ANUGA – A New Free & Open Source Hydrodynamic Model

E Rigby, Rienco Consulting, Wollongong NSW 2500 Australia
E-mail: ted.rigby@rienco.com.au
R VanDrie, Balance Research & Development, Wollongong NSW 2500 Australia
E-mail rudy@balancernd.com.au

Abstract

In December 2006, the Australian National University and Geoscience Australia released their 2D hydrodynamic model ANUGA to the free and open source software world. While initially developed to simulate coastal inundation by a tsunami or storm surge, ANUGA is a fully generalised hydrodynamic model with potential as a tool for simulation of any flow that is predominantly 2D in nature. Given this flexibility, the Authors of this paper became interested in and have trialled the model as a conventional hydrodynamic 2D flood model on both a complex urban system and a simpler rural system. This paper presents their findings in terms of the complexity in constructing and running these two trial models and comments on ANUGA's applicability to flood modelling generally. The urban model included a dam break scenario with floodwater passing through a residential area. The ability to construct a model with elements varying in size to suit the features being modelled permitted flow behaviour to be simulated realistically and at a level of local detail that structured grid models cannot practically reproduce ANUGA is mostly coded in Python, a very powerful but easy to learn (free and open source) object oriented language. As such, extensions or amendments to the model are very easy to develop and implement. The authors have recently added the ability to model distributed rainfall and infiltration and are currently exploring options for accommodating underground networks and structures such as bridges and culverts. The paper concludes that ANUGA is a model worthy of serious consideration whenever a problem arises involving simulation of flooding in streams, estuaries or as a result of coastal inundation.

1. INTRODUCTION

ANUGA is a general fluid flow modelling tool developed to simulate the effects on the built environment from hydrological hazards such as tsunami, storm surges or dam breaks. It is based on a research prototype developed at the Australian National University throughout the nineties. This prototype was used by Zoppou and Roberts (1999) to investigate the impact of the collapse of a water reservoir on a downstream urban area. The present incarnation of ANUGA is based on lessons learned from this prototype but the model was substantially redesigned and enhanced during 2004 at Geoscience Australia for use with risk analysis of natural hazards.

ANUGA implements a finite-volume method for solving the conserved form of the 2D depth integrated Shallow Water Wave equations. The study area is represented by a mesh of triangular cells in which the conserved quantities of water depth, h, and horizontal momentum (uh, vh), in each volume are tracked. Fluxes across cell boundaries are calculated using the central-upwind scheme of Kurganov, Noelle and Petrova (2001). One advantage of this approach is that the traditional characteristic decompositions and Riemann solvers are replaced by one simple scheme that efficiently addresses super- and sub critical flows, wetting and drying as well as faithful reproduction of planar surfaces. ANUGA uses a second order spatial reconstruction to produce a piece-wise linear representation of the conserved quantities. This surface is allowed to be discontinuous across the edges of the cells, but the slopes are limited to avoid artificially introduced oscillations. As a consequence wave fronts can be arbitrarily steep allowing for stable resolution of bores and hydraulic shocks. The algorithms underlying solution of the shallow water wave equations are discussed further in the user manual and paper by Toro (1992).

ANUGA is written in the object-oriented programming language Python with computationally intensive parts implemented as highly optimised shared objects written in C. Python is known for its clarity, elegance, efficiency, flexibility and reliability. In particular, complex software can be built in Python with relatively few considerations relating to the idiosyncrasies of the underlying software language syntax.

In addition, Python's automatic memory management, dynamic typing, object model and vast number of libraries means that software can be produced quickly and can be readily adapted to changing requirements throughout its lifetime.

The fundamental object in ANUGA is the *Domain* which inherits functionality from a hierarchy of increasingly specialised classes starting with a basic structural mesh to classes implementing the finite-volume scheme described above. Other classes are *Quantity* with each instance representing values of one variable across the mesh along with their associated operations, *Geospatial_data* which represents georeferenced elevation data and a collection of *Boundary* classes which allows the model to be built without having to modify the underlying source code.

