National Modelling Initiative: Progress Report

Dec 2020

M. Herzfeld, F. Rizwi, N. Margvelashvili, D. Wild, N. Mortimer

# Background

A high-resolution 3D operational hydrodynamic ocean model around the entire Australian coastline, or national re-analysis, does not currently exist. To date, coastal case studies have either been performed on BLUElink outputs directly, or downscaled models have been developed which are nested within BLUElink. These latter models often require numerous nests to achieve acceptable nesting ratios when downscaling from the global model (10 km resolution) to the area of interest at the coast (100 s metres). A high resolution coastal model (of order 1 km resolution) would allow these ‘bridging’ nests to be removed when downscaling, and would also provide a suitably coastally optimized platform (which BLUElink is not) within which to conduct coastal oceanographic assessments.

Although an operational national modelling platform does not exist, there have been several attempts to develop such models (e.g. the CSIRO ‘Ribbon’ model, CSIRO ARENA model or UWA ozROMS). The development of such a model must overcome numerous technical and political obstacles in order to be successful. The latter issue relates to participation and potential roles of scientists with a stake in developing such a model, ownership, approach (including modelling platforms) and general agreement on a project plan. The ANSR (Australian National Shelf Reanalysis) initiative was such an attempt to overcome such barriers and garner an inter-institutional collaboration, but served to highlight the depth of these issues and was ultimately unsuccessful to date. The lesson learnt from this exercise is that if a national collaboration is to be achieved on a national model then equitable ownership must be established across all participating institutions.

The technical barriers to a national model are many, and many of the current attempts do not adequately address them. The size of the domain is the dominant obstacle, i.e. how to achieve high coastal resolution without an impossible computational burden. A model that encompasses the rectangular size of the Australian continent is at a minimum approximately 5000 x 4000 km. If this were to be resolved at 1 km resolution a grid of 20 million surface cells results, which is a massive computational burden (most coastal applications being less than 100,000). Additionally, large computational inefficiencies result due to the large amount of land in a rectangular domain that must be integrated over, and there exists areas which are well beyond the coastal margins of Australia and not of interest, but must be included in a rectangular domain. It is advantageous to align cells with the principal directions of flow (alongshore and cross-shore), and this is not possible everywhere in the domain using rectangular meshes. Currently the ozROMS model uses this approach and achieves a resolution of 3-4 km with ~4.7 million surface cells. It is unlikely that a target resolution of 1 km could be achieved with this approach, and if so would not be a cost effective use of the compute resources required to achieve it.

An alternative is to use a cyclic polar grid that wraps around the Australian continental shelves like a ‘ribbon’. This approach naturally excludes any land cells, and all cells in the domain are within the coastal margin and therefore of interest. This approach uses significantly less cells, where all cells from the rectangular approach that are over land or the in areas not of interest can be pushed into the coastal margins. An example of this approach (CSIRO ‘Ribbon’ model) achieved a mean cross-shelf resolution of ~2.5 km with 1 million surface cells, and less than 200,000 of those actually wet and involved in numerical integrations. This approach potentially could achieve a target resolution of 1 km with significantly less compute resources than a rectangular grid, however, the generation of these grids is extremely complex, time consuming, and ultimately does not possess the control to place required resolution in desired regions.

An unstructured model does not suffer these technical limitations. The unstructured mesh can be fitted to the coastline, with any arbitrary limit offshore, hence excluding all land cells. Resolution, and resolution transition, can be exactly placed where desired via the use of a weighting function. This function may itself be a function of bathymetry, distance from the coast, tidal amplitude, population density, eddy kinetic energy, any combination of these, or indeed any arbitrary metric. The unstructured approach is particularly suited to a national modelling exercise, and could practicably produce a mesh with coastal resolution of ~1 km. This project aims to produce such a model.

# Method

The approach of the National Modelling Initiative aims to produce similar products to the eReefs project, specifically a near real-time operational model (capable of short term forecasts if adequate forcing data is available) which appends daily output to a growing historical archive, and supported by a data assimilating re-analysis of that archive. The project also aims to address both the technical and political issues associated with development of such a model. In essence, the national model is proposed to be subdivided into a number of discrete ‘tiles’ representing distinct geographical regions (e.g. Great Barrier Reef, East Australian Current, Tasmania etc.) which are then assembled into a national model using 2-way nesting. Each tile could be operated in isolation for calibration and validation purposes, thus increasing throughput to result in a more constrained calibration, and a calibration that is region specific. These regional tiles would be made available to custodians that would champion the assessment of the tile, which would in turn feed back into the regional framework. The regional custodians would sensibly be institutions that have a history and local knowledge of the region comprising the tile. Depending on interest, an open source delivery of the tiles may be made available so that community development of the tiles is achieved to create a collaboration hub of regional modelling. If diverse uptake of these tiles were achieved, then a national ownership of the final product would result where multi-institutional investment could circumvent the institutional tensions so far observed in progressing a national initiative. The progress to date of this approach are considered in turn.

## Pilot model

The national model developed within this project uses the unstructured model COMPAS (Coastal Ocean Marine Prediction Across Scales; Herzfeld et. al., 2020,

<https://research.csiro.au/cem/software/ems/hydro/unstructured-compas/> ). COMPAS is a coastal ocean model designed to be used at scales ranging from estuaries to regional ocean domains. It is a three-dimensional finite volume hydrodynamic model based on the three dimensional equations of momentum, continuity and conservation of heat and salt, employing the hydrostatic and Boussinesq assumptions. The equations of motion are discretized on arbitrary polygonal meshes according to the dynamical core used by Ringler (2013) in the global MPAS model (Model Prediction Across Scales). This framework is described in Thuburn et al. (2009) and Ringler et al. (2010) and is hereafter referred to as TRiSK (Thuburn Ringler Skamarock Klemp). The TRiSK formulation is a generalization of the standard Arakawa C-grid scheme to unstructured grids. The horizontal mesh must be an orthogonal, centroidal and well-centred ‘primal-dual’ tessellation, typically consisting of collections of Voronoi cells and their dual Delaunay triangles. A choice of fixed ‘z’ coordinates or terrain following  coordinates is available in the vertical. The ‘z’ vertical system allows for wetting and drying of surface cells, useful for modelling regions such as tidal flats where large areas are periodically dry. It also allows the free surface to move through multiple layers in areas of large tidal range with fine vertical discretization. The bottom topography is represented using partial cells. COMPAS has a nonlinear free surface and uses mode splitting to separate the two-dimensional (2D) mode from the three-dimensional (3D) mode. The model uses explicit time-stepping throughout except for the vertical diffusion scheme which is implicit.

COMPAS is based on the framework outlined in Herzfeld (2006); this model is a finite difference model that uses an underlying unstructured coordinate system. The generalization of this unstructured system greatly simplifies the transition from finite differences to TRiSK numerics. As such, the horizontal terms in the governing equations (momentum advection, horizontal mixing and Coriolis) are discretised using the TRiSK numerics, whereas the pressure gradient and vertical mixing are discretised using the finite difference approach outlined in Herzfeld (2006). COMPAS therefore inherits much of the functionality of the host model of Herzfeld (2006), including numerous turbulence closure schemes (k-, k-, Mellor-Yamada 2.0 and 2.5) and open boundary infrastructure. COMPAS is available open source at <https://github.com/csiro-coasts/EMS>, with documentation provided at <https://research.csiro.au/cem/software/ems/ems-documentation/> .

Meshes used with COMPAS are required to be a set of polygonal cells which are orthogonal (cell edges must be perpendicular to the line joining adjacent cell centres), centroidal (triangle vertices must be positioned at the centre-of-mass of each polygonal cell) and well-centred (cell vertices must lie in the interior of their associated dual triangles). These polygons are typically constructed from Voronoi diagrams, of which the dual is a Delaunay triangulation. Additionally, meshes must also conform to the geometry of coastal and open boundaries, and adhere to user-defined rules (a weighting function) that specify the mesh resolution. The generation of grids conforming to these requirements is not trivial, requiring the construction of highly optimised Centroidal Voronoi Tessellations (CVT’s) and ‘non-obtuse’ Delaunay triangulations. COMPAS uses the unstructured meshing library JIGSAW (Engwirda, 2017; <https://github.com/dengwirda/jigsaw>) to accomplish these tasks, resulting in high quality meshes that support the requirements of the TRiSK numerics. JIGSAW is incorporated as an ‘inline’ component in the COMPAS framework, facilitating automated domain generation and model configuration. Options applicable to the domain boundary include sub-sampling, smoothing and linking, and options for the mesh-spacing distribution include mapping distance from the coast, bathymetry or any other variable read from file to mesh resolution via a series of piecewise linear, exponential or cosine functions. Various weighting functions are available inline in COMPAS; in this project we use a distance from the coast weighting. Initially the target coastal resolution is 2 km, with the view that this will be increased to 1 km when any emergent model development issues are streamlined.

The coastal weighting is considered the most appropriate if the objective of the national model is to provide system characterization of coastal processes, and provide downscaling forcing for coastal models at the estuarine or embayment scale. It may be necessary to refine resolution over areas not near the coast (e.g. reefs in the GBR). This is possible in COMPAS, as inline weighting also includes distance from a user defined polygon, or hybrid polygonal and coastal distance weighting.

The resolution map of the 2 km model is shown in Fig. 2.1. Selected examples of high resolution regions are illustrated in Fig. 2.2. While this mesh maintains resolution of ~2 km at the coast, it decreases to as much as 25 km at the open boundaries. The resolution mapping used for this mesh is summarized in Table 2.1. The islands in the Timor and Arafura Sea regions naturally produce high resolution at these northern open boundaries (due to coastal proximity), and this must be over-ridden so as coarser resolution is generated to maintain acceptable nesting ratios for boundary forcing.

Diagram

Description automatically generated

Figure 2.1. Resolution of the National Model mesh.

Map

Description automatically generated

Figure 2.2. Bathymetry and mesh for (a) Port Philip Bay region VIC, (b) Storm Bay TAS, (c) Fitzroy Estuary / Keppel Bay QLD and (d) Darwin Harbour NT.

|  |  |  |
| --- | --- | --- |
| Map | Coastal distance (km) | Resolution (km) |
| 1 | 0.2 | 2 |
| 2 | 1 | 2.5 |
| 3 | 150 | 5 |
| 4 | 300 | 25 |

Table 2.1. Piecewise functions mapping distance from the coast to resolution.

Numerous targeted resolution alterations were made to improve model accuracy. These are:

1. Within the GBR reef matrix the resolution was set to 1200 m within the polygon depicted in Fig. 2.2a. Resolution decreased away from this polygon according to the mapping in Table 2.1.
2. Within Storm Bay in SE Tasmania the resolution was set to 1200 m within the polygon depicted in Fig. 2.2b. Resolution decreased away from this polygon according to the mapping in Table 2.1.
3. Within the Fitzroy River region, the Narrows between Curtis Island and the mainland had a resolution of 1000 m. Similarly, the tidal creeks at the mouth of the Fitzroy River were resolved at 1000 m (Fig 2.2c).
4. The area between Hinchinbrook Island and the mainland was resolved at 1200 m (Fig. 2.2c).

These regionalized resolution enhancements were implemented at known areas of interest. Any future versions of the national model would benefit from such targeted enhancements across the entire national coastline.

Diagram, map

Description automatically generated

Figure 2.3. Polygons for resolution enhancement in the GBR.

The model uses bathymetry from the Geosciences Australia (2002) database, with regions outside this extent filled using the dbdb2 (Naval Research Laboratory Digital Bathymetry Data Base <https://www7320.nrlssc.navy.mil/DBDB2_WWW/>) global database. These are supplemented with the Beaman (2010) 100 m dataset within the Great Barrier Reef, and a 30 m northern Australia dataset (<https://ecat.ga.gov.au/geonetwork/srv/eng/catalog.search#/metadata/121620>). The GBR and NW Australia bathymetry datasets were interpolated onto the mesh using a bi-linear interpolation, and the remainder of the mesh was interpolated using the combined GA and dbdb2 dataset using a natural neighbours non-Sibson method. This allowed for accurate bathymetric values in areas requiring extrapolation (e.g. where 4 data points did not surround a model mesh point as required by bi-linear interpolation). Examples of bathymetry and the mesh are shown in Fig. 2.2.

Minimum depth was 2 m throughout the domain, with exceptions as follows. North west Australia had a minimum depth of 4 m (Fig 2.4a), with Bonaparte Gulf (Fig 2.4d) having 8 m and the Kimberley region (Fig 2.4f) 12 m. The GBR region (Fig2.4b) used a 4 m minimum, as did St. Vincent’s Gulf (Fig 2.4c) and around Phillip Island (Fig 2.4e).

Additionally, the area of the Kimberley depicted in Fig 2.5a was prescribed a depth of 12 m, which extended the minimum depth requirement of Fig 2.4f. A channel of 50 m was inserted within King Sound (Fig 2.5b), 25 m in the passages near Greville Island in the Kimberley (Fig 2.5c) and 15 m in the passages around Storr Island (Fig 2.5d), also in the Kimberley. Finally, the channel into Port Phillip Bay (Fig 2.5e) was deepened to 10 m.

Finally, the North West Shelf region, then the Kimberley region were smoothed with an equally weighted convolution filter using 3 passes, and the entire bathymetry was median filtered to remove spurious ‘holes’.

Diagram, engineering drawing

Description automatically generated

Figure 2.4. Regionalized minimum depth.

Map

Description automatically generated

Figure 2.5. Localized bathymetry deepening.

The model is nested within the global OFAM model (Oke et al., 2008) which supplies currents, sea level, temperature and salinity on the open boundaries. Low frequency sea-level from the global model is superimposed with 8 tidal constituents (M2 S2 N2 K2 K1 O1 P1 Q1) from the TPXO9v1 1/6° global model (Egbert and Erofeeva, 2002; <http://volkov.oce.orst.edu/tides/otps.html>) and applied at the boundary of the regional model. The boundary condition used is that of Herzfeld and Andrewartha (2012), modified for unstructured meshes. Surface fluxes are derived from the Australian Bureau of Meteorology’s operational atmospheric models (ACCESS-A; Puri et. al., 2012, <http://www.bom.gov.au/nwp/doc/access/NWPData.shtml>) at a resolution of 12 km. Heat fluxes applied at the surface boundary condition were computed from standard meteorological variables provided by ACCESS-A (wet and dry bulb temperature, air pressure, wind speed and cloud amount) using short and longwave calculations outlined in Zillman (1972) and the bulk method for sensible and latent heat using bulk coefficients of Kondo (1975). For the surface freshwater fluxes, precipitation was provided by ACCESS-A and evaporation was computed from the latent heat flux. Wind speed was converted to stress using the bulk scheme of Large and Pond (1981).

The model uses 53 layers in the vertical with 1 m surface resolution, the k- turbulence closure model of Burchard et al, (1998) and an unstructured implementation of the Van Leer (1979) advection scheme for tracers. Horizontal mixing was scaled to the mesh using the edge area (edge length x distance between centres; see Section 2.2.6), with the optimum mixing (m2s-1) given by (edge area) / (100Dt), where Dt is the 3D time-step. A regionalized viscosity and diffusivity was applied, with generally more mixing supplied in the energetic regions of north west and east Australia. The regionalization follows the partitioning depicted in Fig. 2.6, with values for each region listed in Table 2.2. Note that the base rate reported is a percentage of the optimum mixing defined above. A bi-harmonic scheme was used for horizontal viscosity. The relationship between Laplacian viscosity (L, m2s-1) and biharmonic (B, m4s-1) is (Griffies and Hallberg, 2000) where here we use the edge length squared for 2.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Horizontal Viscosity | | | | | | | | | | |
| Region | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Base (%) | 50 | 10 | 70 | 10 | 50 | 10 | 10 | 10 | 100 | 100 |
| Smagorinsky |  |  |  |  | 0.1 |  |  |  | 0.1 |  |
| Horizontal Diffusion | | | | | | | | | | |
| Region | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Base (%) | 20 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 50 | 50 |
| Smagorinsky |  |  |  |  |  |  |  |  |  |  |

Table 2.2. Regionalized horizontal mixing parameters. Base rate is a percentage of the grid optimized value, and Smagorinsky is the empirical constant if the Smagorinsky scheme is invoked.

The time-step is 8 s which delivers a runtime of 2.8:1 on 72 processors. While this runtime is acceptable for near real-time operations, it is clearly too slow for extended hindcasts and massive distributed processing is required to increase runtime to acceptable levels. This requires development to improve MPI or OpenMP master-slave parallel processing into a hybrid MPI-OpenMP slave-slave approach (see below).

The model is currently running in near real-time, with an archive created dating to September 2020. Selected outputs are presented in Section 3.

