Deltares

Memo

To

anyone interested

Date	Our reference	Number of pages
March-April, June-July, 2020	SUPSUB pseudo-time stepping	12

Contact person	Direct line	E-mail
Mart Borsboom	+31 (0)88 335 8435	mart.borsboom@deltares nl

Subject

The linearization-error dependent pseudo-time step procedure of SUPSUB revisited (WORK IN PROGRESS)

In SUPSUB the nonlinear 1D shallow-water equations are discretized fully implicitly in time ($\Rightarrow$ no stability restriction on the time step). The resulting nonlinear systems of equations are solved by a modified Newton method. For efficiency, an exact and nearly full linearization is used. For robustness combined with efficiency, linearization-error dependent pseudo-time stepping is added. The technique of adding a pseudo-time step is known as pseudo-transient continuation, see e.g. Kelley and Keyes (1998); Knoll and Keyes (2004); Hicken and Zingg (2009); Savant *et al.* (2011); Ceze and Fidkowski (2013); Mavriplis (2018).

The Newton solution method for nonlinear problems[1] generally converges very fast (quadratic convergence speed), *at the condition* that the initial guess is sufficiently close to the final solution. Newton is therefore an excellent basis for an efficient nonlinear solution method. Straightforward Newton is not robust, however. Its performance depends on a reliable linearization, i.e., on sufficiently small linearization errors. It converges very slowly, or not at all, when the initial guess is too far from the final solution causing very/unacceptably large (initial) linearization errors.

Since linearization errors scale with the solution correction squared (and higher powers), their effect can be made sufficiently small by ensuring small enough solution corrections. The effect of the linearization itself scales, of course, linearly with the solution correction. The purpose of pseudo-time stepping is to reduce the size of the solution correction where and when necessary in order to ensure small enough linearization errors and hence guarantee robustness. In Section 5 of Borsboom (2019), *Iterative solution algorithm*, the condition for the size of the pseudo-time step is derived that ensures a sufficiently small linearization error.

Summary

We present a modified Newton solver designed to solve a nonlinearly implicit time discretization (or nonlinear steady-state discretization) of a flow/transport problem very fast (note that this implies stability), *at the condition* that the Jacobian and its derivatives are sufficiently smooth. This is realized by adding a linearization-error-dependent pseudo-time-step technique to the Newton linearization of the model equations. The technique is generally applicable, and is developed here for the 1D shallow-water equations. It includes several (small) modifications, additions and refinements compared to the very successful method developed for and applied in SOBEK-RE and SUPSUB, cf. Borsboom (1999) and Section 4.2 of Borsboom (2001), *Iterative*

[1] For linear problems Newton reduces to a direct method, i.e., full convergence in one iteration.

solution algorithm. Together, these modifications, additions and refinements are a realization of the improvements mentioned in Section 5 of Borsboom (2019), *Iterative solution algorithm*.

The model equations are presented in Section 1, their time discretization and linearization are outlined in Section 2, while the construction/derivation of the linearization-error-dependent pseudo-time-step is outlined in Section 3.

1 The model equations

The 1D shallow-water equations that are solved in SUPSUB are:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \ , \tag{1}$$

$$\frac{\partial Q}{\partial t} + \frac{\partial Q^2/A}{\partial x} + gA\frac{\partial \zeta}{\partial x} = -g\frac{PQ|Q|}{C^2A^2} + \frac{\partial}{\partial x}\left(\nu_{\mathrm{art}} A \frac{\partial Q/A}{\partial x}\right) \ . \tag{2}$$

The unknowns of this system of equations are wetted cross-sectional area $A = A(x,t)$ and discharge $Q = Q(x,t)$. Cross-section-averaged flow velocity u is equal to Q/A.

Water level ζ and wetted perimeter P are considered functions of A. A and P are determined by the geometry of the channel specified in terms of its width $W = W(x,z)$ as a function of the horizontal coordinate along the channel axis x and the vertical coordinate z. Defining $W = 0$ below the bottom at $z = z_b$, we have:

$$A = \int_{-\infty}^{\zeta} W \,\mathrm{d}z \ , \ P = \int_{-\infty}^{\zeta} \sqrt{4 + \left(\partial W/\partial z\right)^2}\,\mathrm{d}z \ . \tag{3}$$

Because $W \geq 0$, A is a monotonically increasing function of ζ and hence the function $A = A(\zeta)$ is reversible to $\zeta = \zeta(A)$, from which we obtain $P = P(\zeta) = P(\zeta(A)) = P(A)$. Since A and not ζ is the geometry-related unknown, $\zeta = \zeta(A)$ and $P = P(A)$ are the two geometry functions used in SUPSUB.

The viscosity term of SUPSUB (last term in the right-hand side of (2)) is primarily part of the applied two-step numerical modeling technique, cf. Figure 1 in Borsboom (2019), although it can be used for physical modeling purposes as well. The viscosity term with discretization-error dependent artificial viscosity coefficient ν_{art} is part of the regularization that ensures sufficient smoothness and hence ensures a problem that is easy to solve numerically, cf. Section 1.1 of Borsboom (2019), *The beneficial effects of smoothing*.

2 Discretization in time and linearization

When a theta time-integration method is used (the method currently used in SUPSUB), the fully implicit time discretization of (1) and (2) leads to the following system of equations per time step[2] (for steady-state computations set $\theta = 1$ and $1/\Delta t = 0$; the latter has been made

[2]This time discretization is not fully in line with the piecewise linear compatible discretization method that we apply in space. Such a time discretization would be obtained by defining (second-order accurate) piecewise linear function approximations in time (i.e., $Q(t) = ((t^n - t)Q^{n-1} + (t - t^{n-1})Q^n)/\Delta t$, etc., $t^{n-1} \leq t \leq t^n$), followed by a sufficiently accurate (i.e., higher than second-order accurate) integration over each time step, followed by division by the time step to get the discretized equations in a more convenient form.

