AUTOMATIC NESTED REFINEMENT – A TECHNIQUE FOR THE GENERATION OF HIGH QUALITY MULTI-BLOCK STRUCTURED GRIDS FOR MULTI-SCALE PROBLEMS USING GRIDPRO®

Krishnakumar Rajagopalan[1], Peter. R. Eiseman[2]

[1] *Program Development Company, White Plains, NY., U.S.A. krishna@gridpro.com*
[2] *Program Development Company, White Plains, NY., U.S.A. eiseman@gridpro.com*

ABSTRACT

A technique for the generation of conformal adaptive refinement of hex meshes is presented. Automatic Nested Refinement is a technique for generating recursively nested topology automatically. It can be applied inside GridPro's topological paradigm to generate block structured grids which can resolve tiny features in the problem while providing for a smooth and consistent way to transition to a larger scale. The selection of self-similar, 'fractal-like' topological templates makes the selection of number of levels easy, while making the technique feasible for infinite levels of adaptive refinement. The technique has been programmed and integrated into the GUI of GridPro, making it very accessible and easy to use. This method illustrates a way of generating structured grids in an unstructured way, made possible because of the topological paradigm of GridPro. Because the technique uses the topological paradigm, it inherits all the advantages the paradigm offers, including the ability to handle very complex geometries, parametric variation of surfaces, and the ease of use, speed and quality of GridPro. The technique has been illustrated using a variety of applications. This method has proved to be a fast, efficient, automatic and reliable means to perform physical simulations that have a disparity in scale.

Keywords: mesh generation, multi-scale problems, multi-block grids, nested refinement, fractals, conformal adaptive hex meshing, GridPro®

1. INTRODUCTION

The term **multi-scale problems** are usually applied for computational analysis of geometries having widely varying length scales. For example, full-field oil reservoir simulations are usually multi-scale problems. For full field simulations of oil reservoirs, the scale of the field is usually of the order of miles, while the scales of pipes and other such flow regions are of the order of inches. Computational analyses of multi-scale problems are usually hard because of the need to generate a computational mesh which can resolve both the higher end and lower end of the scale of the geometry, and also taking care that the transition is smooth and consistent. At present, there are a variety of methods to generate meshes for such problems. They generally involve hybrid or completely unstructured methods, or structured methods with hanging nodes and other techniques.

Multi-scale problems usually require a different approach to grid generation. To accurately resolve the solution of the field near the smallest spatial scales, the size of the discretized spatial cells need to be of the order of these smallest scales. If the same size cells are used in the large-scale field too, then there will be too many cells at places where it is not needed, and this can cause the computational time of the numerical simulation to be too big for any practical purposes. The field near the small-scale geometries needs to be resolved with small cells, while the rest of the flow field should be discretized with cells of large size. These kinds of problems require special kinds of grids - grids which can transition from a small size near the small features to large cell sizes in the general field.

Multi-scale problems are common in reservoir simulations. But these problems can also occur in other applications such as the analysis of very thin turbine blades, the analysis of flow and heat transfer about/inside thin pipes in biological flows etc. Also, as the need to produce more complete and reliable analysis grows, problems which are not usually multi-scale can become so. For example, the flow past an aircraft or a car is not usually a multi-scale problem. The study of fluid flow around a small appendage such as an antenna is usually considered redundant. But cases can occur where such analysis might be useful and needed, and in such cases, a reliable method to produce grids for such cases needs to be available. Other examples, like the analysis of the flow past a thin structure like a strand of hair, or the analysis of flow over riblets, etc can also be considered as multi-scale problems. Sometimes, such a multi-scale grid may be necessary to reduce the aspect ratio of the cells in certain locations. The modern CFD (computational fluid dynamics) algorithms are generally quite tolerant to high aspect ratios.

But there can be cases when low aspect ratio cells are desired in certain key locations.

In this paper we show a technique by which one can easily generate conformal multi-block grids for multi-scale problems. The basic idea was to stack up elementary topological elements in a certain way so as to handle scale geometries in grids. A recursive way of stacking up a single element was preferred because such a technique could be programmed and hence provide for an easy and automated way to handle multi-scale problems. Such a recursive or 'nested' structure is called a *nested refinement* in that region.

