Simulating Runoff along Soil Slope Using Smoothed Particle Hydrodynamics Method and Comparison with Standard Empirical Formula

L.M. Dakssa
I.S.H. Harahap
Universiti Teknologi PETRONAS (UTP), Tronoh, Perak, Malaysia

Abstract - **Fast-moving water along soil slopes causes severe hydraulic scour and can also wash away slope-protective covers, leading to soil slope instability. In this paper, the application of smoothed particle hydrodynamics (SPH) – a meshfree numerical method - to triggering surface runoff along a soil slope was investigated in view of predicting fast-moving landslides such as debris flows and avalanche. SPH source code of the flow governing equation, in FORTRAN language, was developed to run the simulation. Results and snapshots of the simulation suggest that the SPH scheme effectively captures the important aspects of the surface runoff phenomenon. Comparison with the results obtained using the standard open channel hydraulics equation of Manning's shows the SPH method predicts the velocity of flow reasonably.**

I. INTRODUCTION

During rainfall events, part of the rainwater reaching the earth's surface enters the soil as infiltration, while the other part flows down the slope as surface runoff (i.e. neglecting interception, evaporation, and other losses). The amount of rainwater that goes as infiltration and as runoff depends on the infiltration rate of the soil. Surface runoff along soil slopes, besides creating hydraulic scour, may also modify soil slope's land-use by removing soil slope-protective covers.

Conventionally, mesh-based numerical techniques, such as finite difference method (FDM) and finite element method (FEM), have been used in most geotechnical investigations of civil engineering works. However, these numerical methods (i.e. FDM and FEM) have inherent difficulties, especially, in areas of large deformations because of mesh-distortion in the case of FEM and because of inefficient use of regular grids for irregular geometries in the case of FDM [1]. In the current research, the objective is, therefore, to explore the applicability of the smoothed particle hydrodynamics (SPH) – one of the meshfree (also referred to as meshless) methods – to triggering surface runoff down a saturated soil slope. SPH was, originally invented for simulating astrophysical phenomena and later, its applications to several fields of science and engineering have been reported in the literature (for instance, [1], [2]; see also section IV).

In terms of organization, the present paper deals with brief background information on the flow, SPH formulation of the problem, followed by a numerical example and concluding remarks.

II. GOVERNING EQUATION

In the current research, fluid flowing along soil slope is assumed incompressible and Newtonian, wherein the popular Navier-Stokes (NS) equation can be considered to govern the flow conditions. In compact form the NS equation is given by

$$\rho \frac{D\mathbf{V}}{Dt} = -\nabla P + \mu \nabla^2 \mathbf{V} + \mathbf{F}_b \tag{1}$$

where,
ρ = density
V = velocity vector
P = pressure
F_b = vector of body forces
μ = is the dynamic viscosity
t = time

III. A BRIEF BACKROUND ON SPH

Smoothed particle hydrodynamics (SPH) is a macroscopic numerical approach initially developed for simulating astrophysical phenomena, in 1977 [1]. SPH meshless numerical method is based on integral interpolation theory and two essential concepts dictate its formulation, namely *kernel approximation* and *particle approximation.*

A. Kernel Approximation

Kernel approximation is the technique of approximating the values of both the field function and the derivative of the field function. To do so, the field function needs to be represented in integral form. This is achieved by multiplying an arbitrary field functions with a smoothing kernel function. Therefore, a function *A* of a three-dimensional position vector *x* (or an estimate of the function *A(x)* at location *x'*) can be expressed in integral form

$$A(x) = \int_{\Omega} A(x')\delta(x - x')dx' \tag{2}$$

where, $\delta(x - x')$ is the Dirac delta function, given by

978-1-4673-5968-9/13/$31.00 ©2013 IEEE

$$\delta(x-x') = \begin{cases} 1 & for\ x = x' \\ 0 & elsewhere \end{cases} \quad (3)$$

where, Ω is the volume of the integral containing $\boldsymbol{x}$; x' is a new independent variable.

