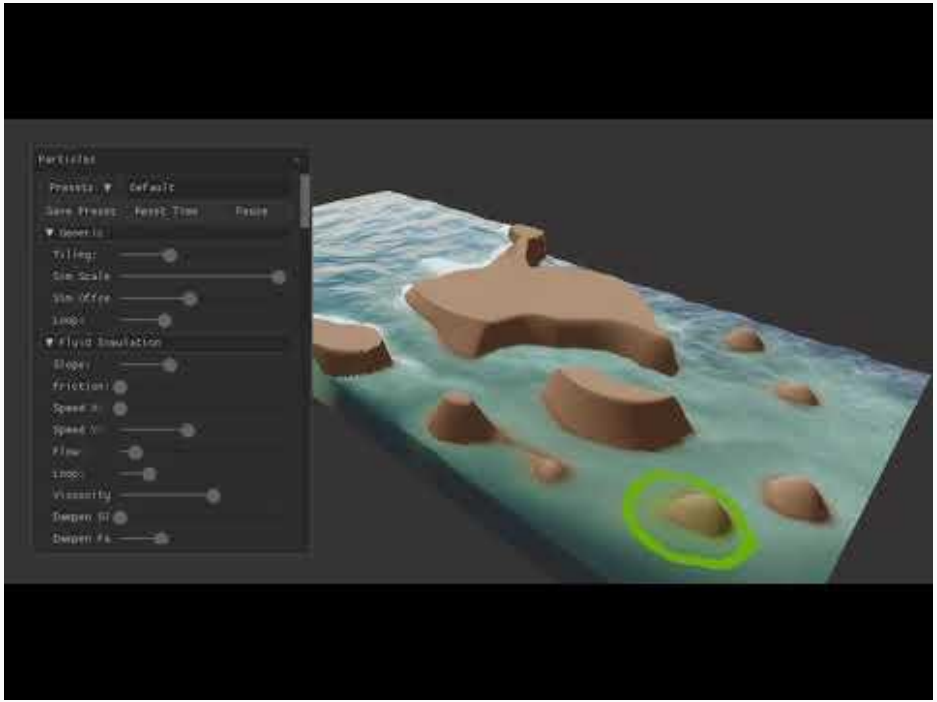
Wave Particles with Interactive Vortices

Final

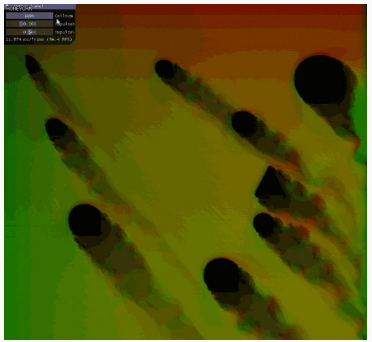
Xiao Zhang and Lan Lou

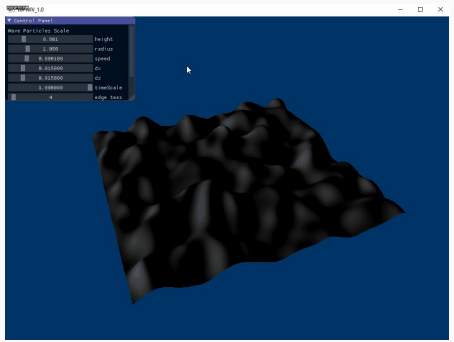
What we want to do

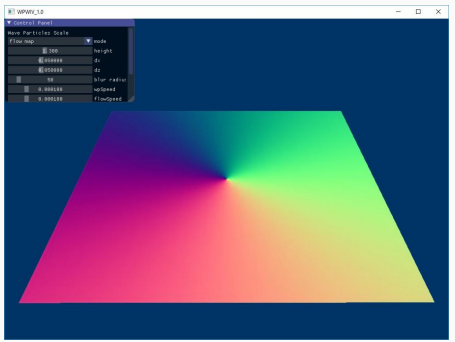


What we did

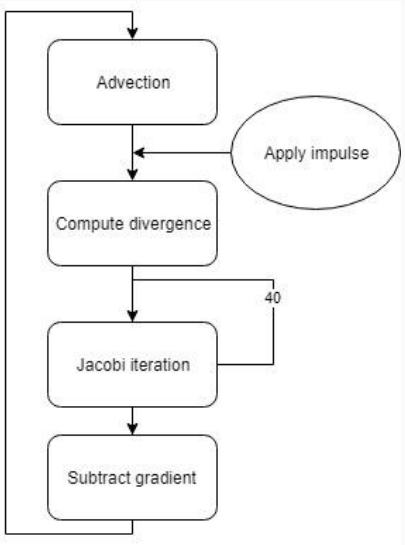
- 2D fluid simulation (GPU Gems)
- Wave particle with flow map (Naughty Dog GDC 2012 talk & Siggraph 2016 course)
- Combine the two above



