Implicit Subtime Stepping for Solving Nonlinear Flow Equations in an Integrated Surface–Subsurface System

Young-Jin Park*, Edward A. Sudicky, Sorab Panday, and George Matanga

A diverse group of problems requires quantification of the entire hydrologic cycle by the integrated simulation of water flow in the surface and subsurface regimes. In a transient integrated simulation of the water cycle, the time step size is a key factor in controlling the solution accuracy and the simulation efficiency for a given spatial discretization. In general, if the time step size is sufficiently small, the resulting solution becomes more accurate but with higher computational cost. Thus, to maintain an acceptable level of solution accuracy in the entire simulation domain, the time step size is restricted by the relatively rapid responses in the surface flow regime. As the relatively rapid responses are typically limited to a small portion of the surface domain compared with the groundwater system, a large portion of the domain tends to be temporally overdiscretized. The implicit subtime stepping approach described here can apply smaller sub-time steps only to the subdomain where the accuracy requirements are needed. In this work, generalized formulations for implicit subtime stepping in the numerical solution of the nonlinear coupled surface–subsurface equations were derived and implemented into the integrated model HydroGeoSphere. Application to several problems showed that implicit subtime stepping can significantly improve the simulation efficiency with minimal loss in accuracy. The methodology was successfully applied to enhance the computational efficiency of an integrated flow simulation in the San Joaquin Valley, California, where the characteristic response time near surface drainage streams is orders of magnitude shorter than in the groundwater regime.

Y.-J. Park and E. Sudicky, Dep. of Earth and Environmental Sciences, Univ. of Waterloo, Waterloo, ON, Canada N2L 3G1; S. Panday, AMEC Geomatix Consultants Inc., 620 Herndon Parkway, Suite 200, Herndon, VA 20170-5400; and G. Matanga, U.S. Bureau of Reclamation, 2800 Cottage Way, Sacramento, CA 95825. *Corresponding author (yyjpark@uwatwaterloo.ca).

Vadose Zone J. 8:825–836
doi:10.2136/vzj2009.0013
Received 17 Feb. 2009.
Published online 28 Oct. 2009.

© Soil Science Society of America
677 S. Segoe Rd. Madison, WI 53711 USA.
All rights reserved. No part of this periodical may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher.

Published November, 2009
In each time span \( t_k \) \((k = 0, 1, 2, \ldots, n)\), the head and saturation distribution in the subsurface and the water depth across the land surface are solved when \( t = t_k \), with given solutions at \( t = t_{k-1} \) obtained from the previous time step. Numerical error in each time span \( [t_{k-1}, t_k] \) is generally a (monotonically increasing) function of the given time step size \( \Delta t_k \) as well as the rate of change of the variable under consideration such as the head or water saturation for subsurface flow, the water depth for surface flow, and the concentrations for a solute transport simulation. In other words, calculated solutions at \( t = t_k \) are more accurate as \( \Delta t_k \) becomes smaller where the head or water depth changes slowly, while the error becomes greater when \( \Delta t_k \) is larger where the hydraulic responses are relatively rapid. If accurate hydraulic responses are to be achieved, the time step size is a key factor in controlling the solution accuracy and the simulation efficiency for a given spatial discretization. In general, the solution becomes more accurate with the use of smaller \( \Delta t_k \), but as the number of time steps increases, there is a higher computational cost.

Because a standard finite-difference approximation of temporal derivatives conventionally uses one fixed time step size for the entire simulation domain, the time step size needs to satisfy the least tolerant accuracy requirement where the hydraulic activity is the highest, for example, near or at active stream channels in the context of a fully integrated simulation framework (Fig. 1). To maintain an acceptable level of accuracy in the entire simulation domain, if the time step size is restricted because a small portion of the surface domain is responding rapidly, then the remainder of the surface and subsurface domains tends to be temporally overdiscretized (Huang and Yeh, 2009). This temporal overdiscretization issue has been addressed using various numerical strategies (VanderKwaak, 1999; Bhallamudi et al., 2003; Park et al., 2008).

In the hybrid implicit–explicit approach suggested by VanderKwaak (1999), an implicit time-marching scheme is applied only to the part of the domain where the stability requirements are stricter, while an explicit scheme is applied to the remainder of the domain. Although VanderKwaak (1999) demonstrated that the hybrid approach can reduce the needed computational effort by decreasing the number of discrete equations to be solved implicitly at each time step, an explicit solution is required to satisfy certain stability conditions as well as an accuracy constraint. Subtime stepping is an alternative fully implicit numerical strategy that applies different time step sizes to one or more subdomains, with each having different accuracy requirements. It was suggested by Bhallamudi et al. (2003) that implicit subtime stepping is most suitable to solve problems where the system response is high in only a small portion of the computational domain. Park et al. (2008) further demonstrated that the approach could significantly enhance computational performance when solving for flow and transport in fractured porous media, where the hydraulic activity in the fracture domain is much higher than that in the matrix but the fracture domain is typically limited to a small portion of the geologic medium. It was also suggested by Park et al. (2008) that subtime stepping can be an advantageous strategy to solve coupled overland and groundwater flow problems because the relatively active zones of the hydraulic responses are typically limited to the area near stream channels, which is also a small portion of the computational domain (Fig. 1). The major difference between flow in fractured porous media and integrated surface–subsurface problems lies in the linearity of the equations to be solved: saturated-zone flow in fractured porous media is linear (excluding, for example, density-dependent flow), while Richards’ equation describing variably saturated flow or the diffusive-wave equation for overland flow is highly nonlinear.

