Final Report for MIT Portugal Partnership (MPP2030) Research Seed Fund Project (2020)

Project Title: A Computationally Scalable Global Ocean Model

C.N. Nguyen and J. Peraire, MIT

Portugal collaborators: P. Revlas and E. Garel, University of Algarve

August 30, 2022

The goal of our research has been the development of a high-order discontinuous Galerkin algorithm to enable large scale ocean modeling. Towards this goal, we have devised a new, three dimensional, non-hydrostatic pressure, free surface model. The model has been tested for two canonical test cases to verify its accuracy. In addition, the model has been recast as a system of conservation laws in incorporated in to the open-source `Exasim` Vila-Pérez *et al.* (2022) framework thus ensuring scalability and portability to a wide range of computer platforms including GPU-based computers. Work was initiated to use the existing model to simulate the complex conditions encountered in the Gulf of Cadiz. Still, the full simulation and validation with experimental data were not completed within the scope of this project and will be the subject of our future work.

1 Research Results

Numerical models are one of the most effective tools to understand the ocean dynamics at coastal, regional, and global scales. They allow for reconstruction of historic states, short term forecasts of ocean states, as well as for climate projections and possible future scenarios. One core requirement for the predictability of such model is that basic hydrodynamic processes are reproduced accurately. The resolution, in terms of horizontal and vertical grid discretization, plays an important role in the definition of the processes that a model is able to resolve. Ocean modeling is a formidable challenge due to the enormous resolution required to resolve turbulent mesoscale eddies down to 0.1 degree horizontally and 10 meters vertically. As most existing ocean models were developed years ago, they are not able to run on accelerators such as GPUs.

We have proposed a high-order, scalable ocean model by taking advantage of the latest advances in algorithmic development and computer hardware. Our group has developed Discontinuous Galerkin Methods for a wide range of applications. DG methods can deliver high-order accurate solutions and can be implemented on arbitrary meshes, as required by complex geometries. Our implicit time integration formulation effectively handles the problem stiffness caused by the different horizontal and length scales but requires the solution of large systems of equations at every timestep. Much work has gone into developing efficient preconditioners to allow simulations of turbulent aerodynamics problems Fernandez *et al.* (2017) over many thousands of CPU processors. However, the matrix storage requirements of the most effective approaches are unsuitable for modern GPU-based architectures. We have recently developed a matrix-free discontinuous Galerkin method aimed at exploiting GPUs Cuong Nguyen *et al.* (2022) and demonstrated the potential to scale these methods to simulations involving billions of unknowns.

1.1 Model Equations

We consider the general nonlinear, non-hydrostatic pressure, free-surface flow model given by,

$$\nabla \cdot \boldsymbol{u} = 0\,, \tag{1}$$

$$\boldsymbol{u}_{,t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \frac{1}{\rho_0}\nabla p + 2\boldsymbol{\Omega}\times\boldsymbol{u} = \boldsymbol{g} + \nabla\cdot(\boldsymbol{\nu}\nabla\boldsymbol{u}) - \boldsymbol{g}\alpha(T - T_0)\,, \tag{2}$$

$$T_{,t} + \boldsymbol{u}\cdot\nabla T = \frac{1}{\rho C_p}\nabla\cdot(\boldsymbol{k}\nabla T) + \frac{J}{\rho C_p} \tag{3}$$

Here, $\boldsymbol{u}$ is the three dimensional velocity vector, p is the pressure, and T is the temperature. In addition, $\boldsymbol{\Omega}$ is the earth angular velocity responsible for Coriolis' acceleration, $\boldsymbol{g}$ is the acceleration of gravity, C_p is the heat capacity of water and α is the thermal expansion coefficient of water. Finally, the density ρ is given by $\rho = \rho_0(1 - \alpha(T - T_0))$, where ρ_0 and T_0 are reference density and temperatures, respectively. Finally the tensors of kinematic viscosity $\boldsymbol{\nu}$ and thermal conductivity are written as

$$\boldsymbol{\nu} = \begin{pmatrix} \nu_H & 0 & 0 \\ 0 & \nu_H & 0 \\ 0 & 0 & \nu_V \end{pmatrix}, \quad \boldsymbol{k} = \begin{pmatrix} k_H & 0 & 0 \\ 0 & k_H & 0 \\ 0 & 0 & k_V \end{pmatrix}$$

once expressed in local horizontal and vertical components. In the expressions above, we have included temperature effects, but note that the effects of salinity, or any other passively advected scalar, such as substance concentrations, can be accounted for in a similar manner.