A program called ***nest*** was developed which takes in a certain input and gives a nested topology as output. The user then loads in this nested topology (as a file), and links it to his existing topology in a few mouse clicks. ***Nest*** operates in an abstract topological level only, and does not need to know anything about the actual surfaces. The grid generation engine in GridPro takes care of topology conforming to the actual surfaces. To make the utility more accessible, a button was added in the GridPro GUI which leads to a dialog box which runs the program to create the nested topology.

2. GRIDPRO'S TOPOLOGICAL PARADIGM

GridPro's topology paradigm is a very powerful and unique technique. This paradigm reduces the problem of generating multi-block grids to that of generating a set of loosely positioned topology corners and their connectivity. The advantages of such a technique are its ease of use and automation – because one does not have to worry about positioning the corners exactly. Also, topology can be used as a template and can be quite independent of the surfaces themselves. This gives rise to an important advantage that topology need be built only once for a grid generation problem. Another main advantage is the automation which can be achieved on the topology paradigm. **Nest** is one example of how this kind of automation can be achieved.

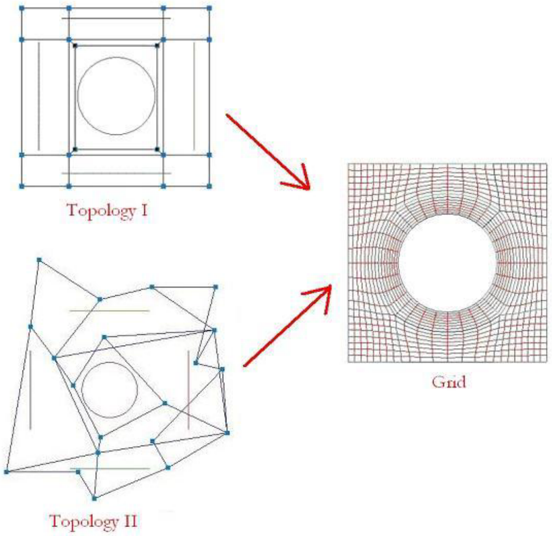


Figure 1. Loose positioning of topology corners

Figure 1 shows a simple example to illustrate how the topology paradigm works in GridPro. Both topology 1 and topology 2 produces almost identical grid point positions. The iterative grid optimization algorithms in GridPro are very robust and ensure the highest quality of the grid for a particular topology. Another example to illustrate the flexibility of topology is shown for a simple case below. The robust optimization algorithms can untwist a folded block as shown in figure 3.

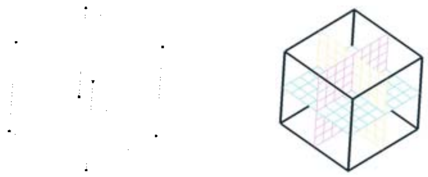
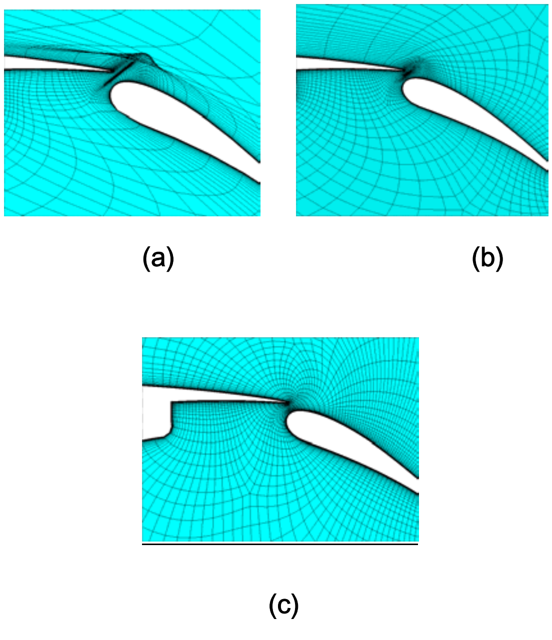


Figure 2. Topology and Corresponding GridPro Grid

Figure 3. Topology corners displaced, but it produces the same grid


(a) (b)

(c)

Figure 4. Grid Optimization in GridPro. Grid for iteration step - (a) 100 (b) 500 (c) 3000

Figure 4 illustrates the grid point movement algorithms by showing pictures of a grid at different iteration steps for the grid generation engine. Again, we see that the folded grid cells in (a) have been unfolded and neatly placed in (c). This illustrates the robustness of the GridPro's grid generation engine.