Diagram, radar chart

Description automatically generated

Figure 2.6. Partitioning for regionalized mixing

## Model enhancements

As COMPAS is a new, relatively untested model, it was necessary to undertake model development in numerous areas in order to optimally operate the national model. A parallel project, the AUSTEN ‘Tidal Energy in Australia – Assessing resource and feasibility for Australia’s future energy mix’ or ARENA project, uses COMPAS in a similar capacity to generate tidal flows at the national scale. This model is only run in 2D mode with tidal forcing only, and uses a mesh with weighting which highly resolves areas with strong tidal currents. However, despite these differences, many of the challenges faced in modelling at the national scale in the ARENA project are also faced within this project, and COMPAS development in both projects complement and leverage off each other. The collective enhancements to COMPAS undertaken in these projects are summarized below.

### Meshing

JIGSAW has undergone improvement as the project has progressed. The author of JIGSAW (Darren Engwirda) has closely collaborated on the project to integrate JIGSAW into COMPAS, and has currently visited the CSIRO laboratories for the past three years to assist with JIGSAW implementation. This has resulted in the provision of JIGSAW as a stand-alone library which has been integrated into COMPAS so that meshes can be seamlessly generated inline. Enhancements to JIGSAW include the generation of stereographic meshes (i.e. meshes built on a sphere rather than a plane) and ‘power meshes’ which provide better mesh uniformity than regular meshes. Darren has also assisted in the development of an unstructured Flux Form Semi-Lagrange advection scheme (see below).

### Weighting options

Numerous inline weighting functions are available from a parameter file specification. Essentially a weighting variable must be defined (or created) which is then mapped to resolution via a set of piecewise functions; i.e. start and end points of the weighting variable are mapped to start and end resolutions, and weighting variables within the start and end points are mapped to resolution via a specified function. These functions may be a linear map which creates a uniform change in resolution with the weighting variable, an exponential function where resolution changes more rapidly at the start point of the weighting function or a cosine function where resolution changes rapidly at the mid-point of start and end weightings. The weighting variable may be bathymetry, or any snapshot of variable in a netCDF file (e.g. a salinity plume) or distance from a point, polygon or the coast. These latter distance weighting variables may be combined such that mapping to resolution use the minimum distance from a point, polygon or coastline. Resolutions or weightings may be over-ridden at specific locations in order to ‘coarsen’ resolution along open boundaries such that nesting ratios are satisfied.

### Bathymetry options

COMPAS now contains more sophisticated bathymetry interpolation options than its structured counterpart, SHOC. These include interpolation methods using sparse (non-gridded) datasets, which is superior for mesh points that require extrapolation rather than interpolation. Gridded interpolations generally require an interpolation point to be surrounded by bathymetry data, and perform poorly for extrapolation. Multiple bathymetry datasets may be supplied, and COMPAS sequentially fills the domain for cells lying within the bounds of each supplied dataset, using the last dataset to extrapolate onto any unfilled cells.

Bathymetry may be modified locally by selecting cells whose distance is minimum to a supplied transect, or lies within a user defined polygon. Selective bathymetry smoothing may be performed in a polygon. The depth at the coast may be regionalized using supplied polygon files. Median filtering of bathymetry has been included. Individual cells may be designated as land and omitted from the mesh.

### 2.2.4 Parallel partitioning (windows)

The specification of cells in an unstructured grid containing non-quadrilateral cells is essentially random, and there are no consistent (i,j) directions that exist in structured models. This makes traditional partitioning approaches of blocking or striping generally unsuccessful using COMPAS, and results in overly fragmented windows. To overcome this, we have implemented several alternative partitioning methods in COMPAS.

1. ‘Grouped’ cell partitions are available where partitions consist of contiguous outward expansion of a given point. These are contiguous in the sense that cells always have at least one neighbour in the same window, however, intersection of the expansion with land may fragment some windows.
2. The user may manually define windows using box model region specifications (created with offline software), and submit the windows via a netCDF ‘region’ file. The PLUM software has been modified to allow ‘region’ files to be created and saved to netCDF for unstructured meshes.
3. There exists specialist public domain software that creates optimum windows by minimizing edge lengths of adjoining windows (which minimizes information transfer, hence increases speed). We have included the METIS software (<http://glaros.dtc.umn.edu/gkhome/metis/metis/overview>) as a library function which COMPAS can call inline to generate the window partitions. Using this specialist software is the preferred, default option.

### 2.2.5 Output

There exists a Climate Forecast (CF) compliant netCDF standard for unstructured models, UGRID 1.0 (<http://ugrid-conventions.github.io/ugrid-conventions/> ), which may be visualized using GODIVA 3 if data is hosted on OpenDAP. COMPAS will output this format if specified, along with structured CF netCDF files if run on structured (quad) grids. Furthermore, output of scalars or cell centered east and north velocity may be output at any arbitrary location in the domain using ‘point array’ netCDF files. For unstructured meshes, this requires interpolation using sparse (non-gridded) data, which is a significant departure from the gridded interpolation used in structured models. Such point output is necessary for information exchange in 2-way nested applications.

### 2.2.6 Horizontal viscosity

Bi-harmonic horizontal viscosity has been included in COMPAS. The relationship between Laplacian viscosity (L, m2s-1) and biharmonic (B, m4s-1) is where here we use the edge length squared for . Biharmonic viscositity can be specified explicitly, or can be input as a Laplacian value which is scaled inline according to the above scaling. Bi-harmonic viscoity is popular in global models, as it selectively dissipates short waves hence can remove unwanted short-wave perturbations while leaving the larger mesoscale processes undissipated.

Horizontal viscosity is input into the model as a single number that relates to a mean cell property (cell area, edge area, edge length, distance between centres) and is scaled inline to each individual cell. The scaling method can impact model performance, since velocities are computed on cell edges, the viscosity schould be also scaled onto an edge. However, as edge length can vary rapidly in resolution transition regions, this can lead to large changes in viscosity, which can be problematic for model stability. Numerous scaling strategies have been included in COMPAS, and we find an edge area based scaling provides a good balance between a smoothly varying viscosity field while still being representative of the edge dimentions the velocity is computed on.

Since the horizontal viscosity is edge centered and therefore difficult to visualize, a cell centered average viscosity diagnostic has been produced.

### 2.2.7 Tide

The TPXO tidal forcing dataset has been integrated into COMPAS, allowing open boundary forcing with tidal elevation or velocity. This uses a netCDF database for tidal amplitude and phase of constituents, which may be updated as better products become available. Either tidal elevation or tidal velocity (or both) may be prescribed on the open boundaries. The tide as predicted by the TPXO database can also be computed at every model cell to provide a tidal height diagnostic. The model sea surface may be relaxed to this diagnostic if required. An equilibrium tidal forcing function is also included.

### 2.2.8 Other enhancements

The code has been arranged so that dedicated momentum tendency arrays are used to update velocity. This not only makes it easier to isolate particular processes in the model (for diagnosis or exclusion) but can also be used for low level parallelisation where mutually exclusive tendencies can be computed simultaneously. The potential vorticity flux used in the nonlinear Coriolis term takes a default implementation of the energetically-neutral form (Eq. 49 Ringler et al., 2010), but the potential enstrophy conserving (Eq. 71 Ringler et al., 2010) or enstrophy dissipating (Eq. 73 Ringler et al., 2010) or ‘Anticipated Potental Vorticity Method’ (Eq. 81 Ringler et al., 2010) have also been implemented for momentum advection. The latter approach appears to better resolve frontal structure. Fatal model instabilities now produce a ‘site’ file that identifies the location of the instability and may be visualized. Regionalization of bottom friction using polygons has been added, and the Manning bottom drag formulation (<http://www.marinespecies.org/introduced/wiki/Bed_roughness_and_friction_factors_in_estuaries>) has been included. Nudging zones for temperature and salinity adjacent to open boundaries and been included, and sponge zones for open boundaries has been improved so as to be specified as a distance (km) from the open boundary rather than some number of cells into the interior. Cyclic open boundaries have been implemented for regular (quad or hex) unstructured meshes. COMPAS can import SST products from their THREDDS servers directly and interpolate into the mesh. This can be from GHRSST or BoM servers; we find the L4 products from GHRSST most useful (e.g. <https://data.nodc.noaa.gov/thredds/dodsC/ghrsst/L4/GLOB/UKMO/OSTIA/> ). If this data is imported, then RMSE against model surface temperature may be produced inline, or a surface heatflux may be specified by relaxing the surface temperature to SST.

The size of the national model mesh results in extremely slow generation of the mapping functions required for the model to operate in unstructured coordinates (up to 1 week processing time). These functions were profiled and optimised to increase efficiency. Specifically, there were numerous routines that required n2 operations (n is the mesh size), e.g. removal of duplicate nodes, which results in ~1010 operations, significantly slowing the model. These routines were re-structured using the quicksort algorithm to result in n.lon(n) operations (~106) which greatly increased efficiency.

# Outputs

The model has successfully run 16 days from 24 Feb to 12 Mar 2017, and this configuration is currently operating in near real-time beginning Sep 2020 on HPSC infrastructure, using 72 processors which delivers a runtime ratio of 2.85:1. Surface temperature with 3D currents and surface salinity with 2D currents are shown in Fig. 3.1 for Mar 2017. The distributions show no spurious behaviour, e.g. checkerboard patterns indicating instability or unrealistically large currents.

Map

Description automatically generated

Figure 3.1. (a) Surface temperature and currents and (b) surface salinity and depth averaged flow for 12 Mar. 2017.

The model is compared with the eReefs GBR4 model for 5 Mar 2017 in Fig. 3.2 and 3.3 (note the different time zones). The broadscale patterns of flow in Fig. 3.2 are comparable, however, the GBR4 model exhibits a stronger North Queensland Current and more structure offshore in the Coral Sea. This is primarily due to the higher resolution (~4 km) over the whole domain in GBR4, whereas the National Model only has higher resolution within the coastal band and is resolved at ~25 km over much of the Coral Sea. The parameter configurations differ between the two models also; notably GBR4 is relaxed to global model density field in deep water to preserve currents driven by dynamic height gradients, whereas this relaxation is currently absent in the National Model. It is these parameter optimisations that require much attention during a calibration phase of the National Model.

The GBR4 flow is more chaotic at the NSW border, with the presence of an eddy in the SE corner, whereas the EAC off Fraser Island is stronger in the National Model. Some boundary specification error likely contributes to GBR4 in this region, since the EAC exits the domain through the southern cross-shelf boundary and is also associated with a mesoscale eddy field. It is difficult to perfectly specify boundary forcing data in these energetic regions, and some over-specification error can result. Being of national scope, the National Model does not need to prescribe these cross shelf open boundaries, and this area demonstrates a clear benefit of modelling the entire shelves continuously. GBR4 must also use an open boundary across Torres St (which again is absent in the National Model), and to a lesser degree we see a deviation in flow patterns in this area between the two models.

Fig. 3.3 compares sea level, salinity and temperature between the two models. Sea level is again comparable, with an ebb tide seen in Broad Sound in both models. The peak tide is weaker in this region in the National Model, also the background sea levels are offset at the offshore open boundary. This indicates some tuning of the tidal forcing is required in the National Model. Differences at the southern and Torres St open boundaries are also observed, with lower sea level associated with the eddy near the SE boundary in GBR4 and weaker ebb tide through Torres St.

Salinity is broadly consistent between the two models, however, the coast is fresher in some regions in GBR4 due to freshwater river inputs which are yet absent in the National Model. This is most notable at the mouth of the Fly River, where low salinity in the National Model is retained from the initial condition. More structure is again seen offshore in GBR4 due to its higher resolution there. This structure is even more evident in the temperature solution, however, that aside, the distributions are again generally consistent. The SWR parameterisation in GBR4 was prescribed using a data assimilation parameter estimation technique, which provided an accurate spatially variable parameter field. In the National Model constant default values are currently used, and this is an area that will require parameter optimisation, preferably using similar techniques to eReefs.

Although resolution is less offshore in the National Model, leading to less meso and sub-mesoscale structure, the opposite is true at the coast and over the reef. Fig. 3.4 shows surface temperature and currents over the reef in an area off the Whitsundays (Fig. 3.4a). GBR4 shows a cooling of water over the reef at this time, and a generally unimpeded southward flow over the reef. The National Model also shows this cooling, with sharper gradients and more structure, but also the flow over the reef shows more variability and is steered by the topography to a greater extent. This is due to more realistic bathymetry resolving of reef flats and channels in the National Model; a result of the 1 km resolution in this region.

Map

Description automatically generated

Figure 3.2. Surface currents for the national model (top) and eReefs GBR4 (bottom) for 12 Mar. 2017. Note the different times due to time-zones used. Left column is eastward current, middle is northward current and right are the flow vectors.

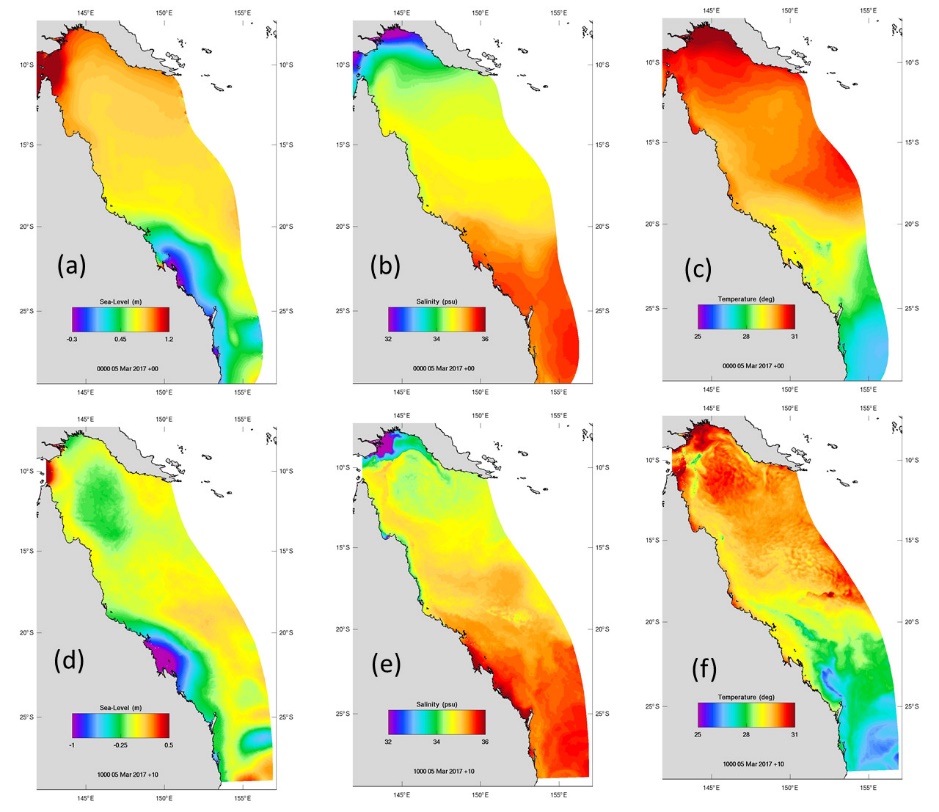


Figure 3.3. Sea level and tracers for the national model (top) and eReefs GBR4 (bottom) for 12 Mar. 2017. Note the different times due to time-zones used. Left column is sea level, middle is salinity and right is temperature.

Chart

Description automatically generated

Figure 3.4. Surface flow and temperature over the reef off the Whitsundays on 28 Feb. 207. (a) Bathymetry, (b) eReefs and (c) national model.

When persistent south-east winds are present along the Bonney Coast in South Australia, surface offshore Ekman transport drives upwelling at the coast, where the thermocline breaches the surface to result in cold water with an associated strong longshore flow. This is observed in early March 2017 (Fig 3.5a), and is successfully replicated by the model (Fig 3.5b). Upwelling extent and magnitude are comparable to observation, as is the SST within the South Australian gulfs.

Map

Description automatically generated

Figure 3.5. Surface temperature and currents in the Bonney Coast region demonstrating upwelling (a) IMOS OceanCurrent image and (b) National Model.

The National Model generally replicates the major features of the GBR4 snapshot and dynamics of SE upwelling, and appears to be behaving adequately in its pilot form. Furthermore, some benefits of the model, e.g. lack of cross-shelf boundaries and increased resolution inshore, are apparent. However, the need for further parameter calibration to observations is also apparent.

# Distributed processing

The 2 km National model grid comprises of 350,000 surface cells and runs at approx. 3:1 on 48 cores/windows on NCI’s latest (2020) super-computer, *Gadi*. Presently, only OpenMP (shared memory) is supported in COMPAS, which severely limits its performance.

The goal of this project is to address scalability in COMPAS by first implementing a distributed memory version, and then to investigate overall efficiency in order to minimise model run-times as much as possible.

