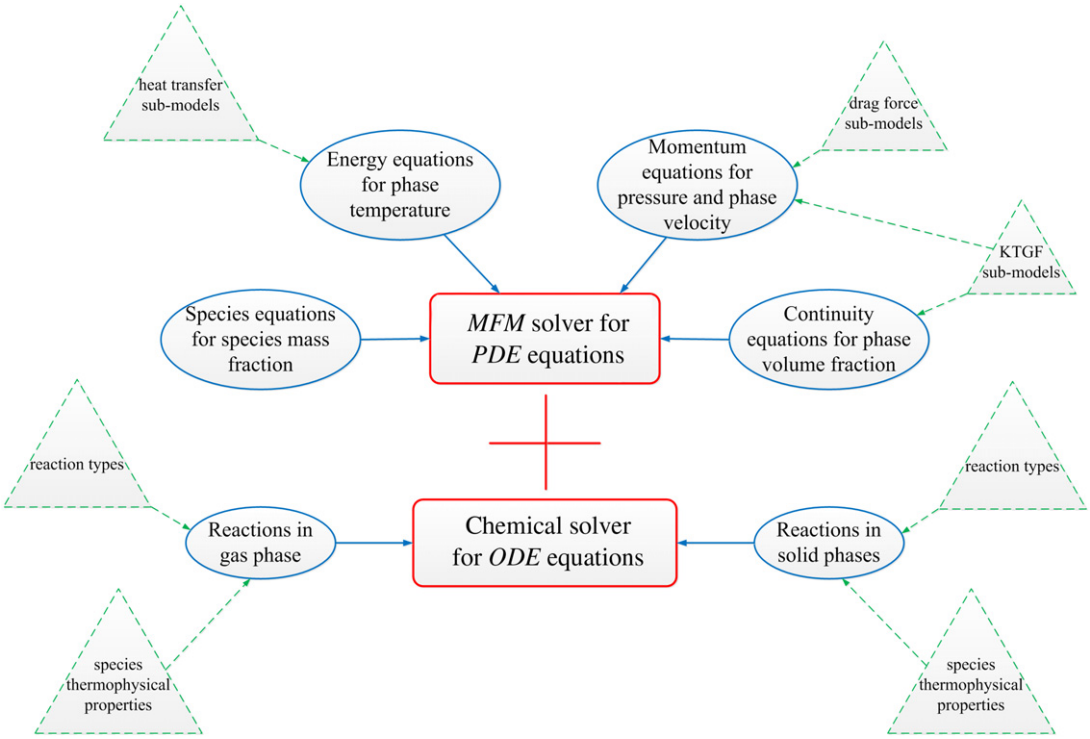

Fig. 2. Code structure of BIOTC.

related field variables to their final states within each time step as

$$\begin{aligned}
&\rho_g += \frac{\sum_{m=1}^{M} R_{gsm}}{\alpha_g}\Delta t, \qquad \rho_{sm} += \frac{R_{sm}}{\alpha_{sm}}\Delta t \\
&T_g += \frac{\sum_{m=1}^{M} \chi_{gsm} + \Delta H_g}{\alpha_g \rho_g C_{pg}}\Delta t, \qquad T_{sm} += \frac{\chi_{sm} + \Delta H_{sm}}{\alpha_{sm}\rho_{sm}C_{psm}}\Delta t \\
&Y_{gk} += \frac{R_{gk}}{\alpha_g \rho_g}\Delta t, \qquad Y_{smk} += \frac{R_{smk}}{\alpha_{sm}\rho_{sm}}\Delta t \\
&\boldsymbol{U}_g += \frac{\sum_{m=1}^{M} \boldsymbol{\psi}_{gsm}}{\alpha_g \rho_g}\Delta t, \qquad \boldsymbol{U}_{sm} += \frac{\boldsymbol{\psi}_{sm}}{\alpha_{sm}\rho_{sm}}\Delta t.
\end{aligned} \tag{31}$$

