Efficient simulation of unsaturated flow in heterogeneous soils using an exponential integrator

Elliot Carr
Joint work with Dr Tim Moroney and Prof. Ian Turner
Queensland University of Technology

QANZIAM, 24-25 October 2009
Exponential Integrators

Consider systems of initial value problems of the form

\[ \frac{du}{dt} = g(u); \quad u(0) = u_0. \]

where \( u \in \mathbb{R}^N \) and \( g: \mathbb{R}^N \supset D \to \mathbb{R}^N \). Exponential integrators solve these systems via the application of a function of the Jacobian matrix \( J(u) = \partial g/\partial u \) on a vector \( v \) – either the matrix exponential or a closely related function:

\[ \varphi(A) = A^{-1}(e^A - I) = I + \frac{1}{2}A + \frac{1}{6}A^2 + \ldots \]

Recently, these methods have found application in the numerical integration of stiff problems.

The attractiveness of these methods is that they can perform very well without the need for preconditioning unlike implicit integrators, which require approximation to \( A^{-1}v \).
Pressure-driven flow in porous media can be modelled using Darcy’s law:

\[ q = -K(h)(\nabla h + e_z), \]

where \( q \) is the Darcy flux vector, \( h \) is pressure head, \( K \) is the hydraulic conductivity and \( e_z \) is the unit vector in the vertical direction, oriented upwards.

Assuming incompressibility, conservation of mass requires that

\[ \frac{\partial \theta}{\partial t} = \nabla \cdot (K(h)\nabla h) + \frac{\partial K(h)}{\partial z}. \]

which gives the well-known mixed-form of Richards’ Equation where \( \theta \) is the volumetric moisture content.
The effective saturation in unsaturated soil is defined as

\[ S_e(h) = (1 + (-\alpha h)^n)^{-m}, \]

where \( \alpha \) and \( n \) are empirically-derived soil parameters and \( m = 1 - 1/n \).

In terms of the effective saturation, the moisture content and hydraulic conductivity are given for unsaturated flow by

\[
\theta(h) = \theta_{\text{res}} + (\theta_{\text{sat}} - \theta_{\text{res}}) S_e
\]

\[
K(h) = K_{\text{sat}} \sqrt{S_e} \left(1 - \left(1 - S_e^{1/m}\right)^m\right)^2
\]

where \( \theta_{\text{res}} \) and \( \theta_{\text{sat}} \) are the residual and saturated moisture content respectively.
Using the finite volume method on a 2D rectangular mesh and integrating over a control volume of volume $\Delta x_i \times \Delta y_j$ the following spatially discrete form of Richards’ equation is obtained:

$$\frac{d\theta_p}{dt} = \frac{1}{\Delta z_j} \left( q_{i,j-1/2} - q_{i,j+1/2} \right) + \frac{1}{\Delta x_i} \left( q_{i-1/2,j} - q_{i+1/2,j} \right),$$

where indices on the flux vector $q$ denote its normal derivative approximated at a representative point on the control volume face.

Enacting the chain rule on the left hand side to obtain

$$\frac{dh_p}{dt} = \frac{1}{C(h_p)} \left[ \frac{1}{\Delta z_j} \left( q_{i,j-1/2} - q_{i,j+1/2} \right) + \frac{1}{\Delta x_i} \left( q_{i-1/2,j} - q_{i+1/2,j} \right) \right],$$

where $C(h) = \frac{d\theta}{dh}$ is known as the specific moisture capacity.
The exponential Euler method is derived by linearising $g(u)$ about $t = t_n$. At each step this results in the linear initial value problem:

$$\frac{du}{dt} = g_n + J_n (u - u_n)$$

where $J_n = J(u_n)$ to advance the solution from $t = t_n$ to $t = t_{n+1}$.

