Robust numerical solution of the reservoir routing equation

Marcello Fiorentini, Stefano Orlandini*

Dipartimento di Ingegneria “Enzo Ferrari”, Università degli Studi di Modena e Reggio Emilia, Strada Vignolese 905, 41125 Modena, Italy

Abstract

The robustness of numerical methods for the solution of the reservoir routing equation is evaluated. The methods considered in this study are: (1) the Laurenson–Pilgrim method, (2) the fourth-order Runge–Kutta method, and (3) the fixed order Cash–Karp method. Method (1) is unable to handle nonmonotonic outflow rating curves. Method (2) is found to fail under critical conditions occurring, especially at the end of inflow recession limbs, when large time steps (greater than 12 min in this application) are used. Method (3) is computationally intensive and it does not solve the limitations of method (2). The limitations of method (2) can be efficiently overcome by reducing the time step in the critical phases of the simulation so as to ensure that water level remains inside the domains of the storage function and the outflow rating curve. The incorporation of a simple backstepping procedure implementing this control into the method (2) yields a robust and accurate reservoir routing method that can be safely used in distributed time-continuous catchment models.

1. Introduction

The term “reservoir routing” (or “storage routing”) is used to denote the propagation of an inflow hydrograph through a reservoir. Storage in surface reservoirs may be natural or man-made [12, p. 100]. Natural storage in the form of lakes, swamps and rivers have a strong influence on the flow from the catchment but does not regulate flow [19, p. 308]. Regulation in the water resources management sense requires man-made control structures. Reservoir storage refers to the part of the reservoir volume which can be used for flow regulation. A reservoir may be required to supply water for power generation, irrigation, water supply, flood protection, and recreational use. Each of these requirements of the reservoir imposes constraints on the operating policy. Many of these requirements contradict each other (e.g., power generation requires a full reservoir or maximum head whereas flood protection requires an empty reservoir). For flood protection in a multipurpose reservoir a certain volume of the storage reservoir is required to be emptied, ready to receive water during the peak of the flood. The flow rate downstream is limited to a predetermined maximum value. The stored water is released after the flood peak has passed. Thus the reservoir receives all the water in excess of the design flow rate and the peak of the hydrograph is skimmed off and is delayed by storage. This mechanism reduces the peak discharge rate of the downstream hydrograph but increases the baselength of the hydrograph [16]. Predictions of the magnitude of the flood peak, its time of occurrence, and the volume of water to be stored are needed for the design of the reservoir. More in general, detailed descriptions of the hydrologic interactions between reservoirs and regulated river systems are needed to determine structural and nonstructural measures for flood protection and water resources management.

A reservoir can be modeled either as a large channel reach or as a pool [9]. While (distributed) channel routing may be needed to describe the propagation of rapidly rising hydrographs (e.g., having time of rise less than 1 h) along very long (e.g., having lengths greater than 80 km) reservoirs, (lumped) pool routing is adequate (e.g., errors non exceeding 10%) in most of the cases [9]. Level pool routing is a procedure for calculating the outflow hydrograph from a reservoir with a horizontal water surface, given its inflow hydrograph and storage-outflow characteristics [4, p. 245]. The equations governing reservoir dynamics can be combined to yield the nonlinear first-order ordinary differential equation

$$\frac{dS}{dt} = I(t) - Q(t,S),$$  

(1)

where $S$ is the volume of water stored in the reservoir, $t$ is the time, $I$ is the inflow discharge, and $Q$ is the outflow discharge, or the equivalent differential equation

$$\frac{dH}{dt} = \frac{I(t) - Q(t,H)}{A(H)},$$  

(2)

where $H$ is the water surface level and $A = dS/dH$ is the water surface area at elevation $H$. A sketch of the quantities involved in level pool routing is shown in Fig. 1. The storage function

* Corresponding author.

E-mail addresses: marcello.fiorentini@unimore.it (M. Fiorentini), stefano.orlandini@unimore.it (S. Orlandini).

URL: http://www.idrologia.unimore.it/orlandini (S. Orlandini).
Robustness, accuracy, and efficiency of numerical methods for the solution of Eqs. (1) and (2) are evaluated in this paper by identifying critical conditions that can be met in time-continuous hydrologic simulations. The case study of the Crostolo River reservoir is presented in Section 2. The methods considered in this study, namely (1) the Laurenson–Pilgrim method [11,16], (2) the Runge–Kutta method (e.g., [4, p. 252]), and (3) the embedded Runge–Kutta method based on Cash–Karp fixed order method [3], are reviewed critically in Section 3. A new strategy for obtaining robust reservoir routing is presented in Section 4.

2. Case study

The case study of the Crostolo River flood control reservoir is considered. This reservoir is located approximately 4 km upstream the town of Reggio Emilia, Italy. Latitude and longitude of the reservoir center are 44°38′22.36″N and 10°35′09.82″E, respectively. Land surface topography is represented by using a 1-m digital elevation model generated from a lidar (light detection and ranging) survey carried out in 2009 by setting the point density equal to 8–10 points/m². The location and the hillshade of a terrain portion including the reservoir are shown in Fig. 2. The reservoir drains a catchment having extension of about 89 km² and time of concentration of about 5 h. The reservoir is formed by a dam in concrete having a height of about 12.40 m (with respect to the base of foundations) and length of about 100 m. The elevations of the reservoir bottom, the spillway crest, and the levee top are $H_B = 102.80$ m above sea level (asl), $H_S = 112.50$ m asl, and $H_L = 116.80$ m asl, respectively (Figs. 1 and 2). The storage function $H–S$ was determined by using the trapezoidal rule to solve the integral

