TREATING THE GEOMETRY OF A FLOATING CAISSON TO OBTAIN A STRUCTURED MESH IN GiD

Benjamín Suárez[1], Antonio Foces[2], José Pereda[2] & Mariano Cacho[2]

[1] *International Center for Numerical Methods in Engineering (CIMNE), Edificio C-1, Campus Norte UPC, C/ Gran Capitán S/N, 08034 Barcelona, Spain.*
[2] *Departamento de Resistencia de Materiales de la Universidad de Valladolid, C/ Paseo del Cauce S/N, 47011 Valladolid, Spain.*

SUMMARY: An algorithm for the generation of the geometry of floating multicellular caissons used in ports and harbours facilities is shown. Data input to analyze these caissons with finite elements using structured meshing in the GiD program is generated. Once the basic dimensions that define the geometry of any caisson (length, breadth, depth, slab extents, diameter and degree of bevelled edges of the cells, thickness of throat and walls) are known, the algorithm divides the whole structure in appropriate small hexahedral volumes and generates the input data file. Afterwards, the GiD program makes the structured meshing of these volumes, obtaining therefore an accurate mesh of elements. Also a considerable saving of time in the generation of the mesh within the stage of analysis in the design process of this type of structures is attained.

KEYWORDS: Floating multicellular caisson, structured meshing, java-algorithm, object-oriented programming.

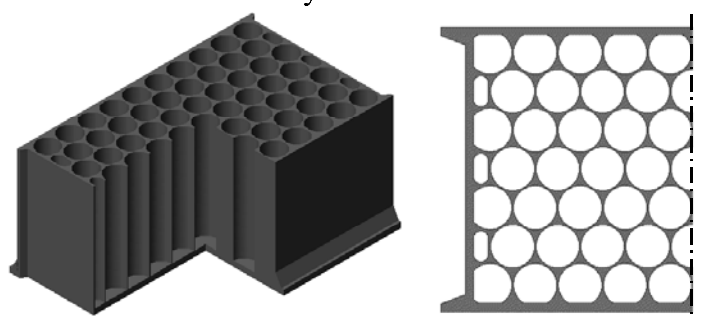
INTRODUCTION

A fast construction of docks may be achieved with the use of floating caissons, which are prefabricated concrete box-like elements drilled with cylinder cavities or cells. Floating box caissons are manufactured upon a floating platform equipped with synchro lift for partially submerging the concrete structure while at the same time erecting it. Then it can be towed out to the final location, where it is then sunk by filling it with sand, gravel, or concrete. The different marine works and harbour constructions in which these caissons can be used include: ports, breakwaters, wharves, berthing facilities and docks, dry docks and slipways.

Description of the typical caisson geometry

In Figure 1.a a cut of a three-dimensional view of a typical floating caisson is shown. The caisson is usually formed by two basic elements. The first one, the slab, is a flat and massive element, with a depth of 0.6 to 1.5 meters that closes the structure bottom. The second one, the shaft, is a parallelepiped block strongly lightened by a series of circular cells arranged properly to gain the larger possible flotation, maintaining suitable values of resistance and stiffness. The usual dimensions of the caisson oscillate between 20 and 30 m of length, 15 and 25 m of breadth (wide) and 12 to 20 m of draft (height).

In Figure 1.b a half of a generic cross-sectional section of the shaft is shown. The circular cells are disposed in a staggered formation to maximize the lightened surface. In the proximities of the lateral walls, where the use of circular cells would create problems of execution and excessive stress concentration in incoming angles, ovoidal cells are used (formed by a rectangle and two half circles). Also, when both the outer dimensions and the frontal and lateral walls thickness are fixed, sometimes it is necessary to bevel the cells next to these walls.



a. Cut from the caisson 3D view. b. Cross-sectional shaft section

Figure 1. Generic views of a typical floating caisson

Proposed algorithm and regions types that define the cross-sectional section

When trying to analyse the resistant behaviour of the caisson to diverse loads by means of the Finite Elements Method, it is necessary to make the meshing of its volume. Geometry is perfectly well-known, reason why a mesh generating program (as GiD) could easily achieve a mesh using tetrahedral elements. Nevertheless, given the peculiar characteristics of the shaft section, the resulting unstructured mesh would not be accurate and would contain a high number of elements.