The exact solution of this problem determines the EEM scheme:

$$u_{n+1} = u_n + J_n^{-1} (e^{\tau_n J_n} - I) g_n$$

$$= u_n + \tau_n \varphi(\tau_n J_n) g_n$$

We note that the method is A-stable and second-order accurate:

Global error $\sim \mathcal{O}(\tau_n^2)$  Local error $\sim \mathcal{O}(\tau_n^3)$
Numerical challenge: How do we compute $\varphi(\tau_n J_n)g_n$?

Project onto the Krylov subspace $\mathcal{K}_m(J_n, g_n) = \text{span}\{g_n, \ldots, J_n^{m-1}g_n\}$:

$$\varphi(\tau_n J_n)g_n \approx \beta V_m \varphi(\tau_n H_m) e_1,$$

where $\beta = \|g_n\|_2$ and Arnoldi’s method is used to generate the decomposition

$$J_n V_m = V_m H_m + \beta_m v_{m+1} e_m^T,$$ (1)

where $V_m$ is a matrix whose column vectors form an orthonormal basis for $\mathcal{K}_m(J_n, g_n)$, $\beta_m = \|v_{m+1}\|_2$ and $e_m$ is the $m$th canonical basis vector in $\mathbb{R}^m$. The difference quotient:

$$J(u)w \approx \frac{g(u + \varepsilon w) - g(u)}{\varepsilon}$$

is used to approximate the Jacobian-vector products required in Arnoldi’s method, removing the need to explicitly form $J_n$. 
Proposition 1

Let the Arnoldi decomposition of $J_n$ hold as defined in (1) and let

$$\tilde{J}_n = J_n - \beta_m v_{m+1} v_m^T.$$ 

The following statements are true:

(i) $\tilde{J}_n V_m = V_m H_m$;

(ii) $\varphi(\tau_n \tilde{J}_n) V_m = V_m \varphi(\tau_n H_m)$ and hence $\varphi(\tau_n \tilde{J}_n) g_n = \beta V_m \varphi(\tau_n H_m) e_1$;

(iii) $\tilde{J}_n g_n = J_n g_n$, provided $m \geq 2$;

(iv) The local error of the scheme

$$u_{n+1} = u_n + \tau_n \beta V_m \varphi(\tau_n H_m) e_1$$

is $O(\tau_n^3)$, provided $m \geq 2$. 

Proof.

(i) Follows from $v_m^T V_m = e_m^T$ and rearranging the Arnoldi decomposition
$$ (J_n - \beta_m v_{m+1} v_m^T) V_m = V_m H_m. $$

(ii) Both results follow from the invariance of $K_m(J_n, g_n)$ under $\tilde{J}_n$.

(iii) Consider
$$ \tilde{J}_n g_n = J_n g_n - \beta_m v_{m+1} v_m^T g_n, $$
but the term $v_m^T g_n$ is zero, since $v_m$ and $g_n = \beta v_1$ are two columns of $V_m$, and hence are orthogonal.
(iv) We analyse the equivalent scheme

\[ u_{n+1} = u_n + \tau_n \varphi(\tau_n \tilde{J}_n) g_n, \]

and expand to obtain

\[
\begin{align*}
  u_{n+1} &= u_n + \tau_n \varphi(\tau_n \tilde{J}_n) g_n \\
          &= u_n + \tau_n \left( I + \frac{\tau_n}{2} \tilde{J}_n + O(\tau_n^2) \right) g_n \\
          &= u_n + \tau_n g_n + \frac{\tau_n^2}{2} \tilde{J}_n g_n + O(\tau_n^3) \\
          &= u_n + \tau_n g_n + \frac{\tau_n^2}{2} J_n g_n + O(\tau_n^3),
\end{align*}
\]

which agrees with the Taylor series expansion of \( u_{n+1} \) about \( t = t_n \) to second order.
Propose to use the difference between $u_{n+1}$ and a second approximate solution $u_{n+1}^{(2)}$, computed using a two-step scheme with half-sized steps:

\[
\begin{align*}
  u_{n+1/2} &= u_n + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} J_n) g_n \\
  u_{n+1}^{(2)} &= u_{n+1/2} + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} J_{n+1/2}) g(u_{n+1/2}),
\end{align*}
\]

where $J_{n+1/2} = J(u_{n+1/2})$.