Now that we have a robust topology engine behind us, we have several possible options. One option is to develop schemes to automatically generate topology. Another is to develop an interactive topology generator. An interactive topology generator called the AZ manager comes with GridPro which has numerous features for automating small parts of topology creation. This provides a kind of semi-automatic approach to topology creation. The fully automatic approach to topology creation is a present research topic in Program Development Company. For more details on topology creation and other aspects of grid generation in GridPro, please refer to the GridPro TIL manual [1] and the GridPro tutorials [2].

3. THE NEED FOR NESTED REFINEMENT

Consider the geometry as shown in figure 5(a) or figure 8. There are many ways we can go about generating a grid for such geometries. But suppose we set quality criteria on the grid. – Set the worst aspect ratio of the grid to be less than 10, and the grid should be smooth etc. A simple way is to just build a structured grid with a lot of grid points to capture the small features in the geometry. If we take this strategy, then the grid in figure 5(a) will have about a million cells. In figure 8, the radius of the tiny tube is about 0.005, and the box dimensions are of the order 10*10*10. This means that a simplistic strategy like the one described above will yield a grid having more than 10 billion cells! Even for such simple geometries, the problem starts becoming intractable using simple methods.

Traditionally, because of the difficulties in generating tractable structured grids for multi-scale problems, triangular/tetrahedral or hybrid meshes, meshes with hanging nodes etc have been the only grids used for such problems. Automatic nested refinement offers to fill in for a structured method which can do multi-scale problems in a reliable and automatic way, offering the advantages of an unstructured approach but a locally structured grid at the same time.

4. AUTOMATIC NESTED REFINEMENT

The basic idea behind nested refinement is simple and has been around for sometime. There are many ways of generating topology to get such grids in GridPro. Some references can be quoted from the GridPro TIL manual [1], where a nested refinement structure has been used to illustrate the idea of components in the topology input language (TIL). Nested refinement can be looked at as a generalization of a structure called ***the clamp***[3].

The following pictures illustrate the idea of a self-similar topological structure which has a high resolution in the bottom and less in the top. A self-similar topological template is chosen and recursively stacked to make such a structure. As mentioned, this process is completely automatic once the top and the bottom surfaces have been specified.

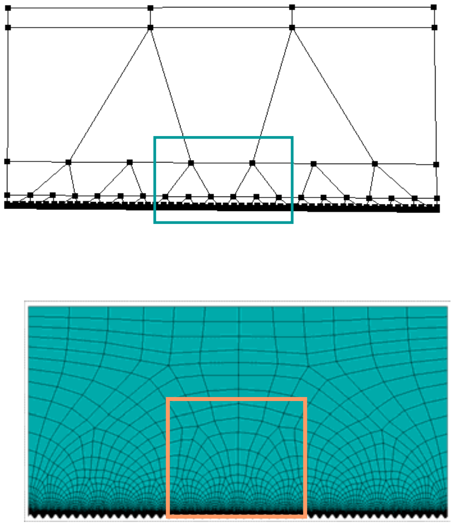


Figure 5(a). Nested Topology and Grid

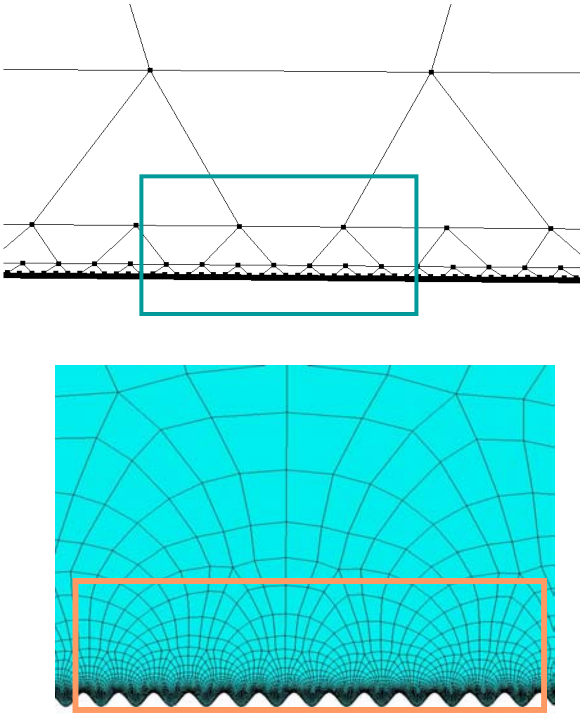


Figure 5(b). Nested Topology and Grid – zooming in to the boxes in 5(a)