Free Surface Boundary Conditions
The following boundary conditions are considered at the free surface

$$p = p_a + \rho g\eta \tag{4}$$

$$(\boldsymbol{\nu}\nabla\boldsymbol{u})\cdot\boldsymbol{n} = \frac{\boldsymbol{\tau}_{\text{wind}}}{\rho_0} \tag{5}$$

$$(\boldsymbol{k}\nabla T)\cdot\boldsymbol{n} = h(T - T_a) \tag{6}$$

where η is the free surface elevation relative to the men free level, $\boldsymbol{\tau}_{\text{wind}}$ is the wind stress, p_a and T_a are the local atmospheric pressure and temperature, and h is the convective heat transfer coefficient.

Boundary conditions at the sea bed, and coastal/open boundaries are problem dependent and discussed together with the applications.

2 Formulation

In order to handle the incompressibility constraint, we replace continuity equation, 1, with

$$\frac{1}{c_0^2}p_{,t} + \nabla\cdot\boldsymbol{u} = 0\,, \tag{7}$$

where c_0 is artificial bulk wave speed.
The kinematic free surface boundary condition takes the form,

$$\eta_{,t} + u\eta_{,x} + v\eta_{,y} - w = 0\,, \tag{8}$$

where η where u, v and w are components of the velocity in the two local horizontal directions, x and y, and in the local vertical direction z, respectively. The next step is the linearization of the free surface boundary condition. To that end, we consider the domain where the problem variables are defined to be fixed and extend form the mean free surface at $z = 0$ to $z = -H$ at the sea bed. Moreover, since we are primarily interested in long waves, we replace equation 8 with

$$\eta_{,t} - w = 0 \ . \tag{9}$$

In order to avoid a non local model, this linearized equation is solved in the whole computational domain thus extending the definition of η to the interior of the domain.

3 Model validation using a linear Airy wave

Assuming small perturbations from the equilibrium position, a constant depth, and, neglecting Coriolis, earth curvature effects, viscous and thermal effects, equations 1 and 2 admit analytical planar waves solutions known as Airy waves. A propagating Airy wave in the x has the form

$$\eta = A\cos(kx - \omega t)\,, \tag{10}$$

$$\phi = \frac{\omega}{k} A \frac{\cosh k(z+H)}{\sinh kH} \sin(kx - \omega t) \ . \tag{11}$$

with the dispersion relation

$$\omega^2 = gk \tanh kH$$

.

Calculating u, w and p, we get

$$u = \phi_{,x} = \omega A \frac{\cosh k(z+H)}{\sinh kH} \cos(kx - \omega t) \tag{12}$$

$$w = \phi_{,z} = \omega A \frac{\sinh k(z+H)}{\sinh kH} \sin(kx - \omega t) \tag{13}$$

$$p = \frac{\omega^2}{k} A \frac{\cosh k(z+H)}{\sinh kH} \cos(kx - \omega t) - gz \ . \tag{14}$$

Also, the extension of $\eta(x, z, t)$ is given by

$$\eta = A \frac{\sinh k(z+H)}{\sinh kH} \cos(kx - \omega t) \ . \tag{15}$$

Since we are dealing with planar waves, we can conduct our validation exercise in two dimensions $(x, z) \in [X_L, X_R] \times [-H, 0]$ as show in in figure 1.

After the simplifying assumptions, our system of governing equations becomes

$$p_{,t} + c_0^2(u_{,x} + w_{,z}) = 0 \tag{16}$$

$$u_{,t} + p_{,x} = 0 \tag{17}$$

$$w_{,t} + p_{,z} = -g \tag{18}$$

$$\eta_t - w = 0 \tag{19}$$

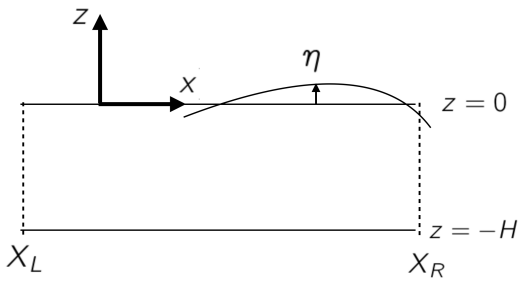


Figure 1: Computational domain for Airy wave simulation.

