Quadrature-free Discontinuous Galerkin Formulation for Shallow-water Equations with Code Generation Features

Sara Faghih-Naini [a, b, *], Vadym Aizinger [a], Sebastian Kuckuk [b], Daniel Zint [c], Harald Köstler [b] and Roberto Grosso [c]

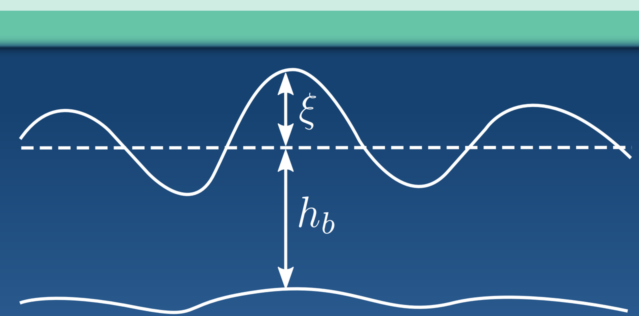
[a] University of Bayreuth, Scientific Computing; * Sara.Faghih-Naini@uni-bayreuth.de

[b] Friedrich-Alexander-University Erlangen-Nuremberg (FAU), System Simulation; [c] FAU, Computer Graphics

Ocean Circulation Models

- Discontinuous Galerkin (DG) method for higher order and accuracy
- Novel quadrature-free scheme for better performance
- Automated code generation for better performance, productivity and portability
- Generated block-structured grids for realistic ocean domains

2D Shallow-water Equations [1]



$$\frac{\partial \xi}{\partial t} + \nabla \cdot (\boldsymbol{U}) = 0$$

$$\frac{\partial (\boldsymbol{U})}{\partial t} + \nabla \cdot (\boldsymbol{U}\,\boldsymbol{U}^T / H\,) + \tau_{bf}\boldsymbol{U} + f_c \boldsymbol{k} \times \boldsymbol{U} + gH\nabla\xi = \boldsymbol{F}$$

ξ : elevation of the free water surface, h_b : bathymetric depth,
$H = h_b + \xi$: total fluid depth, f_c : Coriolis parameter, $\boldsymbol{k}$: local vertical vector,
$\boldsymbol{U} = (U, V)^T$: depth integrated horizontal velocity field,
$\boldsymbol{F}$: forcing term, g : gravitational acceleration, τ_{bf} : bottom friction coefficient

Modification of the System for Quadrature-free Integration with $\mathrm{c} := (\xi, U, V)^T$

Compact representation of SWE in conservative form:

$$\frac{\partial \boldsymbol{c}}{\partial t} + \nabla \cdot \begin{pmatrix} U & V \\ \frac{U^2}{H} + \frac{1}{2}g(H^2 - h_b^2) & \frac{UV}{H} \\ \frac{UV}{H} & \frac{V^2}{H} + \frac{1}{2}g(H^2 - h_b^2) \end{pmatrix} = \begin{pmatrix} 0 \\ -\tau_{bf}U + f_c V + g\xi\frac{\partial h_b}{\partial x} + F_x \\ -\tau_{bf}V - f_c U + g\xi\frac{\partial h_b}{\partial y} + F_y \end{pmatrix}$$

Introducing depth averaged velocity field $\boldsymbol{u} = (u, v)^T$

Recasting of nonlinearities into product form

$$\frac{\partial \boldsymbol{c}}{\partial t} + \nabla \cdot \underbrace{\begin{pmatrix} U & V \\ Uu + \frac{1}{2}g(H^2 - h_b^2) & Uv \\ Vu & Vv + \frac{1}{2}g(H^2 - h_b^2) \end{pmatrix}}_{\tilde{A}(\boldsymbol{c}, \boldsymbol{u})} = \underbrace{\begin{pmatrix} 0 \\ -\tau_{bf}U + f_c V + g\xi\frac{\partial h_b}{\partial x} + F_x \\ -\tau_{bf}V - f_c U + g\xi\frac{\partial h_b}{\partial y} + F_y \end{pmatrix}}_{\boldsymbol{r}(\boldsymbol{c}, \boldsymbol{u})}$$

$$\boldsymbol{u}\,H = \boldsymbol{U}$$

Semi-discrete DG Formulation on Ω_e

$\{\mathcal{T}_\Delta\}_{h>0}$ triangulation of $\Omega \subset \mathbb{R}^2$ with Ω_e elements of $\mathcal{T}_\Delta$, discontinuous polynomial space $\mathbb{V}_\Delta = \{\varphi_\Delta \in L^1(\Omega) : \varphi\big|_T \in \mathbb{P}_p(T), \forall T \in \mathcal{T}_\Delta\}$; seek $\boldsymbol{c}_\Delta \in (\mathbb{V}_\Delta)^3, \boldsymbol{u}_\Delta \in (\mathbb{V}_\Delta)^2$, s. t. for $t \in (t_0, t_{end}), \forall \Omega_e \in \mathcal{T}_\Delta\ \forall \phi_\Delta \in (\mathbb{V}_\Delta)^3$ and $\forall \psi_\Delta \in (\mathbb{V}_\Delta)^2$:

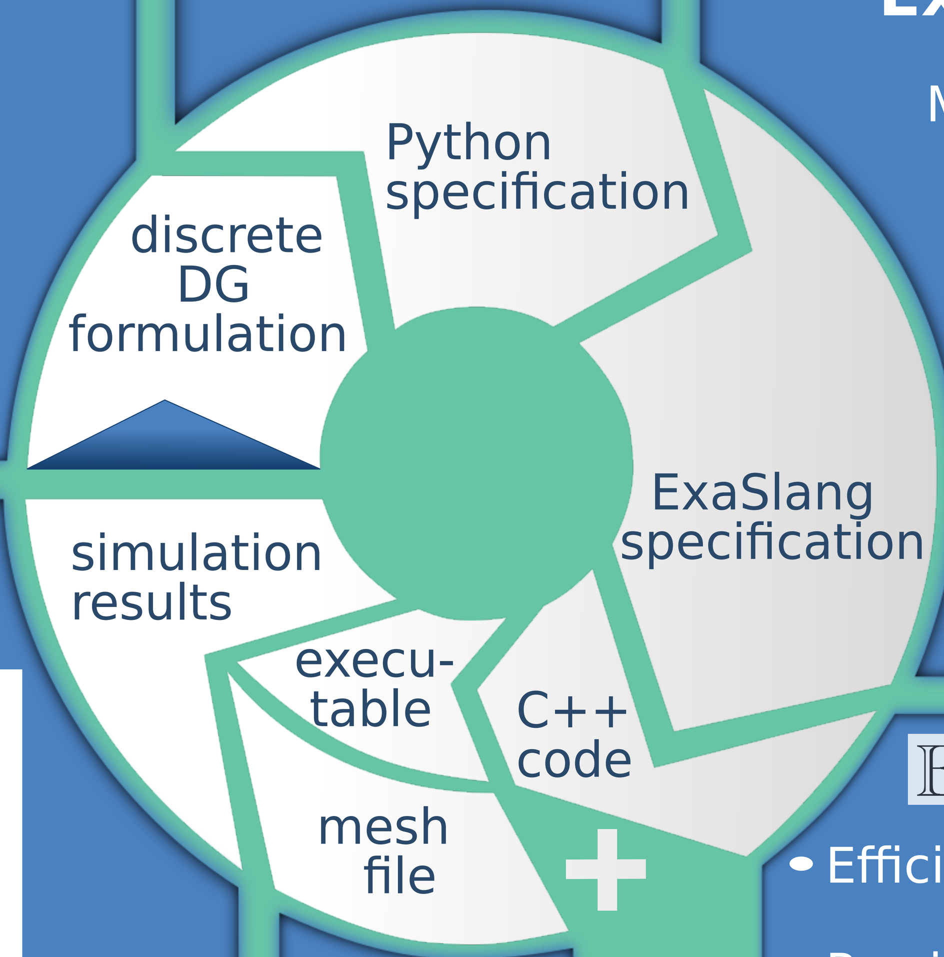
$$(\partial_t \boldsymbol{c}_\Delta, \boldsymbol{\phi}_\Delta)_{\Omega_e} + \langle \hat{\tilde{\boldsymbol{A}}}(\boldsymbol{c}_\Delta, \boldsymbol{u}_\Delta, \boldsymbol{c}_h^+, \boldsymbol{u}_\Delta^+; \boldsymbol{n}), \boldsymbol{\phi}_\Delta \rangle_{\partial\Omega_e} - \left(\tilde{\boldsymbol{A}}(\boldsymbol{c}_\Delta, \boldsymbol{u}_\Delta), \nabla\boldsymbol{\phi}_\Delta\right)_{\Omega_e} = (\boldsymbol{r}(\boldsymbol{c}_\Delta, \boldsymbol{u}_\Delta), \boldsymbol{\phi}_\Delta)_{\Omega_e}$$

$$(\boldsymbol{u}_\Delta \cdot H_\Delta, \boldsymbol{\psi}_\Delta)_{\Omega_e} = (\boldsymbol{u}_\Delta H_\Delta, \boldsymbol{\psi}_\Delta)_{\Omega_e}$$