3. The structure of BIOTC

BIOTC consists of many files in numerous directories, in which each file or directory is designated for a specific purpose. The main structure of BIOTC is shown in Fig. 2. In the main directory, the file named "BIOTC-2.1.x.C" is the main program executing the time loop, in which all functional files are executed in each time step. The file "createFields.H" is for variable declaration that executes definitions and initializations of all field variables. The file "alphaEqn.H" is designed to solve the phase continuity conservation equations. The file "interPhaseCoeffs.H" is executed to calculate the interphase coupling coefficients, i.e., β_{gsm} and h_{gsm}, for the momentum and energy conservation equations. The file "UEqn.H" is to construct the momentum conservation equations for all phases and to obtain the predictor of velocity. The file "pEqn.H" is used to solve the pressure Poisson equation and to correct the velocity using the updated pressure. The file "TEqn.H" is used to construct and solve phase energy conservation equations. The file "YEqn.H" is used to construct and solve mass species conservation equations. The file "ReactUpdate.H" is to carry out the chemical solver to compute reaction rates and reaction heat release and update the field variables based on the resulting chemical reaction rates and heat.

Various directories contain sub-models for users to select. The directory "phaseModels" contains the definition of phase properties, including density, velocity, particle diameter, etc. The directory "dragModels" contains sub-models for interphase momentum exchange coefficients. Similarly, the directory "heatTransferModels" contains sub-models for interphase heat transfer coefficients. The directory "kineticTheoryModels" contains a variety of sub-models for solid phase granular stress, subdivided into categories to calculate p_{sm}, μ_{sm} and λ_{sm}, etc. The sub-models of chemical reactions, including the selection of thermophysical properties of each species, the reaction type, and reaction rate, are stored in the "solidChemistryModel" and "chemistryModel", respectively, for solid and gas phases.

4. Tutorial case

4.1. Numerical setup

A laboratory-scale bubbling fluidized-bed reactor for cellulose fast pyrolysis, located at Iowa State University [43], was simulated. The schematic of this reactor is shown in Fig. 3, which includes the bed's geometrical information. Sand was initially packed with solid volume fraction of 0.59 to the height of 0.055 m. Cellulose was continuously fed from the left-hand-side injector at a mass rate of 0.1 kg/h, while fluidization nitrogen was supplied from the bottom at a velocity of 0.36 m/s. The biomass injector was located 0.017145 m from the bottom of the reactor with inner diameter of 0.00762 m. The temperature of the sidewall was maintained constant at 773 K. The bed was initially heated to 773 K, and the temperatures of injecting biomass and nitrogen were fixed at 300 K and 773 K, respectively. The physical properties of each species in each phase are provided in Table 2, in which the particle diameters are directly measured from the experimental pre-processing and the others are adopted from [38].

