HPC-LEAP Workshop: Shallow Water Equation for Tsunami Simulation

Local Time Stepping

Xiao Xue, Andrew Brockman

HPC-LEAP Marie Curie Action

30 June 2016

Outline

Outline

Shallow water equation and Tsunami

CFL condition

- A necessary condition that makes finite volume and discontinuous Galerkin method stable.
- CFL condition for the n-dimensional case

$$C = \Delta t max_{i=1,\dots,n}(\frac{U_{x_i}}{\Delta x_i}) \leq C_{max}$$

Shallow water equation and Tsunami

CFL condition

- A necessary condition that makes finite volume and discontinuous Galerkin method stable.
- CFL condition for the n-dimensional case

$$C = \Delta t max_{i=1,\dots,n}(\frac{U_{x_i}}{\Delta x_i}) \leq C_{max}$$

Global time stepping scheme

- All cells in domain have same time step size Δt
- May have computational overhead with small Δx_j in domain or wave speed regarding to water hight $U = u + \sqrt{gh}$

Outline

1D shallow water equations

1D shallow water equations

- $h(x,t)$ is the fuild depth, $u(x,t)$ is the fluid velocity,g is the gravity constant,$B(x)$ is relative to sea level.

$$h_t + (hu)_x = 0,$$
$$(hu)_t + (hu^2 + \frac{1}{2}gh^2)_x + ghB_x = 0$$

2D shallow water equations

2D shallow water equations

- $h(x, y, t)$ is the fuild depth, $u(x, t)$ and $v(y, t)$ are fluid velocity in two different directions,g is the gravity constant,$B(x, y)$ is relative to sea level.

$$
\begin{gathered}
h_t + (hu)_x + (hu)_y = 0, \\
(hu)_t + (hu^2 + \tfrac{1}{2}gh^2)_x + (huv)_y + ghB_x = 0, \\
(hv)_t + (huv)_x + (hv^2 + \tfrac{1}{2}gh^2)_y + ghB_y = 0,
\end{gathered}
$$

Godunov & Roe solver

Godunov's Method

- Recall Q_i represents the average of $q(x, t_n)$ over cell C_i

$$Q_i^n \approx \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x, t_n) dx$$

- Godunov's method can be represent as follow, detail in [2,p.311]

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}(F_{i+\frac{1}{2}}^n - F_{i-\frac{1}{2}}^n) \text{ with } F_{i+\frac{1}{2}}^n = \boldsymbol{F}(Q_{i+1}, Q_i)$$

Godunov & Roe solver

Roe Approximation

- Roe's Riemann solver approximation is linearize the nonlinear problem $q_t + f(q)_x = 0$ to
$$\hat{q}_t + \hat{A_{i-1/2}}\hat{q}_x = 0$$
The matrix $\hat{A_{i-1/2}}$ is chosen to be some approximation to $f^{'}(q)_x$ valid in a neighborhood of Q_{i-1} and Q_i
- Roe Approximation:
$$\hat{q} = \begin{bmatrix} \hat{h} \\ \hat{h}\hat{u} \end{bmatrix}$$
where
$$\hat{h} = \frac{h_l+h_r}{2}, \text{ and } \hat{u} = \frac{u_l\sqrt{h_l}+u_r\sqrt{h_r}}{\sqrt{h_l}+\sqrt{h_r}}$$

Outline

Hierarchical LTS Scheme

Hierarchical LTS Scheme

Continue: Godunov & Roe solver

- The Godunov & Roe solver previously introduced is worked on the GTS where Δt is constant at each loop.

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x_i}(F^n_{i+\frac{1}{2}} - F^n_{i-\frac{1}{2}})$$

- Δt will be limited by *CFL condition*: $\Delta t \leq C_{max} min(\frac{\Delta x_i}{U_i})$

Hierarchical LTS Scheme

Continue: Godunov & Roe solver

- The Godunov & Roe solver previously introduced is worked on the GTS where Δt is constant at each loop.

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x_i}(F^n_{i+\frac{1}{2}} - F^n_{i-\frac{1}{2}})$$

- Δt will be limited by *CFL condition*: $\Delta t \leq C_{max} min(\frac{\Delta x_i}{U_i})$

Solution: LTS

- Use Δt_i locally, to avoid the unnecessary calculation.
- Question:
  How to synchronize with different time step per cell?