## 4.1 Progress summary (Jan. 2020)

* Overall distributed memory code design, via MPI, complete
* MPI initialisation phase prototype complete
* MPI sim-loop phase: work-in-progress (estimated finish date: Jun. 2020)
* Fine-grain parallelisation ideas/techniques investigated, including some preliminary proof-of-concept code
* Implementation and testing of these ideas within COMPAS is planned for the latter half of 2020
* Ideas presented at the International Workshop on Multi-Scale (Un)-structured mesh numerical Modelling for Coastal, shelf and global ocean dynamics (IMUM 2019)

###### Update Dec. 2020

* Implementation of distributed init functions 95% complete
  + hd\_init for master and
  + hd\_init\_w for slave windows implemented

## 4.2 Introduction

Coding MPI into COMPAS builds on the previous work done for SHOC. The shared memory parallelisation is already reasonably optimal for both models (via OpenMP), hence the main focus of this work is to address scalability in a distributed memory environment. A secondary, but important, goal is to investigate run-time performance efficiency within each window/process in order to speed-up the overall model runs with a minimum of compute resources. This is especially important in the case of ensemble runs (for DA) as many simultaneous runs are required instead of a single massively parallel run. This is both a challenge, as the CPU speeds have plateaued in recent years, and an opportunity if done well, to be one of the leading performers in the numerical modelling community.

## 4.3 Distributed vs shared memory

A distributed memory system involves inter-process, as opposed to intra-process for shared memory, communication. Unlike threads within a process in which they all have access to the global memory of that process, each process has a completely independent memory map and therefore any communication between them is required to be explicit.

Implementation of the Message Passing Interface (MPI) therefore involves adding extra code to handle the various sending and receiving of data between processes. Each “window” is now run in its own process but the general principal of running the windows in parallel follows a similar paradigm for MPI as OpenMP.

## 4.4 Basic workflow (shared memory)

The modelling workflow can be loosely divided into 2 sections (see Fig. 4.1):

1. Init (hd\_init) – the one off initialisation and setup of the model in memory
2. Sim-loop (hd\_step) – the simulation loop that repeats many, many times over the course of the model run

It is the sim-loop that we’re mostly interested in speeding-up, however, the init relates to the memory footprint of processes which needs to be taken into account.

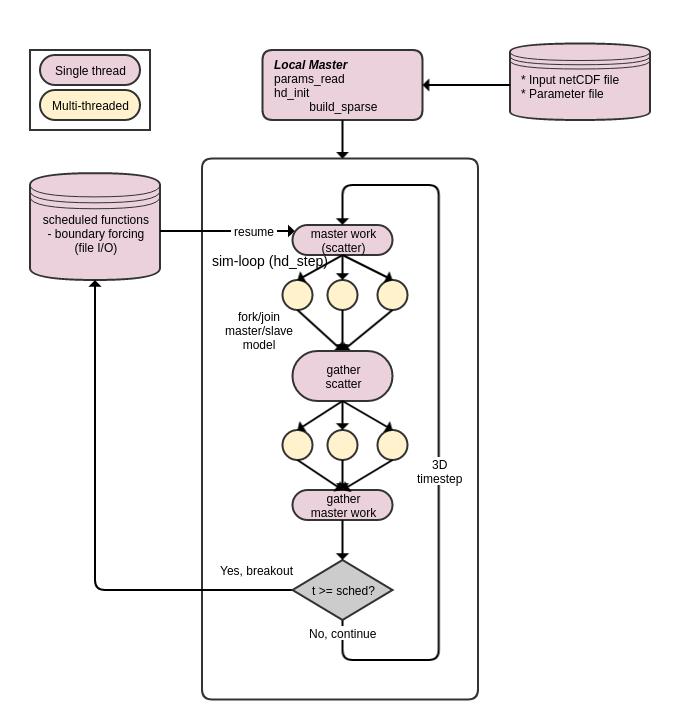


Figure 4.1. Basic modelling workflow (SHOC and COMPAS)

### 4.4.1 Background : MPI implementation in SHOC

The initial MPI work in SHOC implemented the Send/Receive calls in the sim-loop but with one major modification; the master-slave data transfers were turned into direct slave-slave transfers, considerably improving performance, see Fig. 4.2 below. Note: for file outputs, the old *window\_empty\_to\_master* function remained the same.

Unlike OpenMP, where there is only one process, in MPI there are as many processes as windows. To minimise code changes, each process replicated the complete initialisation stage for the entire spatial domain (essentially having a local master on each process). This allowed every process to initialise in parallel at the same time and the scheduled functions to work out-of-the-box. Each process then only copied the data for its own window to run and performed calculations for only the window it was responsible for. Using CSIRO’s *pearcey* supercomputer (20 cores/node), we were now able to utilise multiple nodes with runtimes shown in Fig. 4.3.

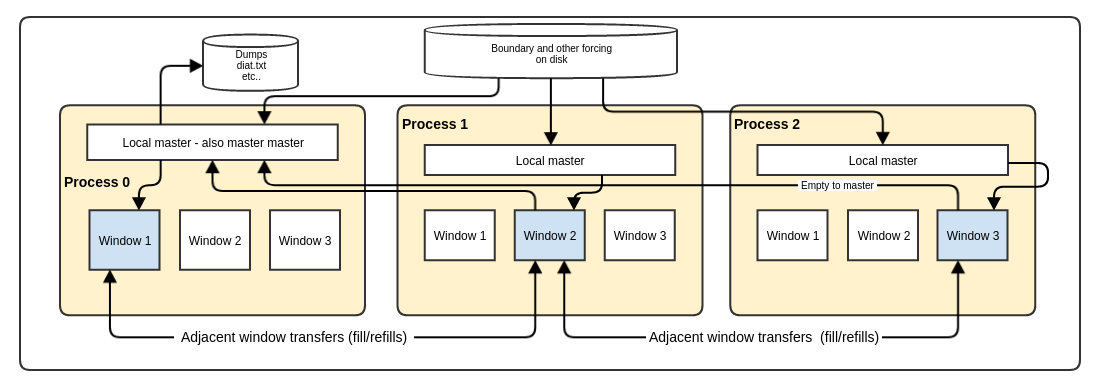


Figure 4.2. MPI implementation in SHOC

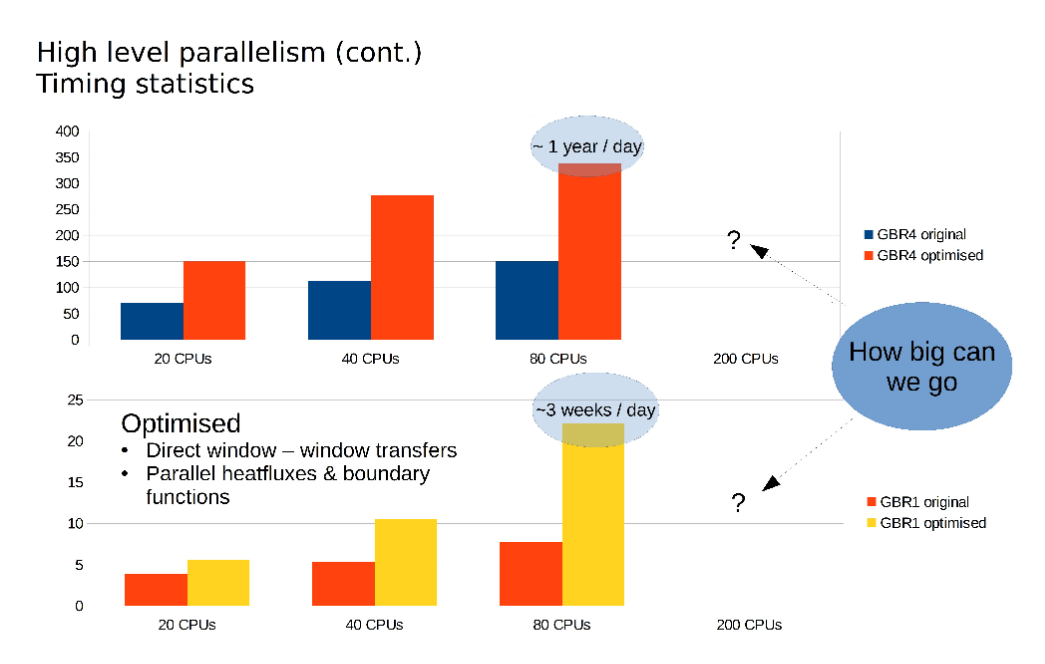


Figure 4.3. Run-time ratios of the GBR models in SHOC

For large models (exactly the ones we’re trying to optimise) the memory footprint of each process was so big that only 2 processes/windows of the GBR1 model (~60GB per process) could fit on each node on *pearcey*, therefore significantly under-utilising the compute cluster. Even on the large memory nodes (512 GB) only 8 out of the available 20 cores could be utilised.

## 4.5 Design consideration for COMPAS

In order to overcome the memory limitation of the SHOC version, we need to either:

1. Significantly reduce the memory footprint of each process
2. Increase the utilisation of the other cores within each MPI process using OpenMP/threads

In addition, look at:

* Upgrading the MPI transfer schemes to use the more optimised; One-sided communication/Remote Memory Access (RMA) – new in MPI v3.0
* Making use of the very latest in computer architectures - efficient use of cache lines, wide AVX registers (SIMD) and GPU’s
* Incorporate all file I/O to be window-based

### 4.5.1 Design #1

In order to address limitation 1 above, we can consider each of the yellow threads in Fig. 4.1 to be an MPI slave process – combining with Fig. 4.2 we get Fig. 4.4.

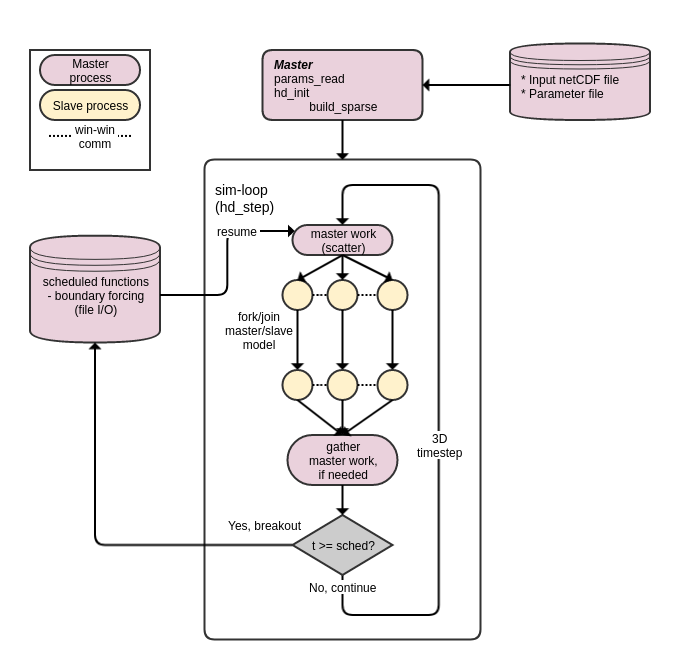


Figure 4: COMPAS MPI design #1, slave process + window-window transfers

Figure 4.4. COMPAS MPI design #1, slave process + window-window transfers

The benefits of this design approach are:

* Similar programming model to OpenMP
* Each slave process now only requires memory for each ***window***

Separate ***master*** process whose job is only file I/O operations (no window hd\_step)

Extensions to this design may include:

* Fold file I/O operations into each slave process so that data interpolation may be performed in parallel (see File I/O section below)
* Take advantage other parallelisation’s (see Fine grained parallelisation section below)

### 4.5.2 Process memory layout

Fig 4.5 illustrates the memory distribution of the MPI processes. The slave processes only allocate memory for their respective window data structures. The master process (always rank 0) only allocates and initialises memory for the master and global geom data structures, the window information is transferred to the slave via MPI during the **hd\_init/hd\_init\_w** phase

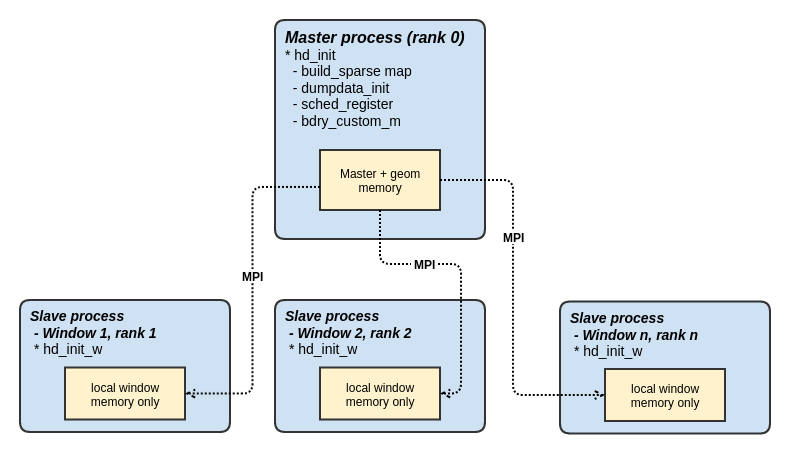


Figure 4.5. COMPAS MPI memory layout

This makes the memory footprint of the entire application as efficient as possible with no redundant memory allocation anywhere. The memory per slave *decreases* as the number of the windows *increase*

One significant advantage of this scheme is that it allows the master to perform boundary read operations in parallel to model computations, ready for parallel distribution at the beginning of the next timestep.

## 4.6 File I/O

One key aspect of the model run missing from the basic 2 stage init./sim-loop workflow description is the file read of boundary forcing data. In EMS, these have been coded in as work done on the master, as low-level file operations are not threadsafe. When transitioned to a multi-process environment, there is not a limitation in using parallel reads, however, there can be some performance degradation when the same file is read from a large number (>100) processes on a cluster (due to operations being potentially serialised by the underlying file system).

A simple way to get around this problem is to use any number of the parallel file I/O libraries, e.g. parallel netCDF/MPI-IO or develop a more sophisticated dedicated file I/O server that allows for double-buffering.

## 4.7 Window partitioning

Spatial partitioning of anything other than “simple” domains is not trivial. The complex curvilinear grids for SHOC have many window edge configurations that together with horizontal LAND/open boundary and bathymetric steps make it extremely difficult to guarantee exact auxiliary cell specifications. Unstructured grids make this even more challenging because of non-uniform number sided polygons. Fortunately, this is a topic of much research in the wider modelling community and as such we’ve incorporated the METIS software for grid partitioning into COMPAS; <http://glaros.dtc.umn.edu/gkhome/metis/metis/overview>. This is quite a sophisticated tool, and is one that is used by the MPAS team, but the 2 main features of interest for us are:

1. Minimising edge lengths; this translates into minimising the data communication between windows.
2. Generation of contiguous windows; this significantly, if not completely, eliminates non-well defined edge boundaries.

## 4.8 Fine grained parallelisation

Computers are getting faster, yet the clock-speed of CPU’s has plateaued for over a decade. This is largely due to the shift towards data parallel paradigm, i.e. more cores, more cache and wider registers to boost SIMD (Single Instruction Multiple Data) performance. GPU’s take this to the extreme, however, unlike their CPU counter parts, require considerable re-write of the code.

Addressing scalability has traditionally been the primary method of choice to increase overall model run-time ratios. However, the advances in recent computer architectures means that these models are not capitalising on the power of recent CPU’s that is essentially available for free. Increasing efficiency per node obviously makes for better overall performance (and reduces cost) but more importantly, allows for more model runs in parallel in the case of ensemble runs.

It is the goal of this project to take advantage of the SIMD paradigm wherever possible.

## 4.9 Next steps

1. Complete MPI implementation so that the COMPAS National model is capable of scaling across as many nodes as needed to yield a reasonable run-time ratio.
2. Investigate, prototype and document SIMD and optimised file I/O concepts.

## 4.10 Online resources

* Original SHOC MPI confluence page : <https://confluence.csiro.au/pages/viewpage.action?pageId=202440791>
* COMPAS Parallelisation : <https://confluence.csiro.au/display/cem/Parallelisation+-+COMPAS?src=contextnavpagetreemode>
* COMPAS MPI code repository : https://bitbucket.csiro.au/users/riz008/repos/ems-compas-mpi/browse

Main EMS Parallelisation: <https://confluence.csiro.au/display/cem/Parallelisation?src=contextnavpagetreemode>

# Visualization – MoVE (Model Visual Explorer)

MoVE is an online application for visualising and exploring both structured and unstructured model output.  It is intended to be a replacement for existing applications such as PLUM and DIVE. MoVE forms part of the broader [DigiCOAST Strategy](https://confluence.csiro.au/display/cem/DigiCOAST?src=contextnavpagetreemode) and utilises the latest technologies currently available in Python and JavaScript to deliver a highly configurable set of components.  It is accessible from any browser without requiring any client-side installation of supporting software.

A number of use cases for MoVE have been identified.  These include:

1. An out-of-the-box fully featured application that permits spatio-temporal inspection and selection of model variables. This application can simultaneously display multiple variable layers from different model runs and grid structures with user configurable levels of transparency and colourisation.

2. A set of high-level components that can be easily arranged with minimal coding to create domain specific applications. Use case (2) is in fact nothing more than a specific example of this.