With Lax-Friedrichs flux $\hat{\tilde{\boldsymbol{A}}}(\boldsymbol{c}_\Delta, \boldsymbol{u}_\Delta, \boldsymbol{c}_\Delta^+, \boldsymbol{u}_\Delta^+; \boldsymbol{n}) := \frac{1}{2}\left(\left(\tilde{\boldsymbol{A}}(\boldsymbol{c}_\Delta, \boldsymbol{u}_\Delta) + \tilde{\boldsymbol{A}}(\boldsymbol{c}_\Delta^+, \boldsymbol{u}_\Delta^+)\right) \cdot \boldsymbol{n} + |\hat{\lambda}|\,(\boldsymbol{c}_\Delta - \boldsymbol{c}_\Delta^+)\right)$

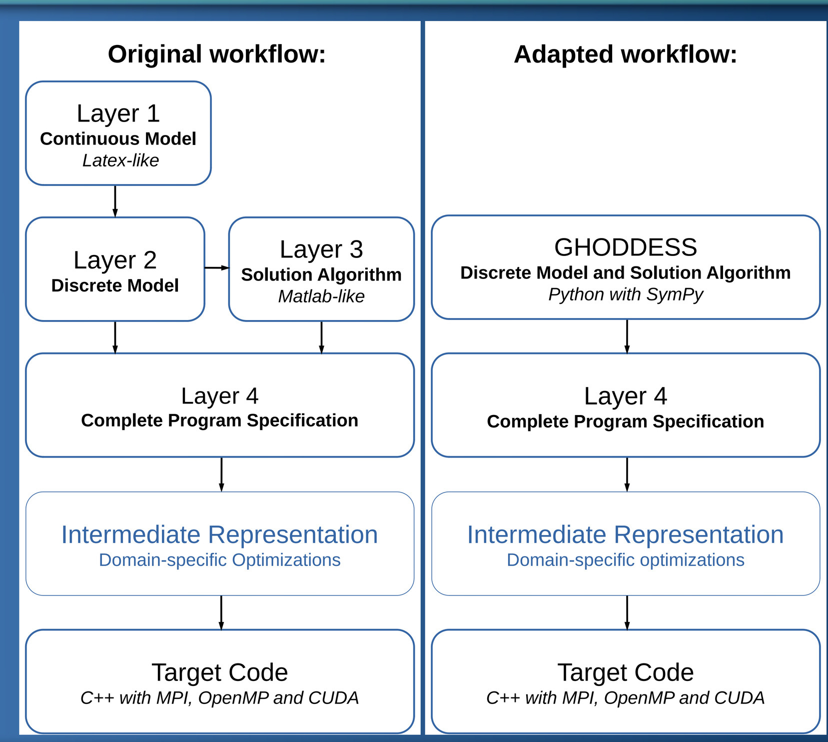
GHODDESS (Generation of Higher-Order Discretizations Deployed as ExaSlang Specifications) [2]

- Uses Python library sympy (analytical differentiation and integral evaluation)
- Contains classes representing triangles and data fields



ExaSlang [3]