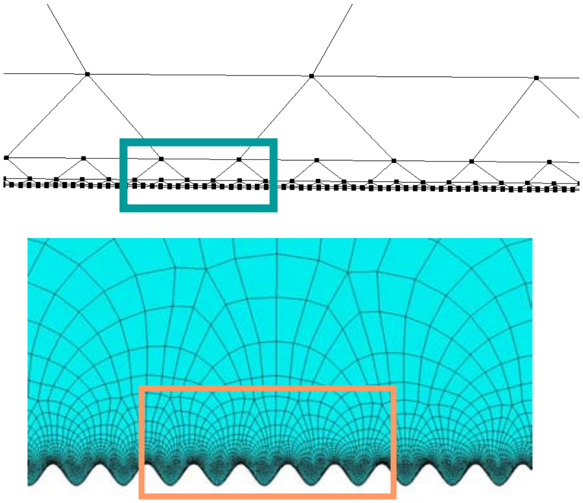


Figure 5(c). Nested Topology and Grid – zooming in to the boxes in 5(b)

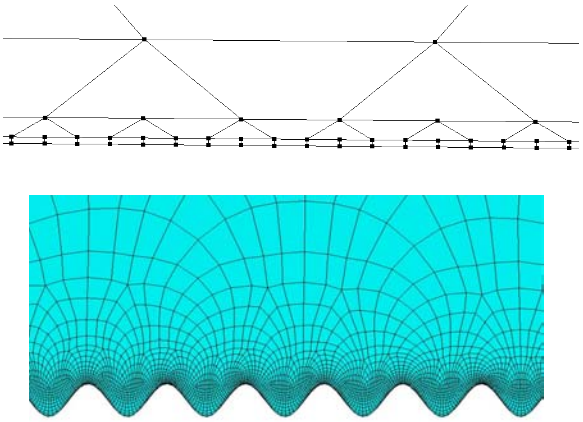


Figure 5(d). Nested Topology and Grid – zooming in to the boxes in 5(c)

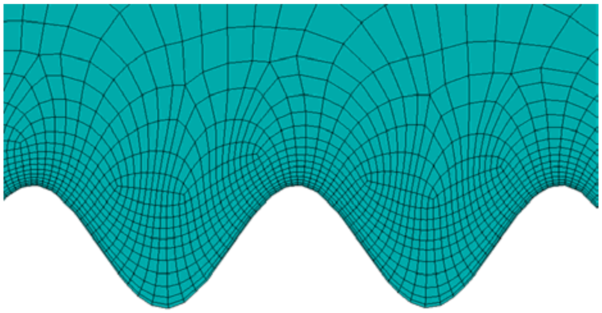


Figure 5(e) Grid near the wall

Figures 5a, 5b, 5c, 5d illustrate how nested refinement can be applied in a topological sense and how this reflects in a grid for the geometry shown. The geometry consists of a regular two dimensional box with tiny sinusoidal waves covering the bottom. In this case, there are 50 sinusoidal waves covering the bottom. For a good resolution, one wave needs to be captured with about 40 grid cells. This means we need about 2000 grid cells in the bottom surface. In the grid shown, there are more than 3000 grid cells in the bottom, while there are only 12 in the top. There are 5 levels of nested refinement in this case. The construction of a self-similar structure makes it possible to produce a grid which expands smoothly out into a low resolution region. Also, the optimization algorithms in GridPro make it possible to get a high quality grid throughout.