$$S(H) = \int_{H_S}^{H} A(h) \, dh. \quad (3)$$

Puls [18] and Goodrich [10] provided a direct solution of the storage Eq. (1) based on tabular storage-outflow functions. This method is known in the literature as traditional (or storage or tabular) routing method [8]. Practical problem in writing a general purpose computer program for tabular routing arise, however, from the requirement to determine a range of outflow discharges sufficiently high to embrace the unknown outflow peak, while still having the points close enough together to represent the storage function accurately [16]. Spurious convergence and numerical oscillations in the outflow hydrograph computed by solving Eq. (1) were investigated by Laurenson [11] and Pilgrim [16]. These authors proposed a method implementing a switching procedure that ensures robust solutions for any nondecreasing storage-outflow function. Fenton [7] evaluated the stability of the numerical solution of Eqs. (1) and (2), concluding that the numerical solution of these equations are to be preferred to the traditional reservoir routing method. Fred and Hsu [9] solved Eq. (1) using an iterative trapezoidal integration method. Li et al. [13] noted some local stability problems in the numerical solution of Eq. (1) and proposed an adaptive method to mitigate these problems. Existing numerical methods for the solution of Eqs. (1) and (2) are adequate for the tasks of the designer. In this case, the structural parameters of the numerical method can be adjusted at will to best meet the robustness and accuracy requirements, because computational efficiency is not a critical factor. It remains, however, unclear whether these methods can be safely incorporated in distributed time-continuous models to describe efficiently the hydrologic interactions between contributing drainage basins, reservoirs, and regulated fluvial systems under highly variable hydroclimatic conditions. This task is required in (real time) flood risk management, Monte Carlo analysis, and (long term) water resources management, where robustness is a critical factor and time steps are constrained as a results of computational efficiency requirements [15,2].
where the water surface area \( A = A(h) \) for \( h = [H_k, H_f] \) is readily determined from the 1-m resolution digital elevation model shown in Fig. 2 by considering all the cells lying within the reservoir and having elevation less than or equal to \( h \). The obtained storage function \( H-S \) is shown in Fig. 3(a). The outflow discharge rating curve \( H-Q \), determined from physical modeling, is shown in Fig. 3(b). It is noted here that the obtained outflow discharge rating curve displays an interval of \( H \) around the value of 104.70 m asl in which the outflow discharge \( Q \) decreases as the water surface level \( H \) increases. This is due to the transition from free surface flow to totally submerged flow through the bottom outlets. The storage volume at the maximum design elevation of 114.68 m asl is about \( 3 \times 10^6 \) m\(^3\), and the outflow discharge released by the reservoir under these conditions is about 950 m\(^3\) s\(^{-1}\) (Fig. 3(a) and (b)).

3. Methods

The numerical methods for the solution of the reservoir Eqs. (1) and (2) can be grouped in two dominant classes. The first class is based on the numerical approximation of the storage Eq. (1) obtained by using the trapezoidal rule, namely

\[
S_{j+1} - S_j = \frac{\Delta t}{2} \left( (Q_{j+1} + Q_j) \right),
\]

where \( j \) is the counter of nodes \( t_j = j \Delta t \) \((j = 0, 1, 2, \ldots)\), along the time domain, \( \Delta t \) is the time step, \( S_j \), \( I_j \), and \( Q_j \) are the storage, the inflow discharge, and the outflow discharge, respectively, at time \( t_j \) \((j = 0, 1, 2, \ldots)\) (e.g., [4, p. 246]). The traditional method mentioned in Section 1 is obtained by putting Eq. (4) in the form

\[
2S_{j+1} + Q_{j+1} = (I_{j+1} + I_j) + \left( \frac{2S_j}{\Delta t} - Q_j \right)
\]

and by determining the unknown values \( H_{j+1}, S_{j+1} \), and \( Q_{j+1} \) from tabular data mapping \( H \) and \( 2S/\Delta t + Q \) [18,10,11,16,4,7]. This traditional method is unconditionally stable, but it displays an accuracy evaluated on the Yejevich’s [20] test case that is less than the second-order Runge–Kutta method [7]. Therefore, the traditional method is not evaluated in the present study. An alternative method based on Eq. (4) was introduced by Laurenson [11] and further developed by Pilgrim [16]. This method, denoted shortly as “LP method,” is described in Section 3.1 and is evaluated in the present study. The second class of methods is composed of explicit numerical solutions of Eq. (2). Runge–Kutta methods belong to this class [4,7]. Explicit Runge–Kutta methods were found by Fenton [7] to provide the best solutions in terms of consistency, accuracy, and stability. The Runge–Kutta method, denoted shortly as “RK method,” and the embedded Runge–Kutta method proposed by Cash and Karp [3], denoted shortly as “CK method,” are described in Sections 3.2 and 3.3, respectively, and are evaluated in the present study.