Hierarchical LTS Scheme

General equation of LTS

- Most part remain the same with GLS scheme instead of Δt_i

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t_i}{\Delta x_i}(F^n_{i+\frac{1}{2}} - F^n_{i-\frac{1}{2}})$$

Where Δt_i is equal to $2^{l-1}\Delta t_{min}$. Δt_{min} is the minum time step calculated by GTS. l is the interger of pre-set level for each cell.

Local CFL Condition

Local Courant Number

- A preliminary cell-based LTS level l_c is assigned to cell i according to the local Courant number Cr_i computed as

$$Cr_i = \frac{\Delta t_{min}}{\Delta x_i} max_{k=1,2}(u_k)$$

Where u_k is the cell wave speed on one of 2 directions.

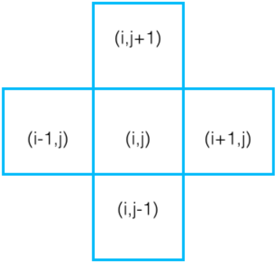
Local CFL Condition

Local Courant Number

- A preliminary cell-based LTS level l_c is assigned to cell i according to the local Courant number Cr_i computed as

$$Cr_i = \frac{\Delta t_{min}}{\Delta x_i} max_{k=1,2}(u_k)$$

Where u_k is the cell wave speed on one of 2 directions.



Local CFL Condition

Local Time Step Level Criterion

- After calculated the local Courant number the local time step level is determined by

$$Cr_0/2^{l-1} \leq Cr_i < Cr_0/2^{l-2}$$

with the exception of level 1 which is controlled by $Cr_i > Cr_i/2$. Cr_0 is the maximum courant number for the simulation.

Local CFL Condition

Local Time Step Level Criterion

- After calculated the local Courant number the local time step level is determined by

$$Cr_0/2^{l-1} \leq Cr_i < Cr_0/2^{l-2}$$

with the exception of level 1 which is controlled by $Cr_i > Cr_i/2$. Cr_0 is the maximum courant number for the simulation.

Create LTS buffer level

- For stability reason, LTS buffer level in neighbour cells should be created.
- LTS level in each cell should be minum LTS level in its neighbour.