	$\|\|e_u\|\|_\infty$	$\|\|e_v\|\|_\infty$	$\|\|e_\eta\|\|_\infty$	$\|\|e_p\|\|_\infty$	$\|\|e_{div}\|\|_\infty$
$c_0^2 = 1.eE+04$	2.1000e-03	5.9773E-04	4.0100E-04	8.0911E-04	1.6472E-05
$c_0^2 = 1.eE+05$	2.1000e-03	6.0580e-04	4.6529e-04	8.0852E-04	1.7859E-06

Table 1: Solution errors at $T = 1$, after 250 time steps as a function of the artificial compressibility coefficient. Solutions computed with cubic polynomials, second order timestepping, $g = 1$, $H = 1$.

with boundary conditions $p = g\eta$ at $z = 0$, $w = 0$ at $z = -H$ and periodic boundary conditions on all variables between $x = X_L$ and $x = X_R$. The simplified set of equations 16-19, is easily coded in `Exasim` and solutions are computed on a 20×10 rectangular mesh for different problem parameters and polynomial degrees. Some results are shown in figure 2.
To highlight the non-hydrostatic pressure distribution, we show in figure 3 the total pressure distribution and the non-hydrostatic component of the pressure obtained by subtracting the hydrostatic pressure from the total pressure, for a typical shallow water solution.

In order to asses the accuracy of the proposed approach, the computed results are compared with the exact solution in tables 1 and 2. We should the errors in the infinite norm in the u and v velocity components, the surface elevation η, the pressure p and the divergence of the velocity. The computation is conducted for a unit of time, $T = 1$, $kH = 3$, $H = 1$ and $g = 1$. In table 1, we study the effect of changing the value of the artificial compressibility. We note that by increasing the value by a factor of 10, the errors in the divergence of the velocity are decreased by a factor of 10, while the errors in all other quantities remain practically unchanged. This indicates that the error in the solution is dominated by the approximation error and the incompressibility condition is sufficiently well satisfied so that the impact on the solution error is minor. In table 2, we look at the approximation errors for different spatial polynomial orders, time integration orders and numbers of time steps. Comparing the first two rows, we note that doubling the number of time steps has no effect on the accuracy, indicating that the solution error is dominated by the spatial approximation. This is confirmed in the last two, when we increase the order of the polynomial and es expected the error is reduced by two orders of magnitude.

4 Model validation using a perturbed Bickley jet

In order to validate the accuracy of the convective terms, we consider a two-dimensional initial value problem consisting of a Bickley jet perturbed by sinusoidal vortices. The domain is doubly-periodic and square with dimensions $x \in [-2\pi, 2\pi)$, $y \in [-2\pi, 2\pi)$. The initial condition is divergence-free, and the initial density or surface displacement perturbations are zero. We introduce a passive dye field, $\sin(y/2)$, to visualize the solution.

A configuration of the Bickley jet Poulin & Flierl (2003), using the shallow water equations, illustrates an