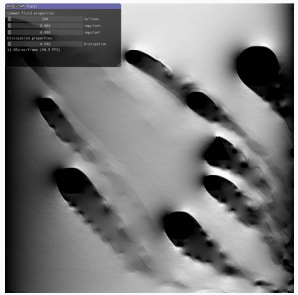


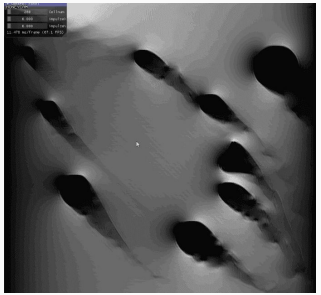


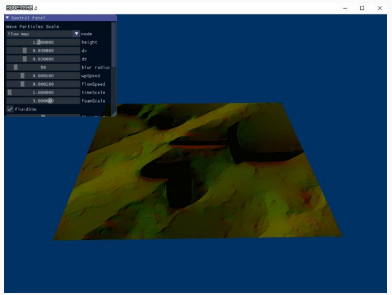
2D fluid simulation



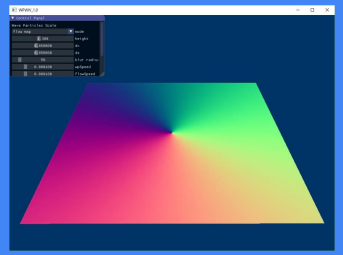
- Stable fluid
  - Not the same as Mr.Grenier's approach, he used LBM, we used Marker-and-Cell Method
- Vorticity features
  - From divergence , Important to our application
  - Best when jacobi iteration is 40

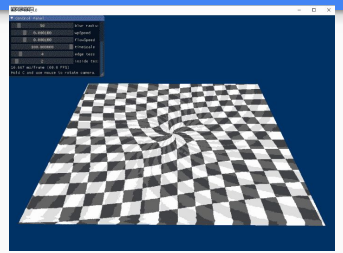




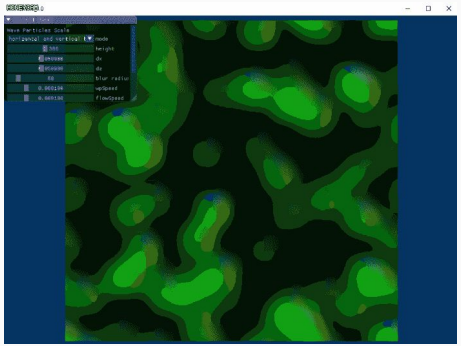


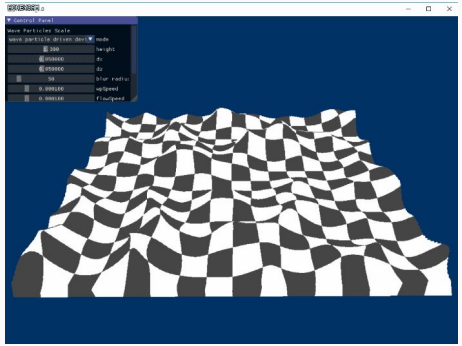
Wave particle with flow map

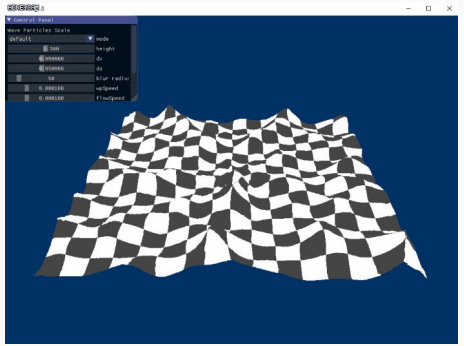




- Wave particle
  - No reflection and subdivision. Just use them to create choppy waves
- Flow map
  - Constantly blending two sets of uv coordinates which are offset by a texture.