For a term like Q^2/A, applying the fourth-order accurate two-point Gauss integration rule (simpler than exact integration), this would lead to $1/2\big((Q^{n-1/2-\sqrt{1/12}})^2/A^{n-1/2-\sqrt{1/12}} + (Q^{n-1/2+\sqrt{1/12}})^2/A^{n-1/2+\sqrt{1/12}}\big)$, and *not* to the evaluation of Q^2/A at $t = t^{n-1/2}$ ($t = t^{n-1+\theta}$) as we apply now. Since we do not know (yet) if application of a compatible discretization in time would be the best thing to do (depending on the application, a DIRK2 or an IMEX time integration scheme may have better properties, cf. Section 3 of Borsboom (2019), *Time integration scheme*), for the time being we stick to the time integration already implemented in SUPSUB.

possible by our division of the time-integrated equations by Δt):

$$\frac{A^n - A^{n-1}}{\Delta t} + \frac{\partial Q^{n-1+\theta}}{\partial x} = 0 \,, \tag{4}$$

$$\begin{aligned}
&\frac{Q^n - Q^{n-1}}{\Delta t} + \frac{\partial (Q^{n-1+\theta})^2 / A^{n-1+\theta}}{\partial x} + g A^{n-1+\theta} \frac{\partial \zeta^{n-1+\theta}}{\partial x} \\
&= -g \frac{P^{n-1+\theta} Q^{n-1+\theta} |Q^{n-1+\theta}|}{C^2 (A^{n-1+\theta})^2} \\
&\quad + \frac{\partial}{\partial x} \Big(\nu_{\text{art}}^{n-1+\theta} \big(\frac{\partial Q^{n-1+\theta}}{\partial x} - \frac{Q^{n-1+\theta}}{A^{n-1+\theta}} \frac{\partial A^{n-1+\theta}}{\partial x} \big) \Big) \,,
\end{aligned} \tag{5}$$

with $A^{n-1+\theta} = (1-\theta) A^{n-1} + \theta A^n$ and $Q^{n-1+\theta} = (1-\theta) Q^{n-1} + \theta Q^n$, and with the A- and Q-dependent variables $\zeta^{n-1+\theta} = \zeta(A^{n-1+\theta})$, $P^{n-1+\theta} = P(A^{n-1+\theta})$, and $\nu_{\text{art}}^{n-1+\theta} = \nu_{\text{art}}(A^{n-1+\theta}, Q^{n-1+\theta}, \zeta^{n-1+\theta})$.

Once discretized in space (which we omit for convenience), the equations (4) and (5) form a nonlinear system of equations for the unknowns A^n and Q^n per grid point at the next time level or steady state n. NB, the application of another fully implicit time integration method may be more suitable, cf. Section 3 of Borsboom (2019), *Time integration scheme*, but will lead to a similar system of equations for the unknowns A^n and Q^n.

Nonlinear system of equations (4) and (5) cannot be solved directly[3]. An iterative solution method is required where subsequent approximations $A^{n,m}$ and $Q^{n,m}$, $m = 1, \ldots$, of A^n and Q^n are determined until convergence, starting from some suitably chosen initial guess $A^{n,0}$ and $Q^{n,0}$, cf. Section 5.1 of Borsboom (2019), *Initializations*. Hence we seek a procedure to determine a sequence of iterative solution corrections $\Delta A^{n,m} = A^{n,m} - A^{n,m-1}$ and $\Delta Q^{n,m} = Q^{n,m} - Q^{n,m-1}$, $m = 1 \ldots$. The corrections should be such that the iterands $A^{n,m} = A^{n,m-1} + \Delta A^{n,m}$ and $Q^{n,m} = Q^{n,m-1} + \Delta Q^{n,m}$, $m = 1 \ldots$, converge as quickly and as efficiently as possible to the solutions that are sought, i.e., to A^n and Q^n for all time levels or steady states $n = 1, \ldots$ of all problems of practical interest. Such a procedure is obtained from a suitable linearization of (4) and (5), constructed from the linearization of its components:

$$A^{n-1+\theta} \approx A^{n-1+\theta,m-1} + \theta \Delta A^{n,m} \,, \tag{6}$$

$$Q^{n-1+\theta} \approx Q^{n-1+\theta,m-1} + \theta \Delta Q^{n,m} \,, \tag{7}$$

$$\begin{aligned}
\zeta^{n-1+\theta} \approx \zeta^{n-1+\theta,m-1} &+ \theta \left. \frac{\partial \zeta}{\partial A} \right|^{n-1+\theta,m-1} \Delta A^{n,m} \\
&+ \frac{\theta^2}{2} \left. \frac{\partial^2 \zeta}{\partial A^2} \right|^{n-1+\theta,m-1} \big(\Delta A^{n,m} \big)^2 \,,
\end{aligned} \tag{8}$$

$$P^{n-1+\theta} \approx P^{n-1+\theta,m-1} + \theta \left. \frac{\partial P}{\partial A} \right|^{n-1+\theta,m-1} \Delta A^{n,m} \,. \tag{9}$$

NB, the nonlinear term in (8) (the one that is quadratic in $\Delta A^{n,m}$) is not part of the linearization, but will be used in the next section for the scaling of the pseudo-time step.

[3] Note that (4) is actually linear in the unknowns A^n and Q^n, i.e., this equation could be solved directly if any complementary linear combination of A^n and Q^n would be known, which is not the case. Only the nonlinear combination of A^n and Q^n enclosed in (5) is available, i.e., what matters is the nonlinearity of the system as a whole, which because of our choice of unknowns is entirely due to the nonlinear character of momentum equation (5). If instead of A we would have considered unknown ζ with $A = A(\zeta)$, or if instead of Q we would have considered unknown u with $Q = Au$ (or both), (4) would have been nonlinear as well.

From the first equation in (3) we obtain:

$$1 = \frac{\partial A}{\partial A} = \frac{\partial \zeta}{\partial A} W_{z=\zeta} \;\Rightarrow\; \frac{\partial \zeta}{\partial A} = \frac{1}{W_{z=\zeta}} \;, \tag{10}$$

$$0 = \frac{\partial^2 A}{\partial A^2} = \frac{\partial^2 \zeta}{\partial A^2} W_{z=\zeta} + \frac{\partial \zeta}{\partial A} \left.\frac{\partial W}{\partial A}\right|_{z=\zeta} \;\Rightarrow\; \frac{\partial^2 \zeta}{\partial A^2} = - \left.\frac{W_z}{W^3}\right|_{z=\zeta} \;. \tag{11}$$

The latter result has been obtained using $(\partial W/\partial A)_{z=\zeta} = (\partial W/\partial z/W)_{z=\zeta} = (W_z/W)_{z=\zeta}$.