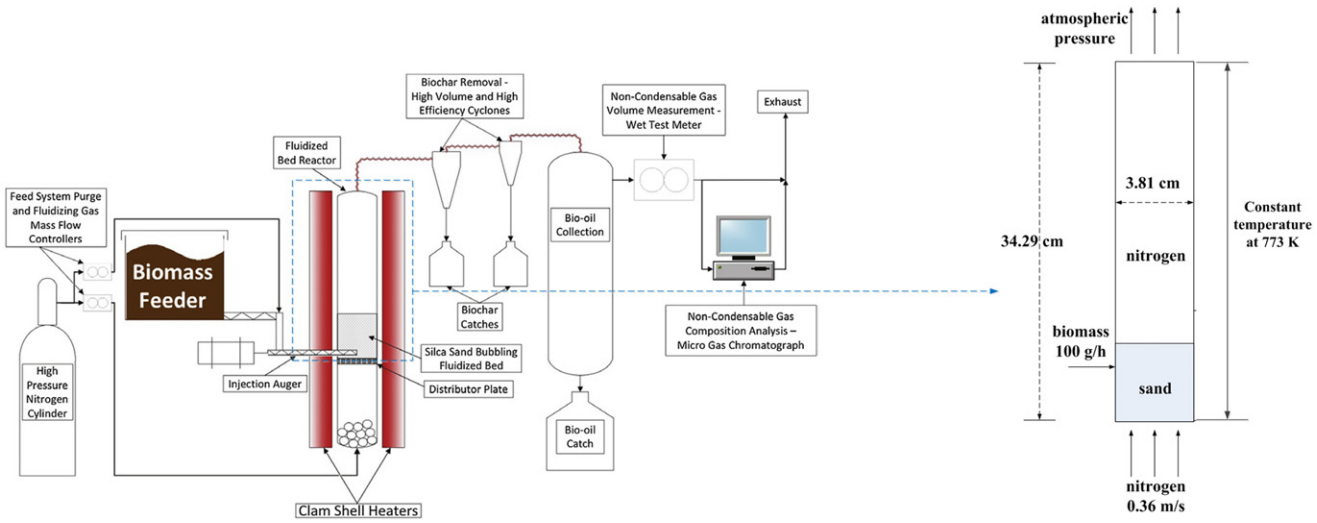

Fig. 3. The schematic of the experimental and numerical setup [43].

Table 2
Physical properties of each species in solid and gas phases [38].

Species	Density ρ (kg/m^3)	Particle diameter d_s (m)	Molecular weight W (g/mol)	Heat capacity C_p (J/kg K)	Dynamic viscosity μ (kg/ms)	Thermal conductivity κ (J/ms K)
Tar			100	2500	3×10^{-5}	2.577×10^{-2}
Gas			30	1100	3×10^{-5}	2.577×10^{-2}
N_2			28	1121	3.58×10^{-5}	5.63×10^{-2}
Cellulose	400	4×10^{-4}		2300		0.3
Biochar	2333	4×10^{-4}		1100		0.1
Sand	2649	5.2×10^{-4}		800		0.27

$$\eta_{tar} = \frac{\int_{t=50}^{t=100} \int_{outlet} (\alpha_g \rho_g \boldsymbol{U}_g Y_{tar}) \cdot d\boldsymbol{A} dt}{\int_{t=50}^{t=100} \int_{outlet} [\alpha_g \rho_g \boldsymbol{U}_g (Y_{tar} + Y_{gas}) + \alpha_{biomass} \rho_{biomass} \boldsymbol{U}_{biomass} (Y_{biochar} + Y_{unreacted-biomass})] \cdot d\boldsymbol{A} dt} \tag{33}$$

Box I.

The devolatilization scheme described in Fig. 1 and Table 1 was used to model the fast pyrolysis of cellulose. Based on the following formula, the minimum fluidization velocity was calculated

$$U_{mf} = \frac{d_s^2}{150\mu_g} g(\rho_s - \rho_g) \frac{\alpha_{mf}^3}{1 - \alpha_{mf}} \tag{32}$$

where α_{mf} is the minimum gas phase volume fraction for bubbling fluidization, and d_s is the diameter of sand particle. Under the current conditions, U_{mf} is about 0.14 m/s. Thus, the supplied nitrogen is able to fluidize the bed. The domain was discretized into 20×180 grid points and the time step was chosen to be 3×10^{-4} s. The simulation was run 100 s and the last 50 s was used for statistical analysis.

4.2. Simulation results

The simulated statistically steady-state solid phase volume fractions at different times are shown in Fig. 4. It can be seen that because of the high density of sand, the volume fraction of gas phase at the bottom of the reactor is low and some bubbles are formed dynamically. These results are in quantitative agreement with the simulations of similar biomass fast pyrolysis reactors using MFIX. [37,42]. The present results demonstrate that BIOTC is able to reproduce the in-bed hydrodynamics of biomass fast pyrolysis reactors.

The product distributions at $t = 100$ s are shown in Fig. 5. It can be seen that because of the quick devolatilization of cellulose, the unreacted cellulose particles exist only near the injector. Tar and gas are blown by the nitrogen towards the outlet of the reactor, resulting in higher concentration at the upper part of the reactor.

The product yields are calculated as follows. During the last 50 s, all the species exiting the reactor are collected and a relative mass fraction is defined for the species yield. For example, the product yield of tar, η_{tar}, is computed as Eq. (33) which is given in Box I.

The comparison of the product yields from experiments and simulation are shown in Table 3. It can be seen that the predicted results match experimental data very well, especially for tar. Therefore, it can be concluded that BIOTC, using combinations of MFM and global reaction kinetics, can predict reasonably well the product yields of biomass fast pyrolysis.

