

Structural analysis of a deployable double-layer tensegrity grid

Gomez-Jauregui, Valentin¹; Sanchez Haro, Javier²; Manchado, Cristina³; Otero, Cesar⁴

ABSTRACT

Different methods have been used recently to discover new families of a type of space frame: Double-Layer Tensegrity Grids (DLTGs). The main objective of this work is analyzing the process for the design and calculation of a specific deployable DLTG (DDLTG) at full scale (4x4x1 m) composed by 16 modules of 1x1x1 m.

Three CAD/CAE tools were used for these purposes: ToyGL, Matlab/Octave, and Midas Civil. It was necessary to create an intermediate platform to exchange data and information between both tools, the ToyGL Processor, based on user-defined functions written in VBA for Microsoft Excel.

The methodology was validated by comparing the results obtained by the first two software tools with those provided by Midas Civil. As a conclusion, ToyGL is reliable and accurate in terms of the definition of geometry, distribution of stresses and forces, reactions, states of self-stress, mechanisms, etc.

Keywords: Structure; deployable; double-layer; tensegrity; grid

1. INTRODUCTION

This research work is framed in the doctoral program of the corresponding author's PhD thesis [1], therefore, some sections of this paper are excerpts of that document.

During the last years, many techniques have been used to calculate and analyze tensegrity structures. There are already some very interesting reviews on the subject [2-4]. Since then, some other form-finding techniques have been explored [5].

The studies carried out to identify the structural behavior of our DDLTG have mainly involved static analysis. Even though using a static approach could mean an overestimation of the load carrying capacity of the structure [6], the purpose of these studies was to make sure that the new structures were stable and that any analytical comparison between the different DLTGs was consistent and trustworthy.

The main CAE methods used for these purposes have been two: ToyGL, developed by Julien Averseng, and some Matlab/Octave codes. More information about both methods will be provided in sections 2.1

¹ EGICAD Research Group. Universidad de Cantabria. (SPAIN). valen.gomez.jauregui@unican.es (Corresponding author)

² Ingeniería de la Construcción y Desarrollo de Proyectos, INGECID (SPAIN). javier.sanchezharo@unican.es

³ EGICAD Research Group. Universidad de Cantabria. (SPAIN). crisrina.manchado@unican.es

⁴ EGICAD Research Group. Universidad de Cantabria. (SPAIN). cesar.otero@unican.es

and 2.3 respectively. It was necessary to create an intermediate platform to exchange data and information between both tools, the ToyGL Processor, based on user-defined functions written in VBA for Microsoft Excel (section 2.2).

In order to be sure about the accuracy of the analysis, the methodology was validated by comparing the results obtained by both software tools. In fact, they were also compared with some commercial software for general purposes (Midas Civil). The results showed that ToyGL is perfectly accurate in terms of the definition of geometry, distribution of tensions and forces, reactions, states of self-stress, mechanisms, etc.

The general scheme of tasks carried out for completing the static analysis of these structures is explained graphically in the Figure 1.

