Imperial College London

A new wetting and drying algorithm using a combined pressure/free-surface finite element method

Simon Funke, Chris Pain, Stephan Kramer, Matthew Piggott

Simon Funke

August 18th, 2010

Grantham Institute for Climate Change,
Energy Futures Lab, I.C.

FUJITSU

AMCG

Overview

1. Introduction
2. Derivation of shared wetting and drying method
3. Testcases
4. Problems/Future work

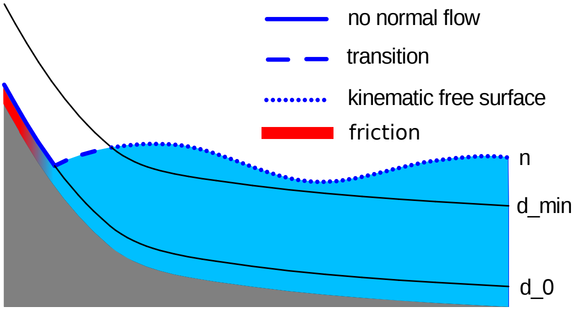
History of wetting and drying

- Interface tracking
- Fixed domain
  - On/off methods
    - Idea: remove dry nodes in the calculation.
    - Robust, but difficult to implement.
    - Timestep limitations.
  - Porosity method
    - Idea: Use negative waterlevel in dry area to kill 2D groundwater flow with friction increasing with negative depth.
  - ...

Imperial College London

New shared wetting and drying method

- Designed for 3D non-hydrostatic models.
- Shared free-surface/pressure variable.
- Horizontally fixed but vertically moving domain.



3D Navier-Stokes Equation

With Boussinesq approximation $\rho = \rho_0 + \rho'$:

$$\rho_0 \left(\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}\right) = -\nabla p + \nabla \cdot \mu \nabla \mathbf{u} - (\rho' + \rho_0) g n_z$$

$$\nabla \cdot \mathbf{u} = 0$$

Subtracting out the hydrostatic pressure gives $\bar{p} = p + \rho_0 g z$ and

$$\rho_0 \left(\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}\right) = -\nabla \bar{p} + \nabla \cdot \mu \nabla \mathbf{u} - \rho' g n_z$$

Combined pressure/free-surface kinematic bc

Kinematic free-surface boundary condition

$$n \cdot n_z \partial_t \eta = n \cdot \mathbf{u} \quad \text{on } \partial_{fs}$$

where n_z is the upwards normal.

Assumption: p is constant on the free-surface, w.l.o.g.:

$$p = 0 \xrightarrow{\bar{p} = p + \rho_o g z} \bar{p} = \rho_0 g \eta \quad \text{on } \partial_{fs} \tag{1}$$

Combined pressure/free-surface kinematic b.c.

$$\frac{n \cdot n_z}{\rho_0 g} \partial_t \bar{p} = n \cdot \mathbf{u} \quad \text{on } \partial_{fs}$$

Imperial College London

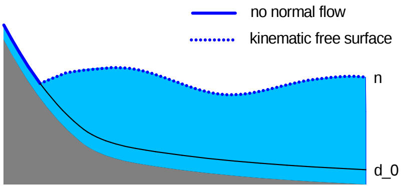
Wetting and drying

So far we had

$$\eta = \frac{\bar{p}}{\rho_0 g} \quad \text{on } \partial_{fs}$$

which is now changed to:

$$\eta = max\left(\frac{\bar{p}}{\rho_0 g}, d_0\right) \quad \text{on } \partial_{fs}$$



Combined kinematic b.c. with wetting and drying

$$\frac{n \cdot n_z}{\rho_0 g} \partial_t \max(\bar{p}, \rho_0 g d_0) = n \cdot \mathbf{u} \quad \text{on } \partial_{fs}$$

Imperial College London

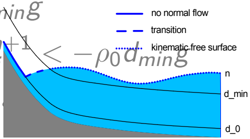
Discretisation and linearisation

Time discretizing the combined kinematic b.c. yields:

$$\frac{n \cdot n_z}{\rho_0 g \Delta t}\left(\max(\bar{p}^{n+1}, \rho_0 g d_0) - \max(\bar{p}^{n}, \rho_0 g d_0)\right) = n \cdot \mathbf{u}^{n+1} \qquad \text{on } \partial_{fs}$$