5. Conclusion

An open-source code, BIOTC, designed to simulate biomass fast pyrolysis at reactor scale, was presented. This code was developed within the framework of OpenFOAM, which provides users freedom to extend this code by inserting their own sub-models, while still maintaining user-friendly interface. The mathematical formulations of the governing equations and their solution strategies were described. The code structure, including the file organization and subroutines for sub-models, were shown. A laboratory-scale bubbling fluidized bed reactor for biomass fast pyrolysis was simulated as a tutorial case to demonstrate the capability of this code. With appropriate modifications, BIOTC can be applied to simulate gasification and combustion of biomass at the reactor scale.

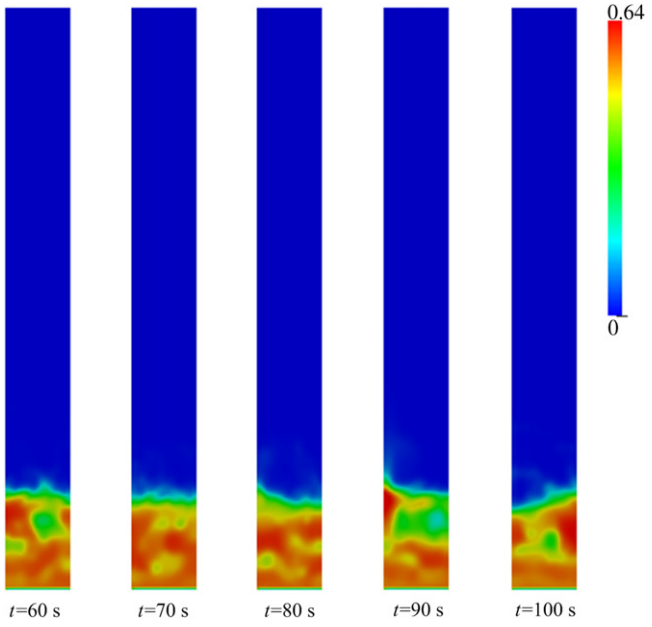

Fig. 4. Instantaneous distribution of solid phase volume fraction at statistically steady state.

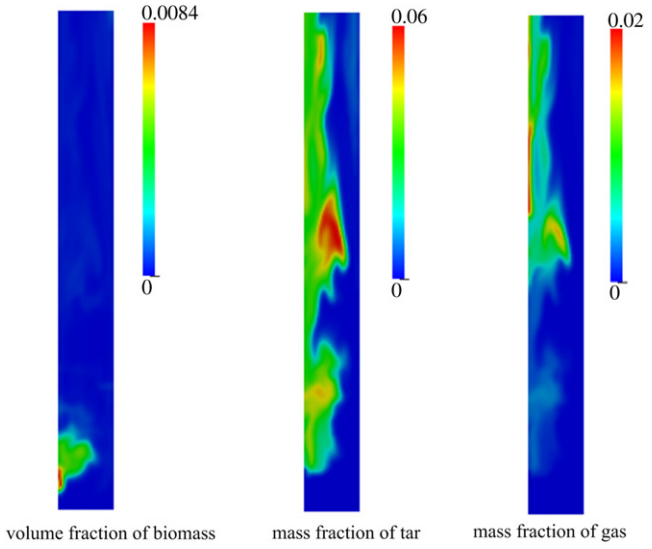

Fig. 5. Product distributions at t =100 s.

Table 3
Comparison of the product yields (%wt) between experiment and BIOTC.

Method	Tar	Biochar	Gas	Unreacted biomass
Experiment	82.1	2.2	12.4	–
BIOTC	79.0	2.4	17.9	0.7

Acknowledgment

This work was supported by the National Science Foundation under Grant number EPS-1101284.