The objectives of this study were (i) to derive a generalized formulation for implicit subtime stepping as applied to fully integrated, nonlinear, surface–subsurface flow problems, (ii) to validate the numerical implementation of the subtime stepping approach in a Newton–Raphson iterative solution procedure as implemented in the integrated model HydroGeoSphere, (iii) to elucidate the accuracy and performance of the subtime stepping strategy, and (iv) to demonstrate the advantages of the approach in a field-scale application. Below we first briefly describe the
governing equations describing integrated subsurface and surface water flow, followed by an overview of the numerical methods; then, the theory for the subtime stepping approach to solve the nonlinear equations is derived for variably saturated subsurface flow and overland flow, with a discussion of results regarding computational performance and accuracy issues. The accuracy and performance aspects are verified using a two-dimensional problem involving an areal surficial flow system in a V-shaped catchment and a groundwater recharge problem involving a macropore and porous-matrix system under variably saturated conditions. Finally, the methodology is applied to a flow simulation at the watershed scale, where the characteristic response time for the river system is orders of magnitude smaller than that in the subsurface regime.

**Governing Flow Equations for an Integrated System**

The modified form of Richards’ equation describing three-dimensional transient subsurface flow under variably saturated conditions is given by:

\[
\nabla \cdot (k_r \mathbf{K} \cdot \nabla h) + Q = \frac{\partial}{\partial t} (\theta_0 S_w)
\]

where \(k_r\) is the relative permeability of the medium as a function of the water saturation \(S_w\), or the pressure head \(\psi\); \(\mathbf{K}\) is the hydraulic conductivity tensor; \(h\) is the total head as \(\psi + z\), where \(z\) is the elevation; \(\theta_0\) is the saturated water content; \(Q\) is an externally applied source or sink of water; and \(t\) is time. The fluid exchange between the surface and subsurface is represented by \(\Gamma\). The storage term in Eq. [1] can be expanded to account for both the change in storage in the unsaturated zone through compressibility effects and a change in saturation in the unsaturated zone (Cooley, 1971; Neuman, 1973):

\[
\frac{\partial}{\partial t} (\theta_0 S_w) \sim S_w \frac{\partial h}{\partial t} + \theta_0 \frac{\partial S_w}{\partial t}
\]

where \(S_w\) is the specific storage coefficient. The governing Eq. [1] is highly nonlinear due to the nature of the constitutive relations between the hydraulic head \(h\), saturation \(S_w\), and relative permeability \(k_r\), which is commonly described through expressions such as the van Genuchten (1980) or Brooks and Corey (1964) relations.

The depth-integrated surface flow equation utilized here involves the following diffusion wave approximation:

\[
\frac{d \Delta h}{d t} = \frac{\partial}{\partial x} \left( d_o K_{ox} \frac{\partial h_u}{\partial x} \right) + \frac{\partial}{\partial y} \left( d_o K_{oy} \frac{\partial h_u}{\partial y} \right) + d_o \Gamma_o + Q
\]

where \(x\) and \(y\) are special coordinates, \(d_o\) is the volume depth, \(h_u\) is the water surface elevation, and \(K_{ox}\) and \(K_{oy}\) are the surface conductances, approximated by the Manning’s equation as

\[
K_{ox} = \frac{d_o^{2/3}}{n_x} \left( \frac{1}{\partial h_u/\partial x} \right)^{2/3}; \quad K_{oy} = \frac{d_o^{2/3}}{n_y} \left( \frac{1}{\partial h_u/\partial y} \right)^{2/3}
\]

where \(n_x\) and \(n_y\) are Manning’s roughness coefficients and \(z\) is taken to be the direction of maximum surface-water slope. Equation [3] is nonlinear because the water depth \(d\) and the surface conductances \(K_{ox}\) and \(K_{oy}\) are complex functions of the dependent variables \(d_o\) or \(h_u = (d_o + z)\). Details can be found in Therrien et al. (2006).

The integration of the surface and subsurface flow regimes can be achieved by explicitly coupling the two governing flow Eq. [1] and [3]. In HydroGeoSphere, the fluid exchange between the surface and subsurface domains is calculated using a first-order leakage relation based on the assumption that the two domains are separated by a thin boundary layer such that

\[
d_o \Gamma_o = -\Gamma = (k_e)_{exch} \frac{b - h_o}{l_{exch}}
\]

where \((k_e)_{exch}\), \(K_{exch}\), and \(l_{exch}\) are the relative permeability, saturated hydraulic conductivity, and coupling length for fluid exchange. Because the governing Eq. [1] and [3] are solved simultaneously together with Eq. [5] in HydroGeoSphere, we refer to the model as a fully integrated type, as opposed to linked or iteratively coupled simulators.

**Subtime Stepping**

A subtime stepping approach was implemented to solve the linear flow and transport equations in HydroGeoSphere (Park et al., 2008). In the following, the basic concept of subtime stepping will be summarized for solving the linear flow and transport equations and it will be extended to the solution of the nonlinear integrated surface–subsurface flow equations by deriving the appropriate residual equations.

