# Coupling Terrain and Building Database Information for Ray-Tracing Applications


Prof. Dr. Ing. F.P. Fontán
*University of Vigo,*
*Department of Signal Theory and Communications ETSE,*
*Vigo, Spain*
*fpfontan@tsc.uvigo.es*

Prof. Dr. Ing. O.E. Ruiz S.
*EAFIT University,*
*CAD/CAM/CAE Laboratory*
*Medellín – Colombia*
*oruiz@eafit.edu.co*

Student Assistant S. Peña S.
*EAFIT University,*
*CAD/CAM/CAE Laboratory*
*Medellín – Colombia*
*spenaser@eafit.edu.co*



## Abstract

*In this paper a methodology for the combination and integration into a single data base of terrain and building data is presented. This study is justified if ray-tracing techniques are to be used in propagation and channel modeling studies. Usually terrain is available in grid or elevation form while building information is normally facet-oriented. Ray-tracing (RT) techniques deal with flat facets and straight edges, if possible in triangular format. To allow the use of RT on urban areas over irregular terrain a common format made up of facets and edges is therefore needed. In this article, the procedure to accomplish this data homogeneity is presented. It is assumed that terrain data is available in two formats: a) grid elevation and b) contour or iso-altitude lines. Building data is assumed available in plant, polygonal planar contour form, a primitive and yet widely used format which is yet to be transformed into 3D entities. The geometric modeling of joint terrain and building data is further difficuledt because union of surface objects (even after achieving a unified format) is undefined from the point of view of 3D boolean operations.*


## 1. Introduction

The modeling of radio propagation (diffraction, reflection, etc.) in the simplest urban scenarios poses considerable computational complexity. Such a computational complexity ([1]) comes from: (a) the translation among modeling schema such as complete / incomplete Boundary Representations (B-Reps), Constructive Solid Geometry, Simplified triangular B-Rep, exhaustive enumeration, etc. b) the difficult solution of geometrical-topological problems, c) the maintenance of consistency in simplified models, and d) the control of explosively large data sets. This paper illustrates the articulation of computational and stochastic geometry techniques ([2]) into the modeling of terrain and buildings to adequately carry out ray-tracing for propagation studies ([3]).

The first concern is the conversion and completion of terrain data. Most geography institutions publish their elevation maps in either grid or contour formats. Conversion from *iso-altitude contour* to *grid* format includes the application of stochastic prediction ([4], [5]) such as "kriging" ([6], [7]) or Principal Component Analysis (PCA [8]). Conversion from *grid* to *iso-altitude contour* format includes the production of a triangle-based mesh as an intermediate step, followed by a parallel slicing of the triangle mesh. The production of a triangle-based mesh is greatly facilitated by the neighborhood information implicit in the grid ([9]) and the assumption that no void spaces are present in the grid. In case the grid has voids, explicit information for the contours surrounding a void region may be generated ([9]). The maintenance of correct topological void information is a more complex task, which requires operations among boundaries, called also "surgical operators" among topologists. In any case, the result of such technique applied to terrain modeling is a 2-manifold (shell) [1] with borders (which means *incomplete*), with $C^0$ continuity if tiled with triangles (where $C^i$ means continuity up to the *i-th* derivative).

A second main task comes with the integration of buildings and terrain information. For many applications, buildings are represented as flat, 2D plant contours, as per architectural drawings. This representation is not a 3D one, and therefore has no topological or geometrical consistency. In the present work 2D plant contours are extruded along the vertical direction to engender complete Boundary Representations, which are solids. If terrain is expressed as a shell with boundaries and the buildings are solids, there exists an obvious incompatibility to reach an integrated geometric model of buildings plus terrain. In this work, such an obstacle is circumvented by (a) extracting from the building solid information its (closed or complete) shell, (b) exploding the building complete shell into several incomplete sub-shells (walls, roof, base), and (c) performing under-determined boolean operations of such shells against the incomplete shells representing the terrain. Operation (c) is not well defined in the realm of Boolean operations, since these have been proposed and implemented for 2-manifolds without border (which represent solid, bounded objects). Fortunately, under-defined Boolean operations between shells have been implemented by computational geometers using heuristics that suffice for terrain modeling. They allow for the “gluing” of buildings to terrain, with elimination of redundant surfaces, and maintenance of consistent incomplete shells. To perform operation (c), the shell representing the terrain must be in Rational Non Uniform B-Spline (NURBS) format. This means, it is necessary to “promote” $C^0$ terrain information to NURB shells, with $C^2$ or higher continuity. In turn, this is possible only for regions with a sampling count of N x M terrain points. Regions with irregular sampling patterns must be represented with $C^0$ continuity (triangular-based) meshes, and algorithms for this primitive type must be used. As a last step, in any case, demotion from $C^2$ to $C^0$ continuity is needed in the finished shell integrating buildings plus terrain as a pre-condition to performing radio ray-tracing. The reason for this demotion is that ray tracing algorithms would be extraordinarily more complex if algebraic surfaces were used instead of flat ones ([10]).