To set up a scenario the user specifies the study area along with any internal regions where increased mesh resolution is required. External edges are labelled using symbolic tags which are subsequently used to bind boundary condition objects to tagged segments of the mesh boundary. The mesh is then generated using ANUGA's built-in mesh generator and converted into the *Domain* object which provides all methods used to setup and run the flow simulation.

ANUGA is based exclusively on free open source software (FOSS) components and was released as such under the GPL license in December 2006. The main advantages of using FOSS include

- The costs of using existing FOSS components are almost zero. This has obvious advantages for any organisation, as funds can be put to other use. Equally important, tools can be assessed and selected quickly given a number of alternatives allowing for convergence to best practices.
- The source code is open and readable by anyone. The software can be analysed, verified and modified by anyone with programming skills, allowing for the addition of features needed by a particular group. Traditionally, there have been many wasted opportunities to re-use perfectly viable code due to the absence of an accepted process and legal framework. FOSS can address this issue. If one government agency is writing a system from scratch and making the final product of such efforts available to other government departments each organisation can leverage a large amount of software and use it for their own purposes.
- Open source software draws its strength from individuals improving, modifying or customising programs and giving them back to the open source community for the benefit of others. The consequence is a rapidly improving software base that advances science and technology faster than most individual organisations can realistically sustain. This is due to a large degree of leverage off pre-existing work, very much as is the case with the traditional scientific process of publication, reproduction of results, peer-review and further development.

In addition to the obvious advantages of using OSS, there are clear advantages in releasing software as OSS as well:

- The product will get more use by others organisations and individuals. Consequently, the amount of feedback, fault reports and suggested improvements is likely to boost the development cycle significantly.
- The level of code reuse across organisations will increase
- Assuming a software product is sufficiently useful, an open source release is likely to increase the profile of the originating organisation and may even enable it to set standards in case the software is widely adopted.
- Releasing core components as OSS increases the transparency of the business process underpinned by the software.
- Releasing software to collaborators and stakeholders is likely to contribute to visible impacts in their business and thus help the strategic positioning the organisation in terms of real outcomes as opposed to simple outputs.

A number of professionals are currently piloting and exploring the scope of applicability and investigating the quality of results obtained from ANUGA by comparison with other models and recorded data. This paper describes one such endeavour, undertaken to ascertain whether ANUGA can be feasibly used to simulate stream and overland flooding.

2. MODEL CAPABILITIES & LIMITATIONS

2.1 Current Capabilities

ANUGA is a library of classes and functions that can be called to facilitate the construction of a triangular mesh based (2D) hydrodynamic model and to simulate water movement associated with various boundary conditions applied to the model domain. Code modules in the present release include;

abstract_2D_finite_volumes	code providing tools for mesh generation and domain creation
alpha_shape	code to create an encircling region around a spatial dataset
caching	code providing caching abilities within ANUGA
coordinate_transforms	code providing coordinate transforms between projections
damage_modelling	code providing damage assessment due to inundation
fit_interpolate	code providing least squares fitting and interpolation
geospatial_data	code providing for the import and export of GISdata
load_mesh	code providing the ability to load an ASCII mesh
mesh_engine	code for creating a triangular mesh within a polygonal region
pmesh	code to assist users create and view triangular meshes
shallow_water	code solving the governing shallow water wave equations
utilities	code for basic numerical tools used in ANUGA

Other modules present in the current release include;

advection	code providing a finite volume solution of the advection equations
documentation	a sub directory containing the user manual

In addition, global configuration parameters are contained in the ANUGA root directory in anuga_config.py and config.py.

