ParCFD'2020
32nd International Conference on Parallel Computational Fluid Dynamics
May 11-13 2020, Nice FRANCE

A PARALLEL ADAPTIVE MESH REFINEMENT LIBRARY AND ITS APPLICATION TO REACTING FLOWS

JIAN FANG[*], UMAIR AHMED[†], JUDICAEL GRASSET[*], CHARLES MOULINEC[*], NILANJAN CHAKRABORTY[†], DAVID R. EMERSON[*] AND R. STEWART CANT[‡]

[*] STFC Daresbury Laboratory
Scientic Computing Department
Warrington, WA4 4AD, UK
e-mail: jian.fang@stfc.ac.uk

[†] Newcastle University,
School of Mechanical and Systems Engineering
Newcastle NE1 7RU, UK
Email: umair.ahmed@newcastle.ac.uk

[‡] University of Cambridge
Department of Engineering
Cambridge CB2 1PZ, UK
Email: stewart.cant@eng.cam.ac.uk

Key words: Adaptive mesh refinement, Space filling curve, Morton code, CFD, reacting flow

Combustion simulations require resolution of the thin reaction layer within the flame where high magnitudes of temperature and species gradients exist. Typically, for simplified chemistry, 10 to 20 mesh points are kept within the flame thickness in order to resolve key thermo-chemical processes within the flame. The mesh resolution requirements become increasingly stringent with increasing complexity of the chemical mechanism [1]. These requirements are more challenging in turbulent reacting flows because the smallest length scales of turbulence need to be resolved in addition to the flame thickness. The need for spatial resolution is exacerbated further in cases of flame-wall interaction (FWI) and localised ignition, especially at elevated pressure because the flame thickness decreases with increasing pressure [2]. In order to carry out a detailed simulation of a reacting flow within a reasonable computational cost, a dynamic adaptive mesh refinement (AMR) library has been recently developed and applied to a CFD solver for reacting flows known as the HAMISH code. The AMR technology enables fine cells to be used only where required, e.g. near the wall and within the flame.

The numerical framework of HAMISH relies on a finite-volume approach for spatial discretisation together with a Runge-Kutta algorithm for time stepping. An unstructured Cartesian mesh allows for adaptive local refinement and de-refinement operations based on requirements prescribed by the numerical solution. The data structure is built by representing the unstructured mesh as a bitree/quadtree/octree (in 1-D, 2-D and 3-D respectively). Spanning the tree with the help of the Morton-ordered space filling curve then allows efficient cell addressing and parallel domain decomposition. Within HAMISH, a Morton code is defined with 18 possible levels together with a level descriptor, as shown in Figure 1. The encoding of each level is represented as a 3-bit binary number or octet, and the level descriptor is defined as a 6-bit binary number. An 8-byte integer number is used to contain a Morton code, for example, the Morton code in Figure 1 is represented as the decimal integer number 1080863910571494l6.

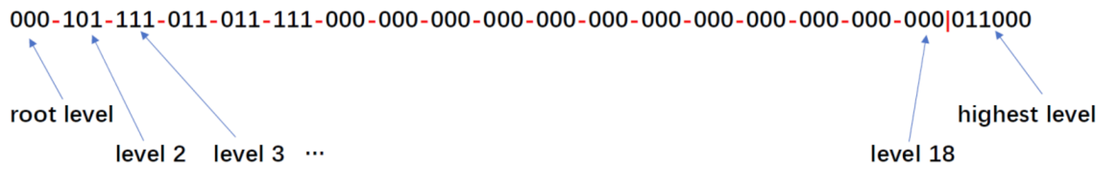

Figure 1: Example of a binary Morton code defined in the HAMISH code

By using an array of Morton codes, an AMR mesh can be generated. A sketch of the Morton encoding of a 2-D three-level mesh is presented in Figure 2, from which we can see that the highest level is used to determine the size of a cell, and encoding defines the spatial coordinates of a cell, i.e. the i,j,k integer coordinate of a cell, which can be efficiently obtained using binary operations.

CELL	M-code
1	00-00-00|01
2	01-00-00|01
3	10-00-00|01
4	11-00-00|10
5	11-01-00|10
6	11-10-00|10
7	11-11-00|11
8	11-11-01|11
9	11-11-10|11
10	11-11-11|11

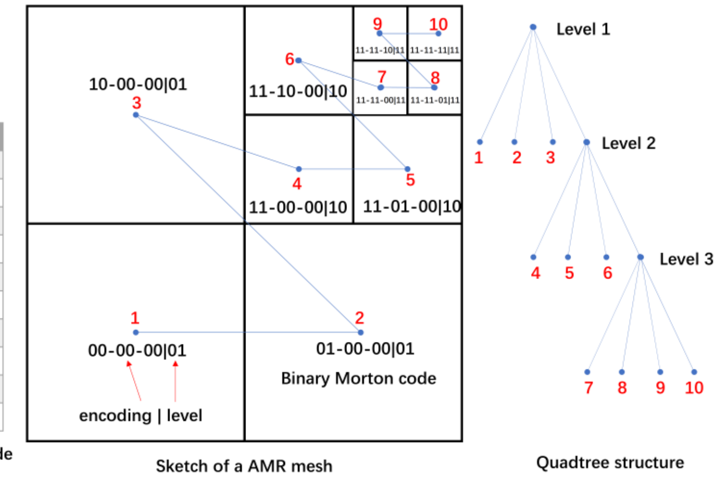

Figure 2: Sketch of a 2-D AMR mesh and quadtree structure generated using an array of Morton codes.