3.1. Laurenson–Pilgrim method

The LP method solves Eq. (4) by using the regular–false (or false position) method [17, p. 347]. The LP method is represented by the general equation

\[
X_{j+1/2}^{(1)} = \psi^k(X_{j+1/2}) = X_{j+1/2}^{(2)} - X_{j+1/2}^{(1)} - \psi^k(X_{j+1/2}^{(2)})
\]

where \( X \) can be either the outflow discharge \( Q \) or the storage \( S \). \( i \) is a counter of the iterations \((i = 0, 1, 2, \ldots)\), \( j \) is a counter of the nodes along the time domain \((j = 0, 1, 2, \ldots)\). \( \psi^k(X) \) is a function of \( Q \) incorporating the direct \( S-Q \) function \( S = F(Q) \), as given for instance by

\[
\psi^k(X_{j+1/2}) = Q_{j+1/2}^{(1)} \frac{\Delta t}{2} + F(Q_{j+1/2}) + Q_{j+1/2}^{(1)} \frac{\Delta t}{2} - (I_{j+1} + I_j) \frac{\Delta t}{2} - F(Q_{j+1/2})
\]

or a function of \( S \) incorporating the inverse \( S-Q \) function \( Q = F^{-1}(S) \), as given for instance by

\[
\psi^k(X_{j+1/2}) = F^{-1}(S_{j+1}) \frac{\Delta t}{2} + S_{j+1}^{(1)} + F^{-1}(S_{j+1}) \frac{\Delta t}{2} - (I_{j+1} + I_j) \frac{\Delta t}{2} - S_{j+1}
\]

The function \( \psi^k(X) \) is clearly defined on the basis of Eq. (4). When Eq. (4) is verified, then (and only then) \( \psi^k(X) = 0 \). Eq. (6) provides an approximation of the zero of the function \( \psi^k(X) \) obtained by considering the secant for the points \( (X_{j+1/2}, \psi^k(X_{j+1/2})) \) and \( (X_{j+1/2}^{(2)}, \psi^k(X_{j+1/2}^{(2)})) \) and the (approximate) condition \( \psi^k(X_{j+1/2}^{(2)}) = 0 \). At each time interval \( j \Delta t \), the LP method starts the iterations \((i = 0)\) with Eqs. (6) and (7) (i.e., with \( X = Q \)) by using \( Q_j \) and \( (Q_{j+1/2} + I_{j+1/2})/2 \) as initial guesses for \( Q_{j+1/2} \) and \( Q_{j+1/2}^{(2)} \), respectively. Further details on the method can be found in Pilgrim [16, p. 137]. The use of Eq. (6) with \( X = Q \) was originally introduced by Laurenson [11]. However, it was noted that the regular–false does not converge if the storage-outflow discharge function displays limbs in which \( S \) increases at constant \( Q \) or \( Q \) increases at constant \( S \). Laurenson [11] suggested, therefore, to alternate the use of Eqs. (7) and (8) for ensuring the convergence of Eq. (6). This convergence is achieved when \( \psi(X_{j+1/2}) < \varepsilon_0 \), where \( \varepsilon_0 \) is a fixed tolerance.

3.2. Runge–Kutta method

The RK methods solve Eq. (2), written in the form

\[
dH/dt = f(t, H)
\]

with

\[
f(t, H) = \frac{I(t) - Q(t, H)}{A(H)}
\]

Fig. 3. (a) Water level-storage (\( H-S \)) relationship obtained by applying Eq. (3) to the digital elevation model reported in Fig. 2, and (b) water level-outflow discharge (\( H-Q \)) relationship obtained experimentally from a physical model of the Crostolo River reservoir. The inset in (b) highlights the region in which the \( H-Q \) function is nonmonotonic as a result of the transition from (open-channel) free surface to pressurized pipe flow through the bottom outlets.
by evaluating \( q \) times the local function \( f(t, H) \) and by using a suitable linear combination of the kind

\[
H_{j+1} = H_j + \Delta t \sum_{n=1}^{q} c_n k_n.
\]

where \( H_j \) is the water level at time \( t_j \), \( \Delta t \) is the time step, \( c_n \) \( (n = 1, \ldots, q) \) are coefficients, \( k_n \) \( (n = 1, \ldots, q) \) is the evaluation of the function \( f(t, H) \) given by (9) at the stage \( n \), namely

\[
k_1 = f(t_j, H_j), \quad n = 1
\]

and

\[
k_n = f(t_j + \tau_n \Delta t, H_j + \Delta t \sum_{m=1}^{n-1} \beta_m k_m), \quad n = 2, \ldots, q,
\]

where \( \tau_n \) and \( \beta_m \) are coefficients [5]. Coefficients \( c_n, \tau_n, \beta_m \) can be organized to form of “Butcher Tableau” as reported in Table 1. In the present study, the fourth-order explicit RK method is used. The related coefficients are reported in Table 2.

### 3.3. Cash–Karp method

The CK method is an embedded, or nested, RK method in which the accuracy of the solution is controlled by comparing solutions obtained from RK methods having different orders [6,3,17]. A first solution, namely \( H_{j+1}^{(1)} \), is computed by using a fifth-order RK method represented by Eqs. (10)–(12) with \( q = 6 \) and coefficients given by Table 3. A second solution, namely \( H_{j+1}^{(2)} \), is computed by employing an embedded fourth-order RK method as

\[
H_{j+1}^{(2)} = H_j + \Delta t \sum_{n=1}^{6} c_n' k_n,
\]

where coefficients \( c_n' \) are given in Table 3 and evaluations \( k_n \) are given by Eqs. (11) and (12). Note that coefficients \( c_n' \) \( (n = 1, \ldots, 6) \) reported in Table 3 display different values depending on the order of the method [3]. As reported in Press et al. [17, p. 710], the difference \( \Delta = H_{j+1} - H_{j+1}^{(1)} \) between the two numerical estimates is a convenient indicator of truncation error. This difference, namely

\[
\Delta = \Delta t \sum_{n=1}^{6} (c_n' - c_n) k_n
\]

is used to adapt the time step. If \( \Delta \) is less than or equal to a given tolerance \( \Delta_0 \), the time step \( \Delta t \) is adapted (normally increased or left as is) using the equation

\[
\Delta t_{\text{new}} = SF \Delta t_0 \Delta_0^{0.20}
\]

where \( \Delta t_{\text{new}} \) is the new (adapted) time step, and SF is a safety factor less than or equal to 1. If \( \Delta > \Delta_0 \), the time step \( \Delta t \) is adapted (normally decreased) using the equation

\[
\Delta t_{\text{new}} = SF \Delta t_0 \Delta_0^{0.25}
\]

The value \( SF = 0.90 \) is used in the present investigation [3].