References

[1] B. Babu, A. Chaurasia, Chem. Eng. Sci. 59 (2004) 1999–2012.
[2] O. Boutin, M. Ferrer, J. Lédé, Chem. Eng. Sci. 57 (2002) 15–25.
[3] E. Theuns, B. Merci, J. Vierendeels, P. Vandevelde, Fire Mater. 29 (2005) 195–212.
[4] W.C. Park, A. Atreya, H.R. Baum, Combust. Flame 157 (2010) 481–494.
[5] H. Lu, E. Ip, J. Scott, P. Foster, M. Vickers, L.L. Baxter, Fuel 89 (2010) 1156–1168.
[6] B. Peters, C. Bruch, J. Anal. Appl. Pyrolysis 70 (2003) 233–250.
[7] A. Gomez-Barea, S. Nilsson, F. Vidal Barrero, M. Campoy, Fuel Process. Technol. 91 (2010) 1624–1633.
[8] I.P. Boukis, P. Grammelis, S. Bezergianni, A. Bridgwater, Fuel 86 (2007) 1372–1386.
[9] P. Kaushal, J. Abedi, J. Ind. Eng. Chem. 16 (2010) 748–755.
[10] P. Baggio, M. Baratieri, L. Fiori, M. Grigiante, D. Avi, P. Tosi, Energy Convers. Manage. 50 (2009) 1426–1435.
[11] K. Ng, E. Ng, Chem. Eng. Sci. 104 (2013) 960–974.
[12] L. Pakzad, F. Ein-Mozaffari, S.R. Upreti, A. Lohi, Chem. Eng. Sci. 101 (2013) 642–654.
[13] J. Bruchmuller, B. van Wachem, S. Gu, K. Luo, R. Brown, AIChE J. 58 (2011) 3030–3042.
[14] J. Bruchmüller, K. Luo, B. van Wachem, Proc. Combust. Inst. 34 (2012) 2373–2381.
[15] A. Anca-Couce, N. Zobel, H.A. Jakobsen, Fuel 103 (2013) 773–782.
[16] P. Lamarche, M. Tazerout, F. Gelix, S. Köhler, K. Mati, F. Paviet, Fuel 106 (2012) 118–128.
[17] M. Simone, E. Biagini, C. Galletti, L. Tognotti, Fuel 88 (2009) 1818–1827.
[18] R. Vi ̂jeu, L. Gerun, M. Tazerout, C. Castelain, J. Bellettre, Fuel 87 (2008) 3292–3303.
[19] C. Dupont, L. Chen, J. Cances, J.-M. Commandre, A. Cuoci, S. Pierucci, E. Ranzi, J. Anal. Appl. Pyrolysis 85 (2009) 260–267.
[20] K. Papadikis, S. Gu, A.V. Bridgwater, Chem. Eng. J. 149 (2009) 417–427.
[21] Z. Luo, S. Wang, K. Cen, Renew. Energy 30 (2005) 377–392.
[22] S. Sun, H. Tian, Y. Zhao, R. Sun, H. Zhou, Bioresour. Technol. 101 (2010) 3678–3684.
[23] Y.B. Yang, A.N. Phan, C. Ryu, V. Sharifi, J. Swithenbank, Fuel 86 (2007) 169–180.
[24] M. Christodoulou, G. Mauviel, J. Lédé, P. Beaurain, M. Weber, H. Legall, F. Billaud, J. Anal. Appl. Pyrolysis 103 (2012) 255–260.
[25] K. Luo, A. Wei, Z. Wang, J. Fan, Powder Technol. 245 (2013) 115–125.
[26] Q. Xiong, B. Li, F. Chen, J. Ma, W. Ge, J. Li, Chem. Eng. Sci. 65 (2010) 5356–5365.
[27] Q. Xiong, B. Li, J. Xu, X. Wang, L. Wang, W. Ge, Comput. & Fluids 70 (2012) 86–94.
[28] Q. Xiong, B. Li, G. Zhou, X. Fang, J. Xu, J. Wang, X. He, X. Wang, L. Wang, W. Ge, J. Li, Chem. Eng. Sci. 71 (2012) 422–430.
[29] Q. Xiong, B. Li, J. Xu, X. Fang, X. Wang, L. Wang, X. He, W. Ge, Chin. Sci. Bull. 57 (2012) 707–715.
[30] Y. Xu, J. Padding, M. van der Hoef, J. Kuipers, Chem. Eng. Sci. 104 (2013) 201–207.
[31] T. Li, P. Gopalakrishnan, R. Garg, M. Shahnam, Particuology 10 (2012) 532–541.