The above code library provides users with the tools to create a 2D model of flood behaviour that;

- Accommodates input data in many different formats
- Accommodates a layout of user defined, variably sized, triangular elements
- Accommodates a layout in which surface roughness can vary spatially across the model
- Accommodates a wide range of boundary conditions
- Accommodates temporal changes in everything but the mesh layout (elevation, roughness etc)
- Allows super-sub critical flow transitions /shocks to form and propagate
- Is highly stable (due in the main to the finite volume formulation)
- Permits caching of intermediate data between runs

2.2 Current Limitations

While well developed as a tsunami model for experienced users within Geoscience Australia and the ANU, the present model has limited documentation and support, and requires at least basic python programming skills to create and run a simulation. Pre-processing of data is in addition rather cumbersome as the code does not provide a Graphical User Interface to assist users construct a model.

From a flood modelling perspective, the present model also has limitations in that it;

- Lacks a strong GIS interface for data I/O
- Is difficult to set up when there are with many areas of differing roughness
- Is difficult to set up and assign boundary conditions to tagged boundary segments
- Does not include consideration of kinematic viscosity in the solution of the governing equations

- Has no ability to simulate flow through structures (culverts-bridges-underground networks etc as 1D links or in 2D)

2.3 Ongoing Development

All of the above limitations are the subject of ongoing development. In particular; current development of relevance to flood modellers includes;

- Code to handle 1D links (culverts, underground networks and bridges) between elements
- Code to handle bridges in 2D recognising the advent of pressure and weir flow
- The inclusion of kinematic viscosity in the solution of the governing equations
- The development of much stronger links to GIS for pre and post processing of data

Python skills will always be needed by those wishing to explore and/or alter the underlying code. It is envisaged however that run file templates and a GUI run file builder will be developed over the coming year that will reduce the need for users to interact directly with the python run time code

Support is clearly an increasing matter of concern as the user base expands.
A user forum is currently available at https://sourceforge.net/forum/forum.php?forum_id=593114
and a mailing list at https://sourceforge.net/mailarchive/forum.php?forum_name=anuga-user .
The ANUGA User Manual is available as a separate download from sourceforge.net at https://downloads.sourceforge.net/anuga .

3. RURAL FLOOD MODEL TRIAL

3.1 Generally

Macquarie Rivulet, was chosen as the trial catchment for flood modelling in a rural environment. This catchment is located in coastal New South Wales, some 80 km to the south of Sydney covering an area of approximately 107 km^2 of mostly forested or grazing land (refer Figure 1). There is some urban development in the lower reaches of the catchment, around Lake Illawarra (refer Figure 2). This catchment was chosen as a trial catchment, as it had been previously modelled by the Author (with HEC 2, RMA2, Fst2dh and Tuflow) and good quality calibration data was available for a significant flood event in June 1991.