### 4. Robust reservoir routing

#### 4.1. Background

Pilgrim [16], Fenton [7], and Fread and Hsu [9] highlighted that the shapes of characteristic curves and inflow hydrographs play a critical role on determining the robustness and accuracy of the numerical solutions of reservoir routing Eqs. (1) and (2). This is a relevant issue especially when these methods are parts of distributed hydrologic models for time-continuous simulations of complex hydroclimatic scenarios. Existing numerical methods are found to fail when: (a) the outflow discharge rating curve \( H–Q \) is nonmonotonic, or it displays high gradients of either \( Q \) with \( H \) or \( H \) with \( Q \), and/or (b) large time steps (as commensurate with the rate of change of water level in the reservoir) are used as a result of computational constraints. An example of numerical failure of the LP method under the circumstances (a) is reported, for demonstration, in Fig. 4(a) and (b). In this example, the problem displayed by the LP method can be overcome by using a monotonic approximation of the “true” \( H–Q \) function as shown in Fig. 5(a) and (b). The case shown in Fig. 4(c) and (d) highlights, however, that the LP method may be unable to handle complex \( H–Q \) functions. Examples of numerical failure under the circumstances (b) are illustrated, for demonstration, in Fig. 4(c) and (d), for the RK method, and in Fig. 4(e) and (f), for the CK method.

In the case shown in Fig. 4(c), the fourth-order RK method provides (inaccurate) predictions of the water level falling beyond the elevation of the reservoir bottom \( H_b \). The points labeled 1–4 in the inset of Fig. 4(c) represent the values of the second argument of \( f(\cdot) \) in Eqs. (11) and (12) (i.e., \( H_j, H_j + \Delta \sum_{m=1}^{3} \beta_m k_m \) if \( n = 2, 3, 4 \). The point labeled 4 \((n = 4)\) highlights that the second argument of \( f(\cdot) \) in Eq. (12) falls outside the domains of the functions \( H–Q \) and \( H–S \), namely \( H_j + \Delta \sum_{m=1}^{3} \beta_m k_m \ < H_b \). As a consequence, the values of \( k_4 \) cannot be computed and the RK method cannot reach a solution. This problem can be overcome by selecting a smaller time step \( \Delta t \) as reported below.

An empirical criterion for the determination of a suitable time step \( \Delta t \) for the solution of the reservoir equation was provided by Pilgrim [16], who suggested to set the time step equal to or less than a quarter of the smallest rising limb baselength observed in

\[
H_{j+1} = H_j + \Delta t \sum_{n=1}^{6} c_n' k_n,
\]

where coefficients \( c_n' \) are given in Table 3 and evaluations \( k_n \) are given by Eqs. (11) and (12). Note that coefficients \( c_n' \) \( (n = 1, \ldots, 6) \) reported in Table 3 display different values depending on the order of the method [3]. As reported in Press et al. [17, p. 710], the difference \( \Delta = H_{j+1} - H_{j+1}^{(1)} \) between the two numerical estimates is a convenient indicator of truncation error. This difference, namely

\[
\Delta = \Delta t \sum_{n=1}^{6} (c_n' - c_n) k_n
\]

is used to adapt the time step. If \( \Delta \) is less than or equal to a given tolerance \( \Delta_0 \), the time step \( \Delta t \) is adapted (normally increased or left as is) using the equation

\[
\Delta t_{\text{new}} = SF \Delta t_0 \Delta_0^{0.20}
\]

where \( \Delta t_{\text{new}} \) is the new (adapted) time step, and SF is a safety factor less than or equal to 1. If \( \Delta > \Delta_0 \), the time step \( \Delta t \) is adapted (normally decreased) using the equation

\[
\Delta t_{\text{new}} = SF \Delta t_0 \Delta_0^{0.25}
\]

The value \( SF = 0.90 \) is used in the present investigation [3].

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Coefficients of the Runge–Kutta (RK) method organized in form of “Butcher Tableau”.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>( a_n )</td>
</tr>
<tr>
<td>1</td>
<td>( a_1 )</td>
</tr>
<tr>
<td>2</td>
<td>( a_2 )</td>
</tr>
<tr>
<td>...</td>
<td>( a_q )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Coefficients of the fourth-order Runge–Kutta (RK) method organized in form of “Butcher Tableau”.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>( a_n )</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>( \frac{1}{5} )</td>
</tr>
<tr>
<td>3</td>
<td>( \frac{1}{10} )</td>
</tr>
<tr>
<td>4</td>
<td>( \frac{1}{5} )</td>
</tr>
<tr>
<td>5</td>
<td>( \frac{1}{2} )</td>
</tr>
<tr>
<td>6</td>
<td>( \frac{11}{30} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Butcher Tableau of Cash–Karp (CK) method</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>( a_n )</td>
</tr>
<tr>
<td>1</td>
<td>( \frac{7}{12} )</td>
</tr>
<tr>
<td>2</td>
<td>( \frac{1}{2} )</td>
</tr>
<tr>
<td>3</td>
<td>( \frac{5}{24} )</td>
</tr>
<tr>
<td>4</td>
<td>( \frac{17}{48} )</td>
</tr>
<tr>
<td>5</td>
<td>( \frac{15}{32} )</td>
</tr>
<tr>
<td>6</td>
<td>( \frac{11}{30} )</td>
</tr>
</tbody>
</table>