## 2. Methodology

The integration of Terrain and Building Data Bases (TDB and BDB respectively) implemented in this work for the purposes of Ray-Tracing in Wave Propagation included the following 4 steps (Figure 1): (i) Terrain and Buildings data acquisition, (ii) data pre-processing and schema conversion to ensure topological and geometrical compatibility, (iii) utilization of under-defined boolean algorithms to joint building and terrain data, with correct disposal of “dangling” faces and edges, which are side-products of the operation. (iv) Post-processing ([1,11]) of resulting shell to improve its geometrical and topological quality, to reduce its size, and to enforce a selective level of data simplification, according with the Wave Propagation downstream applications.

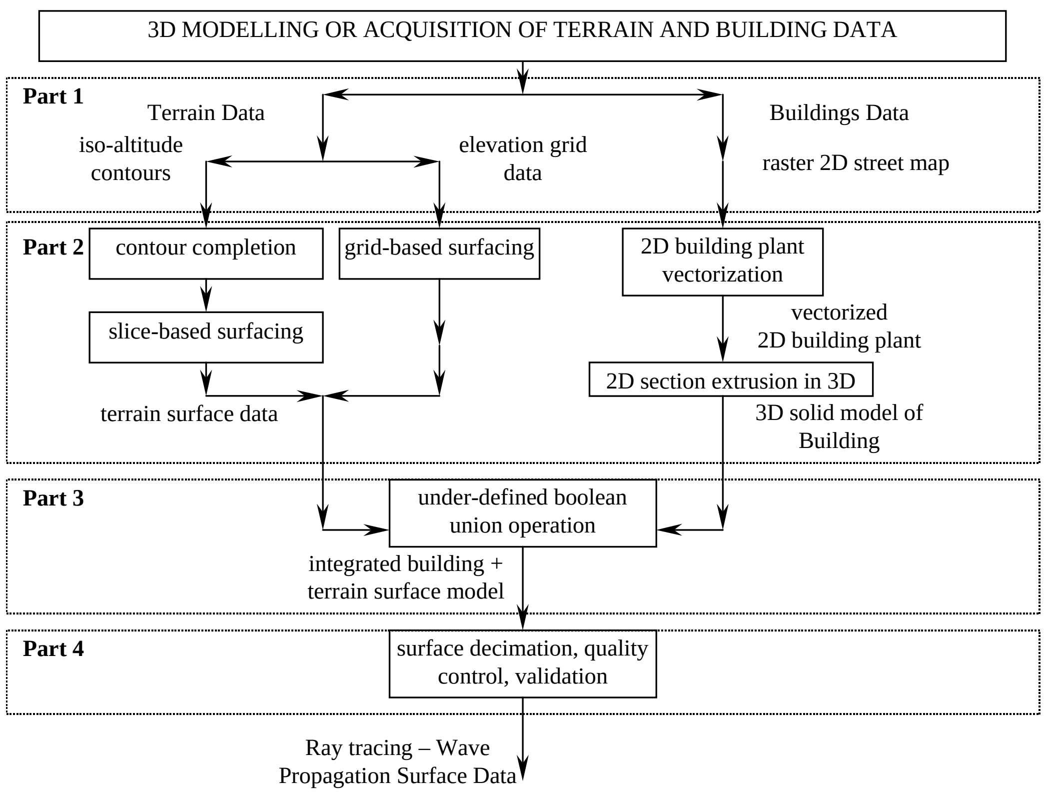


**Figure 1. Methodology. Flow diagram.**

### 2.1. Part 1. Format Conversion of the Terrain and Buildings Data

Format conversion refers in this context to changes in the data format, to make it operable under certain algorithms and / or applications. Data conversion is different from schema conversion in that the first implies parsing, scanning, translation from hard-copy to raster data, etc. Schema conversion implies calculation of topological relations and geometrical parameters, to ensure completeness and consistency of the particular sample at hand. A typical case of schema conversion is the translation from grid to implicit surface, iso-altitude into implicit surface, implicit surface into iso-altitude, etc. Notice that, because of insufficiency in data, schema conversion is not always possible, and not always renders unique results.