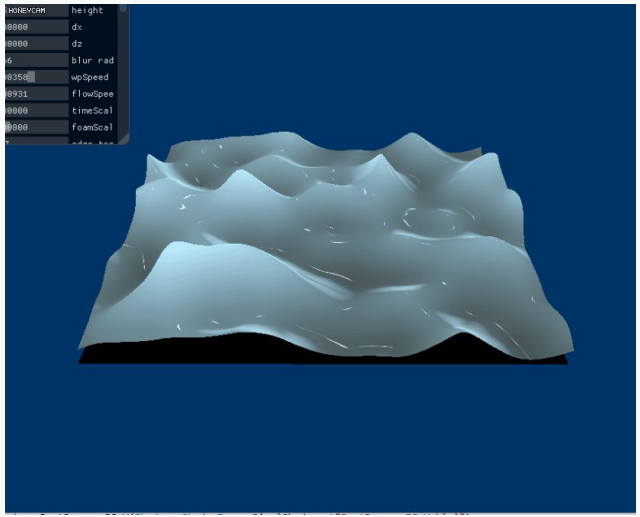






Combine 2D fluid simulation with flow map driven wave particle

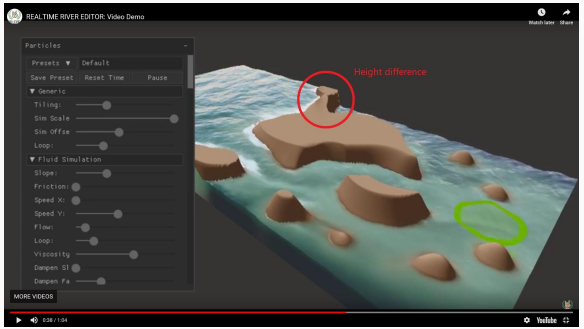
- 2D fluid sim
  - Currently only using velocity field, and divergence field
- Flow map driven wave particle
  - Use velocity field as flow map
- Rendering
  - Use divergence to spawn foam and brighten water
  - Use exponential integral to approximate color bleeding
  - Fresnel and specularity



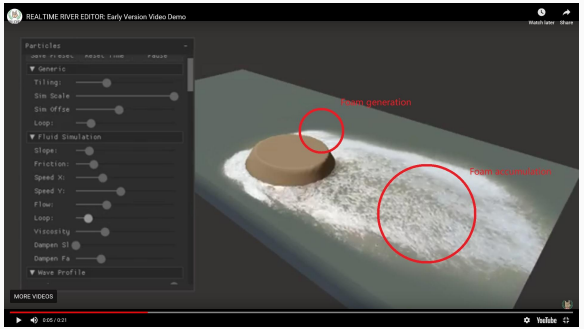
Live demo

Limitations

- Height difference is hard to reproduce
  - Because our height is only related to wave particle and wave particle is moving randomly.

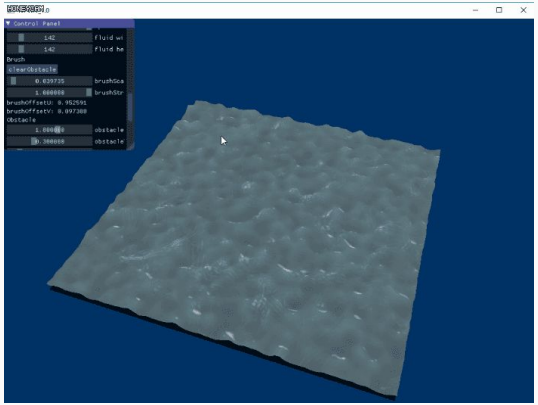


- Foam does not accumulate
  - Because we use divergence to create foam. Divergence is calculated per frame so it does not accumulate.



Future works

- Advect height property
  - Either advect a separate scalar property and add on top of the wave particle displacement or advect wave particle displacement directly
- Advect foam amount
  - Use either divergence or pressure(or friction) to generate foam and advect it in fluid simulation
- Flow map
  - Need more experiments to decide whether to drop it or keep it



Reference

- River Editor: Water Simulation in Real-Time, written by Jean-Philippe Grenier
- Water Technology of Uncharted, presented by Carlos Gonzalez Ochoa from Naughty Dog on GDC 2012
- Rendering rapids in Uncharted 4, presented by Carlos Gonzalez Ochoa on Siggraph 2016 Advances in Real-Time Rendering in Games course
- Implementing Wave Particles for Real-time Water Waves with Object Interaction, written by Cem Yuksel, Donald H. House and John Keyser from Texas A&M University
- Real-Time Water Waves With Wave Particles, written by Cem Yuksel