Note: The values in the tables are represented in the form of fractions to maintain precision and accuracy in the calculations.
the inflow hydrograph. An analytic criterion based on the Lipschitz condition was provided by Fenton [7] in the form

$$\Delta t \frac{dQ}{dS} < 2.$$  \hspace{1cm} (17)

When the outflow discharge \( Q \) is not controlled, the time step \( \Delta t \) ensuring robust reservoir routing can be readily determined from Eq. (17) by using the storage-outflow discharge function \( S-Q \) [7]. A numerical criterion for the determination of the time step \( \Delta t \) was derived by Laurenson [11] by considering the inflow and the change of storage over the first half of the time step \( \Delta t \) as given by

$$\left| \frac{l_j + l_{j+1/2}}{2} \Delta t - \frac{Q_j + Q_{j+1/2}}{2} \Delta t \right| - \left| F(Q_j) - F(Q_{j+1/2}) \right| \leq \frac{\epsilon_s}{2}, \hspace{1cm} (18)$$

where the subscript \( j + 1/2 \) refers to the value linearly interpolated at the midpoint of the time step, \( F(\cdot) \) is the direct \( S-Q \) function \( S = S(Q) \), and \( \epsilon_s \) is a given tolerance. The time step \( \Delta t \) is reduced (generally to the half) iteratively until the condition (18) is met. Another numerical criterion is given by Eqs. (15) and (16) of the CK method. Time steps obtained from Eq. (17) may be very small, and this may imply that robustness of the reservoir routing equation is achieved in exchange for excessive computational costs. The other criteria do not ensure that the reservoir routing equations are accurately solved for all the conditions met in time-continuous hydrologic simulations. There is, therefore, a need for a reliable criterion ensuring accurate and computationally efficient solutions of the reservoir routing equations.

Fig. 4. Critical conditions found when ((a) and (b)) the Laurenson–Pilgrim (LP) method with a nonmonotonic \( H-Q \) function is used \((\lambda = 10, \Delta t = 30 \text{ s}, \text{ and } \epsilon_s = A H_0 \text{ with } \Delta_0 = 10^{-4} \text{ m})\), ((c) and (d)) the fourth-order Runge–Kutta (RK) method with large \( \Delta t \) are used and \( H \) approaches \( H_B \) \((\lambda = 10, \Delta t = 900 \text{ s})\), and ((e) and (f)) the Cash–Karp (CK) method with large time steps \( \Delta t \) are used and \( H \) approaches \( H_B \) \((\lambda = 10, \Delta t = 900 \text{ s}, \text{ and } \Delta_0 = 10^{-4} \text{ m})\).
are provided with a temporal resolution has comparable magnitude with the 10^-6 through the factor f/C_0^1 is the resulting scaled hydrograph, is ob-
P = P = 10^5 < are computed (Eq. (12)). As sketched in Fig. 6, if it is found that t_B is computed and t_D is reset to the original (not backstepped) value for the next computation. The backstepping procedure given by Eqs. (19), (21), and (22) provides a simple and efficient means for obtaining robust reservoir routing while also avoiding an excessive computational burden caused by the selection of a small time steps over the entire simulation period. The method described in this section is denoted as RK method with backstepping (RK-B). The problems shown in Fig. 4(c)–(f) are overcome by using this method as illustrated, for demonstration, in Fig. 7. In the inset of Fig. 7(a), the points labeled 1–4 represent the values of the second argument of f (+) in Eqs. (11) and (12) (i.e., H_j if n = 1, and H_j + ΔH^t-1_0 βmkm if n = 2, 3, 4).

5. Numerical experiments

Numerical experiments are carried out by using the inflow hydrol-

graph observed on April 18–20, 2009 as a base case and by scaling this hydrol-

graph through the relationship

\[ l_i = \min(l_i, l_i^{(1)}) + \lambda (l_i - \min(l_i, l_i^{(1)})) \]  

(23)

where l_i is the resulting scaled hydrol-

graph, \lambda is a scaling factor, l_i is the base case hydrol-

graph (reobtained for \lambda = 1), and l_i^{(1)} is a thresh-

old value. By using Eq. (23), flow discharges l_i less than or equal to l_i^{(1)} are not altered, whereas flow discharges l_i greater than l_i^{(1)} are altered by scaling the excess l_i - l_i^{(1)} through the factor \lambda. The value of l_i^{(1)} was set equal to 10 m^3 s^-1. The scaling factor \lambda varied from 1 to 15. For \lambda = 10, the peak of l_i has comparable magnitude with the flow discharge having average recurrence of 1000 a. For \lambda \geq 1 the inflow hydrographs l_i display gradually varying flow discharges occurring normally when the water level in the reservoir is low, combined with rapidly varying flow discharges occurring when the water level in the reservoir is either low or high. These circumstances are helpful to identify critical issues in the numerical solution of the reservoir routing Eqs. (1) and (2) that do not come out by using smoother analytic cases such as that considered by Yevjevich [20]. The inflow hydrographs l_i are provided with a temporal resolution

\[ H_j + \Delta H^t \sum_{m=1}^{n-1} \beta_m km < H_B, \]  

(19)
given by Eq. (23) by varying the scaling factor. The CPU time $t$ goes to zero [7], it is assumed that the numerical solution obtained from the RK method by using a very small time step is a valid benchmark solution. A benchmark solution is, therefore, obtained by applying the fourth-order RK method with constant time step $\Delta t = 0.1$ s. The reservoir characteristic relationships $H - S$ and $H - Q$ used to determine this benchmark solution are those shown in Fig. 3. The benchmark water levels and outflow discharges are denoted as $H^{(b)}$ and $Q^{(b)}$, respectively (Figs. 4, 5, and 7).