The format conversions performed in this work are: (i) Parsing of iso-altitude data from DXF, TXT, MAP format into MATLAB internal data ([12]). Iso – altitude files ([13, 14]) are received from government SIG systems ([15]). (ii) Parsing of elevation – grid data into MATLAB internal format. (iii) Import-Export of VRML (Virtual Reality Markup Language) from MATLAB format, (iv) Import-Export of VRML from ARX (AutoCAD Runtime Extension, [16]) format. In any case, the information parsed is filtered, since many attributes may be present in the file, which are not relevant to the issues of Wave Propagation (color, line pattern, layer, line thickness, etc.). The goal of this whole set of format conversions is to profit from definite capabilities inherent to the geometric Kernel of the CAD packages (for example, Boolean operations between incomplete shells, in Rhinoceros [17]) or numerical libraries (for example MATLAB). In addition, programming on the Application Programming Interface (API) of Rhinoceros and AutoCAD was realized, at three different levels, allowed by them: (a) a stream of commands, stored in a text file (called *script* in AutoCAD and *command file* in Rhinoceros), (b) A higher level command language, at the level of interpreter with no direct access to the geometric database, (called ADS in AutoCAD and script in Rhinoceros), and (c) a direct and complete access to the Rhino or AutoCAD database and Geometric Kernel from C or C++ languages ([18]) (interface called *ARX* in AutoCAD and *SDK* in Rhinoceros). All three levels were used, in order to pragmatically articulate the maximum of built-in commands with essential home-made applications, specific for the project.

As shown in Figure 2 , Terrain information is available in iso-altitude contours and elevation regular grids (Figure 3). Architectural housing information is still commonly represented as planar 2D vectorized contours showing the plant view of the construction (Figure 4). Notice that even this information may not be available in not-so-wealthy city councils. Therefore, a digitization of hard copy drawings may be necessary to obtain it.

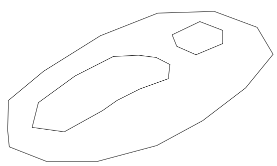

**Figure 2. Iso-height contour lines.**

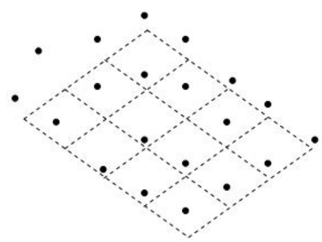

**Figure 3. Elevation, regular grid matrix.**

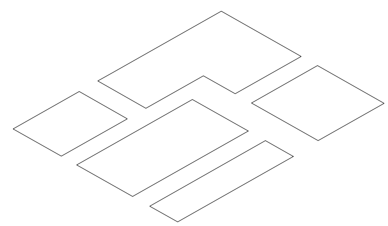

**Figure 4. Planar 2D vectorized contours representing building plant views.**

## 2.2. Part 2. Data Meshing

A central goal in the geometric work for simulation of Wave Propagation is to integrate housing and terrain information. This integration means that buildings are modeled as protrusions in the original terrain shell. The shell need not be closed (in fact, it would be senseless to close it), but it needs to have a unique border (the external one). The condition of unique border precludes internal holes, as well as folds, T-joints, dangling edges and faces, etc. To reach his point, a set of context-defined boolean operations were used. These are under defined boolean operations among surfaces which require: (i) all the participant objects to be bordered or incomplete shells, and (ii) dangling faces or edges (by-product of surface intersection and splitting) to be eliminated using user-provided heuristics. These two conditions determine the efforts realized. In particular, condition (i) was satisfied by creating NURBS surfaces, of $\boldsymbol{C^2}$ or higher continuity, even for flat data. After the under-defined boolean operations a demotion into $\boldsymbol{C^0}$ surfaces was performed, to facilitate downstream export, visualization and calculation tasks.

**2.2.1. Terrain Representation for Wave Propagation.** The treatment of terrain data aims to convert Geographic Information System (GIS) data into a Boundary-Representation scheme. As mentioned before, conversion between schema is not always possible, usually because insufficient information or ambiguity at either side of the conversion process. The B-Rep scheme prescribes that a "solid" is the "interior" of a closed "surface". With no formal definition on those terms, it is underscored here that the Boundary Representation also serves to represent incomplete surfaces, with the understanding that when the surface is incomplete, no solid is being represented. For the purpose of terrain and housing representation, a partial shell is adequate, if additional information is provided (for example, what is "inside" or "outside"). The surface to be created starting from GIS data may be either triangle ($\boldsymbol{C^0}$) or NURBS ($\boldsymbol{C^2}$) type. Two types of GIS data are processed in this project:

(i) Iso-altitude contours. An iso-altitude contour is a Piecewise Linear approximation of a planar curve, considered in a coordinate system in which the planar curve will have constant $Z$ (usually called "altitude") value. The GIS data is commonly a set of iso-altitude curves, confined to a rectangular region in the X-Y plane (see Figure 5), with the $Z_i$ values of the iso-altitude $\boldsymbol{C_i}(u)$ curves usually forming a monotonically increasing, uniformly spaced sequence in Z axis. The $\boldsymbol{C_i(u)}$ curves are supposed to be a cross sectional sample of a non – self intersecting surface (in this case, the terrain itself). The open contours $\boldsymbol{C_i(u)}$ represent portions of originally closed cross sections being interrupted by the artificial grid of the GIS. This is one of the insufficiencies of this scheme, and is to be overcome by using the rectangle of the grid itself to complete the missing part of the contours, therefore closing them. This process is not a trivial one, since the operation of completing contours from pieces of the rectangular grid is not completely defined, and produces a number of possible outcomes. In this particular instance, this process is assisted by a human operator, who uses an understanding of the terrain to complete the contours (Figure 6). This step is required to be able to reconstruct a $\boldsymbol{C^0}$ surface from the closed, oriented contours. Algorithms for this purpose are usually based in Delone Triangulations and Voronoi Diagrams ([19]). A variation, equipped with 2D-shape similarity reasoning, is found in [20].

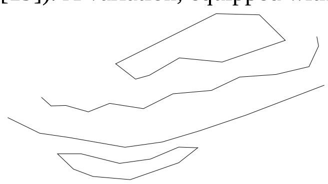

**Figure 5. Open contour lines.**

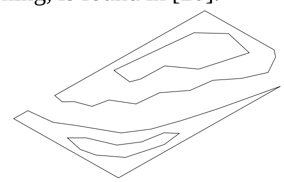

**Figure 6. Closed contour lines.**

(ii) Elevation grid data. A grid, elevation data set represents a function $f: R \times R \rightarrow R$ (Figure 7). One assumes that for a particular pair $(x, y)$, the function has a unique value $f(x, y)$. Therefore, it is adequate to represent most of terrain data. In this case, $f(x, y) = z(x, y)$, an altitude value. It is usual that the $(x, y)$ couples be sampled from a regular rectangular grid, therefore having an $(N \times M)$ number of samples. This formalism is widely used, even with the $f(\,)$ function being temperature, humidity, pressure, etc. Also, in range pictures for 3D digital optical sampling the same information is stored. The surface reconstruction algorithm takes advantage of the neighboring information implicit in the grid itself, and builds a $\boldsymbol{C^0}$ set of triangles interpolating the $f(\,)$ values at the grid intersections (Figure 8). When points of the $(x, y)$ grid register no value of $f(\,)$, a void is produced in the triangle mesh. This situation is not uncommon since the optical device may not register the point due to optical or atmospheric conditions. In those cases explicit information on the external and internal borders of the shell is required.

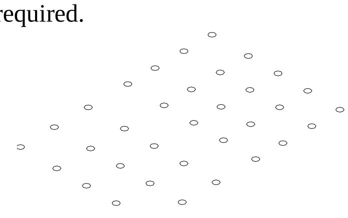

**Figure 7. Grid intersections (x, y).**

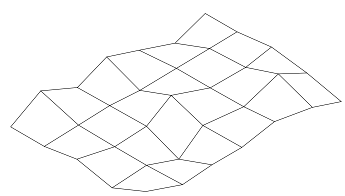

**Figure 8. Meshed terrain surface from grid-elevation data.**

NURBS Production. At this point, a parallel shell representation is devised, depending on the algorithms that later on will be commissioned to calculate Boolean operations between shells. If the Boolean operators are not able to perform intersections or unions between triangular meshes, an alternative NURBS representation must be used, as in the present case. The conversion (Figure 9) from grid-elevation data to NURBS format was achieved by writing an export module in MATLAB, able to produce *command* or *script* files to be executed in Rhino. This undertaking made use of the second level of interaction with the API of geometric modelers, mentioned above to produce the NURBS version of the terrain data (Figure 10).

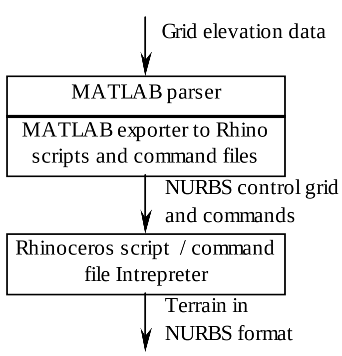


**Figure 9. Procedure for NURBS representation of grid-elevation terrain data.**

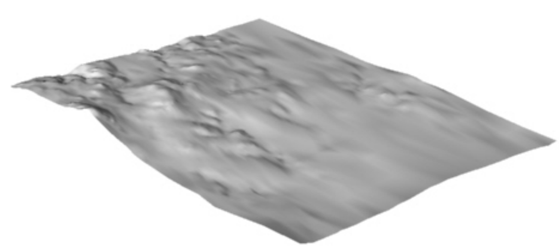

**Figure 10. Result of NURBS representation for grid-elevation terrain data.**

**2.2.2. Housing Representation for Wave Propagation.** As said before, architectural drawings are commonly 2D plant views of buildings, in the form of closed, non – self intersecting polygonal contours (Figure 4). This representation scheme is incomplete and ambiguous. In order to upgrade it and obtain 3D geometric models of buildings such sections are extruded in the vertical (Z) direction, a distance dependent on the height of the building. The result of this operation is indeed 3D solid models (Figure 11), which includes the closed shell model of the boundary. The solid model is therefore decomposed and its shell information, in NURBS format obtained (Figure 12).