3. A set of high-level components that can be easily imported and accessed in other environments such as Jupyter Notebooks. This will provide the user with an easy to use and consistent mechanism for interrogating and visualising model output as part of a wider data analysis workflow. By utilising the same set of MoVE components, model data storage structures and grid structure discrepancies are encapsulated thereby providing a single interface to all underlying data sources.

## 5.1 Component Architecture

The MoVE system is organised as a set of de-coupled software components that together provide a Model-View-Controller system for model inspection and visualisation. Key to its flexibility is the broadcast signalling mechanism provided by the [Blinker](https://pythonhosted.org/blinker/) library. The signalling mechanism means that simply creating an instance of a component is all that is required to have that component interact with other components in the system. No hard-coded links are required for component events and data transfer to occur. Access to individual components is via simple python imports within the top level move package name space.

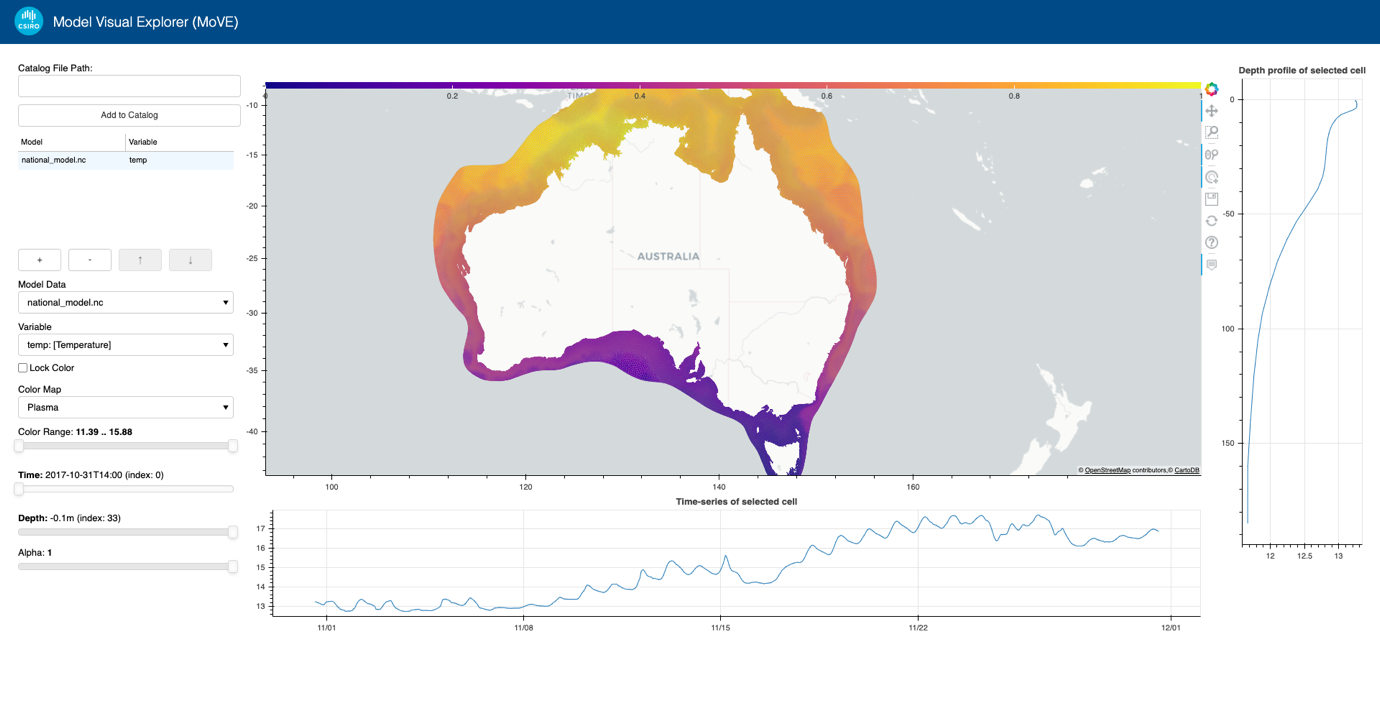
Client side GUI development and synchronisation with the server is achieved through the use of [Bokeh](https://bokeh.org/). Bokeh provides a system of powerful and flexible charting components, user interface widgets and client server synchronisation while allowing almost all code to be written with the Python language.

The core components are:

* DataModel (move.model): A generic interface for accessing gridded model data. Xarray is used to load NetCDF with the intention of incorporating Dask in the near future.
* DataCatalog (move.model.catalog): A mechanism for high level management of multiple data sources. Stores links to the location of data sets. Maintains a list of available data sources. Manages the creation and supply of DataModel objects.
* DataCache (move.model.cache): A utility class for caching to disk or memory, frequently used, expensive operations such as the generation of unstructured grids.
* Signals (move.signals): A global set of signals that provide the mechanism for communication of events and data between the various components in the system.
* Signal Logging (move.signals.logging): A utility class for tracking the flow of signals in the system. This is primarily for debugging purposes and can be switched off when not needed.
* Settings (move.settings): Client and server level settings storage.
* Controller (move.controller): The ‘business logic’ part of the system. The controller is the link between the server-based data model and the client based GUI view; e.g. the client presses a button in the view to load a dataset. This triggers a signal which is received by the controller. The controller makes the decision to send a signal requesting data. This signal is received by the DataCatalog which loads a new DataModel and sends a signal once loaded. The view receives this signal and updates the list on the screen.
* View (move.view): The part of the system that is presented on the screen. The view represents the current state of the system. It responds to and emits signals but all logic and calculations are handled by the controller. There are many advantages to this approach including the ability to connect multiple views to one controller e.g. a table view, a 2D view and a 3D view. Since they all receive the same signals from the same controller, changes to one view will automatically be reflected in the other views with requiring any extra coding.
* Selection (move.view.selection): A set of selection tools that work consistently across different model grid structures.
* Jupyter (move.jupyter): This a convenience package that provides a single entry point to the MoVE system when using components within external applications such as Jupyter Notebooks. For example,

from move.jupyter import \*

* Application (move.application): This is the default application for interogating and displaying model data. It is included as part of the MoVE system allowing a user to install and use without any programming. It also serves as an example of how MoVE components can be assembled into a complete application.

Figure 5.1. Model Visual Explorer (MoVE) web interface.

## 5.2 ParaView

Developing the ability to visualize large scale model output often results in complex code. However, there exist commercial off-the-shelf packages that could be expanded to fit our visualisation needs. ParaView (<https://www.paraview.org/> ) is such an open source package that is popular in the modelling community, primarily for structured grids, however, a plugin exists for UGRID compatible netCDF on triangular meshes. No plugin exists for hexagonal meshes, and we aim to develop such an application here. An initial evaluation of ParaView on a structured SHOC rectilinear grid (Fig. 5.2) shows promise, but also highlights the steep learning curve of the package. For the national mesh, test scripts have been produced that allow the COMPAS unstructured mesh to be converted to VTK format for loading into ParaView (Fig. 5.3). Preliminary experiments have demonstrated the potential for using this visualisation package with COMPAS output; our next task is to write a plugin to load COMPAS output directly into ParaView.

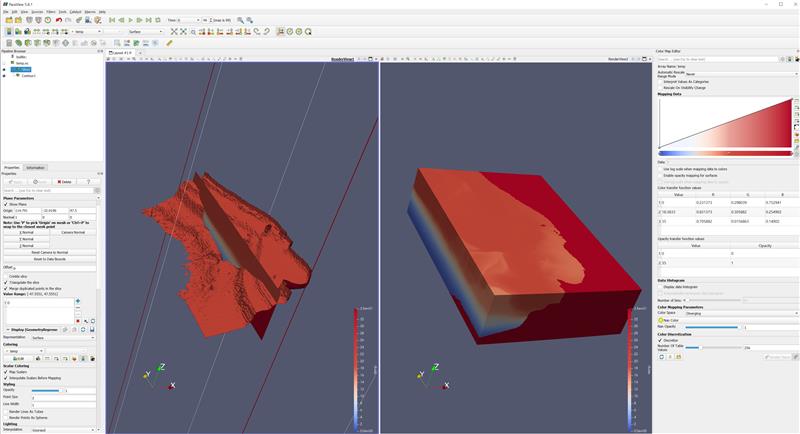


Figure 5.2. Rectilinear grid SHOC output in ParaView.

Graphical user interface, application

Description automatically generated

Figure 5.3 National model grid in ParaView. (a) Whole mesh and (b) Port Lincoln zoom

# Transport model

The transport model uses offline velocity fields to transport (advect and diffuse) tracers throughout the domain. Using semi-Lagrangian approaches for advection it is possible to implement unconditionally stable (or not bound by restrictive time-step constraints) schemes that allow for long time-steps. This is attractive for simulation of biogeochemistry or sediment transport, where the number of tracers can be large and computational costs high leading to slow runtimes. An unconditionally stable scheme can boost runtime in these circumstances by orders of magnitude. While the structured model has a flux-form semi-Lagrange transport scheme (Gillibrand and Herzfeld, 2015), the implementation of such a scheme in unstructured models is far more complicated, and currently one does not exist. It is, however, necessary in order to simulate BGC efficiently, hence we aim to develop such a scheme which is detailed below.

The layout of a generic (hexagonal) mesh is illustrated in Fig. 6.1, where red dots denote the cell centre and triangles the edges. We require to track a streamline back from each edge and compute the streamline integrated tracer value. This value is then multiplied by the volume flux through that edge to obtain the mass flux. A new tracer value at the next time-step is determined via the mass flux divergence. This is the essence of the Flux Form Semi-Lagrange (FFSL) method described by Leonard et. al., (1996) and Lin and Rood (1996).

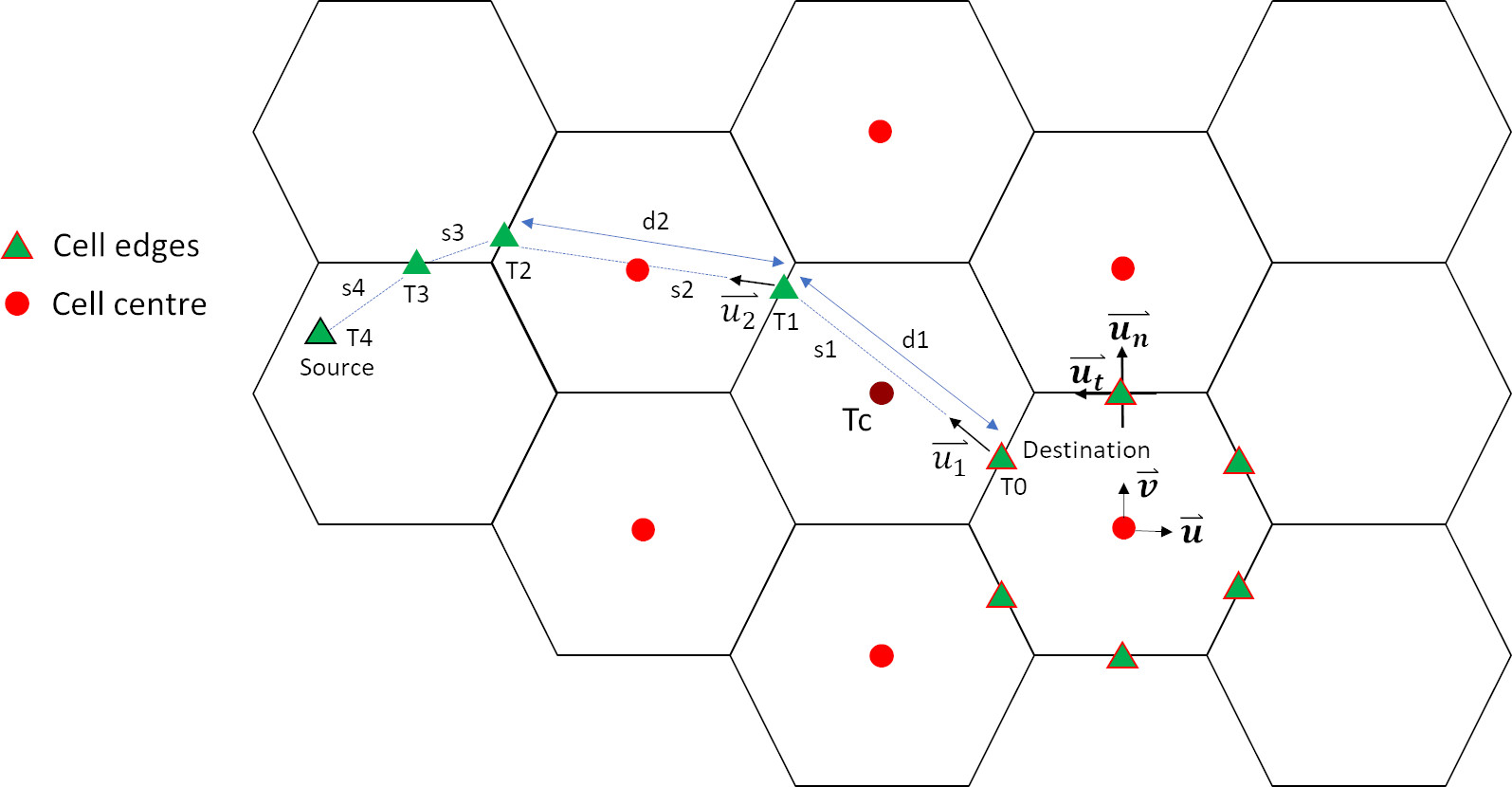


Figure 6.1. Geometry of streamline trajectory.

The streamline from edges is represented as a combination of piecewise linear segments, where each segment is the distance travelled during a sub time-step , and the sum of these sub-steps is equal to the model time-step . The sub time-step is computed so that the streamline traverses a mesh cell exactly in one step. Given the velocities at the location corresponding to the start of the segment (), the distance (e.g. for the first segment, ) is given by:

The velocities generally need to be interpolated onto the location of the start of the segment. We do this using cell centered east and north velocity components (u, v), interpolated using a quadratic least squares scheme. The direction of the segment is given by:

where and are east and north velocity components respectively at the segment start. The tracer value at the end of the segment (e.g. T1 for the 1st segment) is also interpolated using the least squares method (currently linear). The cell centred tracer values used in this least squares interpolation includes the upwind cell centre corresponding to the edge and one ‘ring’ of centres surrounding this centre (e.g. the red dots for the location T1). The segment integrated value is given by the integral of the linear segment using tracer start and end tracer values, e.g.

The next segment (and segment integrated tracer value) is computed using a new sub time-step and this process is repeated until the sum of sub time-steps equals the model time-step and the end of the streamline (the source location, e.g. T4) is located. The streamline integrated tracer value to be applied at the destination edge (T0) is then the sum of segment integrated values divided by the sum of distances of the segments. The same procedure can be performed in the vertical to get vertical fluxes.

This approach follows that of Gillibrand and Herzfeld (2016), which is an adaptation of the Flux Form Semi-Lagrange scheme originally proposed by Leonard et al, (1996) and Lin and Rood (1996). The essence of these schemes is that transverse terms of the advection equation must be accounted for, which represents the fact that as a streamline tracks back in one axis direction, it encounters tracer values due to advection in other axis directions in a multi-dimensional problem. Leonard et al, (1996) accounted for this by effectively solving the advection problem twice – once using advective form to obtain the transverse solution and again using the flux form with transverse solutions to obtain the final solution. In our case we only track the streamline horizontally, and the vertical transverse terms must also be accounted for in the same manner as undertaken by Leonard et al, (1996). We compute both the vertical transverse terms for the horizontal streamline, and horizontal transverse terms for the vertical flux divergence. The latter is computed using mean edge interpolated values from the horizontal streamline tracing.

Both the velocity and tracer values must be interpolated onto the location where the streamline crosses an edge. For velocity any interpolation scheme may be used. Note that the velocity components at the destination edge (T0) need not be interpolated; we use the east and north rotated velocity vector through that edge, computed from normal and tangential components. For tracers a monotone interpolation scheme must be used. We use a linear least squares interpolation with a slope limiter applied to ensure monotonicity (i.e. no new maxima or minima are introduced).

The timesteps used for this scheme are limited to the Lipschitz condition, which essentially requires that streamlines cannot cross in one timestep. We retain the sub-stepping algorithm used for the standard advection, where if a stability constraint is violated then a smaller timestep is automatically chosen. While the scheme shows acceptable performance, it may be possible to further optimize performance using parallelization. In particular, the streamline integrated tracer value must be recomputed for each tracer (along the same streamline), so this scheme will not scale well with increasing tracers. This is an area for optimization using parallelization.

This scheme was developed in collaboration with Darren Engwirda (NASA), and to our knowledge is the only conservative advection scheme available that operates with long time-steps on unstructured meshes.