In the present study, the attention is focused (1) on the possible nonmonotonicity of characteristic functions $H - Q$ such as that shown in Fig. 3(b), (2) on time steps involved in numerical solutions, and (3) on the shape of inflow hydrograph as represented by the hydrographs $I$, given by Eq. (23) by varying the scaling factor $\lambda$. Point (1) is analyzed by comparing results obtained with the nonmonotonic function shown in Fig. 3(b) and its monotonic approximation shown in Fig. 5(a) as mentioned in Section 4.1 (Figs. 4(a) and (b) and 5(a) and (b)). Point (2) is investigated by considering the inflow hydrographs $I$, given by Eq. (23) with $\lambda$ ranging from 1 to 15 (Fig. 9). In the numerical experiments performed, the time step $\Delta t$ and tolerance $\Delta \theta$ are the essential parameters influencing the numerical solutions of Eqs. (1) and (2). The tolerance $\Delta \theta$ is used (directly) in the CK method (Eqs. (15) and (16)). The same tolerance $\Delta \theta$ is used (indirectly) in the LP method to estimate the corresponding tolerance $\epsilon_S$ as

$$
\epsilon_S \approx \Delta \theta.
$$

This equation comes up by observing that the interval of water level $\Delta \theta$ is related to the corresponding interval of the storage $\epsilon_S$ through the equation $\epsilon_S/\Delta \theta \approx ds/dh$ (Fig. 3(a)) and that $ds/dh = A$ on the basis of Eq. (3).

Robustness, accuracy, and efficiency of numerical methods are evaluated by considering three indices: (1) CPU time $t$, (2) relative error on outflow volume $\epsilon_V$, and (3) relative error on peak water level $\epsilon_H$ (Figs. 8 and 9). The CPU time $t$ evaluates the efficiency of simulations. The values of $t$ reported in this study are obtained by using codes written in Matlab and an Intel Core i5–460 M processor (3 MB cache, 2.53 GHz). These values are especially informative in relative terms as the CPU time clearly depends on the performance of algorithm and processor. Relative errors on outflow volume and peak water level with respect to the reservoir bottom are computed as

$$
\epsilon_V = \frac{V - V^{(b)}}{V^{(b)}}
$$

and

$$
\epsilon_H = \frac{H_p - H_p^{(b)}}{H_p^{(b)} - H_b},
$$

where $V$ and $H_p$ are the outflow volume and the peak water level of the numerical solution examined, respectively, $V^{(b)}$ and $H_p^{(b)}$ are the corresponding values in the benchmark solution, and $H_b$ is the level of the reservoir bottom. The relative error $\epsilon_V$ provides a measure of the method robustness. The error $\epsilon_H$ is normally significant when the method has numerical problems related to discontinuity points or breakdown. The relative error $\epsilon_H$ is an indicator of the model accuracy. The quantities $t$, $\epsilon_V$ and $\epsilon_H$ are plotted against the time step $\Delta t$ (Fig. 8) for different values of $\Delta \theta$, namely $\Delta \theta = 10^{-3}$ m in Fig. 8(a)–(c) and $\Delta \theta = 10^{-4}$ m in Fig. 8(d)–(f). The variations of $\epsilon_V$ and $\epsilon_H$ with the scaling factor $\lambda$ are reported in Fig. 9 using a time step $\Delta t$ equal to 30 s (Fig. 9(a) and (b)) or equal to 900 s (Fig. 9(c) and (d)), with $\Delta \theta = 10^{-4}$ m in both the cases.

Robustness and efficiency of the RK-B method are further demonstrated in two additional experiments. Robustness is demonstrated by the comparison between Yevjevich’s [20] analytical solution obtained for a test case reproducing the broad features of the Crostolo River reservoir, the corresponding numerical solution obtained from the RK-B method, and the numerical solution obtained from the RK-B method when the relationship $H - Q$ is perturbed by using a step function as indicated by the inset of Fig. 10. This comparison between analytical, numerical, and perturbed solutions is shown in Fig. 10. Efficiency is demonstrated by a Monte Carlo simulation of peak water levels $H_b$ in the Crostolo River reservoir. A simple hydroclimatic characterization of the drainage basin contributing inflow to the reservoir is provided. Spatially and temporally averaged rainfall hyetographs having a fixed return period $T$ are obtained from the intensity–duration–frequency relationship $i_T(d) = \text{ARF}_d \cdot d^{\alpha - 1}$, where $i_T(d)$ is the rainfall intensity associated to the rainfall duration $d$. ARF is the areal reduction function, $\alpha$ is a scaling coefficient, and $\eta$ is an exponent [4, p. 453]. Excess rainfall hyetographs, obtained by applying a constant runoff coefficient $\phi$, are transformed in (direct) runoff hydrographs by using the concept of time-area curve [1, p. 448]. The contributing areas are determined by assuming that surface runoff concentrates with constant velocity $v$ along topography-based surface flow paths [14]. The values used, for demonstration, in the present study are: $T = 200$ a, ARF = 0.92, $\phi_{200} = 74.12$ mm h$^{-1}$, $n_{200} = 0.349$, $\phi = 0.65$, and $v = 1.13$ m s$^{-1}$. The Monte Carlo simulation is performed by generating initial levels $H_b$ and rainfall durations $d$ randomly from uniform probability distributions on the
The results obtained are shown in Fig. 11. 30,000 points are computed by using the RK-B method with $\Delta t = 1800$ s in approximately 4 h. These points are classed on the basis of four intervals of $H_0$ and represented using different colors as indicated by the bottom left legend of Fig. 11.