Problem: max operator is not linear! Use approximation:

$$\max(\bar{p}^{n+1}, -\rho_0 g d_0) - \max(\bar{p}^{n}, -\rho_0 g d_0) = \begin{cases} \bar{p}^{n+1} - \bar{p}^{n}, & \text{if } \bar{p}_{li}^{n+1} \geq -\rho_0 d_{min} g \\ \frac{\bar{p}_{li}^{n+1} + d_0 g}{-d_{min} g + d_0 g}(\bar{p}^{n+1} - \bar{p}^{n}), & \text{if } -\rho_0 d_0 g < \bar{p}_{li}^{n+1} < -\rho_0 d_{min} g \\ 0, & \text{if } \bar{p}_{li}^{n+1} \leq -\rho_0 d_0 g \end{cases}$$



where the subscript *li* stands for last nonlinear iteration.

Discretisation and linearisation V2

We can do better! Linearisation V2:

$$\max(\bar{p}^{n+1}, -\rho_0 g d_0) = \begin{cases} \bar{p}^{n+1}, & \text{if } \bar{p}_{li}^{n+1} \geq -\rho_0 d_0 g \\ -\rho_0 g d_0, & \text{if } \bar{p}_{li}^{n+1} \leq -\rho_0 d_0 g \end{cases}$$

Discretisation V2

Integration with node based linearisation:

$$\int_\Delta \Phi_j \max\left(\bar{p}^{n+1}, -\rho_0 g d_0\right) \overset{dis.}{=} \int_\Delta \Phi_j \max\left(\sum_{i=1}^{N_{loc}} \bar{p}_i^{n+1}\Phi_i, -\rho_0 g \sum_{i=1}^{N_{loc}} d_{0_i}\Phi_i\right)$$

$$\overset{lin.}{=} \int_\Delta \Phi_j \sum_{i=1}^{N_{loc}} \max\left(\bar{p}_i^{n+1}, -\rho_0 g d_{0_i}\right)\Phi_i$$

$$\overset{quad.}{=} \sum_{\delta=1}^{N_{quad}} \Phi_j \omega(\delta) \sum_{i=1}^{N_{loc}} (\max\left(\bar{p}_i^{n+1}, -\rho_0 g d_{0_i}\right)\Phi_i)(\xi_\delta)$$

Note that the linearisation of the max operator is only exact at the *loc* points, but the quatrature evaluates the valuates at the quadrature points. (TODO...)

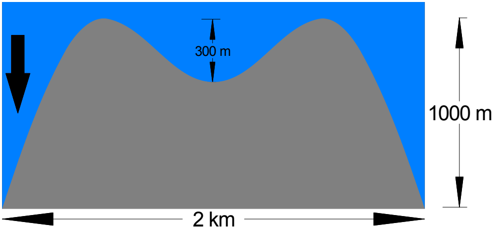
Integration with quadrature based linearisation:

$$\int_\Delta \Phi_j \max\left(\bar{p}^{n+1}, -\rho_0 g d_0\right) \overset{dis.}{=} \int_\Delta \Phi_j \max\left(\sum_{i=1}^{N_{loc}} \bar{p}_i^{n+1}\Phi_i, -\rho_0 g \sum_{i=1}^{N_{loc}} d_{0_i}\Phi_i\right)$$

$$\overset{quad.}{=} \sum_{\delta=1}^{N_{quad}} \omega(\delta)\Phi_j \max\left(\sum_{i=1}^{N_{loc}} \bar{p}_i^{n+1}\Phi_i, -\rho_0 g \sum_{i=1}^{N_{loc}} d_{0_i}\Phi_i\right)(\xi_\delta)$$

Now use the Linearisation V2 of the max operator for each quadrature point. For each quadrature point, if the max operator returns the first argument, then stick the term into matrix M_1, else in M_2.The result is a term of the form $M_1\bar{p}_i^{n+1} - M_2\rho_0 g d_0$ with $M_1 + M_2 = M$,where M is the full mass matrix.

Testcase: Lake in island



→ **Video**: Lake in island

Friction term

$$\rho_0 \left(\partial_t u + u \cdot \nabla u + \sigma \alpha u\right) = -\nabla \bar{p} + \nabla \cdot \mu \nabla u - \rho' g n_z$$

with

$$\alpha = max\left(d_0 - \frac{\bar{p}_{fs}}{\rho_0 g}, 0\right), \quad \bar{p}_{fs} \text{ is pressure on } \partial_{fs}$$

and σ a user defined friction coefficient.