A discrete mass-balance equation for saturated groundwater flow is, for a control volume associated with the \(i\)th node (\(v_i\)),

\[
\frac{v_i S}{\Delta t} \left( b_i^{+\Delta t} - b_i^{-\Delta t} \right) = \sum_{j \in F_i} \gamma_{ij} \left( b_j^{+\Delta t} - b_j^{-\Delta t} \right) + Q_i
\]

where \(\gamma_{ij}\) is a set of nodes connected to the \(i\)th node and \(\gamma_{ij}(b_j^{+\Delta t} - b_j^{-\Delta t})\) represents the flow from the \(j\)th node to the \(i\)th node, with \(\gamma_{ij}\) defined as the conductance between the two nodes (Therrien and Sudicky, 1996; Therrien et al., 2006). Fully implicit time stepping is applied in Eq. [6] and for all the derivations that follow. When the \(i\)th node is subtimed (\(i \in S\), where \(S\) is a set of subtimed nodes), it can be connected to both subtimed (\(j \in S_1\), \(j \in S_2\)) and fully timed (\(j \in F_i\), \(j \in S \cap S_1\)) nodes, where \(\eta = S_1 \cup F_1\) and \(\eta = S \cap S_1\). Thus, a mass balance equation for the \(i\)th node at the \(m\)th subtime level is derived as

\[
\frac{v_i S}{\Delta t} \left( b_i^{+m\Delta t} - b_i^{-m\Delta t} \right) = \sum_{j \in S_1} \gamma_{ij} \left( b_j^{+m\Delta t} - b_j^{-m\Delta t} \right)
\]

where \(\Delta t\) is a subtime step size, given by \(\Delta t/M\), with the full time step size being \(\Delta t\) and the number of subtime steps \(M\) (\(1 \leq m \leq M\)). A same-implicitness factor scheme is used to approximate the flux between subtimed and fully timed nodes in Eq. [7]. The details of the validity and the limitation for the scheme are discussed in Bhallamudi et al. (2003) and Park et al. (2008). When the node is fully timed, a similar discrete equation can be derived:

\[
\frac{v_i S}{\Delta t} \left( b_i^{+\Delta t} - b_i^{-\Delta t} \right) = \frac{1}{M} \sum_{m=1}^{M} \left| \frac{v_i S}{\Delta t} \left( b_i^{+m\Delta t} - b_i^{-m\Delta t} \right) \right|
\]

\[
= \frac{1}{M} \sum_{m=1}^{M} \left| \sum_{j \in F_i} \gamma_{ij} \left( b_j^{+m\Delta t} - b_j^{-m\Delta t} \right) + Q_i \right|
\]
For linear flow problems, the discrete mass balance equations can be directly assembled into global conductance and capacitance matrices to solve for the dependent variables $b_{i,j}^{t}, b_{i,j}^{t+1/2}, b_{i,j}^{t+1}$, ..., $b_{i,j}^{t+MΔt}$. For the case of nonlinear flow, once a residual equation is derived from the flow equation, then it is straightforward to obtain a Jacobian matrix for the Newton–Raphson linearization. Below, the residual equations for variably saturated groundwater flow and overlapped flow are derived when a node is subtimed or fully timed, with the dependent variables associated with the residuals.

**Groundwater Flow under Variably Saturated Conditions**

A discrete mass-balance equation for the control volume associated with a subsurface node $i$ is given as (Therrien et al., 2006)

$$\frac{v_i}{\Delta t} \left( S_{w,i} \right)_{j}^{t+\Delta t} S_{i,j} - S_{i,j}^{t+\Delta t} - S_{i,j}^{t} + Q_i + \Gamma_i = 0 \tag{9}$$

where the flow from the $j$th node to the $i$th node includes a weighted value of relative permeabilities ($\lambda_{i,j}$), evaluated at the interface between nodal volumes for the $i$th and $j$th nodes. From Eq. [9], the residual for a subtimed or fully timed subtime level can be derived as

$$R^i_{s} \left[ b_{i,j}^{t+\Delta t}, b_{i,j}^{t+1/2}, b_{i,j}^{t+MΔt}, b_{i,j}^{t+(m-1)Δt} \right] = \frac{v_i}{\Delta t} \left( S_{w,i} \right)_{j}^{t+\Delta t} S_{i,j} - S_{i,j}^{t+\Delta t} - S_{i,j}^{t} + Q_i + \Gamma_i \tag{10}$$

It is important to note that in the subtime stepping approach, the hydraulic head $h_i^{t+(m-1)Δt}$, will be an unknown and a part of the solution when $1 < m \leq M$, and thus a term $\partial R_i / \partial h_i^{t+(m-1)Δt}$, needs to be incorporated into the Jacobian matrix for $\Delta h_i^{t+(m-1)Δt}$. Likewise, a residual can be derived when the $i$th node is fully timed such that

$$R^i_{f} \left[ b_{i,j}^{t+\Delta t}, b_{i,j}^{t+1/2}, b_{i,j}^{t+MΔt}, b_{i,j}^{t+1} \right] = \frac{v_i}{\Delta t} \left( S_{w,i} \right)_{j}^{t+\Delta t} S_{i,j} - S_{i,j}^{t+\Delta t} - S_{i,j}^{t} + Q_i - \Gamma_i \tag{11}$$