6. Discussion

Existing numerical methods for the solution of Eqs. (1) and (2) are not always robust. The LP method (Section 3.1) is unable to handle complex, nonmonotonic outflow rating curves such as that shown, for demonstration, in Fig. 3(b) (Fig. 4(a) and (b)). In fact, the
regula-falsi used in the LP method does not overcome totally convergence problems of root finding algorithms as highlighted by Laurenson [11] and Pilgrim [16] by considering the Newton–Raphson method [17, p. 340]. In Eq. (6), convergence may not be reached when (1) \(X\) increases at constant/nearly-constant \(\Psi(X)\) or (2) \(\Psi(X)\) increases at constant/nearly-constant \(X\). The case (1) implies that the denominator \(\Psi(X_{i+1}^{\text{pert}}) - \Psi(X_{i+1})\) is equal-to close-to zero and results are inaccurate due to spurious oscillations. The case (2) produces slow convergence and large diffusion of numerical errors due to high gradients of the goal function \(\Psi(X)\). In addition, when characteristics functions are nonmonotonic, the terms \(X_{i+1}^{\text{pert}} - X_{i+1}\) and \(\Psi(X_{i+1}^{\text{pert}}) - \Psi(X_{i+1})\) in Eq. (6) may have opposite sign, allowing the second term of the right hand side to be positive. This causes in turn spurious numerical oscillations of the solution \(X_{i+1}^{\text{pert}}\) (Fig. 4(a) and (b)), which prevent the convergence criterion \(\Psi(X_{i+1}^{\text{pert}}) < \epsilon_t\) mentioned in Section 3.1 from being satisfied. This problem occurs even when small time steps (for instance, \(\Delta t = 30\) s in the case shown in Fig. 4(a) and (b)) are used, and it can only be overcome by approximating the nonmonotonic \(H-Q\) function (such as that reported in Fig. 3(b)) with a monotonic function (such as that reported in Fig. 5(a)). This produces, however, a loss of detail in the description of the outflow hydrograph. This loss of detail is not very important in the case shown in Fig. 5b, but it may be more significant in other cases (see, for instance, the cases reported in the insets of Fig. 9(a), (b) and (d)).

When a large time step \(\Delta t\) (larger than 660 s in this application) is used (as a result of computational constraints), the RK and CK methods (Sections 3.2 and 3.3, respectively) may fail to describe critical conditions in which the inflow is rapidly varying and the water level is low (Fig. 4(c)–(f)). These issues may not be relevant as long as the problem of reservoir design is addressed. In this case, time steps may be reduced in order to obtain accurate determinations of the design variables. However, the same issues become relevant when reservoirs, pools, lakes, and depression areas are modeled using a distributed time-continuous model in which the reservoir routing equation is solved in combination with the equations describing the (interacting) surface and subsurface flows (e.g., [2]). In these cases, large time steps are necessary to ensure the numerical efficiency of the overall hydrologic model and robust reservoir routing is required for all the inflow and storage conditions displayed. The RK method (Section 3.2) displays accurate and robust solutions as far as the time step \(\Delta t\) is small, less than or equal to 720 s in this application (Fig. 8). When larger time steps are used, however, the RK method may fail to provide a solution (Figs. 4(c) and (d), 8, and 9). Failures were found to occur when the computed points fall outside the domains of the characteristic functions \(H-S\) and \(H-Q\) (see, for instance, the inset of Fig. 4c). The CK method (Section 3.3) was found to be unable to prevent these problems (Figs. 4(e) and (f), 8, and 9). A direct solution of the problem is therefore provided by introducing a backstepping procedure handling all the situations in which the computed data points are found to fall outside the domains of characteristic functions (Figs. 6 and 7). The resulting RK-B method was found to handle accurately all the cases considered in this study while also displaying a numerical efficiency comparable with that of the RK method and significantly better than that of the CK method (Figs. 7–9).