Replacing $J_{n+1/2}$ with $J_n$, and $g(u_{n+1/2})$ with its orthogonal projection $V_m V_m^T g(u_{n+1/2})$ onto the space $K_m(J_n, g_n)$ results in the following modified scheme

\[
\begin{align*}
  u_{n+1/2} &= u_n + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} J_n) g_n \\
  u_{n+1}^{(2)} &= u_{n+1/2} + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} J_n) V_m V_m^T g(u_{n+1/2}).
\end{align*}
\]
This allows all matrix function terms to be approximated using the existing Arnoldi decomposition:

\[
\varphi(\frac{\tau_n}{2} J_n)g_n \approx \beta V_m \varphi(\frac{\tau_n}{2} J_n)e_1
\]
\[
\varphi(\frac{\tau_n}{2} J_n)V_m \approx V_m \varphi(\frac{\tau_n}{2} H_m),
\]

and we obtain the final two-step scheme:

\[
u_{n+1/2} = u_n + \frac{\tau_n}{2} \beta V_m \varphi(\frac{\tau_n}{2} H_m)e_1
\]
\[
u_{n+1}^{(2)} = u_{n+1/2} + \frac{\tau_n}{2} V_m \varphi(\frac{\tau_n}{2} H_m)V_m^T g(u_{n+1/2}).
\]

**Conclusion**

The value of \(u_{n+1}^{(2)}\) can be approximated using a single additional function evaluation!
Let the Arnoldi decomposition of $J_n$ hold as defined in (1) and let
\[ \tilde{J}_n = J_n - \beta_m v_{m+1} v_m^T \] as in Proposition 1. The local error of the scheme:

\[
\begin{align*}
  u_{n+1/2} &= u_n + \frac{\tau_n}{2} \beta V_m \varphi(\frac{\tau_n}{2} H_m) e_1 \\
  u^{(2)}_{n+1} &= u_{n+1/2} + \frac{\tau_n}{2} V_m \varphi(\frac{\tau_n}{2} H_m) V_m^T g(u_{n+1/2}),
\end{align*}
\]

is $O(\tau_n^3)$ provided $m \geq 2$. 
Proof.

We analyse the equivalent scheme:

\[ u_{n+1/2} = u_n + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} \tilde{J}_n) g_n \]
\[ u_{n+1}^{(2)} = u_{n+1/2} + \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} \tilde{J}_n) V_m V_m^T g(u_{n+1/2}). \]

Expanding:

\[ \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} \tilde{J}_n) g_n = \frac{\tau_n}{2} g_n + \frac{\tau_n^2}{8} J_n g_n + O(\tau_n^3) \]
\[ g(u_{n+1/2}) = g_n + \frac{\tau_n}{2} J_n g_n + O(\tau_n^2) \]
\[ V_m V_m^T g(u_{n+1/2}) = g_n + \frac{\tau_n}{2} J_n g_n + O(\tau_n^2) \]
\[ \frac{\tau_n}{2} \varphi(\frac{\tau_n}{2} \tilde{J}_n) V_m V_m^T g(u_{n+1/2}) = \frac{\tau_n}{2} \left( I + \frac{\tau_n}{4} \tilde{J}_n + O(\tau_n^2) \right) g(u_{n+1/2}) \]
\[ = \frac{\tau_n}{2} g_n + \frac{3\tau_n^2}{8} J_n g_n + O(\tau_n^3) \]

Substituting:

\[ u_{n+1}^{(2)} = u_n + \tau_n g_n + \frac{\tau_n^2}{2} J_n g_n + O(\tau_n^3) \]
Stepsize adjustment

Conclusion

$\Delta_n = u_{n+1} - u_{n+1}^{(2)}$ is an order equivalent approximation to the local error incurred by EEM during one integration step.