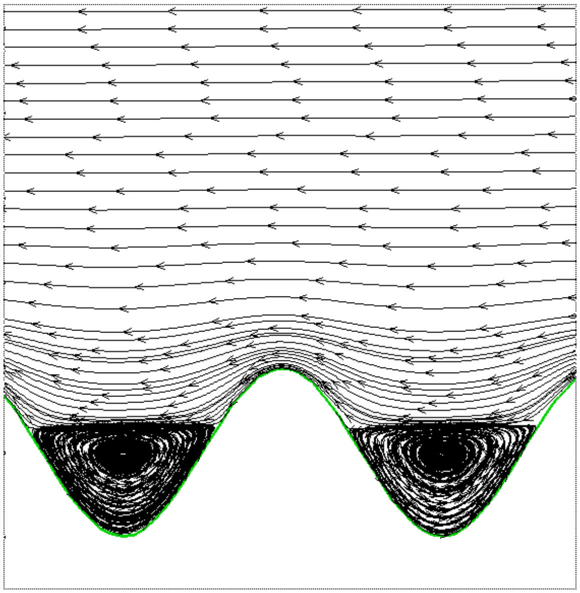


Figure 5(f) Flow field solutions

Figure 5(f) shows flow field solution for the wavy wall problem computed using Fluent®. Comparisons between flow field solutions using the nested grid (shown in figures 5a through 5e) and a overpopulated grid on the same geometry shows that the nested grid runs much faster but gives the same results. For more details on this comparative simulation, refer to [1].

This concept can be easily extended into three dimensions. **Nest** does this by simply extruding a two dimensional topology. **Nest** can be applied consecutively in two transverse directions to get a true three dimensional nested refinement.

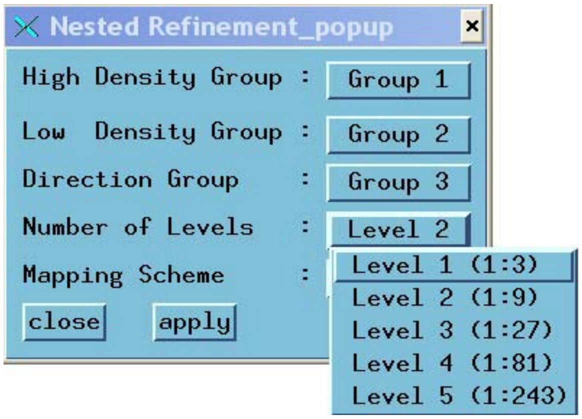
5. IMPLEMENTATION OF NEST

Nest can be accessed from the AZ manager, the Graphical User Interface of GridPro. Clicking on the **nest** button will pop up a dialog box like the one shown in figure 6.

In GridPro, arbitrary groups of topology corners can be identified as a group. At the time of this writing, there can be 32 such separate groups of corners. The first line in the dialog box in figure 6 specifies the group of corners which identifies the region of the grid where a high resolution is required. Similarly, lines 2 and 3, called the low density group number and direction group numbers respectively, specifies the group of corners which identifies the low density region and a special group called the direction group. The direction group is necessary because the nested refinement is done in one of two possible directions at a time, and the direction group of corners identifies the direction.

The number of levels is a crucial number which needs to be chosen with care by the user. This is the number of times the self-similar structure is sub-divided. For now, only 5 groups have been given in the dialog box. But the user can nest up to arbitrarily many levels of nested refinement using the program from the command line. In the limit of infinitely many recursions, it will converge to a fractal structure.

The ratios specified alongside the number give the overall cell size/resolution ratio which will be achieved between the high and low density regions. For example, in the case of the wavy wall shown in figure 5, the number of cells required in the top is 12 and the number of cells we need in the bottom is around 3000. This threshold ratio is close to 200, and hence, the number of levels can be chosen to be 5, which will give an average ratio of 243. The ratios go as $1:3^n$, where n is the number of levels of nested refinement. This can be mathematically proved [1].


Nested Refinement_popup

High Density Group : Group 1

Low Density Group : Group 2

Direction Group : Group 3

Number of Levels : Level 2

Mapping Scheme :

Figure 6. Nested Refinement dialog box

6. APPLICATION OF NEST TO SOME GEOMETRIES

6.1 Island of Oahu, Hawaii

Nest has been used to obtain a grid around the Hawaiian island of Oahu. This is a typical example of a multi-scale problem. The coast line of the island is crinkly and this typically requires a large number of cells close to the island. But the cell size far away from the island (in the far field – the ocean) can be the size of the island itself. The following pictures show the nested grid around Oahu.