**Overland Flow**

A discrete mass-balance equation for the control surface associated with a surface flow node linked to the $i$th subsurface node $o_i$ ($a_{oi}$) is given as (Therrien et al., 2006)

$$\frac{\partial \phi}{\partial t} \left( d_{oi}^{\gamma+\Delta t} - d_{oi}^{\gamma} \right) = \sum_{o_j \in o_i} \lambda^{t+\Delta t}_{o_i,o_j+1/2,\gamma+1/2} \left( b_{o_j}^{t+\Delta t} - b_{o_i}^{t+\Delta t} \right) + a_{oi} \Gamma_{oi} + Q_{oi} \tag{12}$$

From Eq. [12], the residual for subtimed or fully timed overlapped node $o_i$ can be derived as

$$R_{oi} \left[ t^{t+\Delta t}_{o_i,o_j \in o_i}, b_{o_j}^{t+\Delta t}, b_{o_i}^{t+\Delta t}, b_{o_i}^{t+(m-1)\Delta t} \right] = \frac{\partial \phi}{\partial t} \left( d_{oi}^{\gamma+\Delta t} - d_{oi}^{\gamma} \right) - \sum_{o_j \in o_i} \lambda^{t+\Delta t}_{o_i,o_j+1/2,\gamma+1/2} \left( b_{o_j}^{t+\Delta t} - b_{o_i}^{t+\Delta t} \right) - d_{oi} \Gamma_{oi}^{t+\Delta t} - Q_{oi} \tag{13}$$

$$R_{oi} \left[ t^{t+\Delta t}_{o_i,o_j \in o_i}, b_{o_j}^{t+\Delta t}, b_{o_i}^{t+\Delta t}, b_{o_i}^{t+(m-1)\Delta t} \right] = \frac{\partial \phi}{\partial t} \left( d_{oi}^{\gamma+\Delta t} - d_{oi}^{\gamma} \right) - \frac{1}{M} \sum_{m=1}^{M} \sum_{o_j \in o_i} \lambda^{t+\Delta t}_{o_i,o_j+1/2,\gamma+1/2} \left( b_{o_j}^{t+\Delta t} - b_{o_i}^{t+\Delta t} \right) - d_{oi} \Gamma_{oi}^{t+\Delta t} - Q_{oi} \tag{14}$$

**Performance and Accuracy**

As mentioned above, the accuracy of a transient integrated simulation is determined by the time step size during each time step. Because only subtimed nodes use a smaller time step size $\Delta t$, and the remainder of the nodes use $\Delta t = MΔt$, the numerical error in temporal integration follows $O(\Delta t^2)$ and $O(\Delta t^3)$ at subtimed and fully timed nodes, respectively. As a tradeoff to solve more accurately at the subtimed nodes, the computational cost may increase. In the following, computational costs related to subtime stepping are analyzed to assist with determination of the optimal conditions under which to apply it.

The computational cost for a transient simulation is generally determined by the number of discrete equations ($n_{eq}$) to be solved during each time step and the total number of time steps ($n_t$). The cost to invert a coefficient matrix of size $n_{eq}$ by $n_{eq}$ ($C_{m}(n_{eq})$) is known to follow a power law such that $C_{m}(n_{eq}) \propto (n_{eq})^{3}$, where $1 < \alpha < 3$ in general and $1 < \alpha < 1.5$ for an iterative sparse matrix solver with incomplete LU preconditioning and a Bi-CGSTAB acceleration technique (Clift et al., 1996; Van der Vorst, 1992; Benthall, et al., 2003). Thus, the total computational cost for a transient simulation ($C_T$) is given as

$$C_T \propto n_t \left( n_{eq} \right)^3 \tag{15}$$

In a standard time stepping approach, the number of equations to be solved during each time step is the same as the number of nodes in the domain ($n_n$), and the number of time steps can
be approximated by the ratio of the total simulation period to the time step size ($T/\Delta t$). Thus, the cost is calculated as

$$C_T(\Delta t) \propto \frac{T}{\Delta t} n_n^\alpha \quad [16a]$$

To achieve an accuracy of $O(\Delta t)$, the cost will increase as

$$C_T(\Delta t) \propto \frac{T}{\Delta t} n_n^\alpha \quad [16b]$$

Thus, when $\Delta t = M\Delta t_s$, $C_T(\Delta t)/C_T(\Delta t) = M$. If the time step size is reduced by a factor of $M$, the number of time steps increases by a factor of $M$, as does the computational cost.