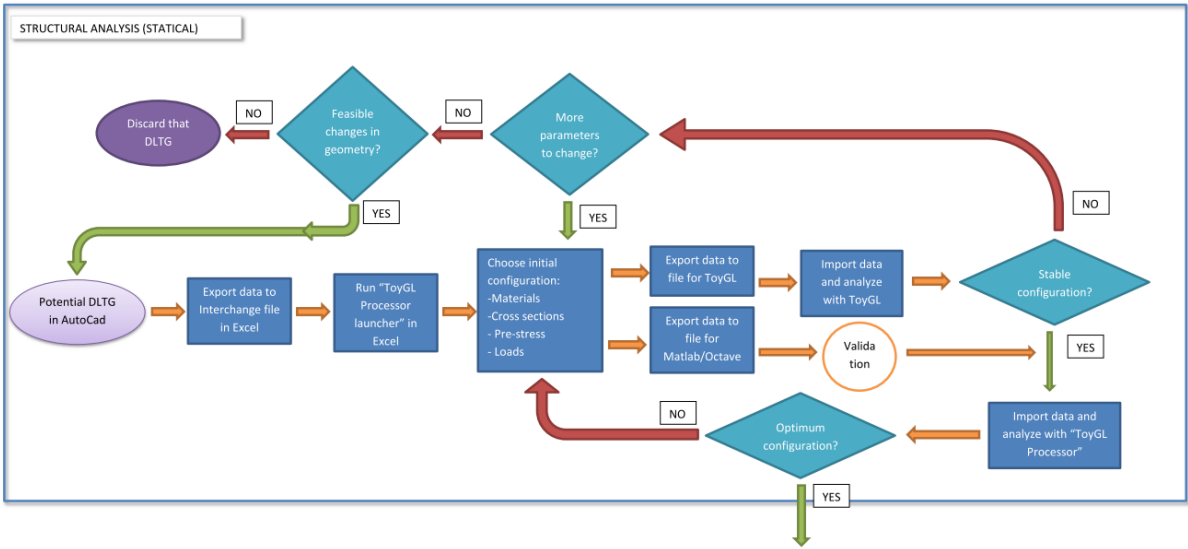


Figure 1. Workflow of the analytical process carried out during this research work

2. SOFTWARE TOOLS

During this work, the use of software tools has been critical, especially those related to CAD/CAE. In fact, new code has been written in order to implement some of the functions and capacities of the CAD/CAE programs. Therefore, in this section, the already existing programs and the new software tools will be discussed in order to explain their functionalities and their relevance to the general scheme of this research.

2.1. ToyGL

ToyGL is a graphical simulation program developed by Averseng [7] used to perform the form-finding and calculations of most of the structures presented in this thesis. It permits traditional and new structures to be designed and, once defined, it allows interaction in real time by changing, with the

mouse and keyboard, the length of members, weights, supports, and position of the nodes. More information about this CAE tool can be found in [7].

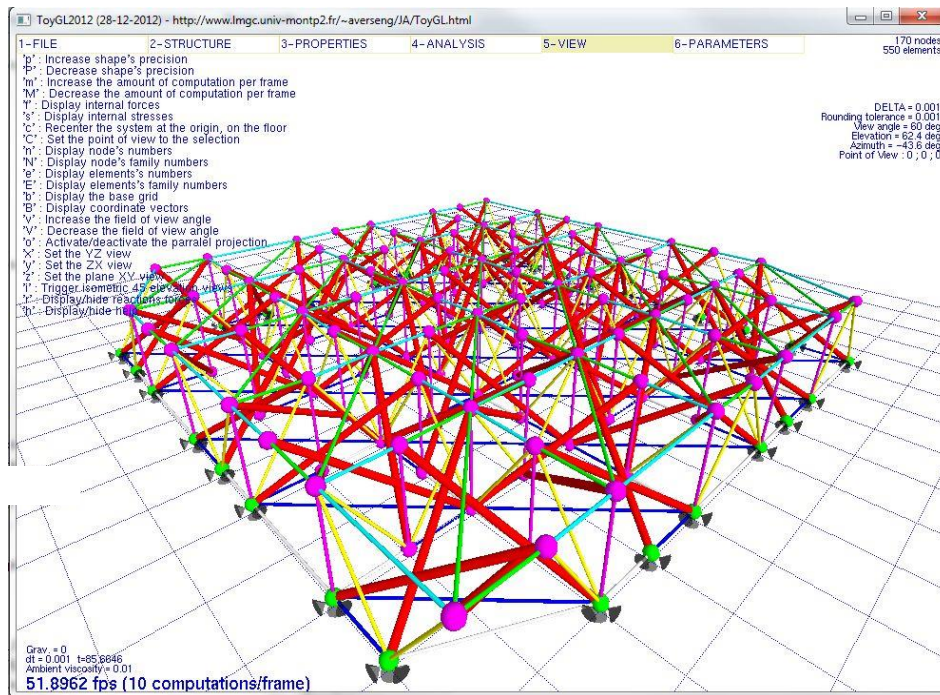


Figure 2. Representation of a DDLTG in ToyGL

The first version used during this research work was the one released in December of 2009. Since then, many improvements and revisions were implemented; in fact, some of the structures created for this thesis were used for the debugging and developing of the software tool. The last and current version was released in December of 2012 (Figure 2).

2.2. ToyGL Processor

Even if ToyGL is extremely useful and intuitive, there were some needs that were not addressed by this tool. For instance, the geometric definition of a whole structure from scratch is much more easy and effectively carried out by conventional CAD programs, like AutoCad. ToyGL would perform much better during the edition of the structure once it has been generated and imported.

Therefore, an additional software tool, “ToyGL Processor”, was created to operate and exploit the benefits of ToyGL in a more efficient manner. It was designed by means of user-defined functions (with Visual Basic for Application or VBA) applied to Microsoft Excel (Figure 3).