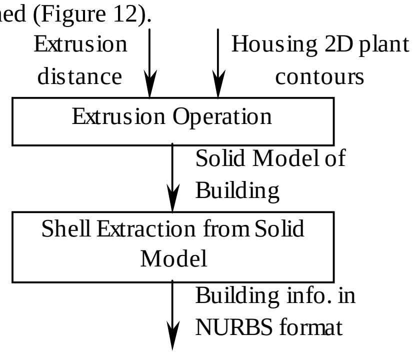


**Figure 11. Synthesis of housing data into NURBS format of building shells.**

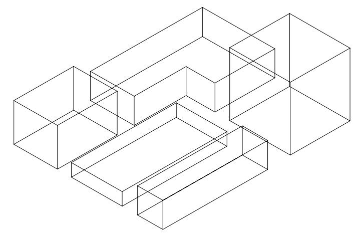

**Figure 12 Resulting shells of buildings in NURBS format.**

## 2.3. Part 3. Integration of Terrain and Building Data.

**2.3.1. Specification.** As said before, Wave Propagation calculations require that housing data be integrated within terrain data. This integration must be complete (Figure 13, Figure 14), in such a away that buildings become protrusions on the terrain shell. The housing – terrain shell must be $C^0$-continuous (no holes) at the junctions building – terrain. There can be no creases, interruptions, folds or dangling edges, non-manifold conditions (redundant "T" surfaces) or faces in such junctures (Figure 14). The surface must be perfect at that neighborhood, with the only concession being that the continuity accepted in such junctions is $C^0$, while all the other places on the building- and terrain- NURBS are $C^2$-continuous (smooth up to the second derivative).

**2.3.2. Shell Union.** The Union operation specified above is not strictly a Boolean one since these operations (in the context of geometric solid modeling) require (a) closed meshes, and (b) a convention that defines the interior and exterior of the closed mesh. Also, Boolean operations unite, intersect or subtract the whole set of points in the interior of solid objects. In strict sense, for example, the intersection of two shells would render a set of curves in the space: the intersection of two surfaces. This is a non-intuitive result, as is the union, subtraction, etc. of shells. Therefore the operation used for the present application, can be defined as follows.

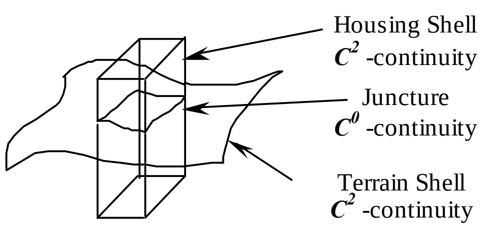


**Figure 13. NURBS shells for building and terrain data before shell integration.**

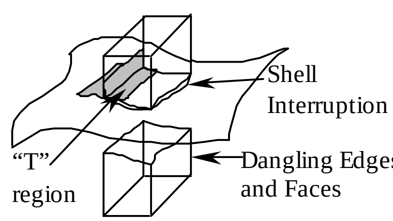


**Figure 14. Violations of manifold properties: dangling edges and faces, shell interruption, redundant ("T") neighborhoods.**

Given $A$, $B$, 2-manifolds (shells) in $R^3$ with $n_A(\,): R^3 \rightarrow \{-1,0,1\}$ and $n_B(\,): R^3 \rightarrow \{-1,0,1\}$ functions in $R^3$ which take value $-1$ at an arbitrary side of $A$, value $0$ on $A$ and $+1$ at the remaining side. $n_A(\,)$ divides the space $R^3$ in ***inside/ on / outside A*** (Figure 15) **.** Equivalent definition may be made for $n_B(\,)$. The operation $\cup^*$ (a very particular type of Boolean union) is $A \cup^* B = \{(p \in A) \text{ or } (p \in B) \mid n_A(p) \geq 0 \text{ and } n_B(p) \geq 0\}$ (Figure 16). This operation basically keeps all portions of surface $A$ at one "side" of $B$ and neglects the points at the other side. It also keeps all points of $B$ at one side of surface $A$, and neglects the others. There is obviously a 4-choice combination of what to keep and what to neglect. Intuitively, it can be thought that $B$ divides $A$ into two parts ($A_1$ and $A_2$) and vice versa in $B$ ($B_1$ and $B_2$). The new surface may be built in 4 different ways: $\{A_1,B_1\}$,$\{A_1,B_2\}$,$\{A_2,B_1\}$,$\{A_2,B_2\}$ (Figure 16).