In the subtime stepping approach, the number of equations to be solved at each time step is not the same as the number of nodes because $M - 1$ additional unknowns arise at each subtimed node. When there are $n_{ns}$ subtimed and $n_{nf}$ fully timed nodes among $n_n$ ($= n_{ns} + n_{nf}$) nodes, the total number of equations becomes $n_{eq} = n_{nf} + Mn_{ns}$. Therefore, the cost is calculated to be

$$C_T(\Delta t/\Delta t_s) \propto \left(\frac{T}{\Delta t_s}\right) (n_{nf} + Mn_{ns})^\alpha$$

[17]

The condition that the computational cost to achieve $O(\Delta t)$ using subtime stepping Eq. [17] will be less than the cost to achieve the same accuracy using standard time stepping is given as

$$\frac{C_T(\Delta t/\Delta t_s)}{C_T(\Delta t)} = \frac{1}{M} \left( \frac{n_{nf} + Mn_{ns}}{n_n} \right)^\alpha < 1$$

[18]

When the ratio of the number of subtimed nodes to the total number of nodes is small or $R_{ns} = n_{ns}/n_n \ll 1$, then Eq. [18] can be rearranged as

$$\frac{C_T(\Delta t/\Delta t_s)}{C_T(\Delta t)} = 1 + \alpha R (M-1)$$

[19]

Equation [19] indicates that the subtime stepping is always more efficient than the standard time stepping when $R_{ns} \ll 1$, as the inequality in Eq. [19] ($R_{ns} < 1/\alpha$) is always true under the constraint $R_{ns} \ll 1$.

When most of the nodes in the domain are subtimed, or $R_{ns} = n_{ns}/n_n \approx 1$, Eq. [18] is reduced to

$$\frac{C_T(\Delta t/\Delta t_s)}{C_T(\Delta t)} \approx M^{\alpha-1} < 1$$

[20]

The inequality condition is not likely to be achieved as $M \geq 2$ and $1 < \alpha < 1.5$ and thus subtimed stepping is not efficient compared with standard time stepping.

In summary, subtime stepping becomes most efficient when the hydraulic activities are high only in a small portion of the domain ($R_{ns} \approx n_{ns}/n_n \ll 1$) and the computational cost to achieve the same accuracy is approximately given by Eq. [17]. If the response is uniform across the entire domain, the approach is not desirable to use.

**Numerical Results**

Implementation of the subtime stepping approach was validated for two-dimensional overland flow, three-dimensional variably saturated flow, and coupled surface and subsurface flow.

**Subsurface Flow under a Variably Saturated Condition: Groundwater Recharge through Macropores**

To validate the subtime stepping implementation for variably saturated flow in the subsurface and also to demonstrate its accuracy and performance, an infiltration problem in a domain containing a macropore was simulated by using standard time stepping with two different time step sizes and subtime stepping. Within a 2-m wide and 2-m deep two-dimensional domain, a 1-cm-wide vertical macropore was located in the middle of the domain (Fig. 2a). The domain was discretized into rectangular elements with a uniform vertical dimension of 1 cm but a variable width: 0.5 cm in the macropore, gradually increasing from 0.25 cm near the macropore to 1 cm in the matrix. Hydraulic conductivity values in the macropore and the matrix were given as 10 and 0.01 cm d$^{-1}$, respectively. For the given contrast of hydraulic conductivity values between the macropore and matrix systems, most of the infiltration was expected to occur through the macropore although the macropore occupied only a small portion of the domain. This suggested that subtime stepping could be an efficient strategy to solve this problem.

The initial saturation was set at 0.35 in the entire simulation domain. The pressure head was specified as zero at the top and bottom boundaries. Four observation points were located along the macropore at 1, 10, 30, and 50 cm from the top (OB1–OB4) and one observation point was located in the matrix 10 cm from the top and 5 cm from the axis of the macropore (OB5) (Fig. 2a). Infiltration of water through the macropore and the matrix was simulated for 10 d using standard time stepping with time step sizes of 0.25 and 1.0 d and also by subtime stepping with a full time step size of 1.0 d and a subtime step size of 0.25 d at the macropore nodes to a depth of 80 cm below the ground surface.
where the response was expected to be relatively rapid. Figure 2b shows the saturation distribution after 3 d and clearly illustrates that the majority of the infiltration occurred through the macropore.

Figure 3 compares the time–pressure head results from two standard time stepping simulations and one subtime stepping simulation at the five observation points shown in Fig. 2a. It is clear in Fig. 3 that in the subtime stepping approach, the solution in the subtimed zone (four observation points OB1–OB4) was determined by the subtime step size (0.25 d) and it was as accurate as the solution by standard time stepping with the small time step size (0.25 d). The accuracy in the fully timed nodes within the matrix (OB5) was the same as that obtained by standard time stepping with the larger time step size (1.0 d).

The CPU costs (a dual Intel Xeon 3.2 GHz CPU with 4 Gb of RAM) for the standard time stepping simulations were 265 and 652 s when the time step sizes were 1.0 and 0.25 d, respectively. Although the number of time steps was four times more when $\Delta t = 0.25$ d, the CPU cost at each time step was less for the iterative solver utilized in the model because the coefficient matrix became more diagonally dominant with the use of a smaller time step size (Van der Vorst, 1992; Clift et al., 1996). When the subtime stepping approach was used, the CPU cost was 344 s, implying that the CPU cost during each time step was greater than that of standard time stepping when $\Delta t = 1.0$ d because the number of equations to be solved increased with subtime stepping.

It was clearly demonstrated from this example that the accuracy of the subtime stepping approach is determined by the use of two different time step sizes as expected, and it is more efficient (by a factor of about two) at achieving the same accuracy as the standard time stepping simulation. We also note that the accuracy and the performance of subtime stepping depends on the number of subtimed nodes and the matrix solver used, as well as the characteristics of the problem.