In the above expression, the function $A(\boldsymbol{x})$ is exact or rigorous, so long as the Dirac delta function is used and $A(\boldsymbol{x})$ is continuous in Ω. In SPH, however, the Dirac delta function needs to be replaced by the smoothing (weighting) function $W(x-x',h)$ in which case it will become an approximate representation of $A(\boldsymbol{x})$. The SPH form of a function approximation (or kernel approximation) is, therefore

$$A(x) = \int_{\Omega} A(x')W(x-x',h)dx' \quad (4)$$

where, W is called the kernel or smoothing function; h is the smoothing length, and it demarcates the influence area of the smoothing function.

It needs to be noted, however, that (4) gives an approximate representation of the integral of a field function as long as W is not a Dirac delta function; and, hence, the name kernel approximation.

Kernel approximation of derivative of a function, on the other hand, refers essentially to approximating the values of the gradient and/or the divergence of the field function. As equations of computational fluid dynamics problems are mostly partial differential equations (PDEs) of second degree [1], an appropriate approximation of the function derivatives is of profound importance. Accordingly, in SPH method, such approximations are usually performed by simply replacing the function A(x) in (4) by $\nabla A(x)$, for gradient and, $\nabla.\mathrm{A(x)}$ for the divergence. More specifically, the kernel approximation of the divergence of the field function A(x) (for vector quantity) is

$$\nabla \cdot A(x) = \int_{\Omega} \nabla \cdot A(x')W(x-x',h)dx' \quad (5)$$

After applying the divergence theorem, it is always the case that the divergence operation on the primed coordinate in (5) is transferred to the gradient of the smoothing function in SPH numerical approach, which entails re-writing (5) as

$$\nabla \cdot A(x) = -\int_{\Omega} A(x') \cdot \nabla W(x-x',h)dx' \quad (6)$$

Note that a *dot product* is used in (5). Similarly, the *gradient* of the function (for scalar quantity) is expressed as

$$\nabla A(x) = -\int_{\Omega} A(x')\nabla W(x-x',h)dx' \quad (7)$$

It can be said, therefore, that the spatial derivative of a field function can be evaluated from the values of the field function and the spatial derivative of the smoothing function. It should also be noted that the negative sign outside the integral sign in (6) and (7) can be removed if the spatial derivative of the kernel function is taken with respect to **x** instead of the primed **x** (i.e., x').

B. Particle Approximation

Similarly, particle approximation consists in approximating the field function and its spatial derivatives (gradient and divergence). Particle approximation is another key operation in SPH numerical formulation and is the means of transforming the continuous kernel approximation (in integral form) into the summation over all particles at the discrete points in the support domain. Particles carry mass (m), velocity (**v**), and other quantities specific to the given problem. The points are regarded as interpolation points, analogous to the grid nodes in mesh-based numerical methods. Therefore, equations that govern the evolution of fluid quantities are expressed as summation interpolants with the help of smoothing function. Equation (4) can be approximated in a summation form as

$$A(x) \approx \sum_{j=1}^{N} A(x_j)W(x-x_j,h)\frac{m_j}{\rho_j} \quad (8)$$

where N is the total number of particles in the support domain; m_j and ρ_j are the mass and density of particle j, respectively. And also, it should be noted that the infinitesimal volume d**x'** is replaced by the finite volume $\Delta V = m_j / \rho_j$.