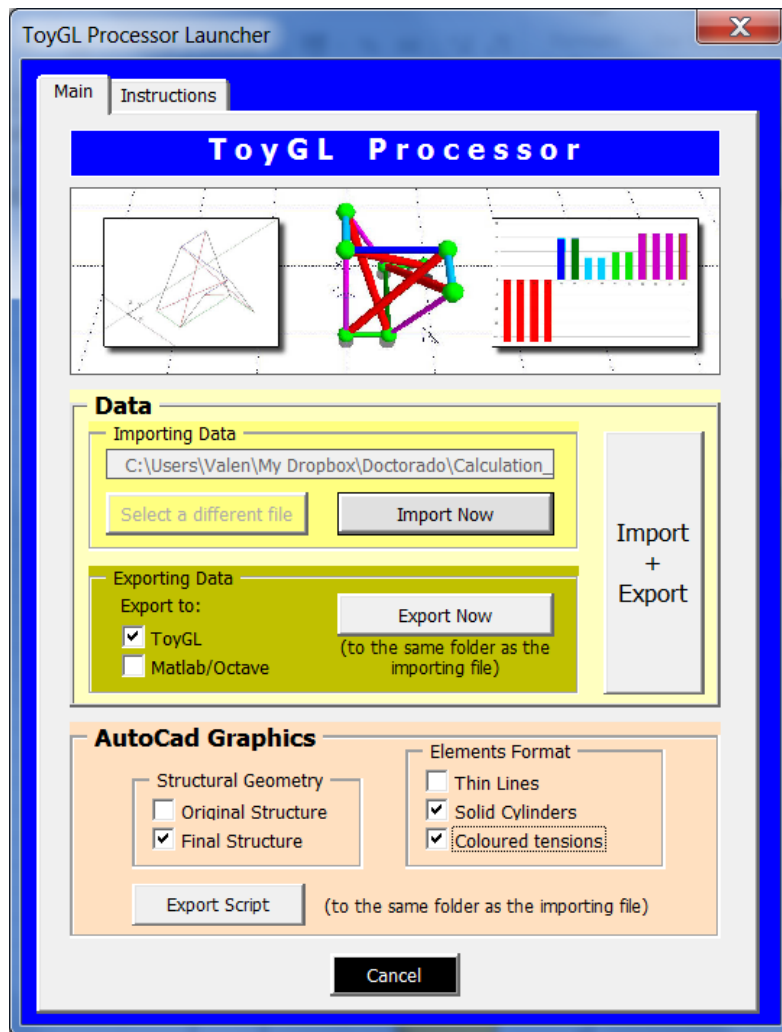


Figure 3. Main dialog box of ToyGL Processor

The ToyGL Processor was an essential link in the process of designing, calculating and representing the existing and new DLTGs analyzed in this thesis, as was illustrated in section 2.2.6, Workflow of analytical process.

Some of the features of this tool are given as follows:

- Importing the geometry of the structure from AutoCad to MS Excel, by means of the interpretation of its geometry, topology, boundary conditions, types and properties of the elements, etc.
- Editing the structure and its elements, changing parameters like geometry, topology, supports, sections, representation (color, thickness, etc.), pre-stress, etc.
- Exporting the geometry and properties of the structure to .txt files that can be read and managed by either ToyGL or Octave/Matlab.

- After the analysis of the structure with those calculation programs, importing the results and the modified structure again to MS Excel.
- Analyzing those results (state of self-stress, stresses, forces, deflection, buckling, etc.) in a visual and direct manner.
- Representing the results by means of lists or graphics (distribution of forces, stresses, etc.).
- Exporting the final geometry and properties of the modified or deflected structure to a script file that can be read by AutoCad in order to represent it as a 3D wireframe, 3D solid model or 3D solid model with visualization of stresses through color gradient.

2.3. Codes for Matlab / Octave

GNU Octave is a high-level interpreted language, primarily intended for numerical computations. It provides capabilities for the numerical solution of linear and nonlinear problems, for performing other numerical experiments and for data visualization and manipulation. The Octave language can be used for free, being quite similar to Matlab so that most programs are easily portable.

Several codes written with this language have been used for analyzing different types of structures. Due to their extension and capabilities exceed the scope of this section, just a summary of their performances and applications will be enumerated in following lines:

- Own codes: Several codes were written for this thesis in order to implement the analytical methodologies and algebraic calculations proposed by different authors [8 - 13]. In summary, the main results obtained from each structure were their mechanisms and states of self-stresses (number and distribution), unidirectional mechanisms, forces and stresses in elements, form-finding, etc. Furthermore, specific functions were created to calculate the minimum distances between the struts of the Quastrut's family or to visualize graphically the mechanisms of the structures.
- Code by IMAC at EPFL [14] generously shared by Rhode Barbarigos and Smith. This software tool for Matlab was designed to perform a nonlinear static analysis by dynamic relaxation for clustered tensegrity structures with discontinuous or continuous cables, springs and bars.
- Code by Gómez-Estrada. Included as an appendix of his Master's thesis [15], these functions were an implementation of a form-finding method, as a proof of concept, which calculates a vector of tension coefficients to self-equilibrate a tensegrity structure in 3D.