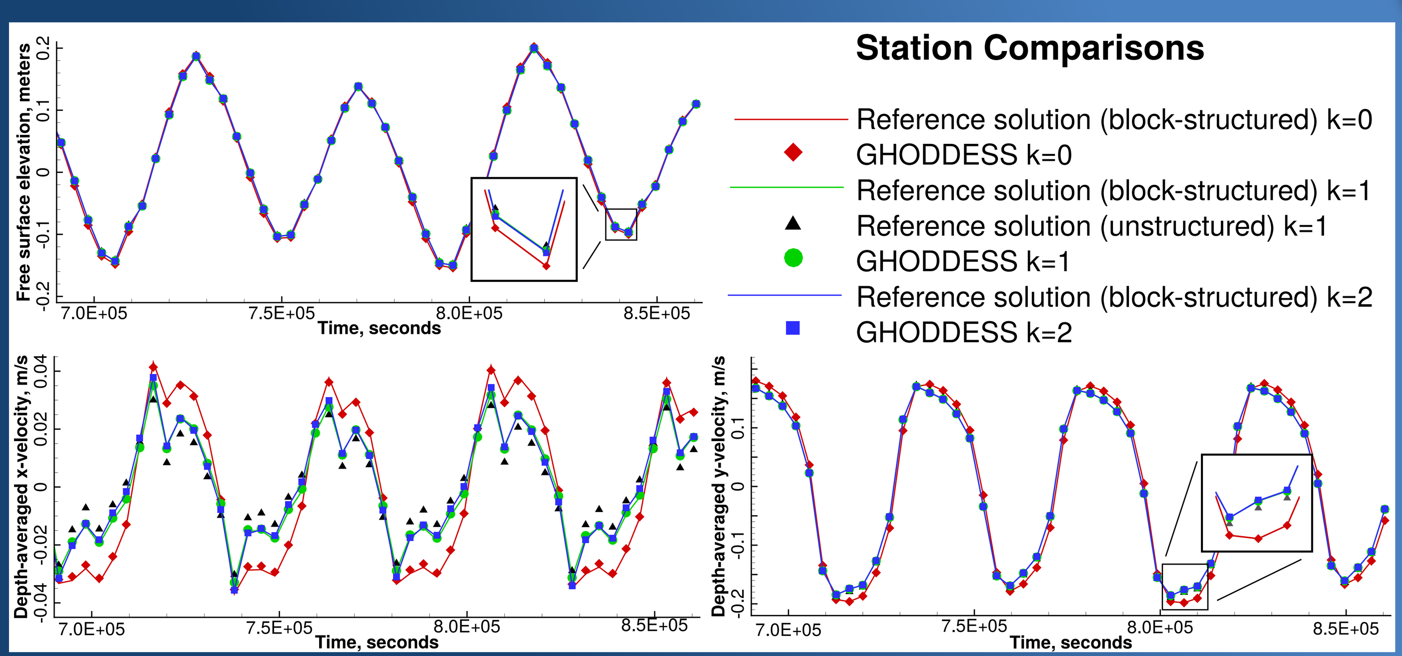
Multi-layered Domain-Specific Language

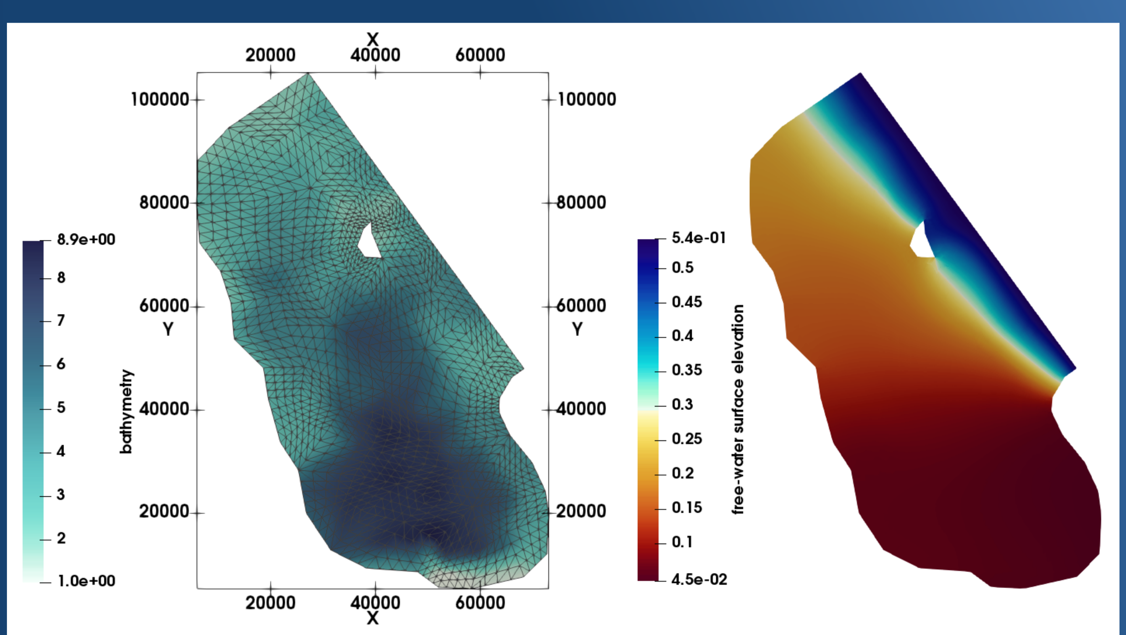


ExaStencils SPPEXA [4]

- Efficient code generation for massively parallel applications
- Parsing ExaSlang code, transforming code elements for low-level optimizations
- Maps to C++ code with CUDA, MPI and OpenMP