Overland Flow in a V-Shaped Catchment

The subtime stepping implementation for two-dimensional areal surface flow was verified using the rainfall–runoff example of Di Giammarco et al. (1996), VanderKwaak (1999) and Therrien et al. (2006) presented the details of the simulation along with the results obtained by various surface water flow simulators benchmarked against this problem. The problem involves two-dimensional surface flow from a tilted V-catchment (Fig. 4), generated by a 90-min-duration, 3 $\mu$m s$^{-1}$ intensity rainfall event. Only one-half of the domain needed to be simulated due to symmetry. The simulation domain therefore consisted of a 1000 by 800 m slope connected to a 1000-m length of channel that was 10 m wide. The surface slopes were 0.05 and 0.02, perpendicular and parallel to the channel, respectively. Manning’s roughness coefficients of 0.15 and 0.015 were applied to the slopes and the channel, respectively. A critical depth boundary condition was applied at the downstream end of the channel.

The domain was discretized into 10- by 10-m rectangular elements. Only surface flow was simulated, with rainfall being applied at the rate of 3 $\mu$m s$^{-1}$ for 90 min, followed by no rainfall for the second stress period of 90 min. Two standard time stepping simulations with $\Delta t = 150$ and 600 s, and a subtime stepping simulation with $\Delta t = 600$ s and $\Delta t_s = 150$ s were used to compare the downstream discharge results. For the subtime stepping simulation, those nodes located in the channel and the slope within 100 m from the channel were subtimed (Fig. 4a).

Figure 5 compares the predicted discharge values from the three HydroGeoSphere simulations with those from the other simulators. The results presented in Fig. 5 indicate that the simulation performed using standard time stepping with the smaller time step size ($\Delta t = 150$ s) was more accurate than the results when $\Delta t = 600$ s, and that the results from the subtime stepping simulation are comparable to the case when $\Delta t = 150$ s.
The CPU costs for the standard time stepping simulations were 82 and 193 s when $\Delta t = 150$ and 600 s, respectively, and is 115 s for the subtime stepping simulation. Again, to achieve a similar level of accuracy for the stream discharge, it is clear that subtime stepping is much more efficient than standard time stepping.

An Illustrative Surface–Subsurface Simulation: Abdul (1985) Field Experiment

VanderKwaak (1999) and Therrien et al. (2006) used the plot-scale rainfall–runoff experiment of Abdul (1985) as a test problem to verify the integrated surface–subsurface modeling approach (Fig. 6). Details of the site, its characteristics, and the experimental setup can be found in VanderKwaak (1999). An artificial channel lies approximately 1.2 m below the surrounding grassy land. The channel was initially dry before the application of the artificial rainfall via irrigation sprinklers. The initial water table was around 22 cm below the streambed, with the artificial rainfall being applied at a rate of 2 cm h$^{-1}$ for 50 min using a sprinkler system.

The domain was discretized areally using 1372 nodes and 2651 triangular elements. Vertically, the grid was distorted to conform to the topographic elevation. Fifteen layers of elements were used, with a fine discretization of 0.1 m near the surface that was increased to 1 m near the bottom of the domain. The rainfall input was at the rate of 5.56 $\mu$m s$^{-1}$ for a first stress period of 50 min, with zero rainfall for the second 50-min stress period. A critical depth boundary condition was applied around the entire upper surface boundary of the domain. Two standard time stepping and one subtime stepping simulations were performed to compare the discharge calculated at the outlet. In the subtime stepping simulation, those surface nodes and the subsurface nodes in the top two layers (Fig. 6a) were subtimed, with the number of subtime steps set to four.

Figure 6b shows that the subtime stepping simulation can provide the same accuracy as the standard time stepping simulation that used the smaller time step size ($\Delta t = 25$ s), although the former required less CPU cost (210 s) than the latter (253 s). The results obtained with the standard time stepping and the larger time step size were less accurate than the results of the other two cases (Fig. 6b).

Application to Integrated Flow at a Watershed Scale Study Area

The Central Valley of California stretches 644 km from Redding in the north to Bakersfield in the south, as shown in Fig. 7. Previously occupied by an inland sea, the Central Valley is mostly a flat area of land nestled between the Coast Ranges in the west and the Sierra Nevada in the east. The Central Valley has an arid climate with <200 mm of rainfall per year in the southernmost regions. Since the development of irrigation systems for crop production, groundwater sources have been heavily relied on to fulfill this need. In 1989, for example, groundwater pumping in the Central Valley accounted for about 20% of all groundwater withdrawn in the entire United States (Williamson et al., 1989). To
provide water for development, complex water storage and conveyance systems consisting of dams, reservoirs, and canals were constructed by the California Department of Water Resources and the U.S. Bureau of Reclamation. The management and operation of these systems for the purposes of water supply, flood management, environmental protection, and recreation are performed under the State Water Project (SWP) and the Central Valley Project (CVP). The reservoirs and dams managed by the SWP and the CVP highly regulate flows in most rivers that originate from the Sierra Nevada mountains.

Anthropogenic activities have drastically altered the Central Valley’s hydrologic system from its natural state. Sierra Nevada snowmelt water is dammed, river flows are controlled, and natural groundwater levels have been lowered from decades of extensive pumping. To better address the challenges of sustainable water resource management, it is necessary to understand the nature, extent, and interaction of the major hydrologic processes occurring within the valley. Computer models have long played a key role in managing the water resources in the Central Valley.