2.4. Autodesk Inventor

Another commercial CAD/CAE software used for the representation and analysis of DLTGs was Autodesk Inventor, version 2012. This is a 3D mechanical design software for creating digital prototypes used in the design, visualization and simulation of structures (in this case).

The main uses of this software tool were for the design, analysis and representation of our structures. However, it is remarkable to admit that this software package permits many more operations and performances with mechanical products in general, although some of them were out of the scope of this research work.

3. DEFINITION OF THE STRUCTURE TO ANALYZE

The typology of the DDLTG Quastrut-S1 had already been designed and calculated (Paper 3.5, Comparison between new families of Double-Layer Tensegrity Grids). The only modification was the decision to use a “closed” or “covered” solution, by addition of cables to the top and bottom layers to “fill the gaps” and reinforce the grid, especially at the boundaries (Figure 17). The final dimensions of the DDLTG were 4x4 m, distributed in 16 equal modules of 1x1 m, 4 in each direction X and Y. The resultant grid had then a total number of 86 nodes and 285 components: 64 struts and 221 cables (62 in upper layer, 62 in lower layer, 76 diagonal bracing cables and 21 vertical tensors).

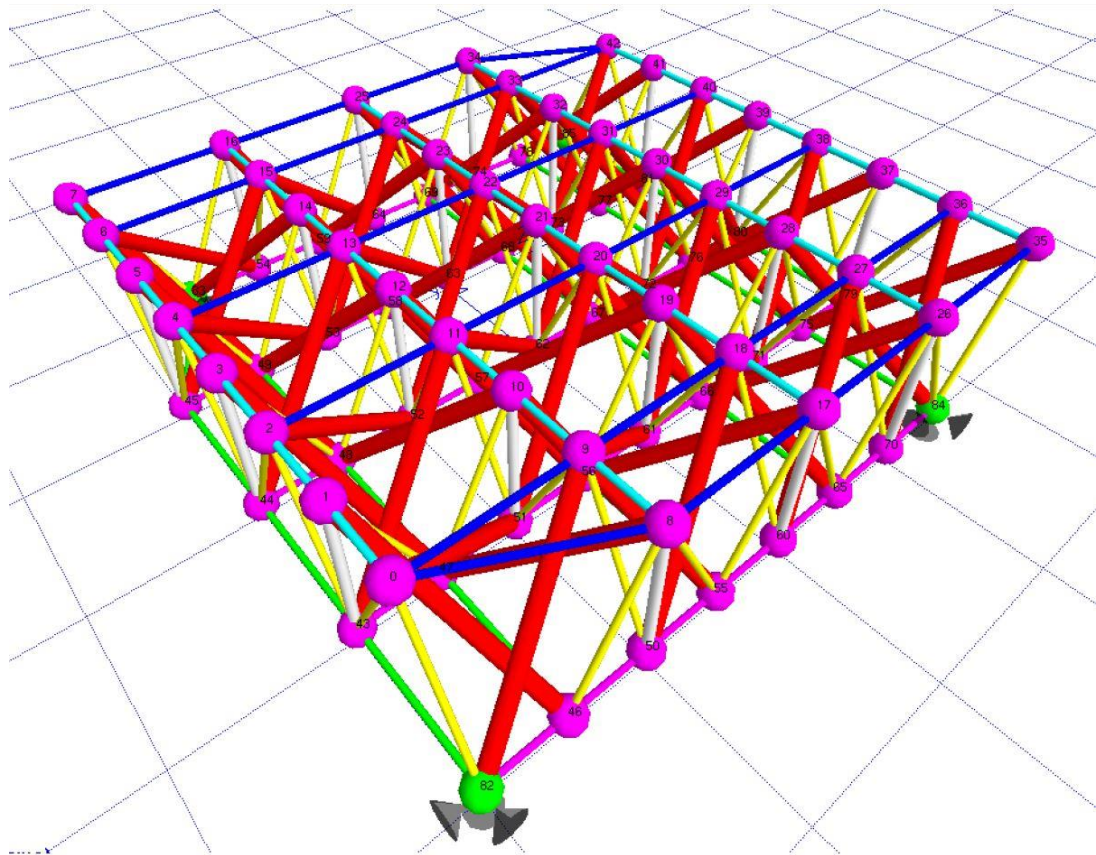


Figure 4. Completed design of the DDLTG after adding additional cables at the boundaries.