From (8), it is possible to infer that the approximate value of a function at any discrete point can be obtained using the weighted average of those values of the function at all other particles in the influence domain of that particle. Following similar argument, the particle approximation for a function at particle (point) i may be written as in (9) [1], [2]

$$A(x_i) = \sum_{j=1}^{N} \frac{m_j}{\rho_j} A(x_j)W(x_i - x_j, h) \quad (9)$$

Particle approximation of the gradient and divergence of a given field function can be formulated following similar fashion. Transformation of the PDEs to the SPH discretized summation form, for instance, can be achieved by different ways. One way is with the help of integration by parts and Taylor series expansion. Suppose A is a scalar field function representing any physical variable and is defined in a given domain of interest. Its gradient can be formulated in a similar manner to (9) as

$$(\nabla A)_i = \sum_{j=1}^{N} \frac{m_j}{\rho_j} A_j \nabla_i W_{ij} \quad (10)$$

Applying some basics of vector calculus, other forms of the gradient equations can also be formulated. For instance, putting ρ inside the gradient operator and applying the chain rule, (11) can be obtained. The introduction of mass and density into SPH particle approximation is to facilitate numerical approaches in hydrodynamics problems where density is a key parameter [1].

$$(\rho\nabla A)_i = \left(\sum_{j=1}^{N} m_j (A_i - A_j)\nabla_i W_{ij}\right) \quad (11)$$

And, the SPH form of the pressure term in the NS equation can be expressed as (for better insight, see [1] or [3]).

$$\left(\frac{1}{\rho}\nabla A\right)_i = \sum_{j=1}^{N} m_j \left(\frac{A_i}{\rho_i^2} + \frac{A_j}{\rho_j^2}\right)\nabla_i W_{ij} \quad (12)$$

C. Smoothing Functions

In the preceding sections, it was indicated that SPH numerical method employs the theory of interpolation as its foundation. Smoothing (or weighting) function is, therefore, at the core of the SPH formulation. Spatial discretization of field variables is based on a set of points (particles, in SPH nomenclature), instead of grid nodes, which are commonly used in mesh-based numerical methods such as FDM and FEM methods. Thus, it is with the help of kernel interpolation that field variables such as velocity, pressure, density, stress, etc., are approximated at any points (i.e., at any discrete points) in the support domain. Accordingly, several kernel functions are being used in SPH numerical method. The use of the piecewise cubic-spline function, commonly known as the *B-spline*, suggested by Monaghan and Lattanzio (as cited in [1]), is popular among SPH numerical modelers. In the current work, however, a more or less similar cubic spline function, effectively applied to different modes of hydrodynamics conditions as in [4], is chosen (see (13)). The same function (with slight variation in equality signs) was used in adaptive smoothed particle hydrodynamics (ASPH) in [1].

$$W(r,h) = \alpha_d \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3; & 0 \le q \le 1 \\ \frac{(2-q)^3}{4} & ; 1 \le q \le 2 \\ 0 & ; q \ge 2 \end{cases} \quad (13)$$

where,

$q = \frac{|x_i - x_j|}{h} = \frac{r}{h}$ is the relative distance between the two points, x_i and x_j, $|x_i - x_j| = r$ is the distance between the two particles. And, α_d is given by $\frac{10}{7\pi h^2}$ and $\frac{1}{\pi h^3}$ for two and three dimensions, respectively.

The glaring shortcoming of spline functions is that their second derivative is a piecewise linear function, and, therefore, the stability properties can be inferior to those of smoother kernels [1=[]. This could, probably, be one of the reasons why the spatial first derivative of the cubic spline smoothing functions is widely used in the emerging literature, instead of the second derivative. The spatial first derivative of *W(q)* for a two-dimensional case, thus, is given by

$$\nabla W(r,h) = \frac{10}{7\pi h^3} \begin{cases} -3q + \frac{9}{4}q^2; & 0 \le q \le 1 \\ \frac{-3}{4}(2-q)^2; & 1 \le q \le 2 \\ 0 & ; \quad q \ge 2 \end{cases} \quad (14)$$

IV. Application to Runoff Simulation

The Navier-Stokes (NS) equation (sometimes also referred to as continuity and momentum equations for Newtonian fluids) in its SPH form was employed for modeling free surface flows [5]. The same was also widely used to model flow through porous media by several researchers, for instance, [6], [7], [8], and [9]. The current study, also, seeks to model surface runoff down a soil slope by solving the same NS equation in its Lagrangian form.