The CK method is found to require significantly higher CPU times than the other methods, especially for \(\Delta t = 10^{-4}\) m (Fig. 8(a) and (d)). In addition, the CK method does not ensure that a solution is obtained for all the time steps \(\Delta t\) lying between 1 and 900 s. The solution is not obtained, for instance, for 315 < \(\Delta t\) < 480 s when \(\Delta t = 10^{-3}\) m (Fig. 8(a)) or for \(\Delta t > 660\) s when \(\Delta t = 10^{-4}\) m (Fig. 8(d)). The smallest CPU times are required by the RK and RK-B methods for a large lower portion of the interval 1 < \(\Delta t < 900\) s (Fig. 8(a) and (d)). While the RK method does not provide a solution for \(\Delta t > 720\) s, the RK-B method is found to provide a solution for all the considered time steps \(\Delta t\) (Fig. 8(a) and (d)). The RK-B methods is, therefore, found to display CPU times that are less than or approximately equal to the other considered methods, by ensuring a solution for all the considered time steps \(\Delta t\) (Fig. 8(a) and (d)). The RK-B methods is, therefore, found to display CPU times that are less than or approximately equal to the other considered methods, by ensuring a solution for all the considered time steps \(\Delta t\) (Fig. 8(a) and (d)).
(Fig. 8(e)). For $\Delta t = 900$ s and $\Delta t = 10^{-2}$ m, the CK method provides $\epsilon_V = 44\%$. The RK and RK-B methods ensure, under the conditions examined in this study, that $|\epsilon_V| < 0.2\%$ and $|\epsilon_H| < 0.04\%$ (Fig. 8(b), (c), (e) and (f)).

The responses of the considered methods to changes of the scaling coefficient $\lambda$ used in Eq. (23) are evaluated in terms of relative error $\epsilon_V$ (Fig. 9(a) and (c)) and relative error $\epsilon_H$ (Fig. 9(b) and (d)). The LP and LP-M methods are found to display large errors for $\Delta t = 30$ s, especially for $\lambda < 5.5$ (Fig. 9(a) and (b)). Under these circumstances, $96\% < \epsilon_V < 337\%$ and $-22\% < \epsilon_H < 39\%$. The inset in Fig. 9a shows that the large errors $\epsilon_V = 337\%$ obtained from the LP method in the case $\lambda = 2.1$ is due to the inability of the method to predict the recession limb of the outflow hydrograph. For $\Delta t = 900$ s, the errors displayed by the LP and LP-M methods are relatively smaller ($-2\% < \epsilon_V < 5.4\%$, $-4.2\% < \epsilon_H < 88.5\%$), but still significant in some cases (Fig. 9(c) and (d)). The insets in Fig. 9(b) and (d) show that the large relative errors $\epsilon_H$ obtained from the LP method in the case $\lambda = 2.0$ are due to the convergence problems displayed where the $H$–$Q$ function is nonmonotonic. The RK-B method is found to display insignificant relative errors $\epsilon_V$ and $\epsilon_H$ ($|\epsilon_V| < 0.5\%$ and $|\epsilon_H| < 2.3\%$) for all the considered values of $\lambda$. It may, therefore, be concluded that the RK-B method is accurate, computationally efficient, and robust for all the considered time steps $\Delta t$ and for all the inflow hydrographs obtained from Eq. (23) by varying the scaling factor $\lambda$ between 1 and 15.

The relevance of the RK-B method in the area of water resources is further demonstrated by the numerical experiments reported in Figs. 10 and 11. The comparison between the analytical, numerical, and perturbed solutions shown in Fig. 10 reveals that the RK-B method is accurate as it agrees with Yevjevich’s [20] analytical solution, and robust at the same time as it handles the case in which the relationship $H$–$Q$ is perturbed using a step function (Fig. 10, inset). The Monte Carlo analysis reported in Fig. 11 shows that the RK-B method is efficient as it requires a reasonable CPU time of about 4 h to perform 30,000 runs, and robust at the same time as it handles highly variable conditions due to the wide spectra of initial water levels $H_0$ ($H_b \leq H_0 \leq H_s$) and rainfall durations $d$ ($1 \leq d \leq 24$ h). The obtained results reveal that peak water level in the reservoir $H_{po}$ is mostly controlled by the initial condition $H_0$ (due, for instance, to a previous flood) for rainfall durations $d$ greater than about 13.6 h, whereas it is mostly controlled by the inflow hydrograph for smaller rainfall durations. The lower bound of the point cloud displays a jump at $d = 13.6$ h that is enhanced by the nonmonotonic behavior of the function $H$–$Q$ highlighted in Fig. 3. The patterns displayed at the lower bound of the point cloud for $d < 13.6$ h are likely to be connected to the interrelationship between the reservoir dynamics and the dynamics of the contributing drainage basin. This interrelationship will be investigated in a future companion paper.

7. Conclusions

The analysis carried out in this paper reveals that the LP method is unable to handle nonmonotonic $H$–$Q$ functions (Figs. 3(b) and 4(a) and (b)). This is a relevant problem in all cases in which a detailed description of the transition between free-surface to pressurized flow through the bottom outlet of a reservoir is required or, more in general, when complex $H$–$Q$ functions have to be considered. The standard RK method fails to provide a solution when large time steps ($\Delta t > 720$ s in this application) are used and $H$ approaches $H_b$ (Fig. 4(c) and (d)). The CK variant of the RK method does not solve this problem (Figs. 4(e) and (f)). Robust reservoir routing is obtained by using the (fourth-order) RK method in combination with a simple backstepping procedure controlling the time step $\Delta t$ when $H$ approaches $H_b$ in such a way that the determined values of $H$ do not fall outside the domains of the characteristic reservoir functions $H$–$S$ and $H$–$Q$ (Figs. 6–11). This strategy yields an accurate, robust, and efficient reservoir routing method that can be safely used in real time flood risk management and in Monte Carlo analysis of complex hydroclimatic scenarios based on the use of detailed distributed catchment models.

Acknowledgments

This study was carried out under the research program PRIN 2010–2011 (Grant 2010JHF437) funded by the Italian Ministry of Education, University, and Research. Paola Gambarelli is acknowledged for participating to the writing of part of the codes used in the present study. The authors thank the three anonymous reviewers for comments that led to improvements in the manuscript.

References