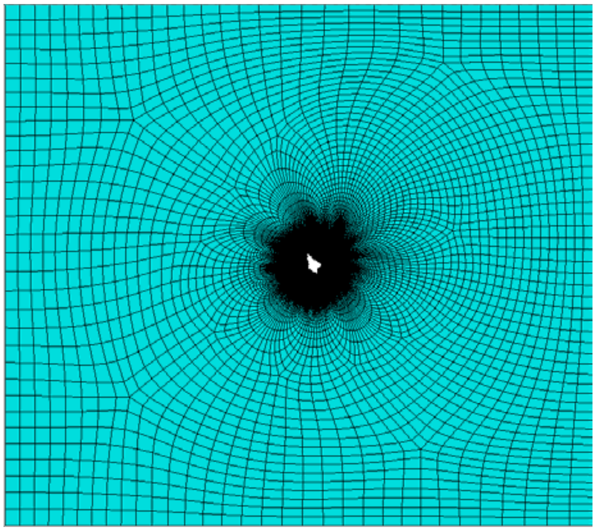


Figure 7(a). Oahu, Hawaii. The complete grid.

In figure 7(a), the island is the tiny white dot in the middle. As we can see the size of the cells far away from island is of the order of the size of the island itself.

Figure 7(b) shows a closer view of the grid where one can note how nested refinement helps in reducing the propagation of many grid lines by redirecting many of them back into the island.

Figure 7(c) shows the grid very close to the island. This is a structured grid which lies inside a cocoon of nested refinement. Zooming in closer the section shown by a rectangle figure 7(c), we note a smooth high resolution grid capturing the coastline of Oahu.