Under these conditions, the theoretical lengths of the struts would be 1,5 m, which was very suitable because the standard length of the tubes supplied in Ireland is typically 7,5 m; thereby allowing 5 struts to be attained from each length and minimizing waste. The cables were contiguous in both directions of the horizontal straight alignments of upper and lower layers and therefore had a total length of 4 m each; the bracing cables were also contiguous, although in zig-zag in their plane, where each diagonal had a theoretical length of 1,118 m.

The material for all the elements was steel ($\delta=7850 \text{ kg/m}^3$). Cables were galvanized high tensile steel wire ropes (tensile strength 1770 MPa), with a nominal diameter of 5 mm, except vertical tensors with 10 mm diameter. Steel struts (quality S235JR, yield point 330 MPa) had a circular hollow structural section HSS 21,3x3. Consistent with all these data, the resulting weight of the structure was 169,35 kg, covering 16 m², that is 10,6 kg/m², which for a space frame is considered a lightweight structure. The data relating to the cables and struts are gathered in Table 1.

All the cables of the DDLTG were tensioned, in such a way that the DDLTG was pre-stressed in order to stay stable and in equilibrium. Cable tensions were 19,6 kN for horizontal and diagonal elements and 78,5 kN for vertical tensors (which corresponds in both cases to approx. 56% of elastic limit).

Table 1. Types and characteristics of the struts and cables of the DDLTG

	STRUTS			CABLES			
	K1-K1	K1-K2	K2-K2	Diagonal	Horizontal ^a	Lower Boundary ^b	Vertical Tensors ^c
Quantity	4	36	24	76	102	22	21
Material	Carbon steel S235JR			Galvanized high tensile steel wire rope			
Density (kg/m ³)	7850			7850			
Strength (MPa)	330 (yield point)			1770 (tensile strength)			
Elastic mod. (MPa)	210.000			100.000			
Cross section	Circular HSS 26,9x3			Strands of 7x19 wires			
Diameter (mm)	21,3			5,0			10
Area (mm ²)	172,47			19,63			78,54
Length (mm)	1500	1475	1450	1118	500 - 1000 -1118		1000

^a Horizontal upper and lower layers
^b Horizontal cables of the lower layer at the boundary
^c Composed by turnbuckle + cable + eyebolt

4. ANALYSIS OF THE STRUCTURE

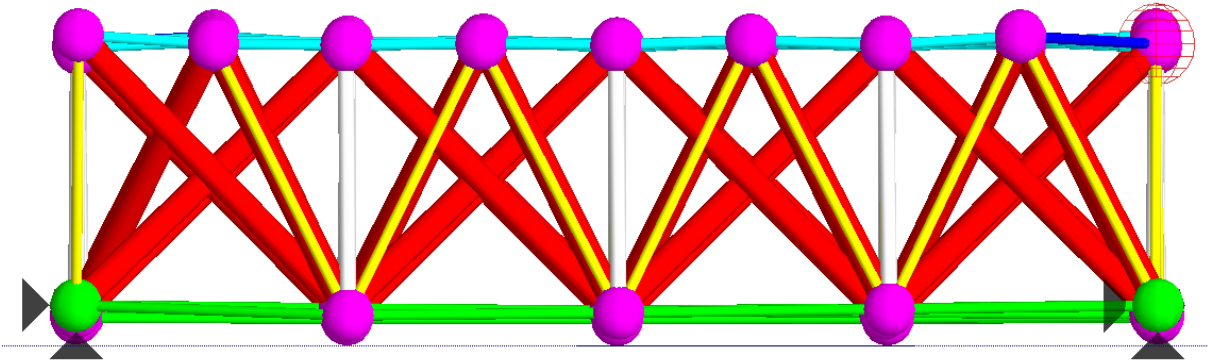


Figure 5. Deflection of the grid under self-weight

As explained above, the structure was calculated with several software tools. In this section, results will correspond to two of those programs: a standard analysis with ToyGL (v. Dec 2012) handled with ToyGL Processor and a linear analysis with Midas Civil (2015 v1.1), which will be conveniently compared. The use of a commercial software for general purposes (Midas) was useful to validate the accuracy of a home-made tool designed for more specific pin-jointed structures and mechanisms (ToyGL). However, Midas (and other similar commercial programs) finds problems to obtain a solution if the structure to calculate has non pre-stressed cables at the initial state. When the cables are not under any tension and they produce a mechanism (characteristic of tensegrity structures), there is not computational convergence and Midas does not reach any solution (even with non-linear analysis and different convergence criteria, like normalized displacements and normalized energetic increments). This fact is determinant, and is the reason why ToyGL could be more suitable in most of the cases.

