The joy of interactive modeling

Gennadii Donchyts, Fedor Baart, Arthur van Dam, and Bert Jagers

Deltares, Netherlands (gennadii.donchyts@deltares.nl)

The conventional way of working with hydrodynamical models usually consists of the following steps:

1) define a schematization (e.g., in a graphical user interface, or by editing input files)
2) run model from start to end
3) visualize results
4) repeat any of the previous steps.

This cycle commonly takes up from hours to several days. What if we can make this happen instantly? As most of the research done using numerical models is in fact qualitative and exploratory (Oreskes et al., 1994), why not use these models as such? How can we adapt models so that we can edit model input, run and visualize results at the same time?

More and more, interactive models become available as online apps, mainly for demonstration and educational purposes. These models often simplify the physics behind flows and run on simplified model geometries, particularly when compared with state-of-the-art scientific simulation packages. Here we show how the aforementioned conventional standalone models ("static, run once") can be transformed into interactive models.

The basic concepts behind turning existing (conventional) model engines into interactive engines are the following. The engine does not run the model from start to end, but is always available in memory, and can be fed by new boundary conditions, or state changes at any time. The model can be run continuously, per step, or up to a specified time. The Hollywood principle dictates how the model engine is instructed from 'outside', instead of the model engine taking all necessary actions on its own initiative.

The underlying techniques that facilitate these concepts are introspection of the computation engine, which exposes its state variables, and control functions, e.g. for time stepping, via a standardized interface, such as BMI (Peckam et. al., 2012).

In this work we have used a shallow water flow model engine D-Flow Flexible Mesh. The model was converted from executable to a library, and coupled to the graphical modelling environment Delta Shell. Both the engine and the environment are open source tools under active development at Deltares. The combination provides direct interactive control over the time loop and model state, and offers live 3D visualization of the running model using VTK library.