Five major basins comprise the Central Valley Aquifer System (Williamson et al., 1989): Redding Basin, Sacramento Valley, Sacramento–San Joaquin Delta, San Joaquin Valley, and Tulare Basin. The associated surface water system is located within Region 18 of the approved boundaries, and numerical codes for the various river-basin units have been developed by the USGS (Seaber et al., 1987). This study will focus on the application of HydroGeoSphere with subtime stepping in the San Joaquin Valley and the northern portion of the Tulare Basin, as shown in Fig. 7. There are six major rivers in this area: Stanislaus, Tuolumne, Merced, Chowchilla, Fresno, and San Joaquin. The major groundwater system is composed of upper and lower aquifers that are separated by a clay aquitard, known as Corcoran, mainly in the western side of the valley. Toward the eastern foot hills, the clay aquitard does not exist. Surface water follows the natural topography of the valley and ultimately discharges to the Pacific Ocean through the San Francisco Bay and the Sacramento–San Joaquin Delta. Historically, upward hydraulic gradients were typical on the western side of the valley, as incoming subsurface flows from the east created artesian conditions, discharging to rivers and wetland areas. The surface water–groundwater connection in the valley has proven essential to maintaining ecosystem health.

**Modeling Layout**

The two-dimensional overland flow domain of the San Joaquin Valley (SJV) was discretized into about 27,400 nodes and 26,700 rectangular elements of 800- by 800-m size. A digital elevation map was used to incorporate the topography of the SJV. The elevations of the nodes associated with six drainage rivers were lowered by 3 m. Along the river elements, a Manning’s roughness coefficient of 0.002 was assigned, with 0.02 being applied to the rest of the domain. The three-dimensional subsurface domain was discretized into about 219,000 nodes and 187,000 brick elements, having the same planimetric dimensions as the surface rectangular grids. Seven evenly spaced vertical layers were used from the ground surface to the bedrock surface. For the application of subtime stepping and to demonstrate its performance, the subsurface was considered to be a single homogeneous aquifer system, although it consists of a relatively thin upper aquifer, a thicker lower aquifer, and a thin clay layer(s) between them. The horizontal hydraulic conductivity was estimated to be 0.28 cm d$^{-1}$ and a vertical value equal to 37 μm d$^{-1}$ was used. Storativity and porosity were estimated as $10^{-7}$ m$^{-1}$ and 0.34, respectively, and it was assumed to have the same van Genuchten retention properties as the Borden sand used for the simulation of the Abdul (1985) rainfall–runoff problem.

We wish to make it clear that the parameterization of the physical system is not intended to represent a calibrated model of the SJV. Instead, we used this watershed-scale problem only to demonstrate the performance of the subtiming scheme when applied to a fully integrated surface–subsurface flow system. Work at the development of a calibrated model of the SJV using the fully integrated surface–subsurface model HydroGeoSphere is ongoing.

**Steady Flow in Response to Average Effective Precipitation**

As a starting point for a subsequent transient simulation, a steady-state solution was first obtained in stages. This involved (i) a steady-state solution for surface water flow with an estimated value of the effective precipitation rate of 284 mm yr$^{-1}$, (ii) a second steady solution for groundwater flow using the same infiltration rate as the effective precipitation rate and with specified head values of 2 m above the elevation of the river nodes, and (iii) an integrated surface water and variably saturated subsurface flow solution with the estimated effective precipitation rate being applied to the surface domain. We defined the effective precipitation as the mean rainfall minus the mean evapotranspiration.

![Fig. 7. San Joaquin Valley location map.](image-url)
The purpose of the steady simulations was twofold: to compare the integrated simulation results to the results of surface-only and subsurface-only simulations, and to illustrate the characteristic hydraulic response times in the different flow regimes.

For the steady-state surface water flow simulation, the estimated effective precipitation was applied to the entire overland domain, and average measured streamflow rates were applied as inputs to the upstream ends of the six rivers. A critical depth boundary condition was applied to the downstream end of the surface drainage network, with a no-flow condition being applied to the rest of the external boundary. The surface of the domain was assumed to be completely dry, with zero water depth at the initial time ($t = 0$). Figure 8 shows the patterns of the simulated steady-state water depths and Manning fluxes across the land surface. Figure 8 shows that precipitation-induced flow was distributed across the entire surface because the interaction with subsurface and infiltration–exfiltration processes were not considered for this simulation. Because the Manning's roughness coefficient in the streams was an order of magnitude smaller than that in the remainder of the domain, the simulated water depths were less in the streams, although they drain most of the water. Precipitation, incoming river flows, and discharges at the downstream end of the main drainage system are plotted with time in Fig. 9. It is evident in Fig. 9 that streamflows take about 10 d to drain water from the inlets to the outlet and the overland runoff component requires about 100 d to drain the system.

For the steady-state subsurface flow simulation, the estimated net precipitation was applied to the top surface of the three-dimensional simulation domain as an effective infiltration, and a specified head boundary condition was applied to the river nodes with the head specified at 2 m above the streambed elevation. The initial head was set to 5 m throughout the entire simulation domain. Figure 10a shows the distribution of simulated steady-state saturations. The result in Fig. 10a indicates that water was infiltrating into the subsurface in the topographically high regions, flowing to the lower elevation regions, and discharging to the rivers where the head is specified. Water flow rates into or out of the system with time are shown in Fig. 10b. The results in Fig. 10b show that steady state was reached after about $10^5$ d.