Figure 6 shows the state of distributions of forces of the DDLTG after being analyzed under the load hypothesis and structural conditions described in the previous section. Even if it is very difficult to appreciate, the set of forces calculated by both software tools are represented in the same illustration (in red and in blue colors); however, as the results are very similar, it is not easy to discern the differences between both graphs. Maximum stress in cables is slightly higher than 30 kN, while maximum compression in struts is always lower than 32 kN.

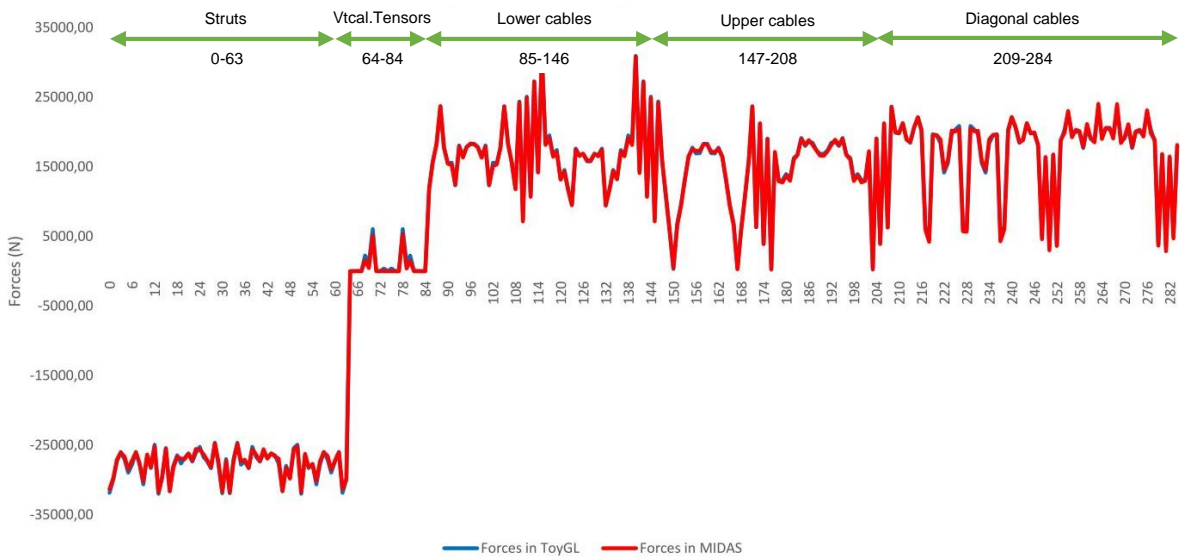


Figure 6. Comparison of member forces (N) between Midas and ToyGL

In Figure 7 it is possible to appreciate the differences of the results related to the member forces obtained with both programs with higher detail. In most of the cases (259 out of 285 elements), these divergences are lower than 3%. There are only 7 cases where that difference is higher than 25% (elements 68, 69, 79, 80, 150, 176, 203) corresponding to those with low force values. It is also significant that the number of slacking cables (vertical tensors) is very similar in both cases: 15 for Midas and 11 for ToyGL (indeed, other four vertical tensors show very low tension in ToyGL).