Create LTS buffer level

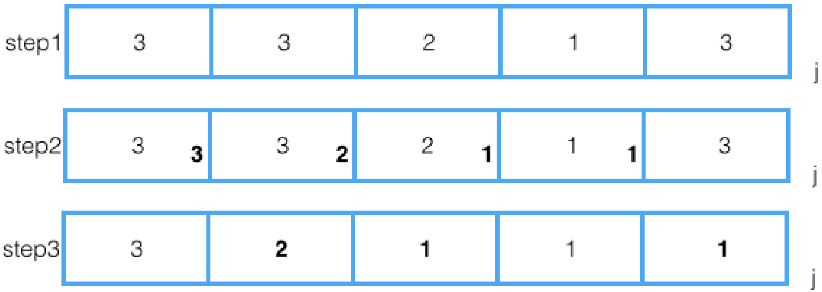


Figure: Create buffer level

Hierarchical LTS Scheme: illustration

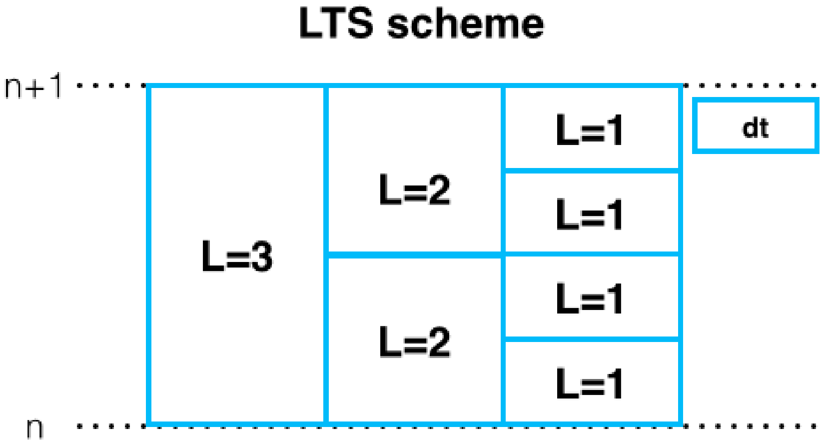


Figure: LTS scheme: first look

Hierarchical LTS Scheme: illustration

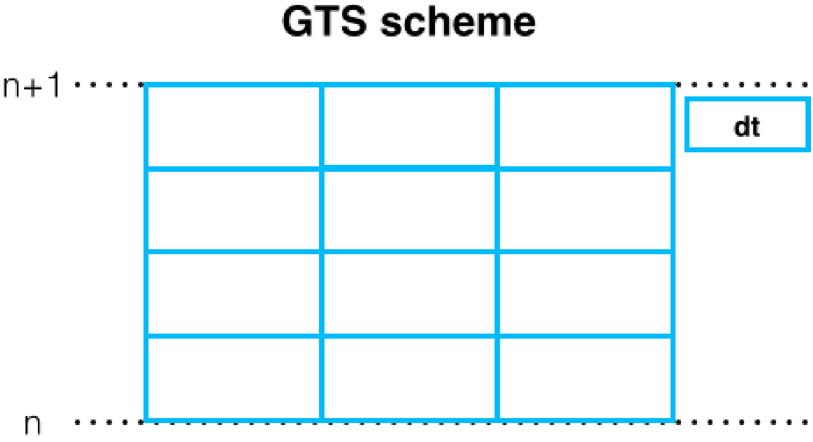


Figure: GTS scheme: computational over head

Hierarchical LTS Scheme: illustration

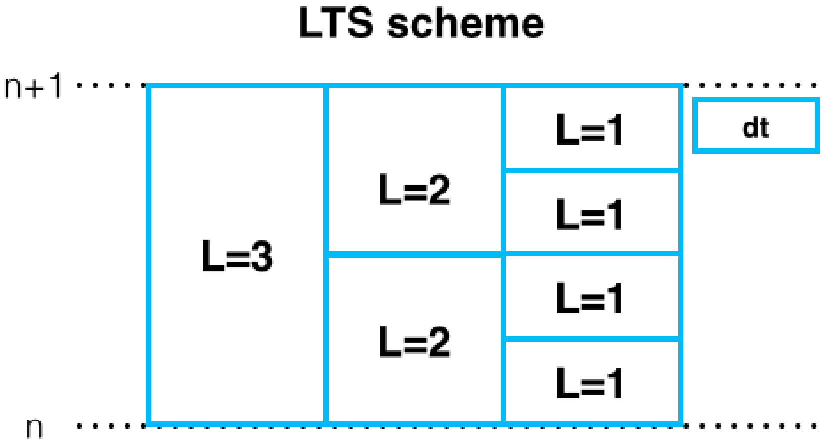


Figure: LTS scheme: benefit