Likewise, from the second equation in (3) we obtain:

$$\frac{\partial P}{\partial A} = \frac{\partial \zeta}{\partial A} \sqrt{4 + \big((W_z)_{z=\zeta}\big)^2} = \frac{1}{W_{z=\zeta}} \sqrt{4 + \big((W_z)_{z=\zeta}\big)^2} \;. \tag{12}$$

We will however use:

$$\frac{\partial P}{\partial A} = \frac{\partial \zeta}{\partial A} (P_z)_{z=\zeta} = \frac{1}{W_{z=\zeta}} (P_z)_{z=\zeta} \;, \tag{13}$$

exploiting the fact that P can be viewed as a function of z whose value is taken at the free surface, i.e., $P(z=\zeta) = P(\zeta(A)) = P(A)$. The reason for this is the interpolation between cross-sectional profiles, required when grid points do not coincide with the locations where cross-sectional profiles are specified. That interpolation may be such that at grid points the second equation in (3) applies approximately but not strictly. In that case linearization coefficient (12) is only an approximation, while expression (13) is always exact. Note that (12) is contained in (13).

Substitution of (6), (7), (8) and (9) in (4) and (5), after substitution of (10) in (8) and of (13) in (9), yields the linearization (omit the superscript n, m of the solution corrections ΔA and ΔQ in the left-hand sides; all other variables without superscript are at $n-1+\theta, m-1$, which denotes variables at time level $n-1+\theta$ with the θ-weighted part at level n at previous iteration level $m-1$):

$$\frac{\Delta A}{\Delta t} + \frac{\Delta A}{\Delta t_{\text{pseu}}} + \theta \frac{\partial \Delta Q}{\partial x} = - \frac{A^{n,m-1} - A^{n-1}}{\Delta t} - \frac{\partial Q}{\partial x} = \text{res}_{\text{cont}}^{n,m-1} \;, \tag{14}$$

$$\begin{aligned}
&\frac{\Delta Q}{\Delta t} + \frac{\Delta Q}{\Delta t_{\text{pseu}}} + \theta \frac{\partial}{\partial x}\Big(\frac{2Q}{A}\Delta Q - \frac{Q^2}{A^2}\Delta A\Big) \\
&\quad + \theta g \left(\Delta A \frac{\partial \zeta}{\partial x} + A \frac{\partial}{\partial x}\left(\frac{\Delta A}{W_{z=\zeta}}\right)\right) \\
&\quad + \theta \frac{g}{C^2}\left(\frac{2P|Q|}{A^2}\Delta Q - \frac{2PQ|Q|}{A^3}\Delta A + \frac{(P_z)_{z=\zeta} Q|Q|}{W_{z=\zeta} A^2}\Delta A \right) \\
&\quad - \theta \frac{\partial}{\partial x}\left(\nu_{\text{art}}\Big(\frac{\partial \Delta Q}{\partial x} - \big(\frac{\Delta Q}{A} - \frac{Q \Delta A}{A^2}\big)\frac{\partial A}{\partial x} - \frac{Q}{A}\frac{\partial \Delta A}{\partial x}\Big)\right) \\
&\quad = - \frac{Q^{n,m-1} - Q^{n-1}}{\Delta t} - \frac{\partial Q^2/A}{\partial x} - gA\frac{\partial \zeta}{\partial x} - \cdots = \text{res}_{\text{mom}}^{n,m-1} \;,
\end{aligned} \tag{15}$$

with in the left-hand side the addition of a pseudo-time step for optimization of the convergence process, cf. the next section.

Remarks/notes:

- At present, values of W_z (including $(W_z)_{z=\zeta}$) computed inside subroutine `geometry` are only used inside that subroutine for the computation of wetted perimeter P, cf. the second expression in (3). Since the $(W_z)_{z=\zeta}$ are not exported outside `geometry` (something that we seem to have copied from SOBEK-RE), they are not available for the computation of the last term in the second line of (15) using $(P_z)_{z=\zeta} = \sqrt{4 + ((W_z)_{z=\zeta})^2}$, i.e., at present the linearization of P is not included in SUPSUB. In retrospect this omission is of course unacceptable[4].
- The $(W_z)_{z=\zeta}$ are also required, but not available hence presently omitted, in the pseudo-time stepping procedure developed/presented in the next Section, cf. (17). In other words, not only the linearization but also the pseudo-time stepping procedure is incomplete, just like in SOBEK-RE. This may have a strong negative effect on the speed and robustness of the convergence process.
  From (17) we obtain that the omission is acceptable provided that $|A(W_z)_{z=\zeta}/(W_{z=\zeta})^2|$ is guaranteed to be always sufficiently smaller than 1 everywhere. It is clear that in pipe-flow simulations this condition will often be violated. For nearly fully filled pipes we have that wetted cross-sectional area A is close to its maximum value, while the width of the pipe at the free surface $W_{z=\zeta}$ is then very small and rapidly decreasing ($(W_z)_{z=\zeta} \ll -1$), i.e., we then have $|A(W_z)_{z=\zeta}/(W_{z=\zeta})^2| \gg 1$. It explains the difficult convergence that we have often observed at locations where a pipe is nearly fully filled. (How nice!)
- The linearization of $\nu_{\rm art}$ is not included in (15). It is not feasible to linearize the very complex expression of the artificial viscosity coefficient. Instead, $\nu_{\rm art}$ is updated explicitly, using underrelaxation[5] to ensure stability.