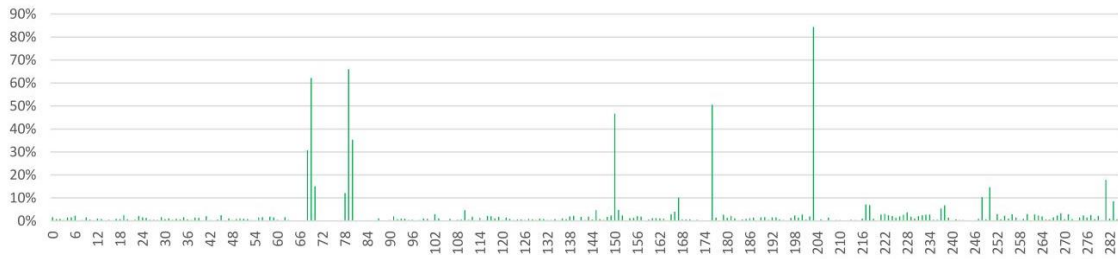


Figure 7. Absolute difference of member forces (%) between Midas and ToyGL

The comparison of displacements of the nodes also shows very similar results. Figure 8 displays the absolute three-dimensional displacement of all the nodes of the structure, and it is clear that both software tools obtain parallel values. In fact, the maximum deflection of the grid has very similar values for both of them: In the upper layer, -0,0201 m (ToyGL) compared to -0,0207 m (Midas), that is 2,91% difference. In the lower layer, -0,0170 m in ToyGL compared to -0,0175 m in Midas (2,62% difference).

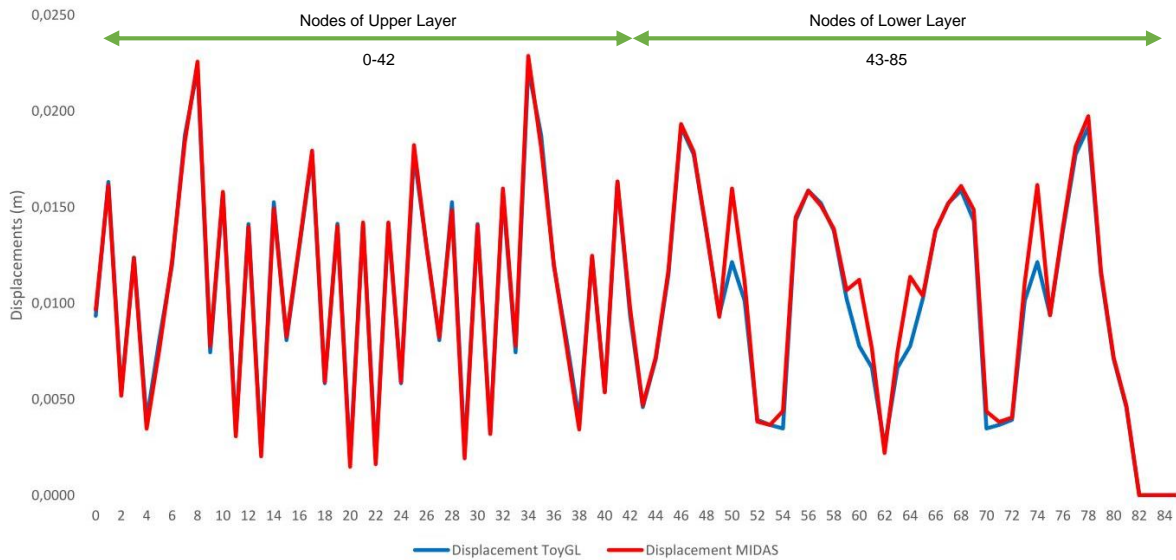


Figure 8. Comparison of absolute displacements (m) of the nodes between Midas and ToyGL

Figure 9 represents the Absolute difference of displacements (%) of the nodes between Midas and ToyGL, where it can be stated that 56 out of 86 nodes have displacement differences of less than 3%. Only the results of six nodes (50, 54, 60, 64, 70, 74) are different in both programs with a percentage higher than 25% (those nodes belonging to the boundary of the lower layer).

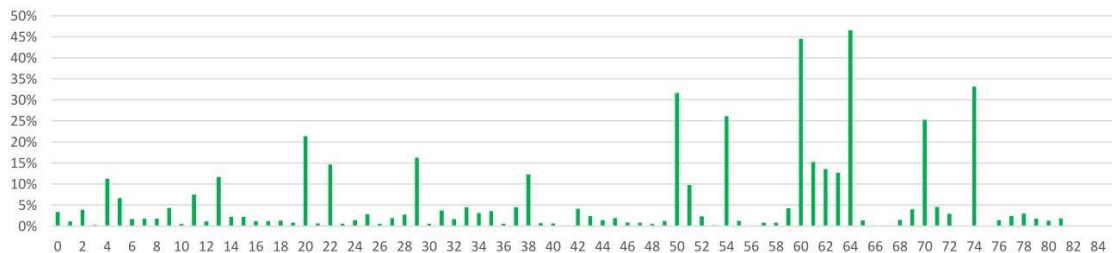


Figure 9. Absolute difference of displacements (%) of the nodes between Midas and ToyGL

5. FOLDABILITY

Foldability, or deployability, is one of the most important advantages of tensegrity structures. Being a type of pin-jointed structures, the elements under compression are, if not isolated, connected with ball joints that act as hinges, permitting folding and stowage in a compact volume. Furthermore, the kinematic indeterminacy of tensegrities is sometimes an advantage. In foldable systems, only a small quantity of energy is needed to change their configuration because the shape changes with the equilibrium of the structure. The analysis of the process for folding and unfolding deployable tensegrities by means of CAD/CAE methods is not straightforward, and it is a challenge that has barely been undertaken until recently.