Tidal Flow near Bahamas

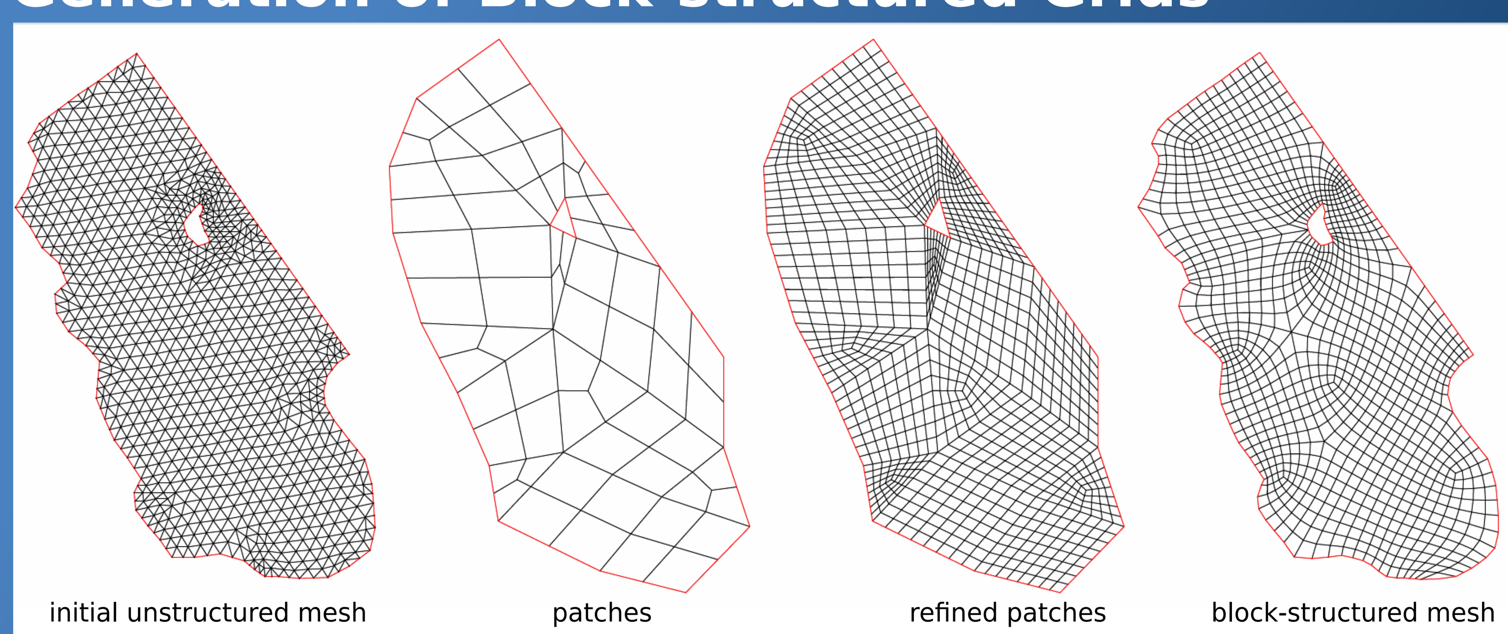




Simulation Details

- Simulation Time: 10 days
- Time step size: 15 seconds
- Block-structured mesh with 2624 elements
- Initial condition: cold start
- Ramped tidal forcing with five tidal components at the open boundary
- DG space: piecewise linear

Generation of Block-structured Grids [5]



Current and Future work

- Incorporate p-adaptivity into the quadrature-free formulation
- Scalability and performance on different hardware architectures
- Transfer approaches to 3D ocean models

References

1 V. Aizinger and C. Dawson. "A discontinuous Galerkin method for two-dimensional flow and transport in shallow water". In: Advances in Water Resources 25.1 (2002)

2 S. Faghih-Naini, S. Kuckuk, V. Aizinger, D. Zint, R. Grosso, H. Köstler, "Quadrature-free discontinuous Galerkin method with code generation features for shallow water equations on automatically generated block-structured meshes". In: Advances in Water Resources 138 (2020)

3 S. Kuckuk. "Automatic Code Generation for Massively Parallel Applications in Computational Fluid Dynamics". PhD Thesis. (2019)

4 C. Lengauer, S. Apel, M. Bolten, A. Größlinger, F. Hannig, H. Köstler, U. Rüde, J. Teich, A. Grebhahn, S. Kronawitter, S. Kuckuk, H. Rittich, and C. Schmitt. ExaStencils: Advanced stencil-code engineering. In: Euro-Par 2014: Parallel Processing Workshops, volume 8806 of Lecture Notes in Computer Science. Springer. (2014)

5 D. Zint, R. Grosso, V. Aizinger, H. Köstler. "Generation of Block Structured Grids on Complex Domains for High Performance Simulation". In: Numerical Geometry, Grid Generation and Scientific Computing. Springer. (2019)