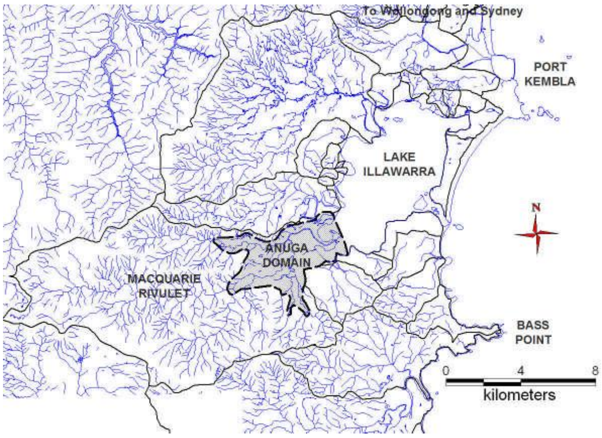

Figure 1 Catchment Location

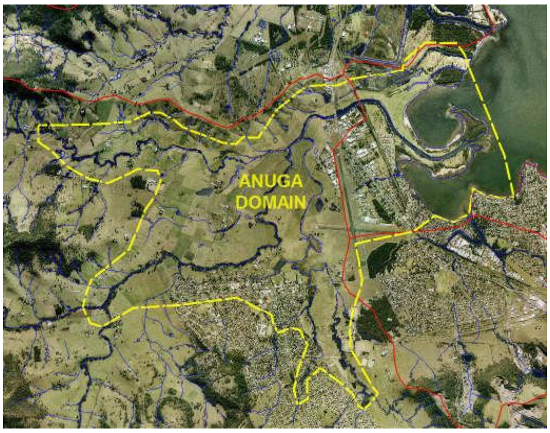

Figure 2 ANUGA Domain

First contact with yet another programming language presented some problems, for the author of this trial, resulting in very slow progress in the initial stages of model construction. While data was available in a (Tuflow-Mapinfo) well attributed GIS format for the catchment, it was necessary to export most data to a less intelligent (csv) format before it could be re-ingested into the ANUGA model. This was a time consuming process and its improvement an early addition to the development wish-list.

The same 2D (active cell) model domain used in Tuflow was used as the boundary of the 2D domain in ANUGA (refer Figure 2). Being an unstructured mesh, this boundary represents the full extent of the ANUGA mesh, providing advantages over grid based models in regard to array sizes.

Notwithstanding these early difficulties, the model mesh was readily generated and the domain created from the mesh with elevation data read in from an XYZ comma separated value (ALS) dataset. When the mesh was first created it was found to contain a large number of very small triangles, drastically reducing the (Courant based) computational timestep and increasing the run time. On investigation it was found that the export of the bounding polygon from GIS had included some near co-incident vertices, forcing the triangulation algorithm to create extremely small triangles to honour these vertices. The triangulation code has since been modified to identify and warn the user about the presence of 'close' vertices in the bounding and regional polygons used in the triangulation process. The final model mesh contained some 190,000 triangles, varying in size from $15m^2$ to $200m^2$ (viz 5.5m to 20m side length). The Tuflow model was by comparison a 10m (685 by 612) grid model of which approximately 175,000 cells were active.

Surface roughness zones were read in from a series of polygonal boundary files obtained from the existing Tuflow-Mapinfo material file and applied to the domain. These zones are reproduced in Figure 2. This also was a somewhat cumbersome process as each of the several hundred polygonal roughness zones had to be individually read in and associated with a particular surface roughness. A more direct GIS oriented input process is needed if the model is to be used for flood modelling on a regular basis.

As a tsunami model, there was little initial interest in or need for a means of inputting rainfall to the model. In flood modelling however, rainfall hyetographs (or runoff hydrographs) are key inputs. Before any work could begin on trialling ANUGA as a flood model, it was necessary to add rainfall inputs to the model code. While different forms of applying rainfall (or hydrographs) are currently under development, the procedure used in this trial involved the application of flow (a hydrograph) to user defined polygons located within the model at user specified locations. The associated hydrograph is then read in from a CSV file, much the same as in Tuflow. This addition to the initial code was a relatively straightforward task.

As the input of time varying water levels is a clear requirement for a tsunami model, there was no difficulty applying a time varying downstream boundary condition, using the present code.

Once the initial problem with triangulation was identified and corrected, the model was run to simulate 36 hours of flooding in a PMF flood event. The run time (using version V1.4733) was problematic, taking 40 hrs on a high end Windows XP Pentium with 4Mb of RAM, compared with Tuflow running the same simulation on the same machine in 18 hrs. In pursuing means of reducing run times, It was noted that ANUGA's computational timestep is automatically calculated within the code at each timestep using a (conservative) Courant number of 1. This resulted in computational timesteps varying from 0.1 to 0.2 seconds during the simulation, considerably less than the fixed 2 sec time-step used successfully in Tuflow. Given the potential reduction in run times from simulation at a higher Courant number, a further series of trials at increased Courant numberers are currently underway.

While the animation tool included in the current release produces an impressive qualitative display of a simulation, it is weak in respect to the display of quantified plots as required in flood modelling. This tool is also the subject of ongoing and current development. In Figure 4, an oblique graphic of a PMF event around its peak is shown, created by the present animation tool, While the ANUGA model has not been calibrated at the time of writing this paper, it is encouraging to observe that with the same roughness values used in the calibrated Tuflow model, the flood extents of the ANUGA model are quite similar.