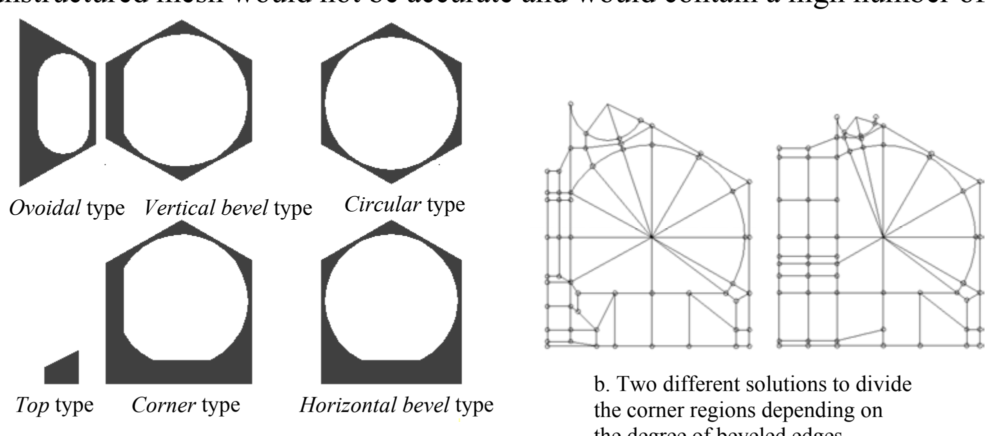
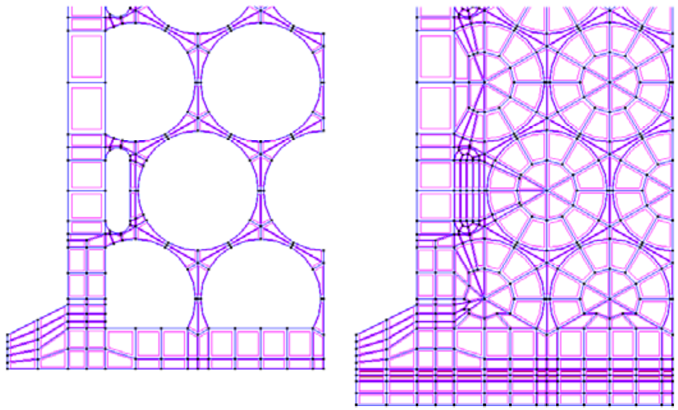


Figure 3. Basic regions that define the cross-sectional shaft section

The proposed algorithm makes a previous geometry processing so that the structured meshing made in GiD gives a very accurate solution. Essentially, it divides the cross-sectional section in basic regions, grouping those of equal geometry in diverse typologies. Each type is treated separately, taking care of its peculiar characteristics. In Figure 3.a the basic regions in which the cross-sectional section of the shaft has been divided are shown.

Later, each region is divided in several zones of four sides (straight and/or curved), so when performing the structured meshing with hexahedra, GiD will perfectly locate these elements in these zones: it will only have to divide properly the zone to give a rough or fine mesh, as desired. In order to determine the zones, nodes must be located in the throats (zones of smaller thickness between adjacent cells), in which the stresses will be more critics. The possible variations of some parameters in each typology have been included in the algorithm, therefore allowing the treating of any degree of lateral and frontal bevelling (from 0 to 100%), symmetry, situation of the tops, etc. In Figure 3.b there is an example showing two possible situations, among others. Figure 4.a shows the result of applying the proposed algorithm to a shaft section.



a. Shaft section b. Slab section

Figure 4. Zonal division of the cross-sectional section

The algorithm allows to generate the points, lines, arcs, surfaces and volumes that define these zones, in such a way that the characteristic parameters that define these entities (coordinates, numeration, centres of gravity, etc) are written in a text file (with extension *.geo*) that can be read and processed directly by GiD.

In the massive prismatic slab, the meshing could be simpler, but to obtain the coupling of nodes and elements with those of the shaft, it is necessary to maintain the same meshing in the slab. Therefore, the same type of basic regions is kept, what implies the filling of the slab regions by making an inner division of the circles in diverse zones of four sides. The Figure 4.b shows the division made in the slab.