Time step ($\tau_n$) adjusted according to the value of $\|\Delta_n\|_{\text{WRMS}}$ with $\| \cdot \|_{\text{WRMS}}$ defined by:

$$
\|y\|_{\text{WRMS}} = \left( \frac{1}{N} \sum_{i=1}^{N} \left( \frac{y_i}{w_i} \right)^2 \right)^{1/2},
$$

where the weights $w_i = RTOL \cdot |y_i| + ATOL$ and $ATOL_i$ and $RTOL$ are the given absolute and relative error tolerances.
Current state-of-play sees the time integration of spatially-discrete forms of Richards’ equation performed using variable-stepsize variable-order implementations of the BDF.

Require the solution of a nonlinear system of equations at each time step:

\[ f(u_{n+1}) \equiv \alpha_{n,0} u_{n+1} - \tau_n g(u_{n+1}) + a_n = 0. \]

In this work, we compared our variable-stepsize implementation of EEM to these methods, as implemented in the CVODE module of the Suite of Nonlinear and Differential/Algebraic Equation Solvers (SUNDIALS)\(^1\).

\(^1\)https://computation.llnl.gov/casc/sundials/main.html
Figure: Schematic representations of the test problems: Forsyth et al.'s problem (left) and Kirkland et al.'s problem (right).
**Benchmark solutions: Forsyth et al.’s problem**

<table>
<thead>
<tr>
<th>Material</th>
<th>$\theta_{\text{res}}$</th>
<th>$\theta_{\text{sat}}$</th>
<th>$K_{\text{sat}}$ (ms$^{-1}$)</th>
<th>$\alpha$ (m$^{-1}$)</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soil 1</td>
<td>0.1020</td>
<td>0.3680</td>
<td>$9.153 \times 10^{-5}$</td>
<td>3.34</td>
<td>1.982</td>
</tr>
<tr>
<td>Soil 2</td>
<td>0.0985</td>
<td>0.3510</td>
<td>$5.445 \times 10^{-5}$</td>
<td>3.63</td>
<td>1.632</td>
</tr>
<tr>
<td>Soil 3</td>
<td>0.0859</td>
<td>0.3250</td>
<td>$4.805 \times 10^{-5}$</td>
<td>3.45</td>
<td>1.573</td>
</tr>
<tr>
<td>Soil 4</td>
<td>0.0859</td>
<td>0.3250</td>
<td>$4.805 \times 10^{-4}$</td>
<td>3.45</td>
<td>1.573</td>
</tr>
</tbody>
</table>

**Figure:** Saturation profiles for the benchmark solutions of Forsyth et al.’s problem at 30 days.
Benchmark solutions: Kirkland et al.’s problem

<table>
<thead>
<tr>
<th>Material</th>
<th>$\theta_{\text{res}}$</th>
<th>$\theta_{\text{sat}}$</th>
<th>$K_{\text{sat}}$ (ms$^{-1}$)</th>
<th>$\alpha$ (m$^{-1}$)</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clay</td>
<td>0.1060</td>
<td>0.4686</td>
<td>$1.516 \times 10^{-6}$</td>
<td>1.04</td>
<td>1.3954</td>
</tr>
<tr>
<td>Sand</td>
<td>0.0286</td>
<td>0.3658</td>
<td>$6.262 \times 10^{-5}$</td>
<td>2.80</td>
<td>2.2390</td>
</tr>
</tbody>
</table>

**Figure:** Saturation profiles for the benchmark solutions of Kirkland et al.’s problem at 12.5 days.
## Solver statistics