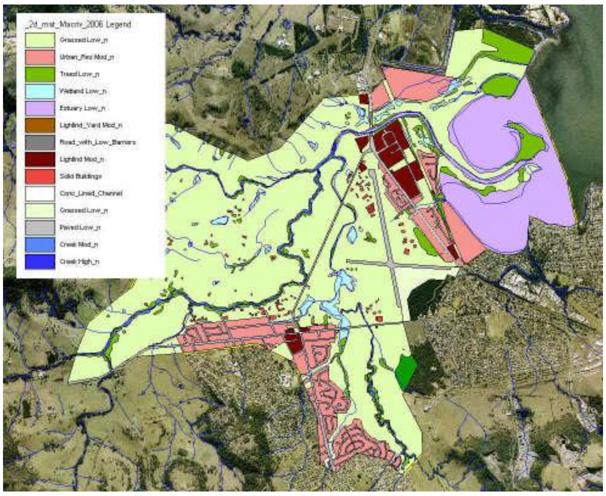

Figure 3 Model Surface Roughness Zones

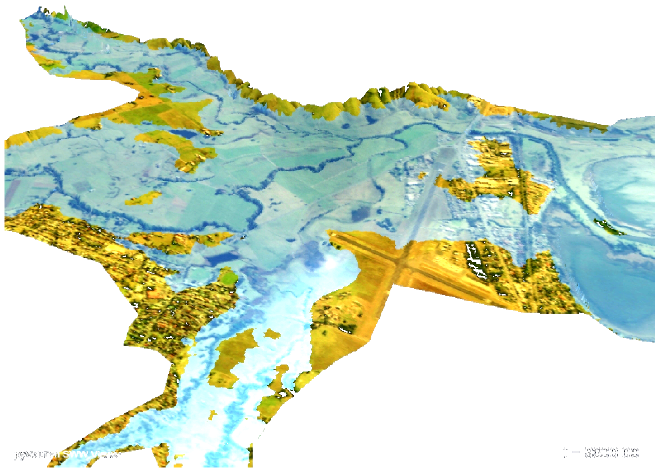

Figure 4 Preliminary PMF Flood Surface

3.2 Strengths

In summary, the model strengths apparent in this trial included;

- Good control of mesh resolution where required to accommodate rapidly changing topography (such as across creek waterways) and/or changes in flow direction and magnitude (such as through bridges
- As the model is of variable resolution, there is no need to link in a 1D model of the rivulet to adequately describe the waterway geometry, avoiding the potentially serious loss in solution quality across the 1D/2D interface
- The ability to produce a stable, reproducible solution
- Source code available to help understand the underlying algorithms

3.3 Weaknesses

- No structures are provided for in the code as yet. In its present form, ANUGA can not correctly simulate flooding where structures significantly impact flood behaviour
- There is no kinematic viscosity (Ke) term included in the governing equations, limiting the ability of the model to correctly model flow behaviour in areas of high velocity gradients
- The post-processing animator has limited output capabilities in respect to the preparation of the qualitative graphics used in flood mapping
- Relative to other, more mature commercially available hydrodynamic flood models, ANUGA flood models are currently time consuming to construct and slow to run

3.4 The Future

In the author's view, ANUGA has excellent potential as a rural flood modelling tool but needs another year in development before it will be realistically able to meet the demands of real world rural flood modelling in an acceptable construction and simulation timeframe**.** Notwithstanding its present limitations**,** ANUGA is, in the Author's view, a model that will feature strongly in future rural flood modelling**.**

4. URBAN FLOOD MODEL TRIAL

4.1 Generally

Ongoing urban development in Shellharbour has encroached around a sizable farm dam (Figure 5) designed and built by the Water Resources Commission in the 1950's. The New South Wales Dam Safety Committee required a Dam Break Analysis to be undertaken to evaluate the potential risk to down stream urban areas. (Figure 6 shows a view from the Dam over the areas down stream as seen in ANUGA).