Figure 5 shows the meshed caisson by GiD with hexahedra elements. Lines have been divided in two, reason why eight hexahedra by each one of the volumes defined are generated.

Classes defined in the proposed algorithm

As the information that describes the caisson presents a clear hierarchic structure, the developed algorithm (programmed in the language JAVA) makes use of the philosophy of the object-oriented programming: An object is a container of the properties and the own variables as well as of the interrelations between the object and the others. In this way, diverse objects, which are instances of classes, are defined. Thus, the class Point contains variables like the numeration, coordinates and number of lines that connect it for a given point. The class Points_Collection groups a series of points that have something in common: they belong to the same basic region. In the same way, the class Line, contains the numeration and number of surfaces that connect it, and the classes initial Point and end Point that define a line. Following this scheme, Arc and

Surface classes are constructed with the respective collections of arcs and surfaces. The class Cell contains, for a given basic region, the different collections from points, lines, arcs and surfaces that define it. Finally, the Cells_Collection class contains all the basic regions that define a cross-sectional section. In the algorithm, two collections of cells are used, one for the shaft and one for the slab.

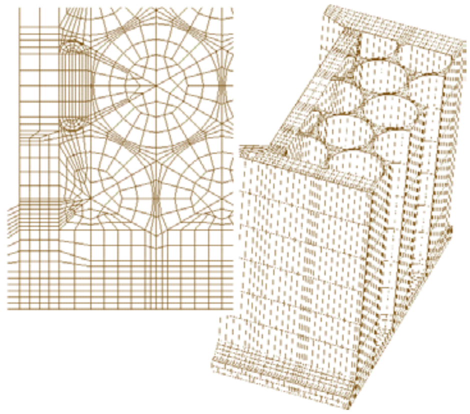


Figure 5. Meshed caisson by GiD

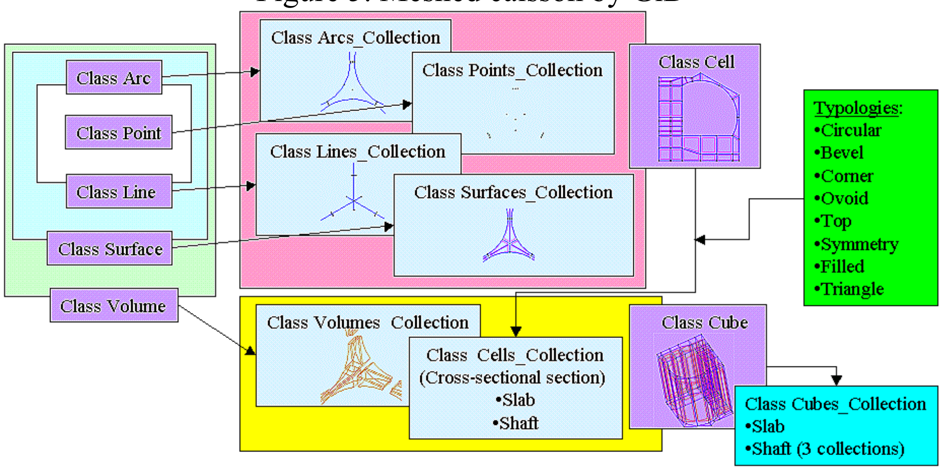


Figure 6. Scheme of classes used in the algorithm

Volume is the class defined from a surface of the collection of cells, the same one translated to a certain height, plus the four vertical resulting surfaces to connect both. The class Volumes_Collection contains all the volumes generated for a cross-sectional section. With the information stored in this collection and the Cells_Collection, the class Cube can be defined. It contains the points, lines, arcs, surfaces and volumes of a basic region. The class Cubes_Collection includes all the cubes between two cross-sectional sections. The algorithm uses one collection of cubes for the slab and up to four for the shaft, for the study of different sections at different heights, as well as to maintain a suitable factor of aspect in the element. Figure 6 shows schematically the relation between the different used classes.

References

1. Eric Comerma Piña, "Diseño automatizado de cajones multicelulares flotantes para obras porturarias", *Tesina de Especialidad*, Universidad Politécnica de Cataluña, Septiembre 1999.
2. Yoshimi Goda, "Random Seas and Design of Maritime Structures", World Scientific Publishing Co., 2000.