→ **Video**: Lake in island with $\sigma = 160$

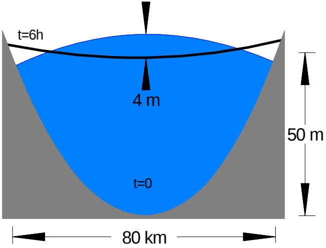
σ	Lake waterlevel after 17h
0.0	-300m
0.2	-160m
2	-16m
160	-0.7m

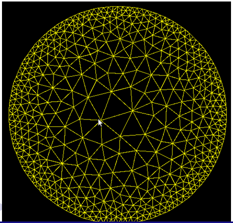
Thacker testcase

Thacker, 1981: Analytical solution.
Settings:

- Discretisation: $P_1^{DG} P_2$
- $d_0 = 1m$
- $d_{min} = 10m$
  Wave speed error: 13%
- $d_{min} = 2.5m$
  Wave speed error: 5%

$\rightarrow$ **Video**

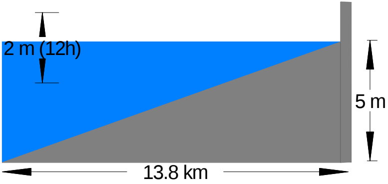




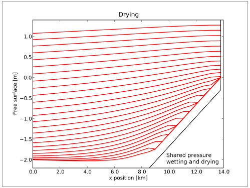
Balzano 1 testcase

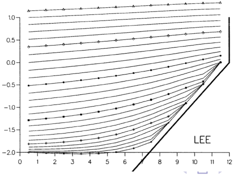
Settings:

- $d_0 = 10cm$, $d_{min} = 20cm$
- Discretisation: $P_1 P_1$
- Element size: 500m
- Timestep: 10min



Result for the drying period, free surface plotted every 20min:



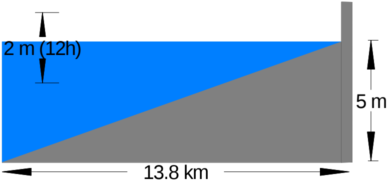


Balzano [1998]

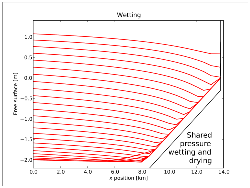
Balzano 1 testcase

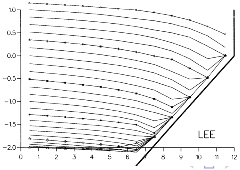
Settings:

- $d_0 = 10cm$, $d_{min} = 20cm$
- Discretisation: $P_1 P_1$
- Element size: 500m
- Timestep: 10min



Result for the wetting period, free surface plotted every 20min:





Balzano [1998]

Real domain testcase: Mersey

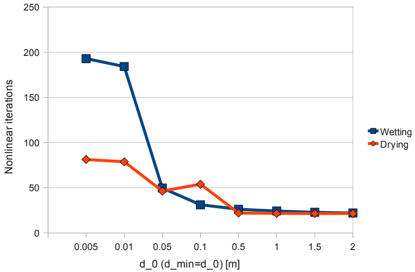
- $d_0 = 10cm$, $d_{min} = 20cm$
- 23 000 elements.
- Discretisation: $P_1 P_1$
- Wetting
  - 2cm per minute
  - Timestep 5min
  - CFL: 8.0

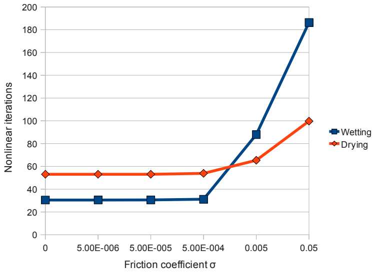
$\rightarrow$ **Video**

- Drying
  - 10cm per minute
  - Timestep 10min
  - CFL: 7.8

$\rightarrow$ **Video**

Problem: Nonlinear solver behaviour





Summary & Future work

- Wetting and drying method based on a shared pressure/free-surface.
- It's simple, accurate and allows big timesteps.

- Optimize computational effort, e.g. non-linear convergence behaviour.
- Apply method on fully unstructured 3D meshes.
- Extend formulation to support wetting and drying on overhanging slopes.

Questions:

What happens to tracers?

Why not mesh movement?