<table>
<thead>
<tr>
<th>ATOL/RTOL</th>
<th>METHOD</th>
<th>ERROR</th>
<th>EVALS</th>
<th>STEPS</th>
<th>FAILS</th>
<th>ERROR</th>
<th>EVALS</th>
<th>STEPS</th>
<th>FAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Forsyth et al.’s problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>EEM</td>
<td>5.77E−3</td>
<td>704</td>
<td>101</td>
<td>0</td>
<td>5.70E−2</td>
<td>6416</td>
<td>267</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>9.50E−3</td>
<td>1313</td>
<td>335</td>
<td>0</td>
<td>1.85E−2</td>
<td>7601</td>
<td>843</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>1.22E−3</td>
<td>991</td>
<td>225</td>
<td>1</td>
<td>5.19E−3</td>
<td>6754</td>
<td>685</td>
<td>1</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>EEM</td>
<td>1.41E−4</td>
<td>2284</td>
<td>441</td>
<td>0</td>
<td>1.73E−4</td>
<td>15023</td>
<td>1316</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>3.83E−4</td>
<td>4689</td>
<td>1515</td>
<td>2</td>
<td>5.54E−4</td>
<td>29046</td>
<td>4431</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>1.18E−5</td>
<td>1748</td>
<td>494</td>
<td>5</td>
<td>8.86E−6</td>
<td>11834</td>
<td>1675</td>
<td>4</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>EEM</td>
<td>9.40E−6</td>
<td>9864</td>
<td>2037</td>
<td>2</td>
<td>4.53E−6</td>
<td>40054</td>
<td>6181</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>1.94E−5</td>
<td>20655</td>
<td>7018</td>
<td>3</td>
<td>3.03E−5</td>
<td>128423</td>
<td>20416</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>5.81E−8</td>
<td>3055</td>
<td>937</td>
<td>10</td>
<td>8.34E−8</td>
<td>21547</td>
<td>3213</td>
<td>18</td>
</tr>
<tr>
<td><strong>Kirkland et al.’s problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>EEM</td>
<td>1.70E−3</td>
<td>1066</td>
<td>158</td>
<td>4</td>
<td>1.43E−2</td>
<td>9936</td>
<td>1112</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>5.29E−3</td>
<td>2287</td>
<td>505</td>
<td>1</td>
<td>7.85E−3</td>
<td>11078</td>
<td>1309</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>1.87E−3</td>
<td>1990</td>
<td>352</td>
<td>1</td>
<td>1.53E−3</td>
<td>9357</td>
<td>1044</td>
<td>4</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>EEM</td>
<td>8.59E−5</td>
<td>3760</td>
<td>720</td>
<td>8</td>
<td>2.73E−4</td>
<td>17223</td>
<td>2185</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>2.04E−4</td>
<td>7874</td>
<td>2269</td>
<td>3</td>
<td>3.12E−4</td>
<td>41780</td>
<td>6112</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>2.65E−6</td>
<td>2973</td>
<td>748</td>
<td>14</td>
<td>2.43E−6</td>
<td>16903</td>
<td>2272</td>
<td>10</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>EEM</td>
<td>5.39E−6</td>
<td>15905</td>
<td>3287</td>
<td>8</td>
<td>7.25E−6</td>
<td>66931</td>
<td>14015</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>BDF(2)</td>
<td>9.47E−6</td>
<td>33659</td>
<td>10566</td>
<td>3</td>
<td>1.28E−5</td>
<td>183209</td>
<td>28663</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>BDF(5)</td>
<td>2.89E−7</td>
<td>5226</td>
<td>1474</td>
<td>17</td>
<td>1.56E−7</td>
<td>31047</td>
<td>4502</td>
<td>16</td>
</tr>
</tbody>
</table>
Summary

- Provided a practical variable-stepsize implementation of the exponential Euler method (EEM).
- Introduced a new second-order variant of the scheme that enables the local error to be estimated at the cost of a single additional function evaluation.
- New EEM implementation outperformed sophisticated implementations of the backward differentiation formulae (BDF) of order 2 and was competitive with BDF of order 5 for moderate to high tolerances.