We have used this scheme in various test cases and found it to be stable, monotonic and conservative. The solutions for a closed basin test case with sloping bottom forced by a constant wind stress is shown in Fig. 6.2. Here we simulate a passive tracer that has an initial concentration of 10 from the surface to 5 m depth, and 20 from 20 to 100 m depth. The model runs for 20 days, and Fig. 6.2 depicts tracer concentration using the full hydrodynamic model with the ULTIMATE QUICKEST advection scheme and a time-step of ½ hour. This is considered the baseline, exhibiting excellent conservation, with the change in mass of passive tracer less than 4x10-7 percent for the simulation. The FFSL scheme in transport mode using a timestep of 1 hour is shown in Fig. 6.3; the tracer distribution compares very well with the baseline and is also highly conservative (< 2x10-6 % change in mass over 20 days). Additionally, the FFSL scheme runs over 8 times faster than the fully coupled model. The time-step for the FFSL scheme may be increased to 12 hours, in this case the model runs over 21 times faster yet maintains conservation and produces a comparable solution (Fig. 6.4). Finally, the test case is run with a basic semi-Lagrange scheme using quadratic least-squares interpolation (Fig. 6.5), which shows a comparable tracer distribution after 20 days, but lacks conservation, with tracer mass steadily increasing to 0.1 % of the initial mass.

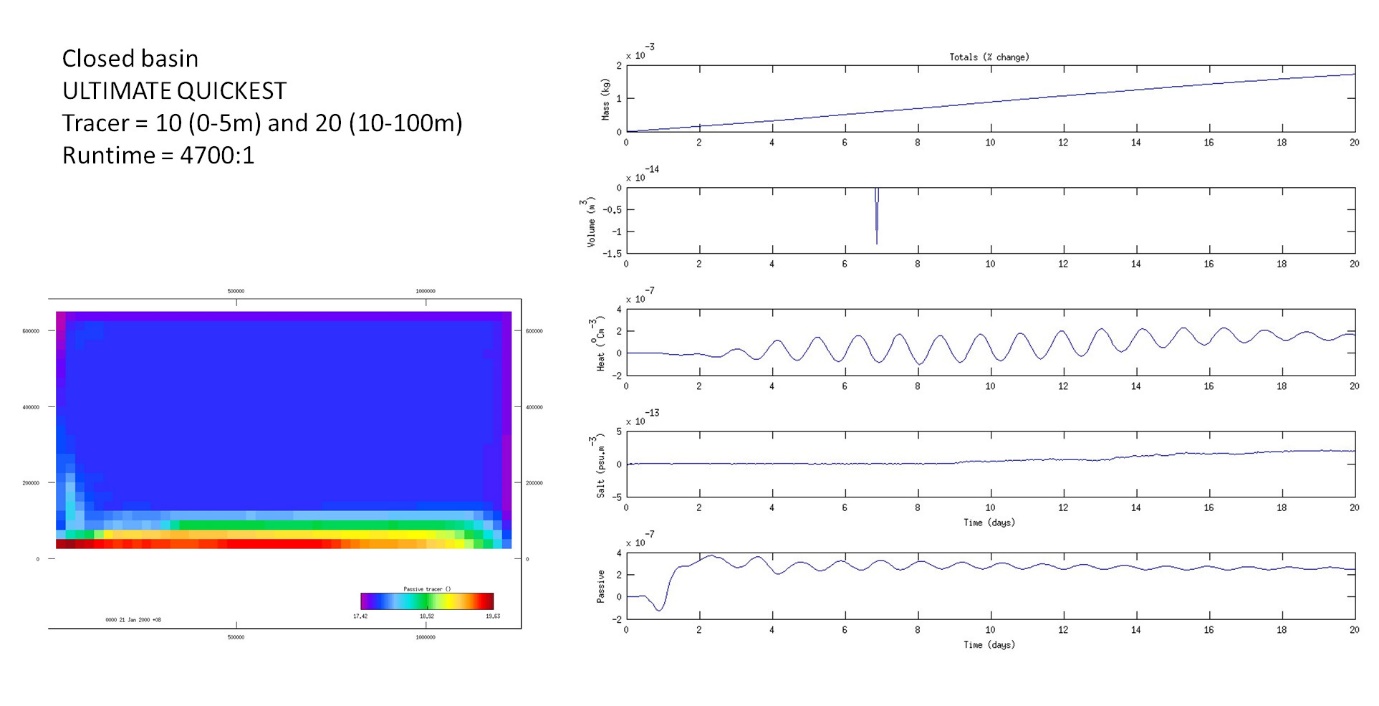


Figure 6.2. Closed basin test case using ULTIMATE QUICKEST.

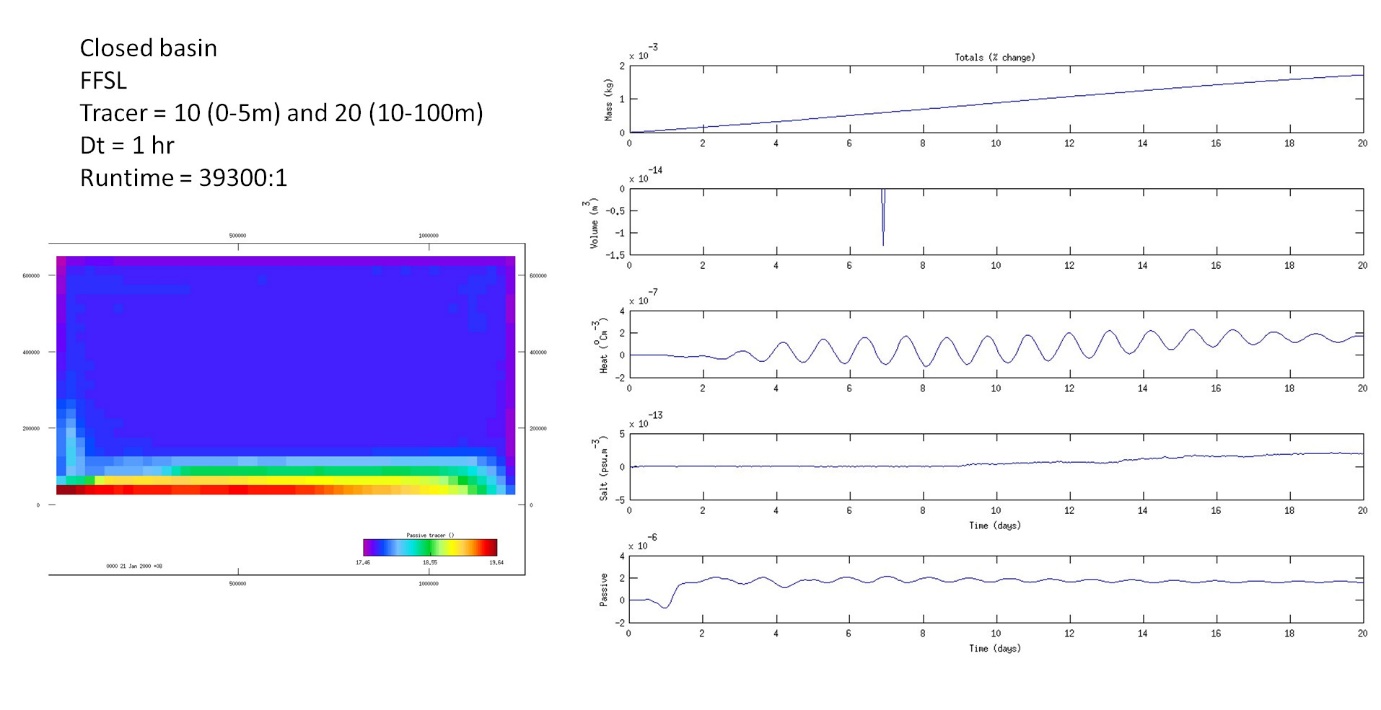


Figure 6.3. Closed basin test case using FFSL with 1 hour timestep.

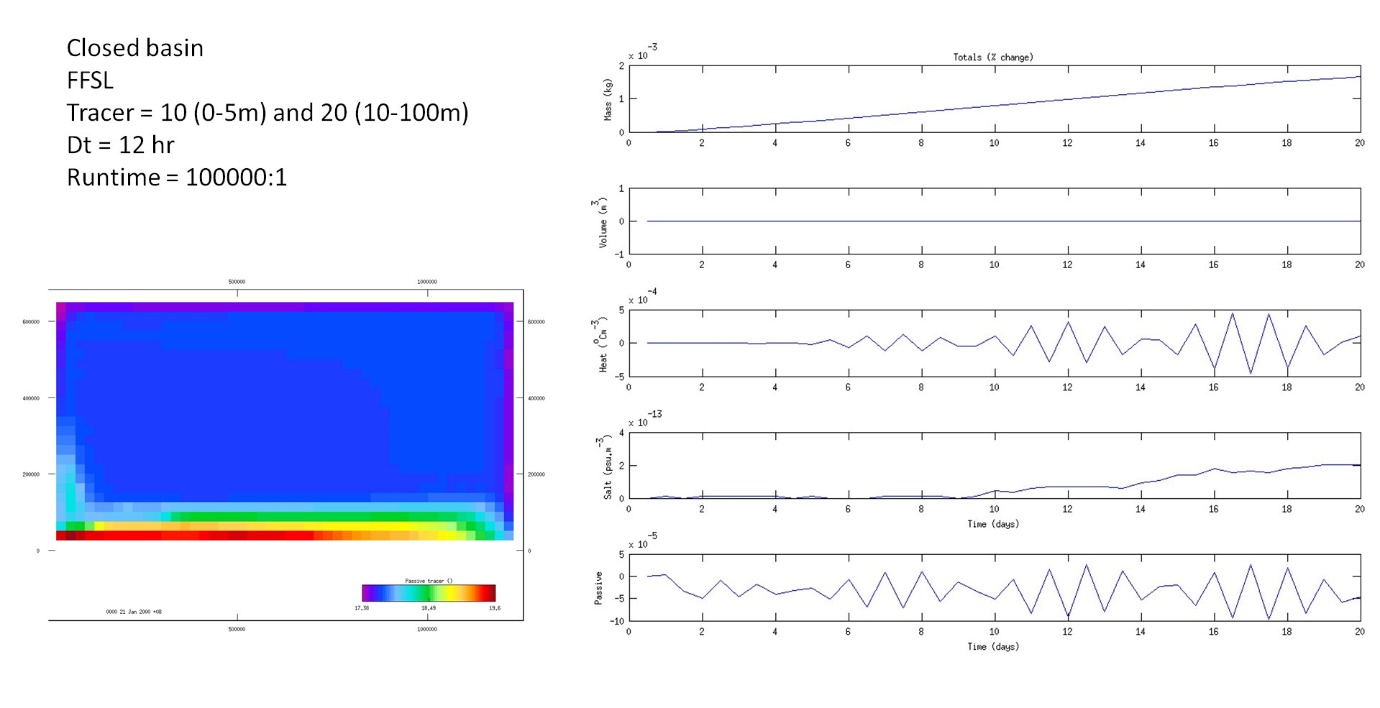


Figure 6.4. Closed basin test case using FFSL with 12 hour timestep.

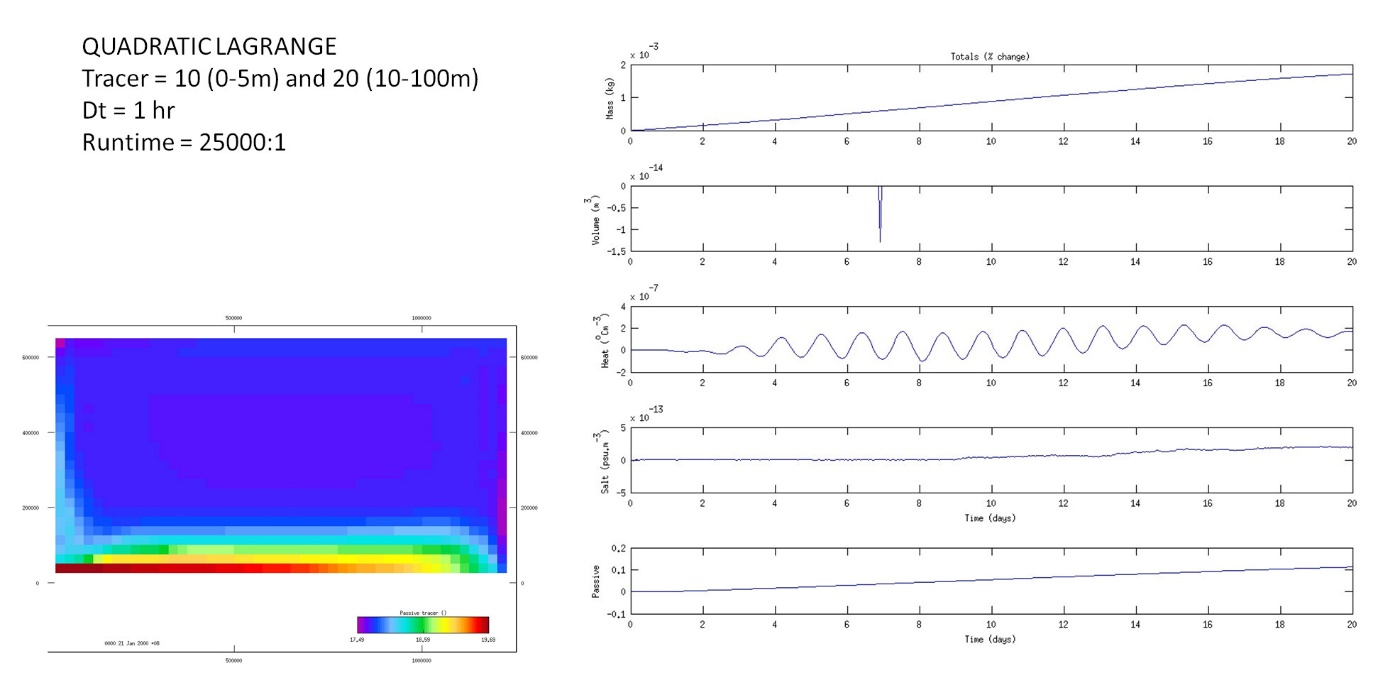


Figure 6.5. Closed basin test case using quadratic Semi-Lagrange with 1 hour timestep.

Another test case in an idealized estuary (Fig. 6.6; freshwater input at one end, tide at the other with a wind stress applied) shows the FFSL scheme resembles the QUICKEST and VanLeer runs, while the Quadratic Lagrange scheme is somewhat degraded.

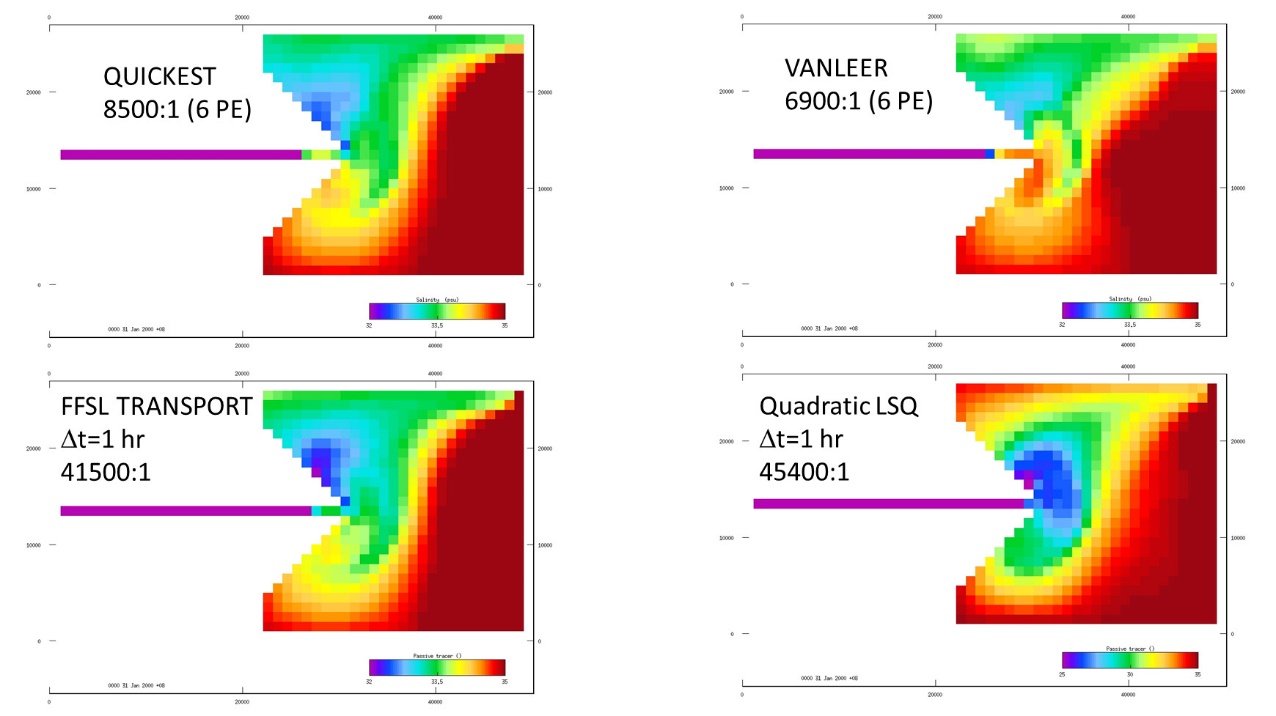


Figure 6.6. Test estuary surface salinity solutions at 30 days.