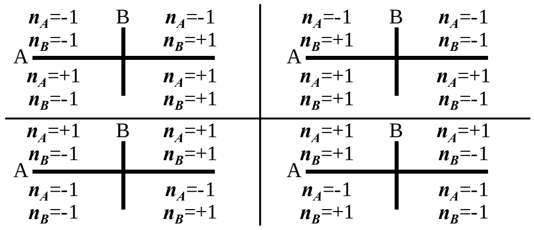


**Figure 15. Four possible orientation combinations in shells A and B.**

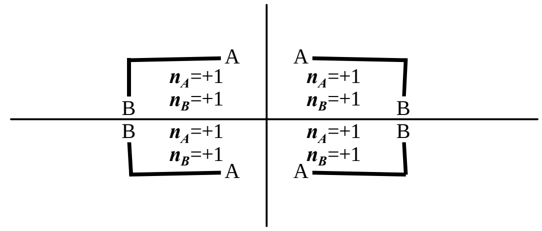


**Figure 16. Result of *U** operation according to the orientation combinations in shells A and B.**

Figure **17** sketches an intuitive result of the U* operation applied to terrain and building shells. The junction presents $C^0$ continuity while other spots of the original terrain and building shells have $C^2$ continuity. As requested, the housing is represented as a protrusion in the terrain shell. For exporting and sake of calculations, the $C^2$ NURBS terrain-housing shell is converted to a $C^0$ triangle or quadrangle-based shell (Figure 18). Technically, this outcome is ready for calculations of Wave Propagation, since it is a 2-manifold, continuous, with only the outermost border (no internal holes).

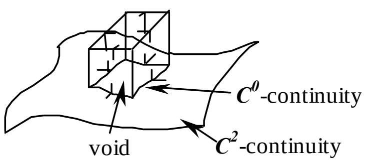


**Figure 17. Detail of the result of the special union U* operation.**

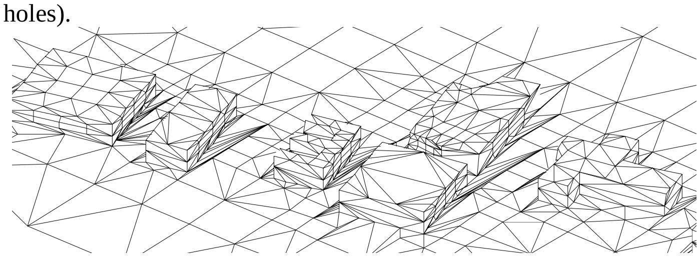

**Figure 18. Triangulation of the result of the U*operation.**

## 2.4. Part 4. Model Simplification

Although the structure from Figure 18 is mathematically adequate for Wave Propagation calculations, its number of triangles, heterogeneity in triangle size and extreme triangle aspect ratios ( $side_i$ / $side_j$ ) greatly impairs any calculation. Therefore a process of quality assurance, aspect ratio and size homogenization is required. This objective was achieved by (i) forming n-sided flat polygons from triangular and quadrangular tiled regions and then (ii) breaking down the n-sided polygons into triangle sets with good geometric characteristics. These steps are discussed next.

**2.4.1. Triangle to Polygon Clustering.** To produce polygonal regions out of triangular and quadrangular ones the procedure is: (i) to collect all triangles or quadrangles $S_k = \{ T_1, T_2, T_3, \ldots, T_m \}$ whose areas are connected and approximately coplanar with a statistically calculated best plane $\pi_k=[p_v,n] \in R^3$, (ii) to build with $S_k$ an n-side flat polygon $P_k$ (possibly with holes). (iii) repeat (i) and (ii) for triangles or quadrangles contained in significantly different planes $P_w$, until all triangles and quadrangles are exhausted. For each $S_k$, there is an outermost straight segment contour, traversed in counterclockwise direction with respect to an outwards pointing vector ($\pm n$) normal to the plane $\pi_k=[p_v,n]$. All inner contours of $P_k$, built in clockwise direction with respect to the normal vector, represent the boundaries of this face with other faces contained in planes significantly different from $\pi_k$. The best (in the statistical sense) plane $\pi_k$ is to be calculated from a cloud of quasi-planar points by using for example Principal Component Analysis ([21, 22]). The result of this step applied to data from Figure 18 is shown in Figure 19.