It is worth noting that SPH was originally invented for modeling flows of compressible fluids and, therefore, its application to incompressible fluid flows needs some treatment to ensure density variation within a certain limit is maintained. In order to circumvent the difficulty of solving the pressure term for incompressible fluids, previous research works have resorted to using equation of state (EoS) as described in the next section.

A. Equation of State

For the standard SPH, for compressible fluids, particle motion is triggered by pressure gradient, which is normally calculated using equation of state. However, for the case of incompressible fluids, applying and solving the pressure using an incompressible fluid EoS dictates the adoption of small timestep [1]. This constraint has led to the adoption of artificial compressibility for solving the pressure gradient in the governing equation, the approach that is dubbed quasi-incompressibility by some researchers. Accordingly, Monaghan [5] modified the EoS suggested for water by Batchelor (as cited in [5]) for describing sound waves and used it for simulating free surface flows. The same equation has been frequently used in emerging literature. In this research, too, the same EoS is used as given in (15). Moreover, [10] applied the same EoS in their formulation of SPH for soil mechanics with successes.

$$P = B\left[\left(\frac{\rho}{\rho_o}\right)^{\gamma} - 1\right] \quad (15)$$

where, γ is constant and is taken to be a unity for incompressible low Reynolds number fluid flows [11] but 7 for other circumstances [5]; ρ_o is the reference density; B is a problem dependent parameter for limiting the maximum density gradient and, in most cases, can be taken as the initial pressure [1], [11]. This paper assumes $\gamma = 1.0$

B. Viscous Term in the NS Equations

The Laplacian for calculating the viscous term in the fundamental NS equation (see (1)) has been dealt with differently by different researchers. Ref. [5], for instance, employed artificial viscosity. In [11] a new equation was developed, which is quite similar to the expression used in [12]. This paper chooses the expression used in [4], which is given as in (16) for calculating the viscous term at particle *i*.

$$\left(\nu\nabla^2\mathbf{v}\right)_i = 2\nu\sum\left(\frac{r_{ij}\cdot\nabla_i W_{ij}}{\bar{\rho}_{ij}\left|r_{ij}\right|^2}\right)V_{ij} \tag{16}$$

where,

v is the kinematic viscosity, r_{ij} is the distance between the *i* and *j* particles, v_{ij} is the velocity difference between particle *i* and particle *j*, and W_{ij} is the smoothing length.

C. Boundary Treatment

Boundary treatment entails special consideration in SPH, as particle deficiency near or on the boundary impairs full exploitation of the scheme. Ref. [5] suggested the use of ghost particles near or on the boundaries so that high repulsive force is created to prevent fluid particles from unphysically penetrating a solid boundary. Such penalty force approach to prevent interior fluid particles from penetrating the boundary is based on the Lennard-Jones molecular force approach. Another approach, in which the Hertzian contact theory is used, was also developed as in [10]. For the current research, we intend to use the approach in [5] as given in (17).

$$PB_{ij} = \begin{cases} D\left[\left(\dfrac{r_o}{r_{ij}}\right)^a - \left(\dfrac{r_o}{r_{ij}}\right)^b\right]\dfrac{x_{ij}}{r_{ij}^2}; & \left(\dfrac{r_o}{r_{ij}}\right)\le 1 \\ 0 & ;\ elsewhere \end{cases} \tag{17}$$

where, as in [1], *a* & *b* are taken to be 12 and 4, respectively, although [5] proposed 4 and 2, respectively, with the conditions that a > b, always. Ref. [5] also suggested that *a* & *b* could also be taken as 12 and 6, respectively, without significant changes in the results. *D* is a problem dependent parameter and is usually taken to be the square of the largest velocity [1], and r_o is selected to be approximately equal to the initial particle spacing.