For the steady coupled surface and subsurface flow simulation, the same boundary conditions as applied to the surface-only simulation (the estimated effective precipitation and average measured streamflow rates) were applied to the surface domain. The steady-state simulation results from the surface-only and subsurface-only simulations were used as an initial condition. Figure 11 shows the distribution of the steady-state surface water fluxes, fluid exchange rates between the surface and subsurface, and the subsurface water saturations. Major differences between the results in Fig. 11 and Fig. 8 and 10 are evident for the amount of runoff occurring across the land surface, the water fluxes in the smaller rivers, and the saturation patterns in the subsurface. In the coupled simulation, a portion of the net precipitation applied to the surface infiltrated into the subsurface, and the remainder of the water flowed over the land surface, depending on the degree of saturation in the subsurface. As a result, the saturation in the subsurface-only simulation was overestimated compared with the results of the coupled simulation.

Figure 12 shows the change in fluid flow rates into or out of the system, along with the infiltration and exfiltration rates between two flow regimes. It can be seen that for the first few tens of days, the total exfiltration rate was greater than the total infiltration rate because the steady-simulation results from the subsurface-only simulation overestimated the water saturations in the subsurface, resulting in excess outflow from the system. Figure 12 also shows, however, that it took about $10^5$ d to achieve steady state. At steady state, the infiltration and exfiltration rates were the same and slightly more than one-half of the total inflow applied to the land surface infiltrated into the subsurface. This implies that the remainder of water flowed over the land surface.
The three steady-state simulations illustrate that the characteristic response time for river drainage, overland runoff, and subsurface flow ranges across four orders of magnitude, from tens of days to $10^5$ d. It is also evident that important features in the integrated flow regime characteristics cannot be captured by performing a surface-only or subsurface-only simulation.

**Transient Integrated Flow in Response to a Single Precipitation Event**

Because the response of an integrated surface and subsurface system is relatively rapid in only a small portion of the domain, such as within the overland flow regime or the rivers, the subtime stepping approach can significantly enhance simulation efficiency. To demonstrate the enhancement in efficiency, the response of the San Joaquin Valley to a single precipitation event was simulated using standard time stepping and subtime stepping approaches, and the results and the computational cost of the simulations were compared.

Figure 13 shows the distribution of surface water depths at different times in response to a single precipitation event with a rate of 25 mm d$^{-1}$ for 1 d. Before the application of rainfall, the results from the steady-state integrated simulation were taken as the initial condition. The rainfall event was applied at $t = 1$ d and the simulation proceeded for an additional 8 d after the precipitation ceased (total simulation period of 10 d).

Two standard time-stepping simulations were performed using time step sizes of 0.1 and 0.5 d. For the subtime stepping simulations, two cases were considered: one in which the entire surface was subtimed and another in which only the river nodes and their neighboring nodes were subtimed. The results in Fig. 14 show that the accuracy of the simulations is strongly influenced by the time step size when using a standard time stepping approach. The subtime stepping simulations yielded the same peak flow rate and peak arrival time, while the peak flow was underestimated by >10% for the case of standard time stepping with a large $\Delta t$ equal to 0.5 d.
The CPU costs for the standard time stepping simulations were 1976 and 680 s for \( \Delta t = 0.1 \) and 0.5 d, respectively. While the number of time steps increased by a factor of five when \( \Delta t = 0.1 \) d, during each time step, the matrix solver converged with fewer iterations, resulting in less than five times the CPU cost. For the subtime stepping simulations, the CPU time was 1517 s when the entire surface was subtimed, while it was 1106 s when about 20% of the active surface nodes were subtimed. Thus, to achieve a similar level of accuracy using standard time stepping compared with the most efficient subtiming case, almost twice as much CPU time was required.

**Summary and Conclusions**

Integrated surface–subsurface flow and transport simulations typically can be fraught with temporal overdiscretization problems because the characteristic response times for streamflow, overland runoff, and subsurface flow can differ by orders of magnitude. The subtime stepping approach can apply different time step sizes to the subdomains that have different response times. It is most suitable for problems where the hydraulic activity is high in only a small portion of the domain such as within or near active stream channels.

A generalized subtime stepping formulation was derived to solve the integrated nonlinear equations describing variably saturated subsurface and depth-integrated surface water flow. The formulation was applied to the Newton–Raphson linearization used in the integrated model HydroGeoSphere. The numerical implementation was validated through a series of test problems ranging in complexity. It was shown that the subtime stepping approach can significantly improve the solution efficiency for solving coupled surface–subsurface problems. Additional efficiency could be achieved by dynamically adjusting the subtimed nodes and the number of subtime steps during the simulation period for a given problem, especially when the zone of shorter characteristic time scale changes during the simulation period, such as spatially and temporally varying streamflow in response to transient precipitation events.

**Acknowledgments**

This work was supported by grants from the Natural Sciences and Engineering Research Council (NSERC) of Canada, by a Canada Research Chair held by E. Sudicky, and by Grant 3-2-3 from the Sustainable Water Resources Center of the 21st Century Frontier Research Program of Korea. Additional funding was provided by support from the U.S. Bureau of Reclamation.
References