[32] S. Wang, Y. Shen, Y. Ma, J. Gao, X. Lan, Q. Dong, Powder Technol. 245 (2013) 314–323.
[33] X. Sun, M. Sakai, Y. Yamada, J. Comput. Phys. 248 (2013) 147–176.
[34] S.H. Hosseini, G. Ahmadi, M. Olazar, Powder Technol. 246 (2013) 303–316.
[35] Q. Wang, J. Xiao, H. Liu, Powder Technol. 249 (2013) 339–352.
[36] Y. Zhang, Y. Bo, Y. Wu, X. Wu, Z. Huang, J. Zhou, K. Cen, Particuology (2013) in press.
[37] D. Lathouwers, J. Bellan, Int. J. Multiph. Flow 27 (2001) 2155–2187.
[38] D. Lathouwers, J. Bellan, Energy Fuel 15 (2001) 1247–1262.
[39] H. Zhang, S. Shao, R. Xiao, Q. Pan, R. Chen, J. Zhang, Energy Fuel 25 (2011) 4077–4084.
[40] H.S. Choi, Y.S. Choi, S.J. Kim, Environ. Prog. Sustainable Energy 28 (2009) 418–426.
[41] R.W. Ashcraft, G.J. Heynderickx, G.B. Marin, Chem. Eng. J. 207 (2012) 195–208.
[42] Q. Xue, T. Heindel, R. Fox, Chem. Eng. Sci. 66 (2011) 2440–2452.
[43] Q. Xue, D. Dalluge, T. Heindel, R. Fox, R. Brown, Fuel 97 (2012) 757–769.
[44] P. Mellin, Q. Zhang, E. Kantarelis, W. Yang, Appl. Therm. Eng. 58 (2013) 344–353.
[45] P. Mellin, E. Kantarelis, W. Yang, Fuel 117 (2013) 704–715.
[46] A. Boateng, P. Mtui, Appl. Therm. Eng. 33 (2012) 190–198.
[47] OpenCFD, OpenFOAM: The Open Source CFD Toolbox User Guide, 2013.
[48] I.A. Cosden, J.R. Lukes, Comput. Phys. Commun. 184 (2013) 1958–1965.
[49] V. Novaresio, M. García-Camprubí, S. Izquierdo, P. Asinari, N. Fueyo, Comput. Phys. Commun. 183 (2012) 125–146.
[50] Q. Xiong, S.-C. Kong, A. Passalacqua, Chem. Eng. Sci. 99 (2013) 305–313.
[51] Q. Xiong, S. Aramideh, S.-C. Kong, Energy Fuel 27 (2013) 5948–5956.
[52] Q. Xiong, S.-C. Kong, S. Aramideh, Environ. Prog. Sustainable Energy (2014) in press.
[53] D. Gidaspow, Multiphase Flow and Fluidization: Continuum and Kinetic Theory Descriptions, Elsevier, 1994. Access Online.
[54] M. Syamlal, W. Rogers, T.J. O'Brien, Technical Note, DOE/METC-94/1004, NTIS/DE94000087, National Technical Information Service, Springfield, VA, 1993.
[55] B. Lu, W. Wang, J. Li, Chem. Eng. Sci. 64 (2009) 3437–3447.
[56] W. Ranz, W. Marshall, Chem. Eng. Prog. 48 (1952) 141–146.
[57] D. Gunn, Int. J. Heat Mass Transfer 21 (1978) 467–476.
[58] J. Li, D. Mason, Powder Technol. 112 (2000) 273–282.
[59] C.K.K. Lun, S.B. Savage, D.J. Jeffrey, N. Chepurniy, J. Fluid Mech. 140 (1984) 223–256.
[60] R. Miller, J. Bellan, Combust. Sci. Technol. 126 (1997) 97–137.
[61] N. Xie, F. Battaglia, R.O. Fox, Combust. Theory Model. 8 (2004) 195–209.
[62] H.G. Weller, Deviation, Modelling and Solution of the Conditionally Averaged Two-Phase Flow Equations, OpenCFD, Ltd., 2005.
[63] R.I. Issa, J. Comput. Phys. 62 (1986) 40–65.
[64] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Taylor & Francis, 1980.