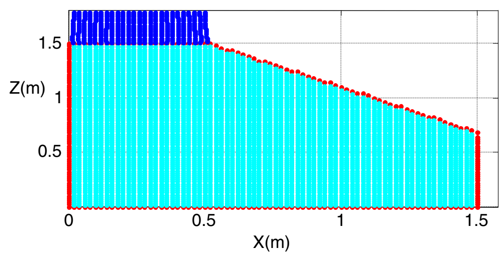
D. Time Integration

There are two major types of numerical integration algorithms - explicit methods and implicit methods. In this paper Predictor-Corrector algorithm is used. In using the predictor-corrector method attempt is made to combine the best aspects of the two methods. The predictor-corrector algorithm consists of a predictor step and a corrector step in each interval. The predictor step predicts a new value, and the corrector step improves the accuracy of that value. The predictor step is undertaken only once while the corrector step is continued until the required level of accuracy is reached. There are several predictor-corrector methods though for the current paper we stick to the Euler predictor-corrector method (some prefer to call it modified Euler method). The procedure in applying the Euler predictor-corrector method is given as follows.

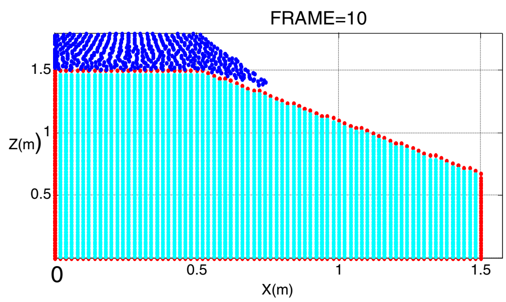
$$\begin{aligned}\phi^{(n+1)^*} &= \phi^n + f\left(t_n,\phi^n\right)\Delta t \\ \phi^{(n+1)} &= \phi^n + \frac{1}{2}\left[f\left(t_n,\phi^n\right) + f\left(t_{n+1},\phi^{(n+1)^*}\right)\right]\end{aligned} \tag{18}$$

For the sake of stability, the timestep, Δt, needs to be checked against several stability requirements. Detailed reading regarding these stability conditions can be made in [2], [12].

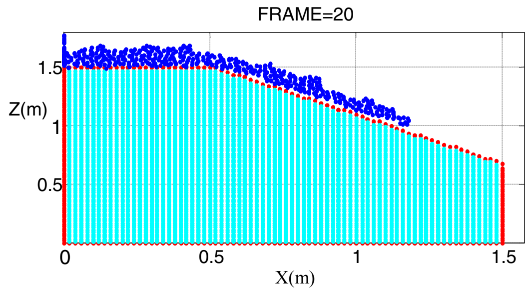
After the above theoretical discussions, the following sections deal with a practical problem of simulating runoff along a soil slope. A numerical example of triggering surface runoff along a soil slope is given in the figures below (see Fig. 1). The numerical formulation was developed based on the concept adopted in simulating free surface flows as in [4] and [5]. In this example, boundary particles were generated to form geometric boundaries for the computation. All the soil boundaries (left, top, right, and bottom) were rendered impenetrable to the moving water particles to avoid rainwater infiltration. A square geometry of 1.5 m x 1.5 m was selected in which soil, having slope of 40 degrees, was placed (because it is a negative slope,-40 degrees is used in the calculations). Fig. 1(a) represents setting the initial conditions.