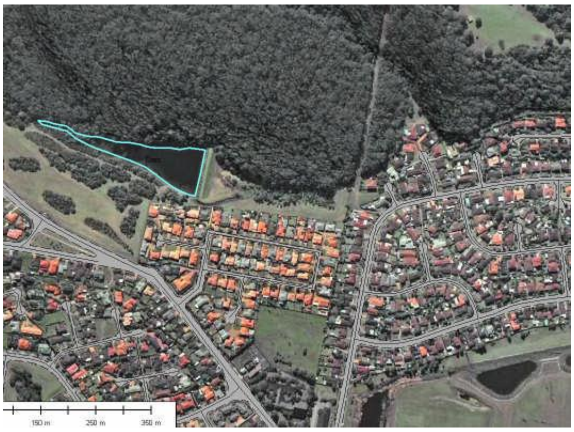

Figure 5 Dam Location

Figure 6 3D- ANUGA Domain View

As with the RURAL example in 3 above this was seen as a good opportunity to trial ANUGA as there were two other 2D models already in existence. The previous models were:

- o A fixed Grid Model Finite Volume Model (Flo-2d) (Refer Figure 7)
- o A Unstructured Grid Finite Difference Model (River-2d) (Refer Figure 8)

Figure 7 Flo-2d Structured Grid l (2x2m)

Figure 8 River-2D Unstructured Grid

In addition there had also been extensive HEC-RAS modelling completed of the site and down stream areas. Finally, as with the rural example, high resolution ALS (LIDAR) data was available covering both the ground surface and buildings. This was important, as there was a specific interest in the influence of the buildings on flow behaviour and more specifically their impact on the definition of hazard. The ALS data set was made up of around 700,000 data points. The data set was imported into the model via its in built Python utilities. It was a relatively simple task to create a highly variable mesh (that was reshaped several times). The mesh was left quite sparse on the boundary of the model extent and refined to very, very fine between buildings. Several polygons were used to specify the resolution applicable to various areas of the model. Ultimately it may be optimised further, to create polygons around the perimeter of each individual building, as there are utilities within the ANUGA toolkit that may allow this to be achieved quite simply.

The Dam Break hydrograph was developed using a parametric method developed by Froelich(1995 and 1987). The outcome of the parametric approach was used to adjust parameters in the hydrologic model to create the hydrograph to simulate the Froelich procedure. The WBNM (Boyd Rigby & VanDrie (1999 & 2007) hydrology model was chosen for its ease of use, its ability to model erodible embankments and thereby its ability to reproduce the required hydrograph shape. Hydrographs were developed for various scenarios including 'Sunny Day' failure, and failure during various events including the PMF event.

Trials included using local hydrographs to model local rainfall, but also the ANUGA source code was adapted to model rainfall falling directly on the computational domain. This addition to the code was a relatively straightforward process. Currently rainfall is limited to a single temporal pattern, however it is planned to add spatial variability in the near future. A relatively simple culvert routine was also added with relative ease. This routine is currently being updated to include momentum transfer, to better model the outflow jet from the culvert.

The overall results compared very favourably with the other models in terms of flood level and flood extent (Figures 9 & 10). ANUGA provided finer detail of shock waves resulting from the flood wave interacting with buildings (Figure 11). ANUGA also provides the ability to export any calculated quantity from the conserved quantities (such as Froude or VxD). These can be directly created as an exported DEM making it easy to develop map overlays for GIS. One outcome of using the very fine mesh around buildings was the identification of raised hazard around the buildings due to the locally increased velocities and in some places, depth.