Hierarchical LTS Scheme: illustration

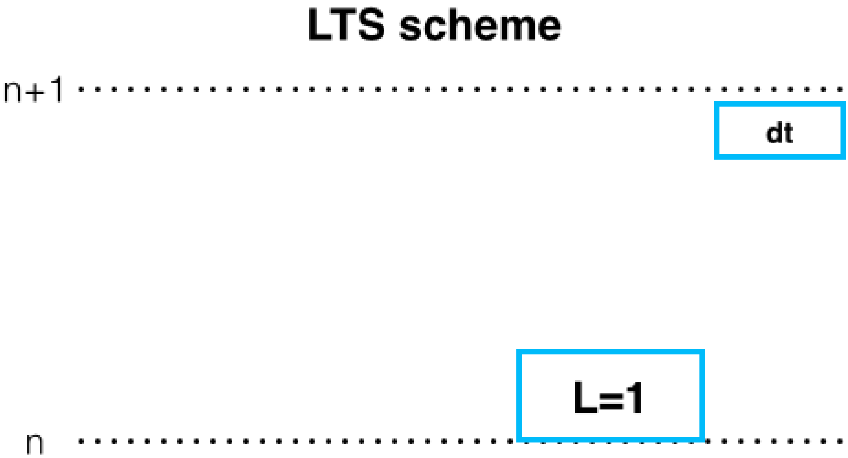


Figure: Local cell update: initial step

Hierarchical LTS Scheme: illustration

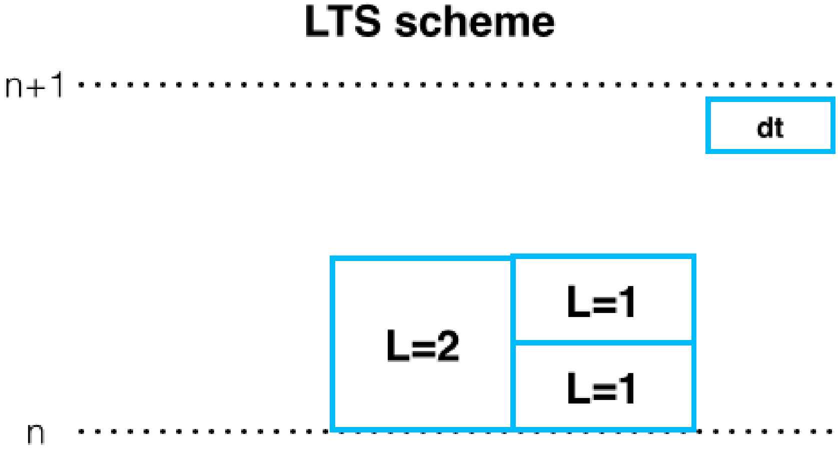


Figure: Local cell update: step 2

Hierarchical LTS Scheme: illustration

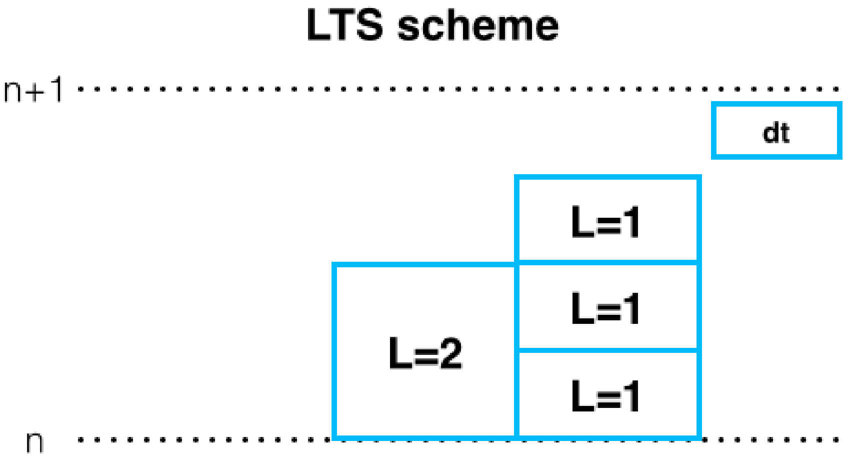


Figure: Local cell update: step 3

Hierarchical LTS Scheme: illustration

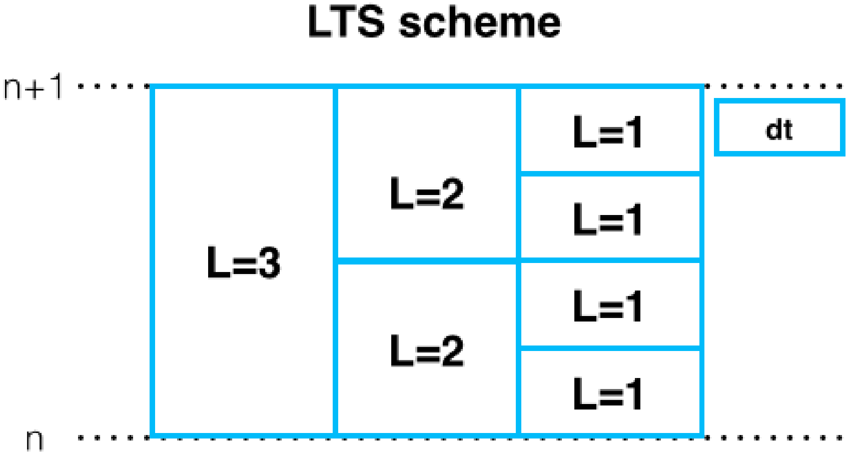
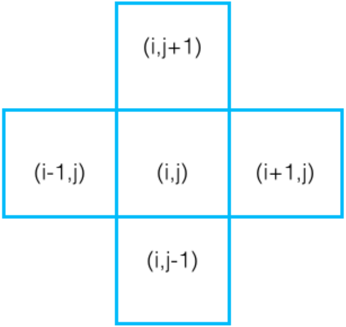