3 Pseudo-time stepping

We are now ready for the construction of the convergence-error dependent pseudo-time stepping. First the assumptions:

- When *all* variables are smooth in space (i.e., solution A, Q, solution correction ΔA, ΔQ, but also geometry variables P, $W_{z=\zeta}$ and $(P_z)_{z=\zeta}$), then *all* terms in (14) and (15) are smooth in space. In that case Newton may be expected to converge nicely. Since the solution A, Q and the geometry variables P, $W_{z=\zeta}$, $(P_z)_{z=\zeta}$ should always be smooth in space (by construction!), it is therefore reasonable to assume that Newton convergence problems can *only* occur when the solution correction ΔA, ΔQ is *not* smooth in space.
  $\Rightarrow$ Only that case needs to be considered.
- The contribution of the bottom friction term to the nonlinear convergence process may be expected to be smooth, for it is free of derivatives of ΔA and ΔQ and tends to dampen perturbations (also those within a convergence process).
  $\Rightarrow$ Not a primary cause for convergence problems due to linearization errors.
- Viscosity terms (likewise for diffusion terms) are primarily smoothing terms. For most computational flow and transport models this applies to the space discretization of such terms as well, because of their (close to) symmetry and diagonal dominance. When in addition the discretization in time is sufficiently implicit, this smoothing property holds regardless of the physical time step Δt. This applies to the discretization of the viscosity term in SUPSUB that is fully and properly linearized, cf. the third line of (15). As a result, ΔA, ΔQ are smoothed in space wherever and whenever some artificial viscosity is present.

[4] It seems best to add the computation of $(W_z)_{z=\zeta}$ and $(P_z)_{z=\zeta}$ (with the latter not necessarily exactly equal to $\sqrt{4 + ((W_z)_{z=\zeta})^2}$, cf. the explanation after (13)) to subroutine `gethypar` that takes care of the interpolation between cross-sectional profiles. This should make it fairly straightforward to ensure that the values of $(W_z)_{z=\zeta}$ and $(P_z)_{z=\zeta}$ at grid points are in agreement with the interpolated values of $W_{z=\zeta}$ and $P_{z=\zeta}$ at grid points.

[5] Underrelaxation of $\nu_{\rm art}$ uses a fixed underrelaxation coefficient. What about the use of a convergence-error dependent underrelaxation coefficient, like we do for the boundary conditions? However, the boundary conditions are solved implicitly (using linearization where required), so for boundary conditions underrelaxation serves as a sort of pseudo-time stepping. Conclusion: for explicitly updated $\nu_{\rm art}$ the use of a fixed underrelaxation coefficient is likely to be the best thing to do.

$\Rightarrow$ Not a cause for convergence problems due to linearization errors either.

Conclusion: for the study of the effect of the leading linearization errors and the scaling of the pseudo-time step it is sufficient to consider (note that ΔA and ΔQ are still at n, m and that all other variables are at $n-1+\theta, m-1$ unless indicated otherwise):

$$\frac{\Delta A}{\Delta t_{\text{pseu}}} + \frac{\Delta A}{\Delta t} + \theta \frac{\partial \Delta Q}{\partial x} = \text{res}_{\text{cont}}^{n,m-1} \,, \tag{16}$$

$$\begin{aligned} &\frac{\Delta Q}{\Delta t_{\text{pseu}}} + \frac{\Delta Q}{\Delta t} + 2\theta\Big(\frac{Q}{A} + \theta\big(\frac{\Delta Q}{A} - \frac{Q\Delta A}{A^2}\big)\Big)\frac{\partial \Delta Q}{\partial x} \\ &\qquad - \theta\Big(\frac{Q^2}{A^2} + 2\theta\big(\frac{Q\Delta Q}{A^2} - \frac{Q^2\Delta A}{A^3}\big)\Big)\frac{\partial \Delta A}{\partial x} \\ &\qquad + \theta\frac{g}{W_{z=\zeta}}\Big(A + \theta\big(1 - \frac{A(W_z)_{z=\zeta}}{(W_{z=\zeta})^2}\big)\Delta A\Big)\frac{\partial \Delta A}{\partial x} = \text{res}_{\text{mom}}^{n,m-1} \,. \end{aligned} \tag{17}$$

The first equation, which is linear in the solution corrections ΔA, ΔQ (hence no linearization-error terms to be added), is equal to (14). The second equation is obtained from (15) by means of the following modifications of its left-hand side:

- omit the contributions from the bottom friction term and the viscosity term (assumed to be not a cause for convergence problems),
- add all terms that are second order in ΔA and ΔQ; these are the lowest-order nonlinear terms omitted in the Newton linearization that together form the leading linearization error when ΔA and ΔQ are small enough, which is to be ensured by a proper scaling of pseudo-time step Δt_{pseu},
- omit all derivatives of A, Q and ζ (negligible because they are smooth in space while the derivatives of ΔA and ΔQ may be highly non-smooth).

The derivation of (16) and (17) is identical to how we obtained similar equations in Borsboom (1999) for the construction of a proper scaling of the pseudo-time step of SOBEK-RE. The difference is due to the different choice of unknowns. They are ζ and Q in SOBEK-RE while in SUPSUB they are A and Q, hence in the SOBEK-RE linearization and its error we have considered A as a function of ζ while here the linearization and linearization error of ζ as a function of A are in order, cf. (8).

Following the procedure as described in Borsboom (1999), we write (16) and (17) in the compact form:

$$\Big(\frac{1}{\Delta t_{\text{pseu}}} + \frac{1}{\Delta t}\Big)\begin{pmatrix}\Delta A \\ \Delta Q\end{pmatrix} + \mathbf{N}\frac{\partial}{\partial x}\begin{pmatrix}\Delta A \\ \Delta Q\end{pmatrix} = \begin{pmatrix}\text{res}_{\text{cont}}^{n,m-1} \\ \text{res}_{\text{mom}}^{n,m-1}\end{pmatrix} \,. \tag{18}$$

The convergence behaviour of this system is determined by matrix $\mathbf{N}$. Introducing[6]:

$$\begin{aligned} u &= \frac{Q}{A} \,, & \Delta u &= \theta\big(\frac{\Delta Q}{A} - \frac{Q\Delta A}{A^2}\big) \,, \\ c &= \sqrt{g\frac{A}{W_{z=\zeta}}} \,, & \Delta c &= \theta g\frac{(W_{z=\zeta})^2 - A(W_z)_{z=\zeta}}{2c(W_{z=\zeta})^3}\Delta A \,, \end{aligned} \tag{19}$$

(NB, ΔA and ΔQ are still at n, m while all other variables are still at $n-1+\theta, m-1$), that matrix can be written as:

$$\mathbf{N} = \theta\begin{pmatrix} 0 & 1 \\ -u^2 - 2u\Delta u & 2u + 2\Delta u \\ +c^2 + 2c\Delta c & \end{pmatrix} \,.$$

[6] Note that Δu and Δc are the linearization errors of u and c.