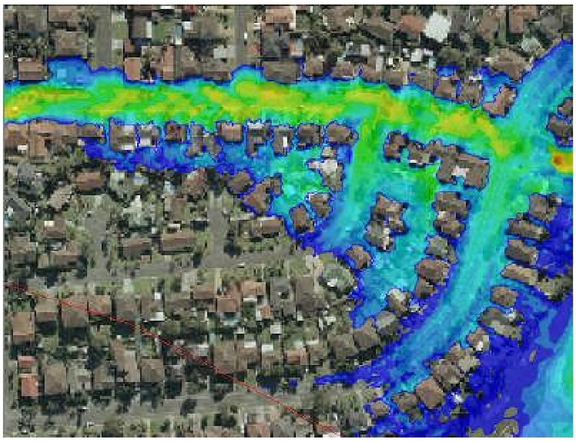

Figure 9 Comparison Result Flood Extent

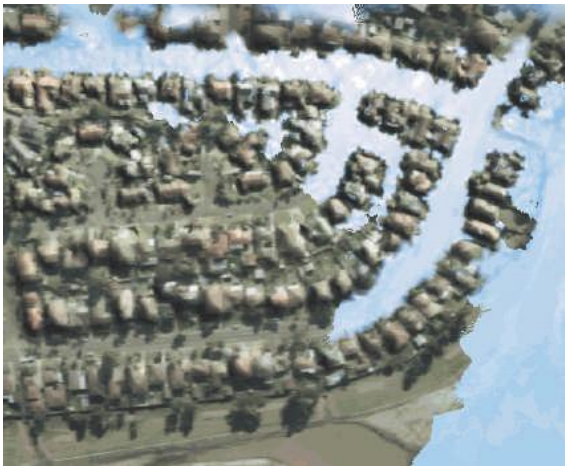

Figure 10 ANUGA Model Flood Extent

4.2 Strengths

In summary, the model strengths apparent in this trial included;

- Extremely stable
- Once familiar with the environment, very easy and quick to establish and alter models
- Flexible import tools and mesh generation tools (utilities)
- Able to model flow between buildings in considerable detail
- Very good 3D visualisation possible with Air Photo Drape
- Flow Behaviour appears more realistic than other models
- Clean, smooth visualisation of flow including shock waves, reflective and moving hydraulic jumps (Figure 11)
- Potential to easily extend capabilities (as shown in Figure 12)
- Source code available to help understand the underlying algorithms

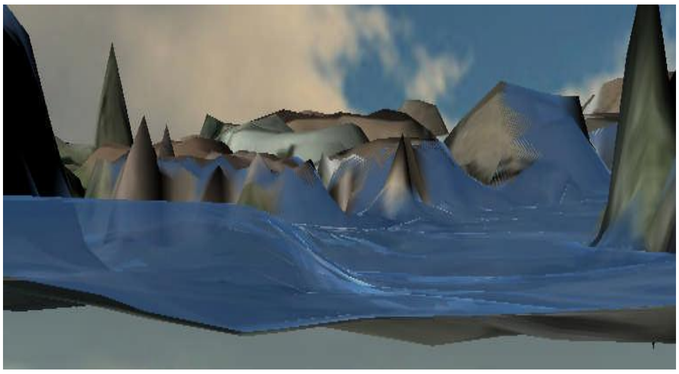

Figure 11 Reflective Shock wave

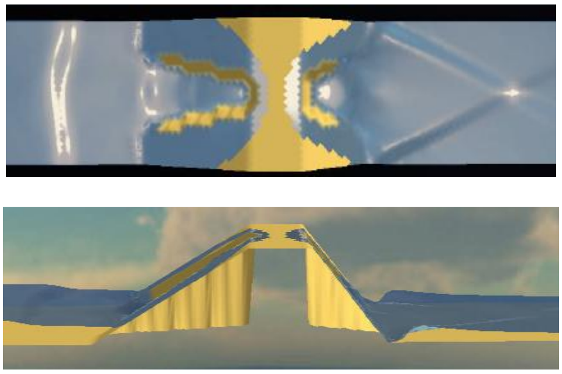

Figure 12 Plan & Elevation of Culvert Trial

4.3 Weaknesses

- No spatially varying rainfall yet
- Not yet straight forward to apply Roughness Variation

4.4 The Future

The author of this segment of the paper was able to develop additional code to enable the application of rainfall over the domain, and to develop a simple culvert routine. The following topics are currently being considered for possible future code enhancements.