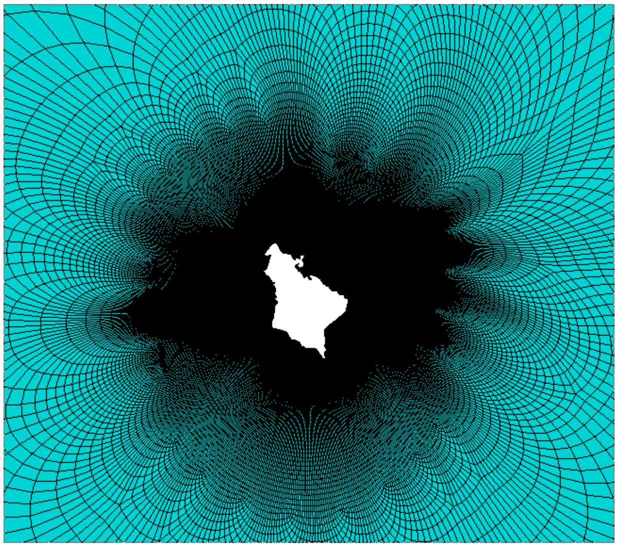


Figure 7(b). Oahu, Hawaii. Zooming in closer…

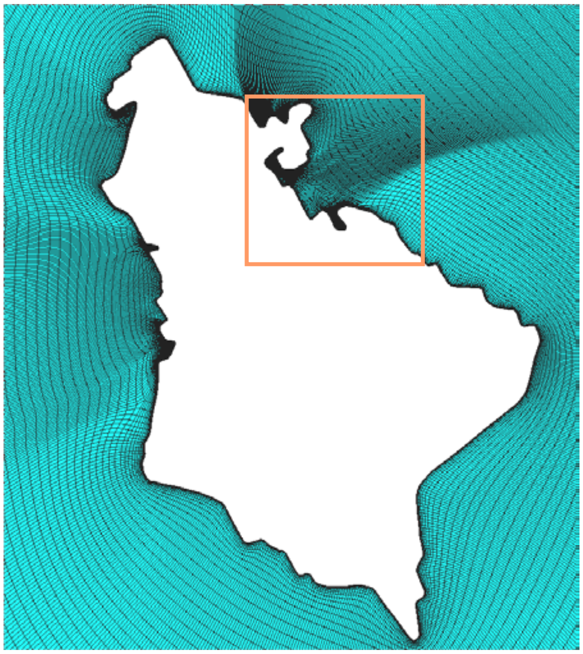


Figure 7(c). Oahu, Hawaii. Zooming in closer…

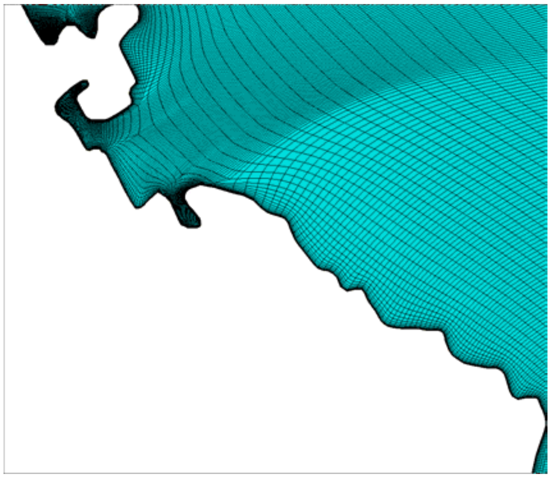


Figure 7(d). Oahu, Hawaii. Zooming in closer…

Figure 7(e) shows a close up of the top right section of the grid in figure 7(d).

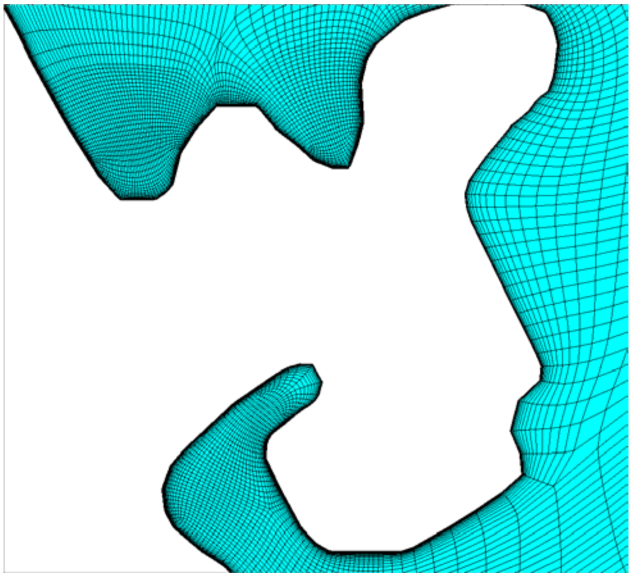


Figure 7(e). Oahu, Hawaii. Zooming in closer…

6.2 Curved Wavy Tube in a Box

A thin curved tube in a box whose radius fluctuates by a small amount is another example of a multi-scale problem. Figure 8 illustrates such a geometry. The arrows indicate a flow direction, and the problem posed can be that of calculating the drag experienced by the tiny tube by such a flow.