Adding some terms of $O(\Delta A^2)$, $O(\Delta A \Delta Q)$ and $O(\Delta Q^2)$ that should be negligibly small (by construction), $\mathbf{N}$ can conveniently be diagonalized by means of:

$$\mathbf{N} = \frac{-\theta}{2(c+\Delta c)} \begin{pmatrix} -1 & 1 \\ c+\Delta c & c+\Delta c \\ -u-\Delta u & +u+\Delta u \end{pmatrix} \begin{pmatrix} u+\Delta u & 0 \\ -c-\Delta c & \\ 0 & u+\Delta u \\ & +c+\Delta c \end{pmatrix} \begin{pmatrix} u+\Delta u & -1 \\ +c+\Delta c & \\ u+\Delta u & -1 \\ -c-\Delta c & \end{pmatrix} . \tag{20}$$

From this result we immediately obtain that the linearization errors in (16) and (17) (and hence the effect of linearization errors on the convergence behavior of (14) and (15)) can be expected to be small when:

$$|\Delta u|, |\Delta c| \ll |u|, c \; . \tag{21}$$

This result is identical to the result obtained in Borsboom (1999).

From (18) we obtain that solution correction ΔA, ΔQ can be made arbitrarily small by ensuring the diagonal $1/\Delta t_{\text{pseu}} + 1/\Delta t$ in the left-hand side to be sufficiently large. This means that condition (21) can be satisfied by taking $1/\Delta t_{\text{pseu}}$ sufficiently large, i.e., by taking the sum of the inverse CFL numbers proportional to (or larger than) the relative solution correction, for example by applying:

$$\frac{1}{CFL_{\text{pseu}}} = \max\left(0 \; , \; \varepsilon_{\text{pseu}} \frac{|\Delta u| + |\Delta c|}{|u| + c} - \frac{1}{CFL}\right) , \tag{22}$$

with $\varepsilon_{\text{pseu}} \ll 1$ some scaling parameter and with $1/CFL = \Delta x/\big((|u|+c)\Delta t\big)$ the inverse CFL number based on physical time step Δt (equal to zero in steady-state computations).

Expression (22) is the same as the one proposed in Borsboom (1999), with the exception of the term $1/CFL$ since back then we did not take the effect of the physical time step into account. It is expected that with this addition the pseudo-time step scaling (22) will be better suited to unsteady computations, i.e., will then lead to faster convergence, especially when $1/CFL$ is rather large. NB, instead of (22) we could also apply (identical at the continuous level, but different at the discretized level):

$$\frac{\Delta x}{\Delta t_{\text{pseu}}} = \max\left(0 \; , \; \varepsilon_{\text{pseu}}(|\Delta u| + |\Delta c|) - \frac{\Delta x}{\Delta t}\right) . \tag{23}$$

In Borsboom (1999) it is suggested to replace $|\Delta u|$ and $|\Delta c|$ in (22) or (23) by $\theta\big(|\Delta Q|/A + |Q\Delta A|/A^2\big)$ and $\theta g/(2cW_{z=\zeta})\big(1 + A|(W_z)_{z=\zeta}|/(W_{z=\zeta})^2\big)|\Delta A|$, i.e., to use instead of $|\Delta u|$ and $|\Delta c|$ the sum of the absolute value of their *components*, cf. (19). That will lead to larger, more conservative values of $1/CFL_{\text{pseu}}$ cq. $\Delta x/\Delta t_{\text{pseu}}$ and hence to larger robustness. Sounds like a good idea, i.e., something that is worth trying if it hasn't been applied yet in SUPSUB (to be verified).

3.1 Discretization in space

As for the discretization in space of the pseudo-time steps in (14) and (15), it may be advantageous to apply them upwind-like for enhanced stability. From the right eigenvectors in (20) we obtain that pseudo-time step $\big((u+c)\Delta A - \Delta Q\big)/\Delta t_{\text{pseu}}$ should be discretized upwind according to the direction of $u-c$, while pseudo-time step $\big((u-c)\Delta A - \Delta Q\big)/\Delta t_{\text{pseu}}$ should be discretized upwind according to the direction of $u+c$. Putting things together, we obtain

the upwind space discretization of the pseudo-time step:

$$\begin{aligned}\frac{\Delta A}{\Delta t_{\text{pseu}}} &= \frac{c+u}{2c}\frac{\Delta A_{u-c}}{\Delta t_{\text{pseu}}} + \frac{c-u}{2c}\frac{\Delta A_{u+c}}{\Delta t_{\text{pseu}}} - \frac{1}{2c}\Big(\frac{\Delta Q_{u-c}}{\Delta t_{\text{pseu}}} - \frac{\Delta Q_{u+c}}{\Delta t_{\text{pseu}}}\Big)\ ,\\ \frac{\Delta Q}{\Delta t_{\text{pseu}}} &= \frac{c-u}{2c}\frac{\Delta Q_{u-c}}{\Delta t_{\text{pseu}}} + \frac{c+u}{2c}\frac{\Delta Q_{u+c}}{\Delta t_{\text{pseu}}} - \frac{c^2-u^2}{2c}\Big(\frac{\Delta A_{u-c}}{\Delta t_{\text{pseu}}} - \frac{\Delta A_{u+c}}{\Delta t_{\text{pseu}}}\Big)\ ,\end{aligned} \tag{24}$$

where the subscripts $u-c$ and $u+c$ indicate variables to be discretized upwind in the direction of $u-c$ and $u+c$ respectively.

The straightforward application of upwind, e.g.:

$$\Delta A_{u\pm c} = \begin{cases}(\Delta A_{i-1} + \Delta A_i)/2 \ , & u \pm c < 0\\ (\Delta A_i + \Delta A_{i+1})/2 \ , & u \pm c \geq 0\end{cases}\ , \tag{25}$$

(similar for $\Delta Q_{u\pm c}$) is not recommended, as it introduces a discontinuous transition in the space discretization of the left-hand side of (14) and (15) at $u \pm c = 0$. Although this does not affect the converged solution (determined by vanishing right-hand sides $\lim_{m\to\infty} \text{res}_{\text{cont}}^{n,m-1} = 0$ and $\lim_{m\to\infty} \text{res}_{\text{mom}}^{n,m-1} = 0$), the abrupt changes in the iteration process that this could lead to are expected to have a negative effect on the convergence process. A smooth transition between the upwind directions is obtained by applying for example:

$$\begin{aligned}\Delta A_{u\pm c} = &\frac{1-\tanh\big((u/c\pm 1)/\alpha\big)}{2}\frac{\Delta A_{i-1}+\Delta A_i}{2}\\ &+\frac{1+\tanh\big((u/c\pm 1)/\alpha\big)}{2}\frac{\Delta A_i+\Delta A_{i+1}}{2}\end{aligned} \tag{26}$$

(similar for $\Delta Q_{u\pm c}$), with $\alpha > 0$ a scaling parameter. In the limit of $\alpha \to 0$ this expression becomes equal to (25).