The Morton code array can be easily interpreted as a bitree/quadtree/octree data structure, as shown in Figure 2, and the cells in the tree can be connected by a searching procedure based on following the path defined by the octets within each Morton code. With information about the cells' connectivity, the finite-volume stencil can be defined, the numerical fluxes can be calculated, and the equations discretised on the mesh can be solved. The Morton codes are stored in memory according to their binary sequence, which traces out a Z-shape as indicated in Figure 2. The mesh refinement/derefinement procedure is handled based on an appropriate length scale dictated by the physical nature of the flow problem. The mesh refinement is enacted by splitting a cubic parent cell of side h into eight cubic child cells of side $h/2$. By contrast, derefinement is carried out when 8 child cells of side $h/2$ combine to form a parent cell with side h. The refinement/derefinement is first conducted on the Morton code by adjusting the level of the Morton code by 1, with new cell addresses provided in the octet at the highest level.

The AMR library has been applied to solve the Navier-Stokes equations with species transport equation and a detailed chemical reaction mechanism. The HAMISH code is tested for several benchmark cases, including, (a) Simulation of 1-D planar laminar premixed flames to demonstrate the capability of AMR in capturing the sharp gradients within the flame (shown in Figure 3); (b) 1-D head-on quenching of laminar premixed flames to demonstrate the capability of HAMISH in dealing with reacting flows in the presence of a wall; (c) 2-D Rayleigh–Taylor instability problem to show the ability of AMR in tracking an interface; (d) 2-D laminar channel flow to show the ability to refine the mesh in the boundary layer next to the wall (shown in Figure 4); (e) 3-D non-reacting Taylor-Green vortex which demonstrates the capability to simulate accurately the vortical motion typical of turbulent flows (shown in Figure 5); (f) 3-D premixed turbulent flame propagation under isotropic homogeneous decaying turbulence to show that this code can be used for Direct Numerical Simulations of turbulent reacting flows in canonical configurations.

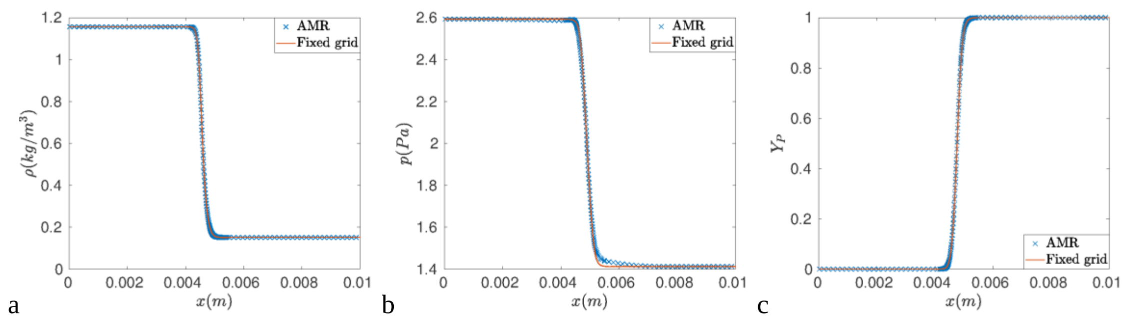

Figure 3: The density (a), pressure (b) and mass fraction of products (c) of the 1-D planar flame.

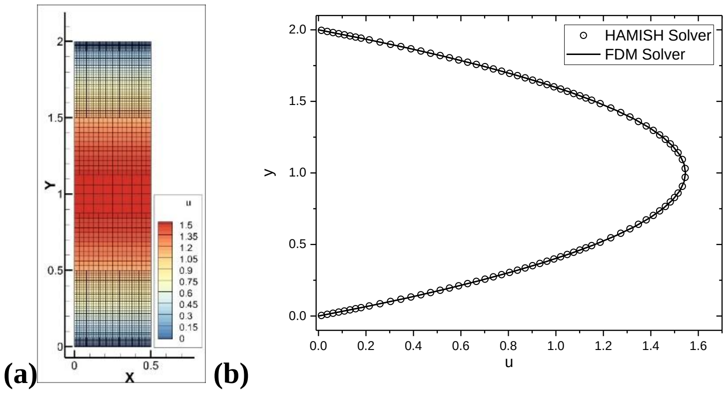

Figure 4: The mesh and velocity profile of the 2-D laminar channel flow

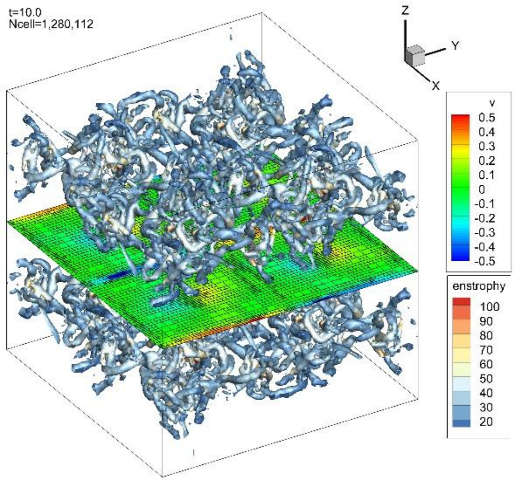

Figure 5: Instantaneous coherent structures in the 3-D Taylor-Green vortex case

REFERENCES

[1] T. Poinsot and D. Veynante, *Theoretical and Numerical Combustion*, R.T.Edwards Inc., Philadelphia, USA, (2001).

[2] S.R. Turns, *An introduction to combustion: concepts and applications*, 3rd Edition, McGraw-Hill, Hong Kong, (2011).