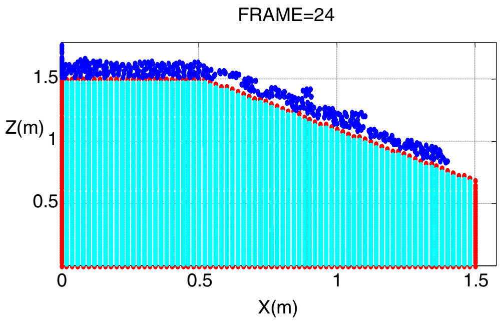
(a)



(b)



(c)



(d)

Fig. 1. Runoff simulation snapshots

A total of 5347 particles consisting of 4747 soil and 600 water particles were used in the simulation. The blue color in the above figures (Fig. 1) represents the moving water particles; the cyan color represents soil particles, and the red, the impenetrable soil particles, which, in turn, represent geometric boundaries of the slope under investigation. Fig. 1(a) depicts the initial conditions in which rainwater of 480 mm depth was placed on the crest of the slope. Initial water particles were placed only on the crest of the slope purely in the interest of simplicity. Figs. 1(b) to (d) represent runoff simulation at successive timesteps for a simulation time of 0.5 seconds. SPH codes were developed in FORTRAN language for the simulation. The figures are self-explanatory in that water particles moving down a soil slope are depicted representing surface runoff. Such analysis maybe utilized in predicting downslope flow velocity and pressure of fast-moving landslides such as debris flows and avalanche. Calculation of flow parameters such as flow velocities, pressures, and others were also undertaken as part of the overall investigation.

In numerical modeling and simulation, verification of the results is vitally important. In the present study, the effectiveness of the SPH method was assessed by comparing the SPH velocity values with the velocity value calculated using one of the standard open channel hydraulics equations. In open channel flows, average discharge and velocity are normally computed using the Manning's empirical equation (see (19)), which is used, prominently, in urban drainage designs [13], [14].

$$V = \frac{1}{\eta} R^{2/3} S^{1/2} \tag{19}$$

where, **V** is the average velocity of flow (m/sec), S is the gradient of the soil slope (m/m), R is the hydraulic radius (m), and η is the Manning's roughness coefficient (dimensionless).

Manning's coefficient of channel roughness is the value accounting for flow retardation due to vegetations and other obstructions on the soil slope. The values of η play an important role in velocity computation. Accordingly, the minimum and the maximum η values were sought so that the minimum and the maximum velocity values would be calculated. Ref. [15] lists different open channels along with their respective η values in which the minimum η value ranges from 0.011 (for concrete lined channel with trowel finish) to 0.110 (for natural streams/floodplains with dense willows). Given the section, taking the average of the two η values (i.e. 0.061), and plugging it into Manning's equation, the average velocity can be obtained. The average velocity for the current section (width of 0.5 m, water depth of 0.48 m) is, therefore, about 4.50 m/s. This maximum average velocity was compared with the maximum velocity obtained using the SPH simulation for the time step of 0.02083 s as given in the table below.

TABLE I.
MAXIMUM VELOCITIES DURING EACH TIME STEP BY THE SPH SCHEME.

Iteration	**Max. velocity in x-direction, m/s**	**Max. velocity in z-direction, m/s**
1	0.74	0.19
2	1.02	0.39
3	1.16	0.60
4	1.23	0.81
5	1.56	1.01
6	1.77	1.52
7	1.96	1.35
8	1.96	1.69
9	2.16	2.17
10	2.75	2.00
11	2.61	2.31
12	2.95	2.85
13	3.35	2.72
14	3.16	3.09
15	3.30	3.06
16	3.28	3.24
17	3.62	3.55
18	3.75	3.75
19	3.53	4.44
20	3.63	4.30
21	4.92	4.39
22	5.33	4.06
23	4.29	4.03
24	5.36	5.40
Ave.	**2.89**	**2.62**
Max. ave. V	**3.90**	

Average maximum velocity can be calculated as:

$\mathbf{V}^2 = (2.89)^2 + (-2.62)^2 \Rightarrow \mathbf{V = 3.90\ m/s}$

Comparing the two average maximum velocities obtained by the Manning's empirical equation (4.50 m/s) and the SPH approach (3.90 m/s), the authors are of the opinion that the SPH approach reasonably predicted the velocity of the flow.