A realistic test case using arbitrary polygons for the mesh was run for 5 days in the EAC region with realistic forcing in December 2014. The hydro model used the ULTIMATE QUICKEST scheme with a time-step of 38 seconds. The FFSL transport model used a time-step of 1 hour. Results for the surface temperature distribution are shown in Fig. 6.7 (a) and an equivalent passive tracer using the FFSL scheme in (b). The QUICKEST scheme runs at 50:1, whereas the FFSL scheme is 18 times faster at 900:1. Solutions are again comparable, however, the longer time-step used in FFSL has the attractive attribute of producing a less diffuse solution. If explicit horizontal diffusion of 500 m2s-1 is included with FFSL (Fig. 6.7 (c)) then the solution compares even more closely with the QUICKEST solution.

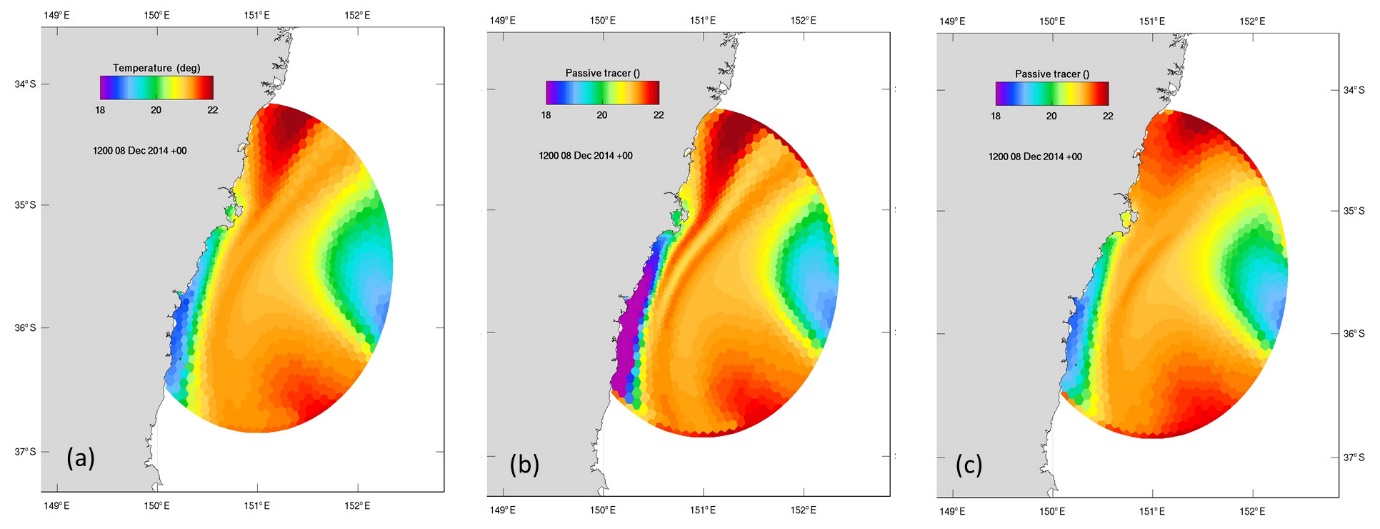


Figure 6.7. Surface temperature solutions for the realistic EAC domain

The FFSL scheme has shown rare cases of non-monotonic behaviour where over and under-shoot are generated in the solution. These instances are not sustained and do not appear to impact the solution in any way. The flux limiter of Thuburn (1995) has been implemented to reduce these instances, and can be optionally invoked. Additionally, clipping may be applied to local minima and maxima in the interpolation stencil at the source cell to render the solution truly monotonic. The process of clipping will violate mass conservation as mass is added or removed from the system. This excess mass may be distributed globally to then render the solution globally conservative. This global filling is different in nature to that used with the Lagrange scheme in SHOC, since the updated FFSL solution is conservative hence the target mass for global filling is exactly known. This is not the case for Lagrange global filling, where mass is only known prior to the update and all sources and sinks of mass (open boundaries, point sources, surface fluxes) must be accounted for to determine the total mass after the update. The estimation of mass fluxes through open boundaries is particularly problematic, hence error is usually introduced to the target mass for global filling. Also, the monotonic clipping in FFSL only occurs at isolated locations (usually in the vicinity of large gradients) and mass that is required to be filled globally is much less than the Lagrange scheme, where every cell is usually subject to non-conservation due to the non-conservative nature of the Lagrange scheme itself (i.e. solving the conservation equation in advective rather than flux form). Although implemented, we currently have not found the need to invoke this clipping.

# Tiling

Once a national mesh is operating, the project aims to partition the mesh into regional tiles that can be operated autonomously while nested with the global model, or ‘stitched’ together via 2-way nesting into a national framework. The partitioning of a national mesh can therefore be achieved such that there is exact overlap of cells across the 2-way interface. This is in essence performing 2-way nesting with no refinement (change in grid size) of the mesh at the nesting interface, and would therefore minimize the spurious short-wave features that are generated and must be controlled when a refinement is present. The technical aspects of tiling the national model therefore revolve around implementation of a robust 2-way nesting system. The approach for this system has been prototyped and assessed (Herzfeld and Rizwi, 2019) and is mature technology that has proven to be suitable for such an application. However, this nesting has only been applied to rectangular grids in a structured model, and some modification may be required for functioning on arbitrary polygons in an unstructured model.

The decomposition of the national model into tiles serves two purposes; firstly, individual calibration can be applied to each tile to result in a more accurate model overall. Each tile can be run in isolation faster than the national mesh, to result in greater throughput and ultimately a better constrained model. Secondly, making tiles available to regional custodians promotes institutional buy-in to result in shared ownership and investment. If this second purpose does not come to fruition due to no external interest, the first purpose remains sufficient to pursue this approach.

The generation of numerous individual models from a national mesh must be efficient and largely automated, as it is anticipated that the tiling process may have to undertaken numerous times in order to optimize the tile limits. With this in mind, an automated tiling system has been developed whereby the tiles are specified via submission of a netCDF ‘region’ file. Upon input file generation, a set of parameter files with accompanying mesh specification is generated for each box in the region file. Included in this are the locations of the open boundaries required to exchange information across the 2-way interface. The level of overlap, or interface separation, of these boundaries may be specified by the user. Also, output dumpfile specifications are created containing the locations required for information exchange in 2-way nesting (temperature, salinity and elevation at cell centres, and normal and tangential boundary velocity at cell edges). Aside from these additions, the tile parameter file is a mirror image of the host model in terms of parameterization and forcing. This allows regional input files to be generated and subsequent model simulation with no additional user optimization. However, it is desirable to optimize the regional parameterization so as to individually calibrate each tile with the objective of improving accuracy. An example a national regionalization is shown in Fig. 7.1, with the associated south east Tasmania tile (region 0) shown in Fig. 7.2.

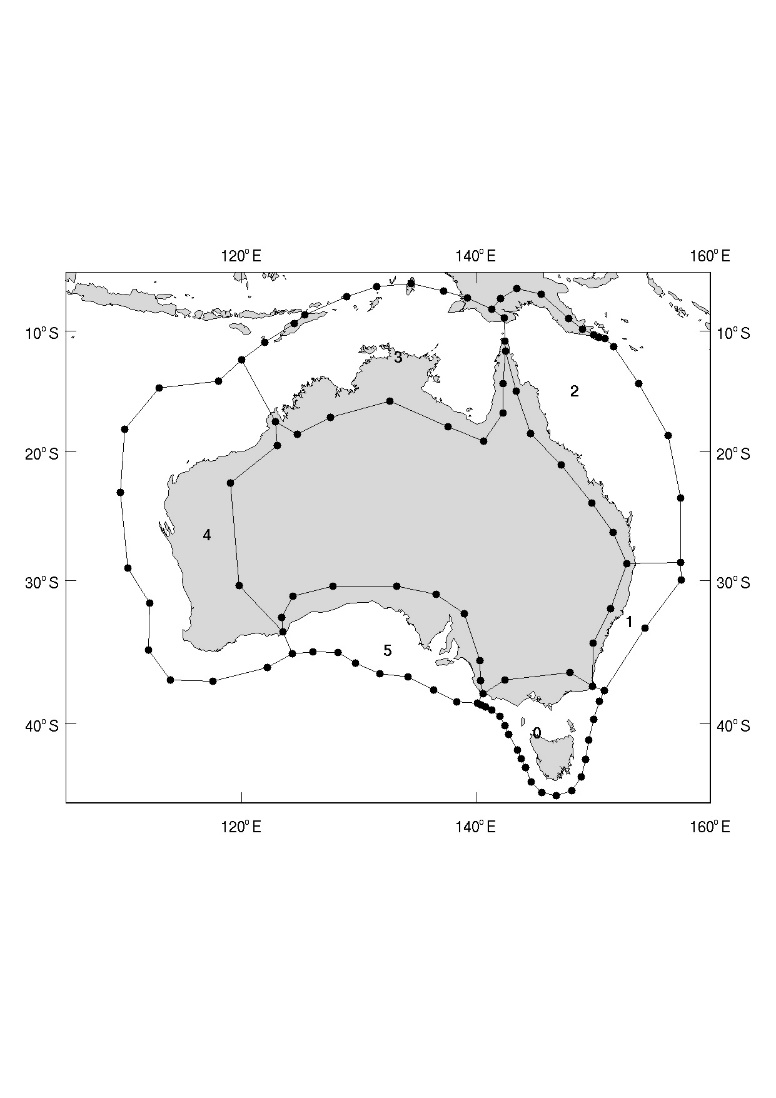


Figure 7.1. Regionalization for tiling.

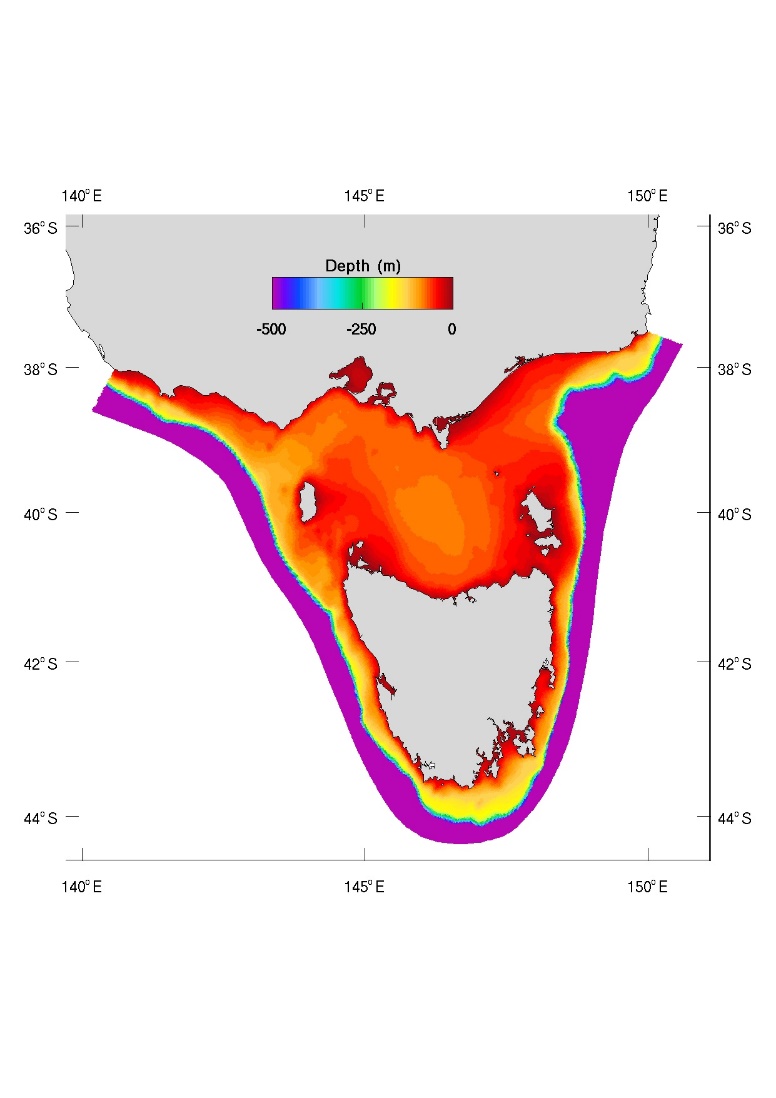


Figure 7.2. Regional Tasmanian (region 0) tile.

The interface separation is an important variable used to maintain stability and accuracy in 2-way nesting. The test estuary used in Section 6 was split in half to assess the 2-way performance. Fig. 7.3. shows two possibilities used for the interface separation, using 1 and 2 cell overlap. In this figure, the most eastern green line in the domain interior corresponds to the eastern open boundary of the western tile (which contains the river), and the western green line is the open boundary for the eastern tile (which contains the tidal boundary). Generally, a larger separation interface will exchange information more representative of the tile interior dynamics, and minimizes over-specification error at the boundary. A separation too large will not contain enough information at the boundary and suffers from under-specification error. Therefore, some optimization of the interface separation is required. The automated tiling allows the interface separation (input as # cells) to be specified in the command line.

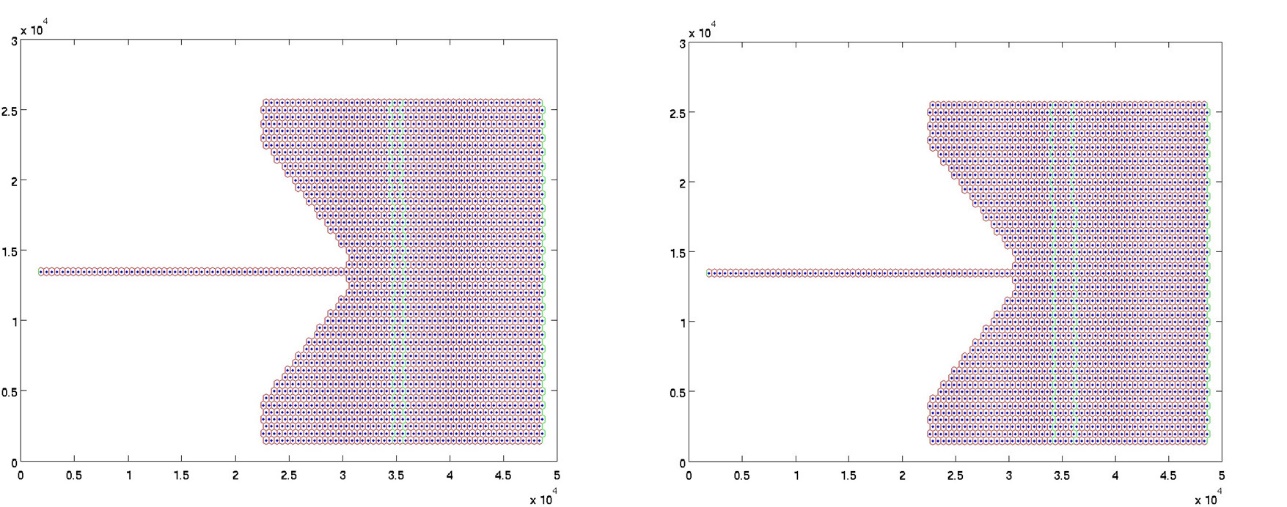


Figure 7.3. Interface separation in the test estuary using 1 cell overlap (left) and 2 cell overlap (right).

An example of the salinity solution for the test estuary with and without 2-way nesting is shown in Fig. 7.4. The 2-way solution is comparable to the non-tiled solution, although differences do occur. This is somewhat expected, as the 2-way algorithm uses open boundaries for information exchange between tiles, and the open boundary equation contains approximations to a seamless solution across the interface. The 2-way parameterisations (e.g. interface separation, type of OBC, restriction operators, sponge zones, barotropic coupling) are being further investigated to optimize these parameters. Importantly, the 2-way solution exhibits a smooth transition across the interface, with no evidence of dis-continuity in flow or salinity at the boundary.

Graphical user interface

Description automatically generated

Figure 7.4. Surface salinity and currents for the tiled (left) test estuary solution at 30 days, and the uncoupled (i.e. baseline) solution (right).