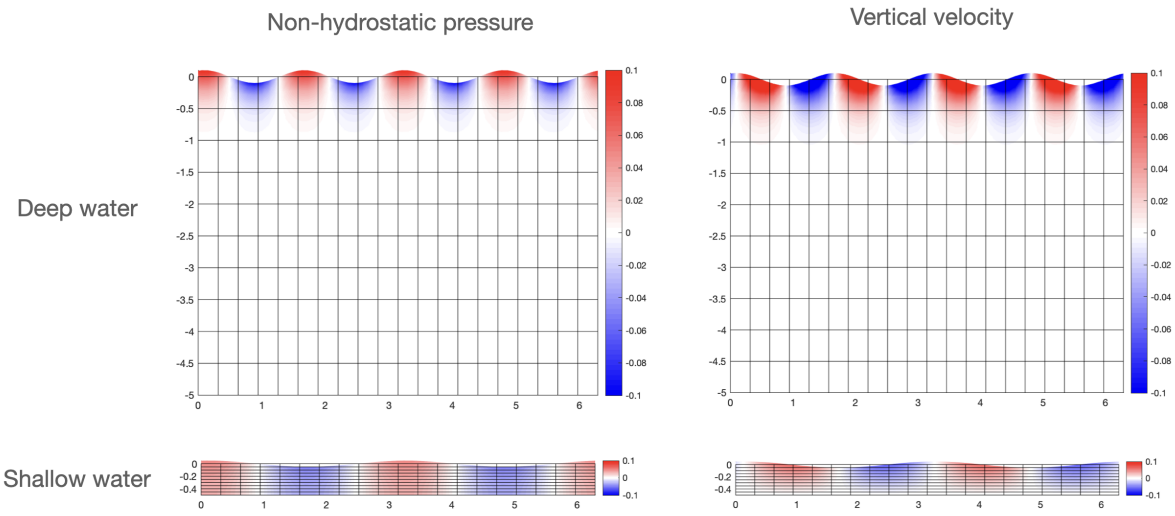
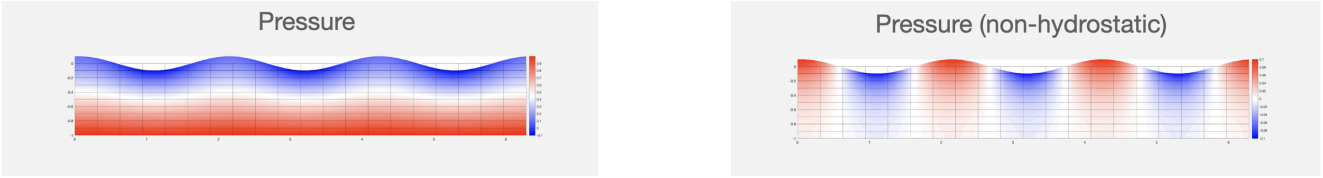


Figure 2: Numerical results computed for the Airy wave. The top row shows the pressure and vertical velocity for a deep water ($kH = 20$) solution, whereas the bottom row shows the same quantities for a shallow water ($kH = 0.8$) solution.