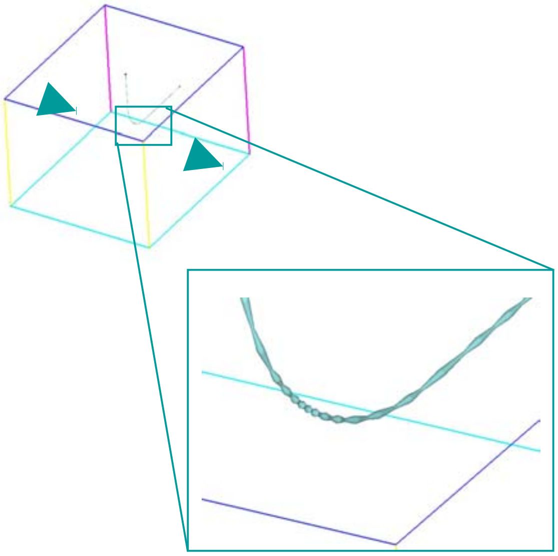


Figure 8. Curved wavy tube in a box

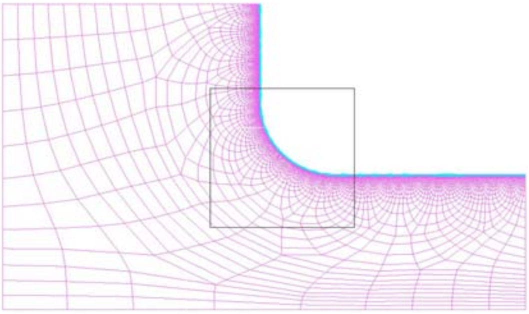


Figure 9(a). Cross-sectional view of a sheet

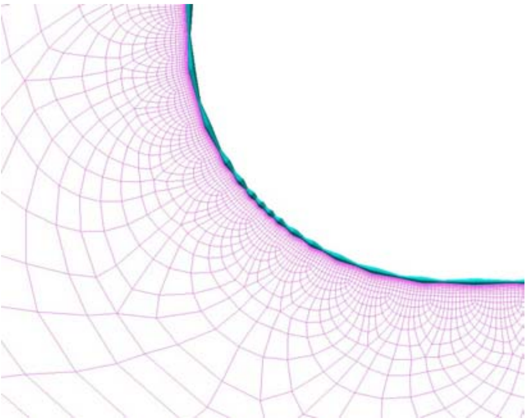


Figure 9(b). Cross-sectional view zoomed in

Figures 9a and 9b show a cross-sectional grid sheet for the grid generated on this geometry using **nest.**

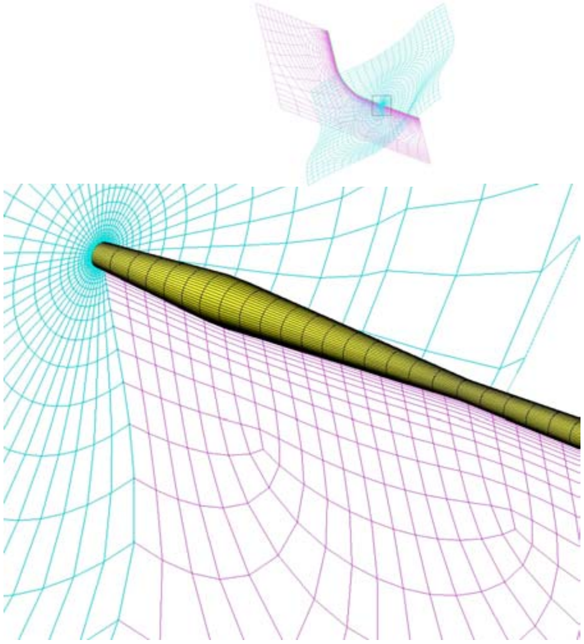


Figure 10 Two cross sectional sheets

Figure 10 shows two cross-sectional sheets perpendicular to each other. This illustrates the resolution of the grid near the wavy tube. Nest achieves the high resolution near the tube and the gentle transition to a lower resolution as one moves away from the tube.

Figure 11. Streamlines of the flow past the tube

Figure 11 shows the results from a simulation performed using Fluent®. Further details of the simulation can be obtained by referring to [1].

7. CONCLUSION

A technique called 'automatic nested refinement' was implemented in the framework of GridPro for addressing multi-scale problems in the context of multi-block structured grids. It offers automation and great flexibility in handling scale differences for grid generation. Future work will involve developing topological structures which will provide for more optimized adaptation and more automation. One means to do this is to use three dimensional templates in a recursive way to do such adaptation [4], [8]. Verification and validation of the use of nested refinement by comparing performance of nested grids with other hybrid and unstructured grid methods will also be an important future direction.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the helpful interactions with Prof. Houston Wood, Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, Virginia.

REFERENCES

[1] Automatic Nested Refinement - – A Technique for the generation of high quality Multi-Block structured grids for Multi-Scale problems using GridPro® - Master's Thesis for Krishnakumar Rajagopalan presented to the University of Virginia, May 2003

[2] GridPro TIL Manual – Program Development Company, White Plains, NY

[3] Parallel Multi-Block Structured Grids – Jochem Hauser, Peter Eiseman, Yang Xia, Zheming Cheng- Chapter 12, Handbook of Grid Generation, CRC Press.

[4] Robert Schneider - Quadrilateral and Hexahedral Meshes, Chapter 21, Handbook of Grid Generation [3]

[5] Peter. R. Eiseman, GridPro and the Topology paradigm, article in SGI world, June 1999

[6] Getting a Grip on Grid generation, Article in NASA SpinOff magazine, 40th anniversary publication, 2002.

[7] S. Balaven, C. Bennis, J.D. Boissonnat, S. Sarda, Generation of Hybrid Grids using Power Diagrams, Proceedings of the 7th international conference on Numerical Grid Generation in Computational Field Simulations, 2000

[8] Ko-Foa Tchon, Julien Dompierre and Ricardo Camarero, Conformal Refinement of All-Quadrilateral and All-Hexahedral Meshes According to an Anisotropic Metric, Proceedings of the 11th international Meshing Roundtable, 2002.