Figure: Local cell update: step 4

Summery for LTS

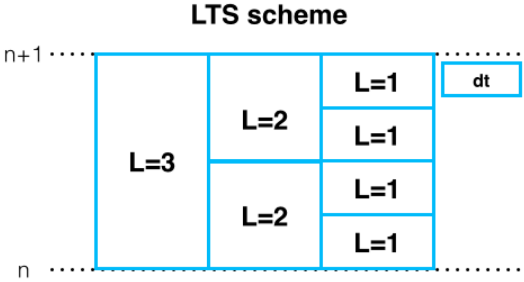
- Step 1: Loop over all cells and set the local time step level based on the local CFL condition. Calculating buffered cell and assigned final LTS level to each cell.



- Step 2: Local update based on the level set before.
- Step 3: Update global time step and go to step 1.

Summery for LTS

- Step 1: Loop over all cells and set the local time step level based on the local CFL condition. Calculating buffered cell and assigned final LTS level to each cell.
- Step 2: Local update based on the pre-seted level.



- Step 3: Update global time step and go to step 1.

Outline

1D SWE & LTS vs. GTS

- The experienment is based on breaking dam scheme in 1D shallow water equations(GTS scheme - timestep 200)

(Loading)

1D SWE & LTS vs. GTS

- The experienment is based on breaking dam scheme in 1D shallow water equations(LTS scheme - timestep 200)

(Loading)

2D SWE & LTS

- The experienment is based on breaking dam scheme in 2D shallow water equations

(Loading)

2D SWE & LTS

- The experienment is based on breaking dam scheme in 2D shallow water equations

(Loading celldt.mp4)

2D SWE LTS Error Measurement

Experienmental Environment

- Domain size: 50 by 50
- Error measurement is using the norm formula in compare with GTS

$$E_2(d_{LTS}) = (\sum_{i=1,\ldots,N}[(d_{LTS})_i - d_{GTS})_i]^2)^{1/2}$$

2D SWE LTS Error Measurement

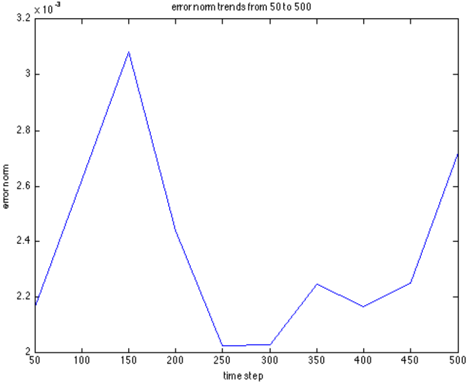


Figure: Norm difference of GTS & LTS with different different time step($L = 3$)

Benchmark

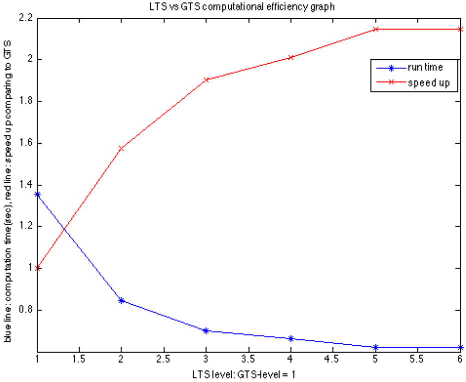


Figure: graph showed runtime difference of GTS & LTS with different LTS levels(1000 time steps)

Outline

References:

[1]R.J. LeVeque. Finite Volume Methods for Hyperbolic Problems. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002

[2] R.J.LeVeque, D.L.George, M.J.Berger. Tsunami modelling with adaptively refined finite volume methods. Acta Numerica, pp.211 - 289, 2011

[3] Bradford, S.F. and Sanders, B.F., Finite-Volume Model for Shallow-Water Flooding of Arbitrary Topography, ASCE Journal of Hydraulic Engineering. 128(3), 289-298, 2002.

[4] Crossley, A.J., Wright, N.G., Whitlow, C.D. Local time stepping for modeling open channel flows. J. Hydr. Engng.129(6), 455462, 2003

[5]Osher, S., Sanders, R. Numerical approximations to nonlinear conservation laws with local varying time and space grids. Mathematics of Computation 41(164), 321336,1983.

Thanks for your attention! Questions?