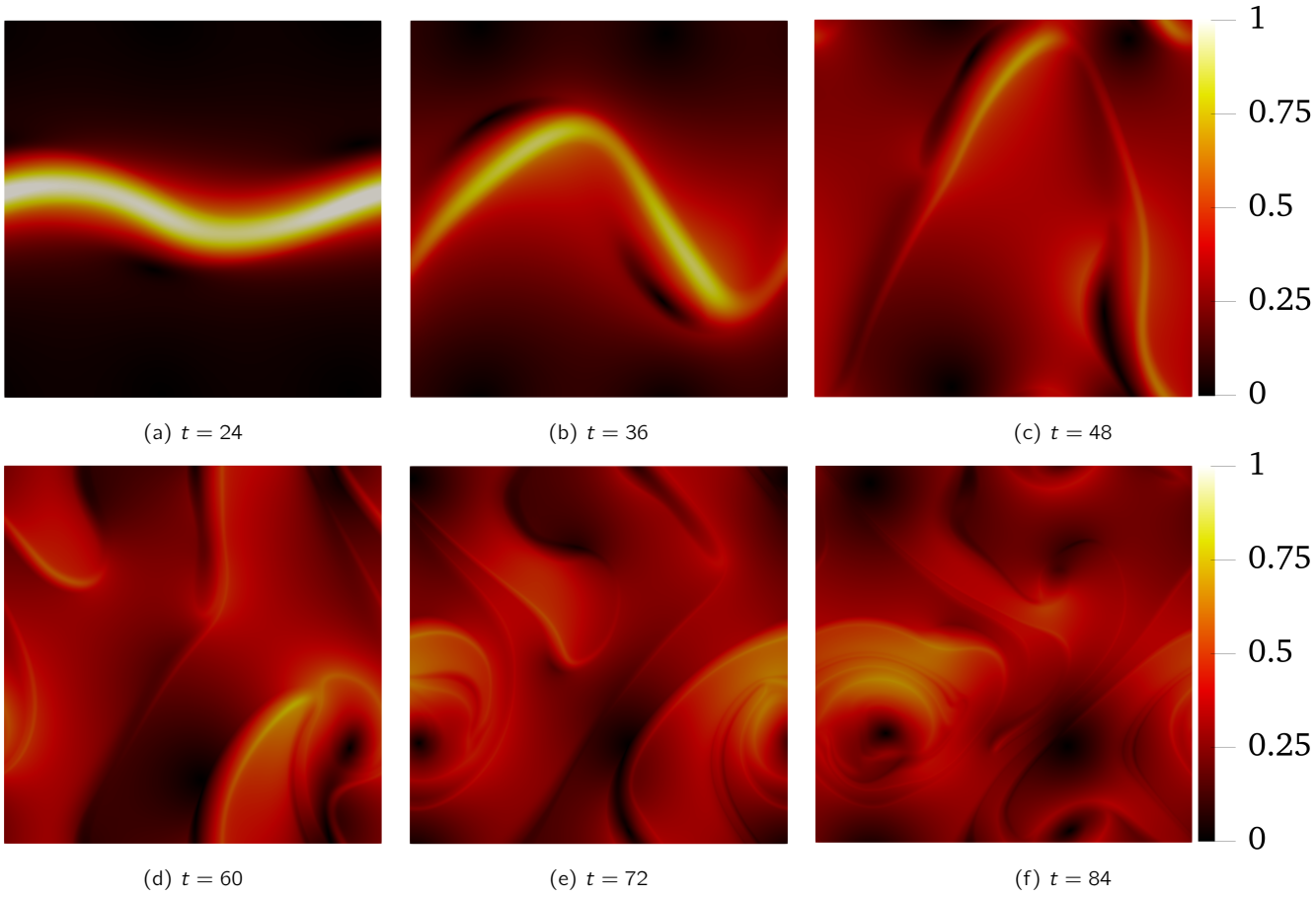
(a) Total pressure distribution

(b) Non-hydrostatic pressure distribution

Figure 3: pressure distributions for a typical shallow water Airy wave.

	$\|\|e_u\|\|_\infty$	$\|\|e_v\|\|_\infty$	$\|\|e_\eta\|\|_\infty$	$\|\|e_p\|\|_\infty$	$\|\|e_{div}\|\|_\infty$
$p = 3, k = 2, N = 250$	2.1000e-03	5.9773E-04	4.0100E-04	8.0911E-04	1.6472E-05
$p = 3, k = 2, N = 500$	2.1000e-03	5.9767E-04	4.0538E-04	8.1934E-04	1.6457E-05
$p = 4, k = 2, N = 250$	1.7942E-05	2.9752E-05	1.6640E-05	1.9308E-05	1.6669E-05

Table 2: Solution errors at $T = 1$ for $c_0^2 = 1.0e + 04$, $g = 1$, $H = 1$. Errors for different values of the spatial polynomial approximation p, timestepping order k and number of timesteps N.



(a) $t = 24$ (b) $t = 36$ (c) $t = 48$

(d) $t = 60$ (e) $t = 72$ (f) $t = 84$

Figure 4: Bickley jet – Magnitude of the velocity field at different simulation times, computed with $p = 4$ polynomials and a DIRK(3,3) temporal scheme.

example of the convection model. The case describes the temporal evolution of a jet flow, $u_{\text{jet}} = \text{sech}^2(y)$, in the square domain $\Omega = (-2\pi, 2\pi)^2$, subject to slight perturbations on the initial velocity field, i.e.

$$\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} u_{\text{jet}} + \varepsilon + u' \\ \varepsilon + v' \end{bmatrix}, \qquad \begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} \psi k \tan(ky) + \psi y/\ell^2 \\ \psi k \tan(kx) \end{bmatrix} \tag{20a}$$

where $\varepsilon = 0.1$ is an isotropic perturbation and u' and v' are the vortical perturbations of the velocity field, with off-center vorticity field

$$\psi = \exp\left(-(y + \ell/10)^2/(2\ell^2)\right) \cos(kx) \cos(ky), \tag{20b}$$

Gaussian width $\ell = 0.5$ and wavenumber $k = 0.5$.

The problem is solved in a 128×128 Cartesian mesh with periodic boundary conditions, given a non-dimensional gravity value of $g = 10^4$. The simulation uses $p = 4$ polynomials and a DIRK(3,3) scheme. Sketches of the velocity magnitude for different times are depicted in Figure 4, which compares well with other published results. `Exasim` solves this case without the need for physical or artificial diffusion, showing a good resolution and propagation of the flow perturbations.