**2.4.2. Polygon to Triangle Fragmentation.** Once n-sided polygons are obtained, Wave Propagation calculations may proceed. However, the algorithmic part of them is more complicated than with triangles. A basic difficulty is that polygons may be non-convex regions, even if they have no holes. In the case in which they have holes, the situation is more complicated. Although intersection, point inclusion, etc. calculations for general polygons with holes are not impossible, it is definitely easier to deal with triangles. Therefore, n-sided polygons are broken down into triangles, which have significant better aspect ratio, size, etc. than the original ones. This step was performed by using the external, Delone Triangulation-based, application Triangle (by J.R., Shewchuk from Carnegie Mellon University [23]), which allows exactly this type of polygon partition by enforcing different geometric characteristics of the final triangle set. This step was carried out by C and C++ programming with direct and complete access to the AutoCAD database through *ARX* Interface (see section 2.1). The result of this step applied to data from Figure 19 is shown in Figure 20.

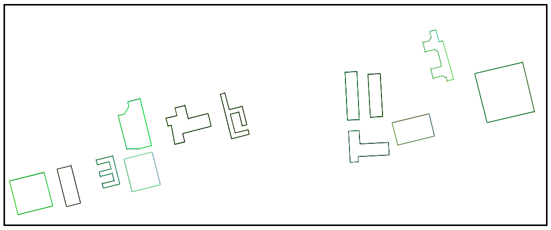

**Figure 19. Zoom of aerial view of terrain and buildings. n-sided polygons obtained in the triangle and quadrangle merging process.**

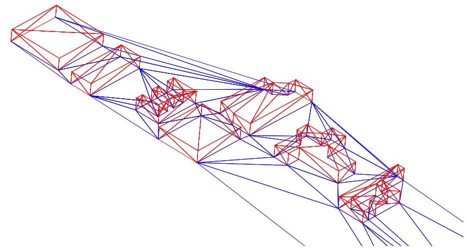

**Figure 20. Isometric view of triangulation of the n-sided polygons from Figure 19.**

## 3. Results

This section discusses three examples which illustrate the aforementioned methodology. For a better evaluation of the results VRML ([24]) and / or STL ([25]) files were exported from the obtained models.

### 3.1. Case 1.

A 16000 $m^2$ terrain and building area was modeled with the procedure mentioned. The U* union operation rendered an integrated mesh with 2983 triangular and quadrangular faces. The step of building n-side polygons from such data produced 112 polygons with 125 contours (indicating the normal and legal presence of internal holes in the polygonal faces). The subsequent step of controlled triangle production rendered 394 triangles. Therefore, the procedure proposed reduced the number of faces to 13.2 % of the initial number. The final result is shown in Figure 21.

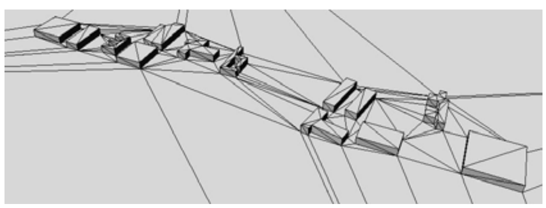

**Figure 21. Terrain and Housing Data set after n-side polygon synthesis and triangle production.**

### 3.2. Case 2.

Grid-elevation data of terrain was parsed and exported to Rhino via scripts of level (b) in section 2.1, to produce a NURBS surface. Next, a triangulation of the terrain was produced, with 5606 triangles and quadrangles. The proposed methodology was applied to such data, and the algorithm of triangle clustering into flat n-sided polygonal regions rendered 208 approximately flat polygons. The controlled triangulation step produced 1658 triangles. Therefore, the final data size was a 29.5% of original one. Figure 22 shows the final result of this procedure.

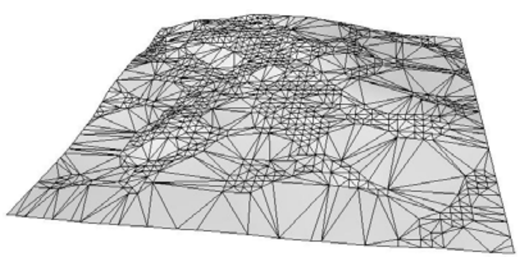

**Figure 22. Irregular terrain after n-side polygon synthesis and controlled triangulation.**

### 3.3. Case 3.

A center of a City was modeled with the procedure mentioned. The U* union operation rendered an integrated mesh with 6332 triangular and quadrangular faces. The step of building n-side polygons from such data produced 328 polygons with 382 contours. The subsequent step of controlled triangle production rendered

2803 triangles. Therefore, the procedure proposed reduced the number of faces to 44.2 % of the initial number. The final result is shown in Figure 23.