The approach of Herzfeld and Rizwi (2019) used open boundary conditions for information transfer between parent and child models, and found that the transmissive properties of boundary conditions could mitigate specification error issues and allow models to be coupled at the baroclinic (3D) timestep only. This means that sea level is held constant on the boundaries over the 2D time-step, and 2D velocities are the depth average of 3D velocities on the boundary; essentially an average of 2D velocities over the previous 2D time-step. We find that in the proposed application for the National Model, where the separate model tiles are coupled tightly at 1:1 ratios in space and time, that coupling at the barotropic (2D) level is necessary. This is not technically an issue to implement, but does mean that more transfers of information are required (elevation and 2D velocities every 2D time-step). A trial 2-way coupling was performed between tile0 (south-east Australia) and tile5 (southern Australia), with solutions presented in Fig. 7.5. The tile5 solution is shown in Fig. 7.5a, which is coupled to the tile0 solution of Fig 7.5b. The continuous National Model solution is shown in Fig. 7.5c. The parameterization is slightly different between the coupled and continuous solutions, with the horizontal viscosity smaller in the tile0 coupled solution. Nevertheless, solutions are comparable, and again no discontinuities are seen across the interface in the coupled solution. The coupled model also appeared very stable, with no indications of developing instabilities across the interface. A further successful example of coupling between the three tiles 0, 1 and 5 is shown in Fig. 7.6. The 2-way coupling appears a viable methodology to decompose the national domain into smaller sub-models. While the methodology appears sound, a larger effort will be required to couple all the tiles of the National Model.

Chart, surface chart

Description automatically generated

Figure 7.5. Two-way nesting of (a) tile5 with (b) tile0. The continuous solution is shown in (c).

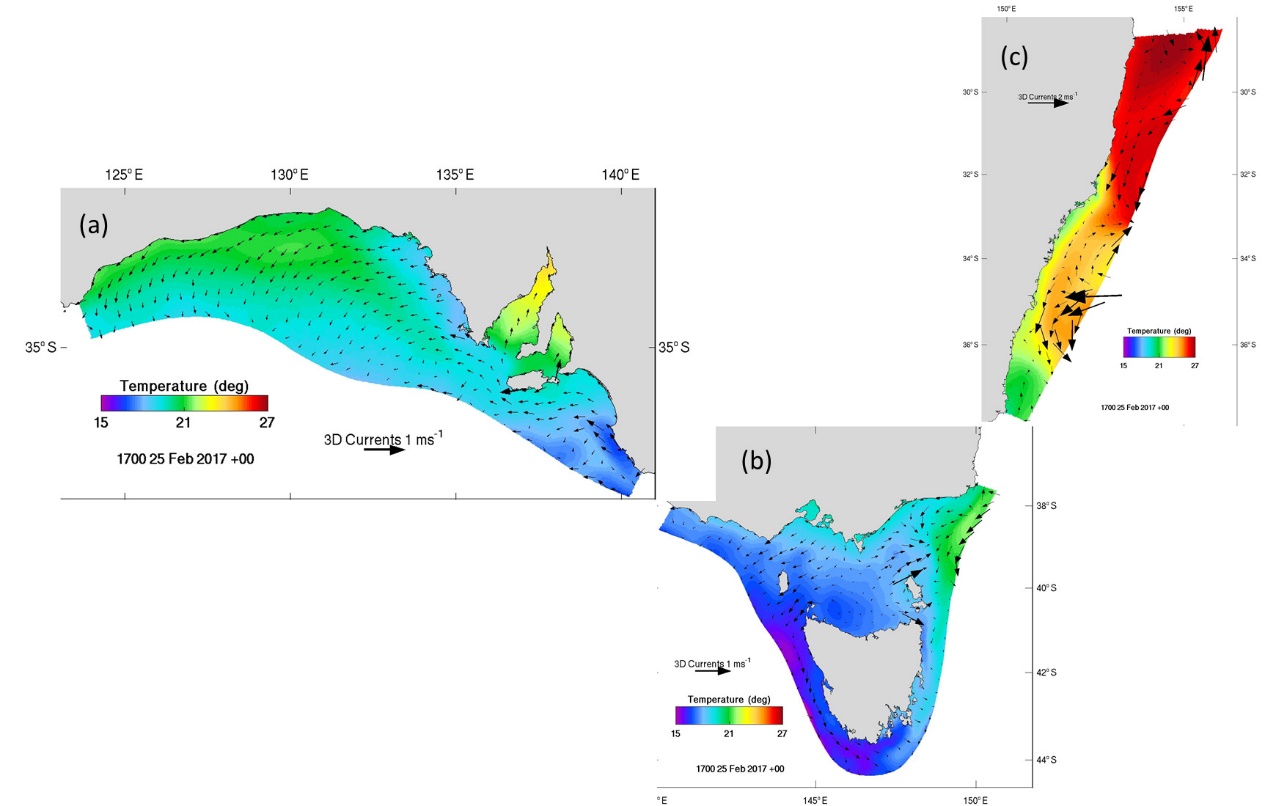


Figure 7.6. Coupling between tiles (a) tile5, (b) tile0 and(c) tile1.

Significant attention has been given to modelling the south-east of Australia for downscaling to case studies along the Tasmanian coastline. The structured model SHOC has been configured to provide a model of this SE Aust. domain (TASC model), for which a snapshot in March 2015 is shown in Fig. 7.7a. This model uses a polar-type curvilinear grid, and it was not possible to make the grid continuous around the Tasmanian coast, resulting in a ‘join’ between Bathurst and Macquarie Harbors on the east coast. Consequently, a discontinuity in the temperature solution is seen at this location. The unstructured model does not suffer these gridding restrictions, allowing a continuous mesh around the coast, hence the discontinuity is absent in an optimized configuration of the national Tasmanian tile (Fig 7.7b). Horizontal mixing has been tuned to minimum values in this model, resulting in sharp frontal boundaries and meso / sub-mesoscale structure along the EAC in the north-east and sub-tropical frontal water in the south-west. This structure appears to be resolved as well, or better, in the national tile than the TASC model. Such mixing parameterization has not been optimized in the National Model (Fig 7.7c – for March 2017), and the temperature solution is smoother with less well-defined structure. The 2-way coupled tile0 (Fig 7.7d) is configured with lower horizontal mixing than the current National Model, which leads to a slight improvement in feature definition, but both models require further tuning of mixing parameterization in order to achieve the results of Fig. 7.7b. Such tuning, along with other critical parameterization, remains to be undertaken for the national domain as part of a comprehensive calibration procedure. Performing such a calibration on the more manageable tiled domains are likely to result in a better constrained model owing to the larger throughput, which is an advantage of the tiled approach.

Chart, surface chart

Description automatically generated

Figure 7.7. Surface temperature for 5 March using (a) TASC model, (b) optimized TILE0, (c) National Model and (d) 2-way coupled tile0.

# Sediment Transport

The national model is to be used to simulate sediment transport using the off-line transport model. This coastal-sediment transport model is to be initialised with a zero concentration of the suspended sediment and the concentration of benthic deposits based on interpolated measurements. The model will be driven with currents simulated by the national model. Sediment resuspension will be enhanced by waves simulated by the unstructured SWAN model. At this stage, an ad hoc sediment concentration will be specified at the river boundaries, to be refined at the later stages of this project. The plan for this year is to establish an early prototype of this mode through the development of the following tasks:

1. Initialisation of the sediment transport model with the realistic distribution of the benthic deposits.
2. Implementation of the unstructured SWAN model to the Australian coast.
3. Implementation of the sediment transport model on the unstructured grids.

### 8.1 Initialisation of the sediment transport model

Fig. 8.1 illustrates the coverage of sampling sites for benthic sediments available from the Geosciences Australia MARS database. The top plot shows sediment concentrations for a number of size-classes (mud, sand, and gravel). The middle and bottom plot shows sampling sites delineating between carbonate and non-carbonate fractions of sediment.

Chart, scatter chart

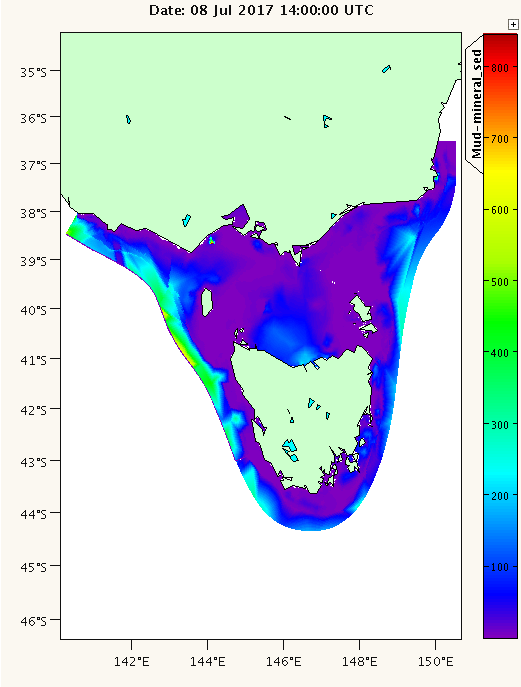
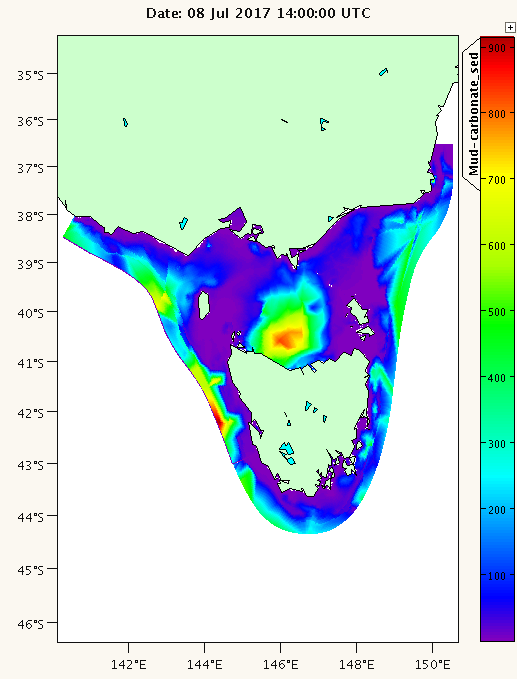
Description automatically generated

Figure 8.1. GA MARS database sediment sampling sites. (a) Gravel, sand and mud, (b) CaCO3 gravel, sand and mud, (c) bulk CaCO3 sampling sites.

According to this data, the coverage of samples with a known content of gravel, sand and mud is quite high and uniform around the Australian coast. There are less sites where we know also the content of carbonates for each of these fractions (middle plot). The number of such samples is particularly low in Gulf of Carpentaria and around Tasmania, whilst the density of such samples is relatively high in the GBR region. The coverage of sites where we know the bulk content of carbonates (but without size fractionation) is fairly high almost everywhere apart from the Gulf of Carpentaria (bottom plot).

Since the carbonate fractions are critical for the accurate evaluation of optical properties in coastal waters, it was decided to keep the fine structure of sediments represented by both carbonate and non-carbonate fractions of sediment classes. For regions where only a bulk fractionation of the carbonate and non-carbonate classes is available, an assumption is made of the proportional allocation of carbonates to each sediment class. For example, in a sediment sample with 20% of bulk carbonates, we assume these carbonates to be made out of 20% of gravel carbonates, 20% of sand carbonates and 20% of mud carbonates.

An example implementation of such an initialisation procedure to the Tasmanian tile is illustrated in Fig. 8.2. Upscaling to the whole Australian cost is the next step.



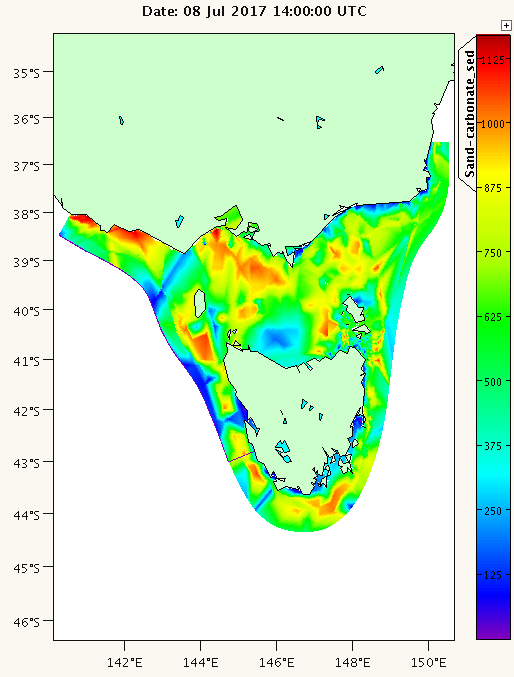
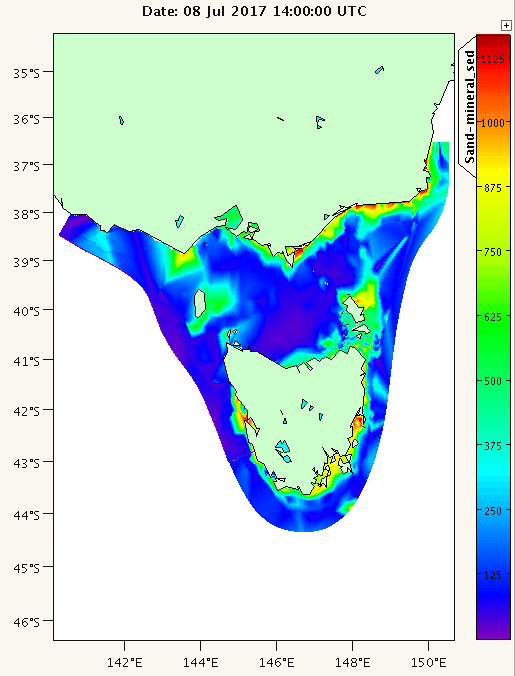
 

Figure 8.2. Carbonate mud (top left), non-carbonate mud (top right), carbonate sand (bottom left) and non-carbonate sand (bottom right) (kg m-3), derived from GA MARS database, Tasmanian tile.

### 8.2 Implementation of the unstructured SWAN model

The sediment transport model requires wave data to simulate sediment resuspension in coastal regions. In eReefs we used BoM Auswaves model to enhance sediment resuspension on the shelf. Auswaves (based also on SWAN with ~10 km resolution) provides an adequate resolution of many shelf-scale processes but does not resolve small-scale coastal features. Numerical schemes based on unstructured grids provide an opportunity to represent both shelf-scale and coastal-scale processes within a single model and with a relatively minor sacrifice in computational efficiency. The plan for this year is to implement SWAN model on such an unstructured grid for a test region and then evaluate the prospect of upscaling it to the national scale.

Two strategies were considered to generate unstructured grids for the wave model. The first one was to generate such grids independently from COMPAS, so that the wave model and COMPAS run on two different grids (e.g. configuration implemented in eReefs). The second option is to run SWAN on the unstructured grid derived from COMPAS. Since SWAN operates on a triangulation rather than a Voronoi diagram, in this case we must use the Delaunay dual of the unstructured mesh for SWAN.

The first configuration (SWAN grid built independently from COMPAS) would make the whole package more modular and flexible since SWAN imposes a number of constraints on the grid structure (e.g. min, max limits on the angles of the triangle, and limits on the number of element connections) which in general may or may not be met by the COMPAS grid. On the other hand, unless an advanced grid generator is used, building and refining a new unstructured grid could be a laborious and time-consuming procedure. Such mesh generation is obviously avoided if the mesh generated for the hydrodynamic model is also utilized for the wave model.

Operating SWAN on unstructured meshes requires two input files; one defining all the grid vertices (‘file.node’) where each vertex coordinates are uniquely indexed, and another one defining the ordering of these vertices into triangle elements (‘file.ele’). These triangle elements are defined using the indices specified in the file.node file and are ordered counter-clockwise. Additionally, a ‘marker’ index is specified for each triangle element, where the marker is set to 0 for the all wet vertices inside domain, 1 for all vertices at the ocean boundary and >1 for vertices at the continental coastline. This is a similar strategy to the definition of the mesh for COMPAS (or JIGSAW). Additionally, the bathymetry data is to be specified in a separate file with a single column, each line of the column showing a water depth at the location of the vertex. The ordering of the depth data in this file must be the same as the ordering of vertices in the file.node.

Fig. 8.3a illustrates an application of the open-source Triangle; package (<https://www.cs.cmu.edu/~quake/triangle.html> ) to generate numerical grid around an idealised low-poly Australian coast. Note that Triangle is generalized meshing package that does not place the numerical constraints orthogonality, well centeredness and CVTs on the dual that JIGSAW does. The size of the grid cells in this mesh is relatively easy to control through the command line parameters of Tringle because of the simplified shape of this domain. For a scenario with a more realistic coastline, the grid again is easy to generate with Triangle and, it may appear suitable (Fig. 8.3b) but because of the small-scale coastal features, Triangle introduces a lot of very small mesh elements when fitting to these features (Fig. 8.3c). Areas with such a high-resolution can slow down considerably the simulation of the model. At the same time, they may not necessarily coincide with the regions of a particular interest for the study (e.g. river channels with a minor impact of waves). Additionally, if the coastline is crossing itself via a side-loop, Triangle completes the triangulation but SWAN does not accept such a grid. Identifying such side-loops manually is not trivial. To summarise, when using Triangle with a complex coastline, a subsequent refinement of the coastline and the mesh is required to improve the initial product. The manual implementation of these refinements could be laborious and time consuming.