Inspection of the SUPSUB code (subroutine `c_flipse`) learns that (26) has been implemented (with hard-coded $\alpha = 0.1$) but is not used (commented out), presumably because of its insufficient performance. The reason for this is not indicated in the code, but may be due to the fact that (26) (same for (25)) would give a pseudo-time step space discretization based on *averages*. Such pseudo-time stepping will hardly be effective in limiting *non-smooth* solution corrections ΔA and ΔQ, while that is exactly the purpose of pseudo-time stepping, cf. the first item in the introduction of Section 3. It may therefore be better to apply:

$$\Delta A_{u\pm c} = \frac{1-\tanh\big(\alpha(u/c\pm 1)\big)}{2}\Delta A_{i-1} + \frac{1+\tanh\big(\alpha(u/c\pm 1)\big)}{2}\Delta A_{i+1}\ ,$$

or perhaps:

$$\Delta A_{u\pm c} = \Delta A_i + \tanh\big(\alpha(u/c\pm 1)\big)\frac{\Delta A_{i+1}-\Delta A_{i-1}}{2} \tag{27}$$

(similar for $\Delta Q_{u\pm c}$). Both are the result of the current review of the SUPSUB iterative solution procedure and have never been considered for implementation. Neither of them looks very attractive though, swaying the upwind pseudo-time step from grid point $i-1$ to $i+1$ while skipping, completely or partly, the pseudo-time step at i. On the other hand, they seem to be an improvement over (26) and may be worth trying.

A more appealing option for the discretization in space of the pseudo-time step is a central discretization with added smoothing, e.g.:

$$\begin{aligned}\frac{\Delta x}{\Delta t_{\text{pseu}}}\Delta A = &\frac{\Delta x_{i-1/2}}{2(\Delta t_{\text{pseu}})_{i-1/2}}\big(\Delta A_i + 2\beta(\Delta A_i - \Delta A_{i-1})\big)\\ &+\frac{\Delta x_{i+1/2}}{2(\Delta t_{\text{pseu}})_{i+1/2}}\big(\Delta A_i + 2\beta(\Delta A_i - \Delta A_{i+1})\big)\end{aligned} \tag{28}$$

(similar for $\Delta Q/\Delta t_{\rm pseu}$), with $\beta > 0$ a smoothing parameter, where we have added how grid size Δx is to be discretized[7].

A favorable property of discretization (28) is that the pseudo-time step that is effectively applied to the wiggly components of solution correction ΔA, ΔQ (components for which we have $-\Delta A_{i-1} \approx \Delta A_i \approx -\Delta A_{i+1}$ and $-\Delta Q_{i-1} \approx \Delta Q_i \approx -\Delta Q_{i+1}$) equals $\approx \Delta t_{\rm pseu}/(1 + 4\beta)$. That is $\approx 1 + 4\beta$ smaller than the pseudo-time step applied to the smooth components of solution correction ΔA, ΔQ (components for which we have $\Delta A_{i-1} \approx \Delta A_i \approx \Delta A_{i+1}$ and $\Delta Q_{i-1} \approx \Delta Q_i \approx \Delta Q_{i+1}$). In other words, by taking β large enough (28) enhances the prevention of non-smooth components in the solution corrections ΔA and ΔQ, which are thought to be the primary cause of perturbations of the Newton convergence process, cf. the first item in the introduction of Section 3. Discretization (28) can be viewed as an improvement over the 'standard' pseudo-time step technique consisting of just adding to the model equations the discretization of a pseudo-time derivative in iterative (i.e., pseudo) time.

The right-hand side of (28) is effectively a finite volume discretization of the conservative implicit smoothing operator $1/\Delta t_{\rm pseu}\Delta A - \beta\partial/\partial x\big(\Delta x^2/\Delta t_{\rm pseu}\partial\Delta A/\partial x\big)$, which defines an implicit second-order low-pass filter. The same type of smoothing/low-pass filtering is applied elsewhere in SUPSUB, e.g., to ensure that artificial viscosity coefficient $\nu_{\rm art}$ is sufficiently smooth but also to ensure that adapted grids are sufficiently smooth. The artificial viscosity term added to the momentum equation (third line of (2)) is in fact also an implicit low-pass filter[8], designed to ensure a sufficiently smooth solution under all circumstances. Simple explicit approximations of such a filter[9] are currently used to ensure reasonably smooth geometry variables $\zeta(A)$ (and hence $W_{z=\zeta}$) and $P(A)$ (computation and subsequent smoothing of $(P_z)_{z=\zeta}$ not yet implemented).

Another favorable property of (28) is that the pseudo-time step space discretization matrix is not only positive definite, but also symmetric. The latter is the result of having a smoothing operator in conservative form. The pseudo-time step with added implicit smoothing currently implemented in SUPSUB equals (28) with $\Delta x_{i-1/2}/(\Delta t_{\rm pseu})_{i-1/2}$ and $\Delta x_{i+1/2}/(\Delta t_{\rm pseu})_{i+1/2}$ both replaced by $\Delta x_i/(\Delta t_{\rm pseu})_i$, which is a finite volume discretization of the smoothing operator $1/\Delta t_{\rm pseu}\Delta A - \beta\Delta x^2/\Delta t_{\rm pseu}\partial^2\Delta A/\partial x^2$ that is in *non*-conservative form[10].