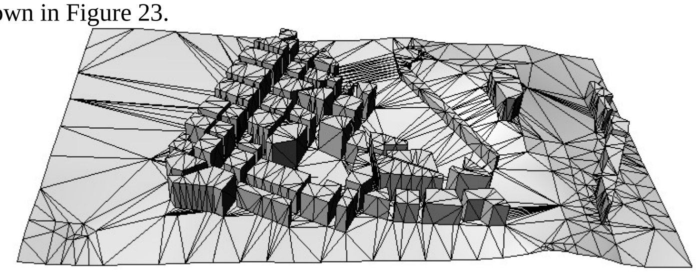

**Figure 23. Centre of a City Data set after n-side polygon synthesis and triangle production.**

## 4. Summary

The methodology demonstrated allows the integration of terrain and housing data coming from very different (and frequently incomplete) representation schema and GIS systems. The process described articulates in a very pragmatic way the diverse capabilities of end-user interfaces in CAD packages, as well as the 3 possible levels of programming onto the geometric Kernels of the same packages, along with computational tools developed in MATLAB for the project. Also, specialized libraries for geometric computations were used in the project. The final result, a good quality triangular mesh, with $C^0$ continuity allows for topological and geometrical queries: areas, normal vectors, locations, as well as neighborhood and boundary information.

## 5. References


[1] Ruiz, O.E., *Understanding CAD / CAM /CG* , American Society of Mechanical Engineers, ASME. Continuing Education Institute. Global Training. ASME Code GT-006. 2002.

[2] De Floriani, L., E. Puppo, and P. Magillo, "Applications of Computational Geometry to Geographic Information Systems", Genova – Italy.

[3] Allen, S.M., K.H. Craig, S. Hurley, W.D. Miners, R.K. Taplin, and M.J. Willis, "A Software Tool for Optimised Millimetric Radio Access Cell Planning", 5, Great Britain.

[4] Sáková, F., "Gis Functions – Interpolation", Periodica Polytechnica Ser. Civ. Eng., Volume 43, pp. 63-86, 1999.

[5] Barbosa, R.L., and J.F. Custódio, "Geração de um Modelo Digital de Terreno por Aproximações Sucessivas", 5, Unesp, São Paulo – Brazil.

[6] Van Beers, W., and J., Kleijnen, "Kriging For Interpolation in Random Simulation", 22, Center - Discussion paper, Tilburg – Netherlands, Oct. 2001.

[7] Gebhardt, A., "PVM kriging with R", 10, DSC 2003 Working Papers, Klagenfurt – Austria, 2003.

[8] Popovici, V, and J.P. Thiran, "PCA in Autocorrelation Space", ICPR 2002, 4, Quebec – Canada, Agu. 2002.

[9] Ruiz, O., and P. Neugebauer, "Topologically Consistent Partial Surface Reconstruction from Range Pictures", Computer Graphics and Imaging (CGIM) conference of IASTED (Intl. Assoc. of Science and Technology for Development). Las Vegas, USA. Nov. 19-23 2000.

[10] Ruiz, O., and P. Ferreira, "Algebraic Geometry and Group Theory in Geometric Constraint Satisfaction for Computer Aided Design and Assembly Planning", IIE Transactions. Focused Issue in Design and Manufacturing, n 28, ISBN 0740-817X, Editor: Chapman & Hall, London, UK, pp 281-294, 1996.

[11] García, M.J., *Lectura Notes in Advanced Numerical Analysis*, Eafit University, Medellín – Colombia, Jan. 2003.

[12] MATLAB www.mathworks.com

[13] Felicísimo, A.M., *El modelo digital de elevaciones*, Oviedo University, Oviedo – Spain.

[14] Felicísimo, A.M., *Modelos Digitales del Terreno*, Oviedo University, Oviedo – Spain, 1994.

[15] IGN www.ign.es

[16] AUTOCAD www.autodesk.com

[17] RHINO www.rhino3d.com

[18] VISUAL C msdn.microsoft.com/visualc

[19] Boissonat D., and B. Geiger, "Three Dimensional Reconstruction of Complex Shapes based on Delaunay Triangulation", SPIE Proceedings Vol. 1905. Biomedical Image Processing and Biomedical Visualization, San Jose, CA, USA, 964-975, Feb 1993.

[20] Ruiz, O., C. Cadavid, and M. Granados, "Evaluation of 2D Shape Likeness for Surface Reconstruction", Journal Anales de Ingeniería Gráfica, ISSN: 1137-7704, No 15, pp.16-24, 2002.

[21] Hastie, T., and W. Stuetzle, "Principal curves", Journal of the American Statistical Association, Vol. 84, pp. 502-516, 1989.

[22] Kegl, B., and A. Krzyzak, "Piecewise linear skeletonization using principal curves", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, N. 1, pp. 59 – 74, 2002.

[23] Shewchuk, J.R., "Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator", Applied Computational Geometry, ACM, pp. 124-133, Philadelphia - Pennsylvania, May 1996.

[24] VRML www.web3d.org/technicalinfo/specifications/vrml97/index.htm

[25] STL rpdrc.ic.polyu.edu.hk/old_files/stl_introduction.htm