An alternative strategy is to either use a more advanced grid generator to build a new mesh from scratch, or extract SWAN compatible grid from the COMPAS grid-generation workflow. In the latter case the SWAN grid and the COMPAS grid would be closely aligned with each other which may provide further benefits in terms of the economy of efforts, easier interpretation, visualisation etc.

Shape

Description automatically generated

Figure 8.3. (a)Triangulation over an idealised low-poly coastline Australia using “Triangle” package, (b) triangulation over a complex coastline, and (c) complex coastline zoomed on Storm Bay.

A national unstructured mesh has been described in Section 2. This mesh has been generated and refined through the application of the advanced grid generator JIGSAW (<https://github.com/dengwirda/jigsaw> ) integrated into the COMPAS libraries. To transform this mesh into a SWAN-compatible mesh, a workflow has been established and successfully tested using the Tasmanian tile of the national model (Fig. 8.4). According to this workflow, COMPAS/JIGSAW generate 3 input files representing the Delaunay dual of the COMPAS mesh for the unstructured SWAN – one file defining the grid vertices, another file defining the ordering of these vertices into triangle elements, and finally the bathymetry file specifying the water depth in the region. This information is readily accessible from the COMPAS mesh topology, and easily extracted and reformatted for SWAN compatibility. The ordering file is then further processed to comply with the SWAN requirement of a counter-clock ordering of the element indices in this file.

Fig. 8.5 illustrates a simulation of the SWAN model on this Tasmanian tile. The SWAN model is driven by ACCESS winds at the surface and WW3 data at the ocean boundary (same configuration we had earlier for a curvilinear SWAN in this region).  The unstructured output is dumped into (non-UGRID) netcdf format and visualised with python script via xarray on NCI (credit to Julian O’Grady and Ron Hoeke).

The next step in this task would be upscaling this workflow to the national scale and running unstructured SWAN around the Australian coast via MPI.

A picture containing diagram

Description automatically generated

Figure 8.4. SWAN compliant triangulation of Storm Bay derived from COMPAS/JIGSAW workflow.

Graphical user interface, application

Description automatically generated

Figure 8.5. A snapshot of unstructured SWAN simulation in Storm Bay.

### 8.3 Implementation of the sediment model on unstructured grid

A number of scenarios have been simulated with the sediment transport model driven by both structured and unstructured hydrodynamic models (SHOC and COMPAS). The goal of simulations was to test the sediment transport model when coupled to the unstructured transport model (Section 6). Both the structured and unstructured applications in a fully-coupled and offline (transport) modes of simulation produce similar distribution of the sediment fields (more detailed description of these experiments is available in a separate report).

# Progress against milestones

Solid progress has been accomplished against the planned milestones. Resourcing has been an issue in some areas, particularly that of data assimilation and parallel processing. The allocation of effort for data assimilation proved unrealistic due to Emlyn’s project and leadership commitments, and consequently the data assimilation component has been removed. Parallelization is progressing, however, previous delays have introduced downstream delays; until this task produces a highly scalable distributed processing model, hindcasts of sufficient length with adequate throughput cannot be accomplished to assess and tune the model.

Within Task 1, Task 1.1, 1.2, 1.3 and 1.7 have been completed; meshing tools have been developed and a national mesh has been developed. A pilot model has been assembled around this and is operating in near real-time. The process of assessment has begun with some areas requiring attention identified. The distributed processing task has made significant progress with a first version anticipated to be trialed early 2021. Similarly, the visualization task has made significant progress, with final products anticipated to be delivered in the first half of 2021. The transport model has been tested and shown to perform extremely robustly and accurately.

Given that a final distributed processing solution is still being developed, model throughput makes detailed assessment challenging. However, several Task 2 objectives have been completed. An automated framework for regional tiling has been developed, and the 2-way nesting framework has been successfully tested. It was considered that extensive consultation over the nature of regional models at this stage would be unlikely to deliver efficient consensus, and a tiling configuration based on existing case studies has been implemented. This partitions the national model into tiles corresponding the eReefs model, the regional model used within the FRDC Storm Bay project, a southern model used for the BP Great Australian Bight project and an area that encompasses WAMSI research interests. This leaves a tile of NSW shelf water, and one spanning northern Australia. As mentioned, the automated tiling now makes generation of alternative regions a routine process.

We have published a manuscript describing COMPAS to Ocean Modelling. Another manuscript ready for submission describes the FFSL advection scheme. Darren Enwirda (NASA) visited the Hobart laboratories in Dec 2019 for 2 weeks to further collaborate on the FFSL development. Rizwi and Herzfeld attended the 18th International workshop on Multi-scale (Un)-structured mesh numerical modelling (IMUM 2019) for coastal, shelf and global ocean dynamics in Santa Fe, New Mexico (24-27 Sep. 2019) where Rizwi gave an oral presentation on distributed processing and Herzfeld gave a keynote presentation on COMPAS.

The progress achieved of the sediment transport model is:

1. Sediment data around the Australian coast has been acquired from the GA MARS database and preliminary evaluation of this data has been carried out.
2. A workflow has been successfully established to implement SWAN on a subset of the unstructured grid. The next step would be to upscale this workflow to the national scale.
3. Sediment transport model driven by an unstructured hydrodynamic model (COMPAS) has been established and applied to an idealised estuary. This model was tested against analogous model driven by SHOC (structured grids). Both models produce similar distribution of the sediment fields in an idealised estuary.

For reference, the project tasks are listed below.

## Task 1.

This task aims to produce a national model operating in near real-time and supporting tools. This national implementation of COMPAS is referred to as the Task 1 model.

### Task 1.1: Mesh generation

Generation of a national mesh with optimized weighting function. The mesh will be optimized with regard to appropriate weighting function, resolution distribution & transition and run-time considerations (number of cells, CFL conditions).

### Task 1.2: Pilot model

Generation of a pilot model. Initial conditions and open boundary and surface forcing fields are to be assembled. A trial period is to be identified and the model run over this period. Model configuration is to be performed to optimize time-steps, open boundary configuration and mixing to achieve a stable run.

### Task 1.3: Initial assessment

Broad assessment of model performance. The national model will be assessed against metrics spread over a broad scale. This would include sea level (tidal and low frequency) at select tide gauge sites, temperature and salinity at National Reference Station sites and seasonal means compared to remote sensing and BRAN output. The goal here is achieve blue water and slope dynamics consistent with the global solutions (boundary currents, mesoscale activity) and consistent seasonal cycles. Tidal dynamics should also be acceptable. Assessment of detailed coastal dynamics should be performed within individual tiles (Task 2.3).

### Task 1.4: Distributed processing

Model performance will be profiled and next generation distributed processing techniques (slave-slave MPI-OpenMP) will be refined. A whole-of-model approach to distributed processing (e.g. including I/O) will be considered rather than optimization of algorithmic routines. The goal is to achieve linear scaling to massive parallelization and minimize the limitations due to Amdahl’s argument.

### Task 1.5: Visualisation

A web based visualization system will be developed capable of displaying both CF structured and UGRID CF netCDF output. This package should include all basic and some advanced features required for data exploration in 4 dimensions. Advance features should include interrogation capability of the underlying mesh coordinate system and mapping functions, with export facilities of that information.

### Task 1.6: Data assimilation

A data assimilating version of the national model will be developed. This will use EnKF assimilation of altimetry, GHRSST and ARGO data. Development is anticipated to allow the current assimilation code to operate on unstructured meshes. Due to the size of the national model, a large amount of compute resource is anticipated hence this task will rely on outputs of Task 1.3 to implement computational efficiencies.

Owing to capacity issues this task has been retracted.

### Task 1.7: Transport model

The transport component of COMPAS will be re-visited to refine interpolation algorithms and include a conservative, unconditionally stable advection scheme. This is anticipated to be a variant of the FFSL scheme available in the structured model.

## Task 2.

This task aims to produce a series of regional tiles with 2-way communication that feed into a national framework. These regional models are referred to as Task 2 models.

### Task 2.1: Regional tiling

The national model will be decomposed into regional tiles. Consultation will be undertaken to identify the geographic extent of the regional tiles, and any prospective custodians of those tiles. A small workshop to achieve this end may be required. Each regional tile will be run in autonomous mode, nested directly into global forcing products, for the simulation period chosen in Task 1.2.

### Task 2.2: 2-way nesting

The decomposed model is required to be assembled into a national framework via 2-way nesting. While the technology is available to do this, the process requires optimization in the form of specification of interface separation, restriction operators and sponge zones to achieve a stable, error-free solution. The message passing required for 2-way nesting will be assessed for performance and refined as necessary.

### Task 2.3: Regional calibration

Each tiled partition created in Task 2.1 will undergo calibration to local data applicable to the particular tile region. The quantity and quality of data available will be dependent on custodianship of the tiles. A minimum dataset comprising that contained in the IMOS database, in conjunction with GHRSST data will be considered for each tile. This should be augmented with tide gauges and wave-rider buoys if not already included in the IMOS database. Model configuration upgrades will be made based on outcomes of the regional assessments.

### Task 3: Sediment transport.

Groundwork to run sediment transport nationally. A map of sediment type and distribution is to be assembled for initial conditions. Bottom stress due to wave motion is required for resuspension of sediments, hence a functioning wave model is required for sediment transport. The SWAN wave model is to be tested on unstructured meshes in preparation for deployment on the national mesh.

### Task 4: Operationalization

The system produced through these Tasks 1 and 2 is to operate routinely in near real-time. At a minimum, the national Task 1 model should operate in this mode, with output appended to a growing archive. To perform Task 2.3 solely within this project would likely involve effort beyond the scope of this project, hence the operationalization model adopted for the Task 2 product is dependent on the level of external interest in the system. The compute infrastructure hosting the regional models is yet to be identified, and will be dependent on the appetite of custodians to run regional applications in-house. Two approaches at differing extremes may be that all tiles are run within this project on NCI or CSIRO infrastructure, or each custodian runs their tile on their local infrastructure and network communication enables the 2-way nesting into a national framework. Given the computational demands of the project, hosting the system on NCI infrastructure appears to be a sensible way forward. Implementing the Task 2 product into an operational framework may be in the form of a roadmap to be realized under a discrete multi-institutionally funded project, rather than a final product within this project. At a minimum, we hope to demonstrate the operationalization of the Task 2 system as a proof-of-concept.

# Next Steps

The technical barriers to developing a national model are on track, and it is anticipated a scalable pilot model and associated regional tiles will be available by the project’s end. However, the current model achieves maximum average resolution of 2 km at the coast, whereas the target resolution is 1 km. Without efficient parallelization, the development of a 1 km model is unrealistic, therefore a slightly coarser version was developed for addressing the technical issues that were required to be overcome. By the end of the project we anticipate to be in a position to revisit the mesh and produce a highly optimized model at the target resolution. We propose to extend the project for 2 years to achieve this, with reduced effort of key personnel allocated over that timeframe. This project is considered to be of strategic importance to O&A, and it is considered that keeping development in-house is advantageous from an IP ownership perspective. Consequently, it has been decided to develop the refined model version (v3) using appropriation funds.

The activities requiring attention for v3 development are:

1. Mesh development. A higher resolution mesh is to built using existing tools. This is now a routine procedure, however, coastline optimization should be revisited to adequately resolve, or omit, all estuaries, channels and bays around the Australian coastline. Enhanced resolution in certain areas is also to be implemented; initially this is anticipated to be the same as v2 with the addition of waters adjacent to capital cities.
2. The coastline optimization is currently a manual process using rudimentary tools. We require a graphical interface to streamline this process, and some visualization effort is required to be dedicated to developing these tools for the efficient coastline generation required by mesh generation. General progress on MoVE and ParaView will also continue.
3. Once a higher resolution mesh is developed, model stability analysis and validation can commence. A near real-time implementation of v3 will follow, with ongoing parameterization improvement via calibration to observations.
4. The regional tiling can be subject to refinement, and a series of regional tiles produced. These may be used in applications under parallel existing projects, e.g. the GBR tile within eReefs.
5. The parallel processing development will continue, where identification and rectification of bottlenecks will be undertaken. Fine grained and I/O parallelization will be implemented within the code where possible. Algorithms that supplement the dynamic core (e.g. the transport model, 2-way nesting) will also be subject to parallelization.

# References

Beaman, R. 2010. Project 3DGBR: a high-resolution depth model for the Great Barrier Reef and Coral Sea. Final Report to Marine and Tropical Sciences Research Facility Final Report, June 2010

Burchard, H., K., O. Peterson and T.P. Rippeth (1998) Comparing the performance of the Mellor Yamada and the k- turbulence models. *J. Geophys. Res.*, 103, 10,543 – 10,554.

Engwirda, D. (2017) JIGSAW(GEO) 1.0: locally-orthogonal staggered unstructured grid generation for general circulation modelling on the sphere, Geosci. Model Dev. 10 (6) 2117.

Egbert, G.D. and Erofeeva, S.Y. (2002) Efficient inverse modelling of barotropic ocean tides. Journal of Atmospheric and Oceanic Technology, 19(2), 183-204.

Gillibrand, P.A., Herzfeld, M. (2015) A mass-conserving advection scheme for offline simulation of tracer transport in coastal ocean models. Ocean Modelling. 101, 1-16.

Griffies, S.M., Hallberg, R.W. (2000) Biharmonic friction with a Smagorinsky-like viscosity for use in large-scale eddy-permitting ocean models. Mon. Wea. Rev., 128, 2935-2946.

Herzfeld, M., Andrewartha, J.R. (2012) A simple, stable and accurate Dirichlet open boundary condition for ocean model downscaling. Ocean Modelling, 43-44, 1-21.

Kondo, J. (1975) Air-sea bulk transfer coefficients in diabatic conditions. *Boundary-Layer Meteorology*, 9, 91-112.

Large, W.G., and S. Pond (1981) Open ocean momentum flux measurements in moderate to strong winds, *J. Phys. Oceanogr.*, 11, 324-336.

Leonard, B. P., Lock, A. P., & MacVean, M. K. (1996). Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Monthly Weather Review*, *124*(11), 2588-2606.

Lin, S.J., Rood, R.B. (1996) Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Wea. Rev.*, 124, 2046-2070.

Oke, P. R.; G. B. Brassington, D. A. Griffin, A. Schiller, 2008: The Bluelink ocean data assimilation system (BODAS), Ocean Modelling, 21(1-2), 46-70.

Puri, K., Dietachmayer, G., Steinle, P., Dix, M., Rikus, L., Logan, L., Naughton, M., Tingwell, C., Xiao, Y., Barras, V., Bermous, I., Bowen, R., Deschamps, L., Franklin, C., Fraser, J., Glowacki, T., Harris, B., Lee, J., Le, T., Roff, G., Sulaiman, A., Sims, H., Sun, X., Sun, Z., Zhu, H., Chattopadhyay, M., and Engel, C. (2012) Implementation of the initial ACCESS Numerical Weather Prediction system, Aust. Met. Oc. Journal, 63, 265-284.

Ringler, T., Petersen, M., Higdon, R. L., Jacobsen, D., Jones, P. W., and Maltrud, M. (2013) A multi-resolution approach to global ocean modeling. Ocean Modelling, 69(C), 211–232. doi:10.1016/j.ocemod.2013.04.010

Ringler, T., Petersen, M., Higdon, R. L., Jacobsen, D., Jones, P. W., and Maltrud, M. (2013) A multi-resolution approach to global ocean modeling. Ocean Modelling, 69(C), 211–232. doi:10.1016/j.ocemod.2013.04.010

Smagorinsky, J. (1963) General circulation experiments with the primitive equations, I. The basic experiment. *Mon. Wea. Rev.*, 91, 99 – 164.

Thuburn, J. (1995) Multidimensional flux-limited advection schemes. J. Comput. Phys., 123, 74-83.

Thuburn, J., T. Ringler, J. Klemp and W. Skamarock, 2009: Numerical representation of geostrophic modes on arbitrarily structured C-grids, Journal of Computational Physics, 2009: 228 (22), 8321-8335. [doi:10.1016/j.jcp.2009.08.006](http://www.sciencedirect.com/science?_ob=ArticleURL&_udi=B6WHY-4X1J78X-2&_user=2493154&_rdoc=1&_fmt=&_orig=search&_sort=d&_docanchor=&view=c&_acct=C000057551&_version=1&_urlVersion=0&_userid=2493154&md5=7b0f9e6bbbf8712ebc27cd213f0d07fe)

Van Leer, B. (1979) Towards the ultimate conservative difference scheme. V: a second order sequel to Godanov’s method. *J. Comput. Phys.*, 32, 101-136.

Zillman, J.W. (1972) A study of some aspects of the radiation and heat budgets of the southern hemisphere oceans. *Bureua of meteorology, Meteorological study no. 26*, Australian Govt. Pub. Service, Canberra.