- A range of options for modelling bridges culverts and piped drainage systems
- The ability to model sedimentation and erosion/ deposition processes
- The ability to simulate various Dam Break Mechanisms (Piping failure etc.)
- An adaptive mesh

A model of the Penrith Lakes White Water Stadium is currently under development as a validation model for flow involving frequent flow transitions, shocks, reflections and rapid variation in flood surface levels. With detail survey of the constructed channel readily available, good control over flow rates and ability to record the flood surface at different discharges, it is expected that the white water course will provide an excellent means of testing most of ANUGA's capabilities.

5. MODEL AVAILABILITY

ANUGA is available as a free download from https://sourceforge.net/projects/anuga/ . As the code is under continuous development users should visit the site at regular intervals to confirm they are operating with the latest code.

6. CONCLUSIONS

Based on our trials of ANUGA as a flood modelling tool, we conclude that ANUGA;

- Is a powerful and highly flexible hydrodynamic model
- Is not a model (yet) for modellers with limited programming experience
- Is well able to simulate flooding where structures are not present
- Will grow rapidly in its capabilities and user friendliness over the next year
- Is well worth investigation by anyone with an interest in flood modelling

7. ACKNOWLEDGEMENTS

The Authors acknowledge with thanks, the considerable assistance provided by Dr Ole Nielsen of Geoscience Australia, in the development of the trial models and preparation of this paper.

8. REFERENCES

Geoscience Australia and the Australian National University (2007), *ANUGA v1.0 User Manual,* https://downloads.sourceforge.net/anuga

Boyd M. J., Rigby E.H. and Van Drie R. (1999), *Modelling urban catchments with WBNM2000*, Institution of Engineers Australia, Water 99 Conference Brisbane July 1999. www.uow.edu.au/eng/cme/research/**wbnm**.html

Boyd M. J., Rigby E.H. and Van Drie R. (2007) *WBNM User Manual*, www.uow.edu.au/eng/cme/research/**wbnm**.html

Froehlich, David C., (1987), *Embankment-Dam Breach Parameters*, Proceedings of the 1987 ASCE National Conference on Hydraulic Engineering, Williamsburg, Virginia, August 3-7, 1987, p. 570-575.

Froehlich, David C., (1995a), *Peak Outflow from Breached Embankment Dam*, *Journal of Water Resources Planning and Management*, vol. 121, no. 1, p. 90-97.

Froehlich, David C., (1995b), *Embankment Dam Breach Parameters Revisited*, Proceedings of the 1995 ASCE Conference on Water Resources Engineering, San Antonio, Texas, August 14-18, 1995, p. 887-891.

Kurganov, A.S. Noelle and G. Petrova (2001) *Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton-Jacob equations*, SIAM Journal of Scientific Computing, 23(3), 707-740

Nielsen 0, Roberts S, Gray D, McPherson A and Hitchman A (2005) *Hydrodynamic modelling of coastal inundation,* MODSIM 2005 International Congress on Modelling and Simulation, Modelling and Simulation Society of Australia & New Zealand, 518-523, www.mssanz.org.au/modsim05/papers/nielsen.pdf

Toro E., (1992), *Riemann Problems and the WAF Method for Solving the Two-dimensional Shallow Water Equations,* Philosophical Transactions of the Royal Society, Series A, 338: 43-68

Zoppou C and Roberts S., (1999), *Catastrophic Collapse of Water Supply Reservoirs in Urban Areas,* ASCE J. Hydraulic Engineering 125(7), 686-695