As far as we have been able to reconstruct from the comments in `c_flipse`, the non-symmetric, non-conservative implicit smoothing has never worked properly. We always seem to have used $\beta = 0$ in SUPSUB computations. As mentioned already, the code for (26) has been commented out, so that one definitely didn't seem to have worked well, probably because of its ineffectiveness to limit non-smooth solution corrections, as we now have realized, cf. the paragraph below (26). We also have implemented an ad-hoc variation on $\Delta x/\Delta t_{\rm pseu}\Delta A = \Delta x_i/(\Delta t_{\rm pseu})_i\Delta A_i$ with an upwind bias scaled with $u/(c + |u|)$. We have not been able to figure out where that one has come from.

[7] In SUPSUB a mass- and momentum-conservative finite volume method is applied, i.e., the integral form of the model equations (1) and (2) (of the equations (4) and (5) after discretization in time, and of the equations (14) and (15) after linearization and the addition of a pseudo-time step) is discretized in space. The finite volume equivalent of adding a pseudo-time step is the addition of a term $\Delta x/\Delta t_{\rm pseu}\Delta A$ to the space-integrated continuity equation, and of a term $\Delta x/\Delta t_{\rm pseu}\Delta A$ to the space-integrated momentum equation.

[8] It is a solution-dependent adaptive filter that varies from first-order for non-smooth solutions to third-order for smooth solutions.

[9] Proper implicit filtering of geometry variables still to be developed.

[10] When $\Delta x^2/\Delta t_{\rm pseu}$ would be constant in space, this form would also be 'conservative', i.e., there would be no difference between this form and the previous one. Because it is very advantageous to scale $\Delta t_{\rm pseu}$ with the local residual (cf. the next Section), this will never be the case in practice, regardless of any variation of the grid size Δx across the domain. Due to the variation in space of the residual, which may be very large and may change considerably from iteration to iteration, pseudo-time step scaling coefficient $\Delta x^2/\Delta t_{\rm pseu}$ is generally strongly non-uniform.

The pseudo-time step space discretization is definitely something that requires further thinking and testing, because it is a crucial element in the performance (robustness and convergence speed) of the iterative solution process. On the other hand, the same applies to the linearization and to the scaling of $\Delta t_{\rm pseu}$, which are both not optimal at present. This may have contributed to the problems that we apparently have had with the design of a proper pseudo-time step space discretization. Anyway, in hindsight it seems best to use either (27) or (28) (or a mixture?), with a strong preference for the latter.

3.2 Estimation of size of pseudo-time step

For the application of (22) or (23), the solution corrections ΔA and ΔQ are required to compute Δu and Δc according to (19). These corrections are obviously not available prior to solving (14) and (15), cf. the chicken-and-egg problem mentioned in Footnote 20 on page 21 of Borsboom (2019). A solution to this problem would be to obtain an estimation of ΔA and ΔQ in terms of the residuals $\mathrm{res}_{\rm cont}$ and $\mathrm{res}_{\rm mom}$ directly from (18). For example, by assuming a wiggly solution correction ΔA, ΔQ, which after all is the type of correction likely to be the biggest problem in assuring a highly robust convergence process (cf. the first item in the introduction of Section 3), we can replace $\partial\Delta A/\partial x$ and $\partial\Delta Q/\partial x$ in (18) by $\Delta A/\Delta x$ and $\Delta Q/\Delta x$, with ΔA and ΔQ now the amplitude of a $4\Delta x$ wiggle (the wiggle mode that gives the largest value of a three-point central discretization of a first derivative). This allows us to solve this equation *locally* after having replaced $1/\Delta t_{\rm pseu}$ by an expression in terms of ΔA and ΔQ using (22) or (23). Note that, since these ΔA and ΔQ are *not* going to be used to correct the iterands $A^{n,m-1}$, $Q^{n,m-1}$ with, accuracy is not required here. All that matters is a procedure that provides a decent approximation of ΔA and ΔQ for the computation of Δu and Δc to be used in (22) or (23). The accurate computation of ΔA and ΔQ then follows from solving (14) and (15).

... (after some thinking) We actually do not need reasonable estimates of ΔA and ΔQ to compute/estimate the value of suitable local $\Delta t_{\rm pseu}$! After all, the idea of (22) and (23) is to take $\Delta t_{\rm pseu}$ as large as possible within the constraints imposed by (21). In other words, what matters is a $\Delta t_{\rm pseu}$ that ensures that (21) holds, i.e., a $\Delta t_{\rm pseu}$ that ensures that $\Delta A_{\max}$ and $\Delta Q_{\max}$, the maximum possible values of $|\Delta A|$ and $|\Delta Q|$, are such that (21) is satisfied.

The conditions (21) obviously require:

$$|\Delta u| \le \Delta u_{\max} = \varepsilon_{\rm pseu}|u| \ , \ |\Delta c| \le \Delta c_{\max} = \varepsilon_{\rm pseu}|c| \ . \tag{29}$$

with $\varepsilon_{\rm pseu} \ll 1$ a sufficiently small scaling parameter, i.e., $\Delta u_{\max}$ and $\Delta c_{\max}$ are considered to be the maximally allowed values of $|\Delta u|$ and $|\Delta c|$. From (19) we obtain:

$$\begin{aligned} \Delta A_{\max} &\approx \frac{2cW_{z=\zeta}}{g\theta\big(1+\max(\delta,-A(W_z)_{z=\zeta}/(W_{z=\zeta})^2\big)}\Delta c_{\max} \ , \\ \Delta Q_{\max} &\approx \frac{A}{\theta}\Delta u_{\max} + \frac{|Q|}{A}\Delta A_{\max} \ , \end{aligned} \tag{30}$$

with $-1 < \delta \le 0$ a parameter.

We will insert these values in the diagonal form of (18), obtained by using matrix decomposition (20), omitting for convenience the small (by construction) nonlinear terms[11]:

$$\begin{aligned} \Big(\frac{1}{\Delta t_{\rm pseu}}+\frac{1}{\Delta t}\Big)\big((u\pm c)\Delta A-\Delta Q\big) + \theta(u\mp c)\Big((u\pm c)\frac{\partial\Delta A}{\partial x}-\frac{\partial\Delta Q}{\partial x}\Big) \\ = \big((u\pm c)\mathrm{res}_{\rm cont}^{n,m-1} - \mathrm{res}_{\rm mom}^{n,m-1}\big) \ . \end{aligned}$$

[11] Taking the second-order nonlinear terms into account would not be a problem, but simply means a small scaling of u and c because of (29). The same effect can be obtained by taking a slightly different value of $\varepsilon_{\rm pseu}$.