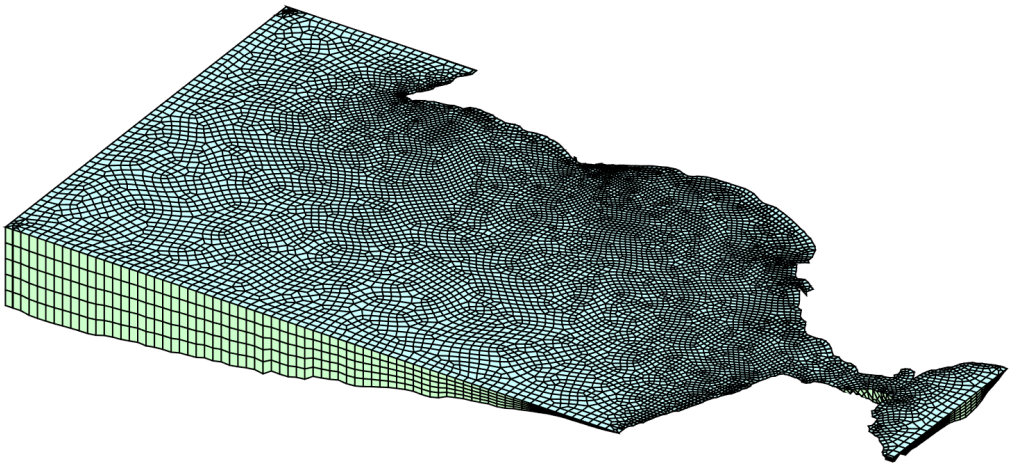


Figure 5: Initial unstructured mesh for the Gulf of Cadiz simulation. The vertical scale has been magnified for display purposes. Note that the mesh already contains the layer of regular elements at the open boundaries to facilitate the imposition of boundary conditions using a sponge layer approach.

5 Initial efforts at modeling the coastal region around the Gulf of Cadiz

We have initiated our efforts at modeling the ocean flow around the coast of Cadiz. To that extent we have implemented a sponge layer boundary condition to facilitate the imposition of the incoming wave while allowing the outgoing wave to leave the domain without reflection at the open boundaries. In addition, we have been able to construct an initial mesh for the simulation. The mesh has been constructed using the bathymetry data from GEBCO (`http://www.gebco.com`). First, an two dimensional mesh of quandrialterals using the the grid generator GMESH Geuzaine & Remacle (2009) has been contructed and then extruded up to the depth dictated by the local bathymetry. The inital mesh constructed is shown in figure 5.

Initial flow simulations have already been conducted, but no useful results have yet been obtained. This will be the focus of our future work, even after the period of the grant has expired.

6 Interactions with our collaborators in Portugal

Initial interactions at the beginning of the project were productive. The Portuguese team provided advice on the nature of the problem and the requirements for the model to be developed. Unfortunately, closing the loop by comparing the computed results with the experimental data has not happened within the period of grant performance. The model development and validation took a significant amount of effort, but we think we are now at a good point to attempt more realistic ocean simulations. We will try this in the future.

References

Cuong Nguyen, Ngoc, Terrana, Sebastien & Peraire, Jaime 2022 Large-Eddy Simulation of Transonic Buffet Using Matrix-Free Discontinuous Galerkin Method. *AIAA Journal* **60** (5), 3060–3077.

Fernandez, P., Nguyen, N. C. & Peraire, J. 2017 The hybridized Discontinuous Galerkin method for Implicit Large-Eddy Simulation of transitional turbulent flows. *Journal of Computational Physics* **336**, 308–329.

Geuzaine, C & Remacle, J F 2009 Gmsh : A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering* **79**, 1309–1331.

Poulin, F. J. & Flierl, G. R. 2003 The nonlinear evolution of barotropically unstable jets. *Journal of Physical Oceanography* **33** (10), 2173–2192.

Vila-Pérez, Jordi, Van Heyningen, R Loek, Nguyen, Ngoc-Cuong & Peraire, Jaume 2022 Exasim: Generating discontinuous Galerkin codes for numerical solutions of partial differential equations on graphics processors. *SoftwareX* **20**, 101212.