V. CONCLUDING REMARKS

The present paper highlights that the Navier-Stokes equation can be effectively used to simulate surface runoff along a soil slope. SPH is capable of capturing the deformations that the flowing water undergoes, which has not been an easy task in FDM and FEM, due to mesh-distortions and other computational related difficulties. Important flow parameters, such as flow velocity and pressure can be easily calculated considering each particle in the computational domain. The boundaries can be made either penetrable (as in the case of porous media) or impenetrable (as in the case of hard stratum\rock). As such, in the present research, the top of the soil slope was rendered impenetrable to water particles so that rainwater infiltration could be prevented. The velocity calculations demonstrate that the average flow velocity obtained by the SPH method is in fair agreement with the average velocity obtained by the Manning's equation. We acknowledge the need for experimental validation of the results. That aspect of the research is ongoing.

ACKNOWLEDGMENTS

The first author would like to acknowledge the financial assistance received from Universiti Teknologi PETRONAS (UTP) through its Graduate Assistantship (GA) Scheme.

REFERENCES

[1]. G.R. Liu, and M.B Liu, Smoothed particle hydrodynamics – a meshfree particle method. World Scientific Publishing, Singapore, 2003.

[2]. H.H. Bui, K. Sako, and Fukagawa, Numerical simulation of soil–water interaction using smoothed particle hydrodynamics (SPH) method. Journal of Terramechanics 44, 2007, 339–346.

[3]. L.M. Dakssa, and I.S.H. Harahap, Investigating rainfall-induced unsaturated soil slope instability: a meshfree numerical approach, WIT Transactions on Engineering Sciences, 73, pp 231-243, 2012.

[4]. M.G. Gesteira, B.D. Rogers, R.A. Dalrymple, A.J.C. Crespo, and M. Narayanaswamy, User guide for the SPHysics code v2.0, Online: http://wiki.manchester.ac.uk/sphysics, 2010.

[5]. J.J. Monaghan, Simulating free surface flows with SPH. Journal of Computational Physics, 110, 399-406, 1994.

[6]. F. Jiang, and A.C.M. Sousa, Smoothed particle hydrodynamics modeling of transverse flow in randomly aligned fibrous porous media. Transp porous med 75, 17-33, 2008.

[7]. J.P. Morris, Y. Zhu, and J.P. Fox, Parallel simulations of pore-scale flow through porous media. Computers and Geotechnics 25, 227-246, 1999.

[8]. G.G. Pereira, M. Prakash, and P.W. Cleary, SPH modeling of fluid at the grain level in a porous medium. Applied Mathematical Modeling 35, 1666–1675, 2011.

[9]. T. Lenaerts, B. Adams, and P. Dutre, Porous flow in particle-based fluid simulations. ACM Trans. Graph. 27, 3, Article 49, August 2008.

[10]. H.H. Bui, R. Fukagawa, K. Sako, Smoothed particle hydrodynamics for soil mechanics. Numerical Methods in Geotechnical Engineering (eds.), Taylor & Francis, 2006.

[11]. J.P. Morris, P.J. Fox, and Y. Zhu, Modeling low Reynolds number incompressible flows using SPH. Journal of computational Physics, 136, 214–226, 1997.

[12]. P.W. Cleary, and J.J. Monaghan, Conduction modeling using smoothed particle hydrodynamics. Journal of computational physics, 148,227-264, 1999.

[13]. Tennessee Department of Transportation (TDOT), design division, drainage manual, hydrology, January 1, 2010.

[14]. X. Fang, D.B. Thompson, T.G. Cleveland, and P. Pradhan, Variations of time of concentration estimates using NRCS velocity method. Journal of Irrigation and Drainage Engineering, ASCE, July/August 2007.

[15]. Tennessee Department of Transportation (TDOT), design division, drainage manual, roadside ditches and streams, May 15, 2011