Multiplying this equation by Δx and replacing $\partial \Delta A/\partial x$ and $\partial \Delta Q/\partial x$ by the largest possible values $\Delta A/\Delta x$ and $\Delta Q/\Delta x$ (obtained for the $4\Delta x$ wiggle mode), we obtain:

$$\left(\frac{\Delta x}{\Delta t_{\text{pseu}}}+\frac{\Delta x}{\Delta t}+\theta(u\mp c)\right)\left((u\pm c)\Delta A-\Delta Q\right)=\left((u\pm c)\text{res}_{\text{cont}}^{n,m-1}-\text{res}_{\text{mom}}^{n,m-1}\right)\ . \quad (31)$$

By replacing ΔA and ΔQ with the maximally allowed values and taking absolute values of the different terms, we obtain from this equation a fairly reasonable estimation of a suitable pseudo-time step:

$$\frac{\Delta x}{\Delta t_{\text{pseu}}}=\max\left(0,\frac{\begin{array}{c}|(u+c)\text{res}_{\text{cont}}^{n,m-1}|+|\text{res}_{\text{mom}}^{n,m-1}|\\+\theta|u-c|(|u+c|\Delta A_{\max}+\Delta Q_{\max})\end{array}}{|u+c|\Delta A_{\max}+\Delta Q_{\max}}-\frac{\Delta x}{\Delta t},\right.$$
$$\left.\frac{\begin{array}{c}|(u-c)\text{res}_{\text{cont}}^{n,m-1}|+|\text{res}_{\text{mom}}^{n,m-1}|\\+\theta|u+c|(|u-c|\Delta A_{\max}+\Delta Q_{\max})\end{array}}{|u-c|\Delta A_{\max}+\Delta Q_{\max}}-\frac{\Delta x}{\Delta t}\right)\ .$$

Hm, this boils down to:

$$\frac{\Delta x}{\Delta t_{\text{pseu}}}=\max\left(0,\frac{|(u+c)\text{res}_{\text{cont}}^{n,m-1}|+|\text{res}_{\text{mom}}^{n,m-1}|}{|u+c|\Delta A_{\max}+\Delta Q_{\max}}+\theta|u-c|-\frac{\Delta x}{\Delta t},\right.$$
$$\left.\frac{|(u-c)\text{res}_{\text{cont}}^{n,m-1}|+|\text{res}_{\text{mom}}^{n,m-1}|}{|u-c|\Delta A_{\max}+\Delta Q_{\max}}+\theta|u+c|-\frac{\Delta x}{\Delta t}\right)\ ,$$

which is definitely not a good idea. Only possible and 'reasonable' alternative is to start from (18) and to assume that ΔA and ΔQ are fairly smooth in space, hence $|\partial \Delta A/\partial x| \ll |\Delta A/\Delta x|$ and $|\partial \Delta Q/\partial x| \ll |\Delta Q/\Delta x|$. It then seems reasonable to neglect the effect of the space derivative in (18), from which we immediately obtain:

$$\frac{\Delta x}{\Delta t_{\text{pseu}}}=\max\left(\frac{\Delta x}{\Delta t},\frac{|\text{res}_{\text{cont}}^{n,m-1}|}{\Delta A_{\max}},\frac{|\text{res}_{\text{mom}}^{n,m-1}|}{\Delta Q_{\max}}\right)-\frac{\Delta x}{\Delta t}\ . \quad (32)$$

The scaling applied here is pretty similar to the one that is used in (23), as it should be. Main difference is that in (32) the pseudo-time step to be applied in some iteration step is determined by the *known* residuals in that step, while in (23) it depends on the solution corrections to be determined in that iteration step, which are yet unknown and therefore need to be estimated using the ΔA and ΔQ from previous steps.

It seems (closer look still required) that the above procedure is in line with the theoretically optimal pseudo-time step derived in Section 6.4 of Deuflhard (2011), *Pseudo-transient Continuation for Steady State Problems*, cf. expression (6.41) of Theorem 6.6.

4 References

Borsboom, M., 1999. "SOBEK, betere schaling van Cflipse." Memo, February 24, 1999.

Borsboom, M., 2001. "Development of a 1-D error-minimizing moving adaptive grid method." In A. Vande Wouwer, P. Saucez and W. E. Schiesser, eds., *Adaptive Method of Lines*, chap. 5, pages 139–180. CRC Press.

Borsboom, M., 2019. "Numerical design of a fast, robust and accurate 1D shallow-water solver for pipe flows with large time scales – proposal (WORK IN PROGRESS)." Memo.

Ceze, M. and K. J. Fidkowski, 2013. "Pseudo-transient continuation, solution update methods, and CFL strategies for DG discretizations of the RANS-SA equations." In *Proc. 21th AIAA CFD Conf.* AIAA. DOI: 10.2514/6.2013-2686.

Deuflhard, P., 2011. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*. Computational Mathematics. Springer.

Hicken, J. E. and D. W. Zingg, 2009. "Globalization strategies for inexact-Newton solvers." In *Proc. 19th AIAA CFD Conf.*, pages AIAA 2009-4139. AIAA. DOI: 10.2514/6.2009-4139.

Kelley, C. T. and D. E. Keyes, 1998. "Convergence analysis of pseudo-transient continuation." *SIAM J. Numer. Anal.* 35 (2): 508–523. DOI: 10.1137/S0036142996304796.

Knoll, D. A. and D. E. Keyes, 2004. "Jacobian-free Newton-Krylov methods: a survey of approaches and applications." *J. Comput. Phys.* 193 (2): 357–397. DOI: 10.1016/j.jcp.2003.08.010.

Mavriplis, D. J., 2018. "A residual smoothing strategy for accelerating Newton method continuation." URL https://arxiv.org/abs/1805.03756v1.

Savant, G., C. Berger, T. O. McAlpin and J. N. Tate, 2011. "Efficient implicit finite-element hydrodynamic model for dam and levee breach." *J. Hydraul. Eng.* 137 (9): 1005–1018. DOI: 10.1061/(ASCE)HY.1943-7900.0000372.