Before working on the full-scale prototype, some simulations were performed with specialized software tools (ToyGL) to find out the best way to fold the structure (Figure 10). The method to create the mechanisms in the grid, so that it was possible to fold it, was very simple: it was only necessary to untie one end of the contiguous cables that conform each diagonal (10 in total) and each vertical tensor (21 in total).

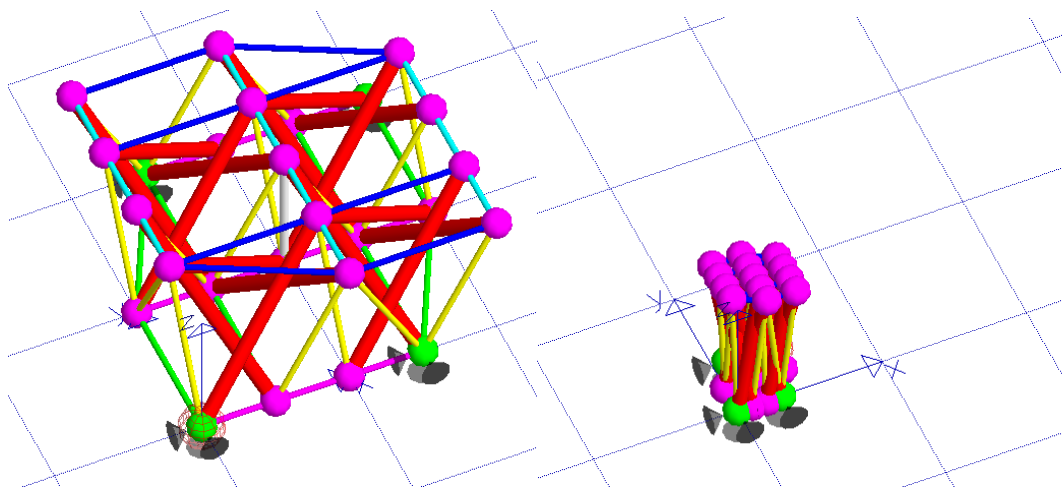


Figure 10. Simulation with software tools: DDLTG 2x2 in its folded and unfolded configuration.

The grid, in its folded and unfolded configuration, can be seen in Figure 10. At the end of the process, the result is a bunch of struts and cables, with a height of a few centimeters more than the length of the struts (1,5 m) and a small diameter.

Thanks to the design of the nodes and grid, the deployment of the grid could also very easy. In the folded configuration, the contiguous cables that make up the diagonal bracing tendons were still inside the holes of the nodes. Therefore, when unfolding the structure, there was no need to re-assemble the connections again because in fact they had not been dismantled.

6. CONCLUSIONS

This paper has briefly described a methodology for the design and calculation of a specific deployable grid, more precisely a DDLTG at full scale (4x4x1 m) composed by 16 modules Quastrut-S of 1x1x1 m.

In order to do so, three CAD/CAE methods were used: ToyGL, Matlab/Octave, and Midas Civil. It was stated that the use of an intermediate platform to exchange data and information between several tools (including AutoCad) was very necessary. For this purpose, the authors created and used ToyGL Processor, based on user-defined functions written in VBA for Microsoft Excel.

The results obtained by ToyGL were compared with those provided by Midas Civil. Both programs are consistent in terms of the definition of geometry, displacements of the nodes, distribution of stresses and forces, reactions, etc. However, for a few elements (especially for those cables that are not carrying a lot of stress) force levels differ slightly. The same happens for certain nodes at the boundary of the lower layer, where displacements calculated by ToyGL and Midas differ more sensibly. This issue proves the difficulties in adapting a general code to all the mechanical characteristics of tensegrity structures, which are very particular due to the high number of infinitesimal mechanisms.

The results of the analysis show that the DDLTG has a correct structural behavior and is stable under self-weight. As it can see in Figure 11, the final stresses of the cables are always under the static limit and the number of slacking cables is not significant.

As a conclusion, the software tools and the methodology for the analysis of tensegrity structures described in this paper have proven to have a high level of reliability, accuracy and consistency.

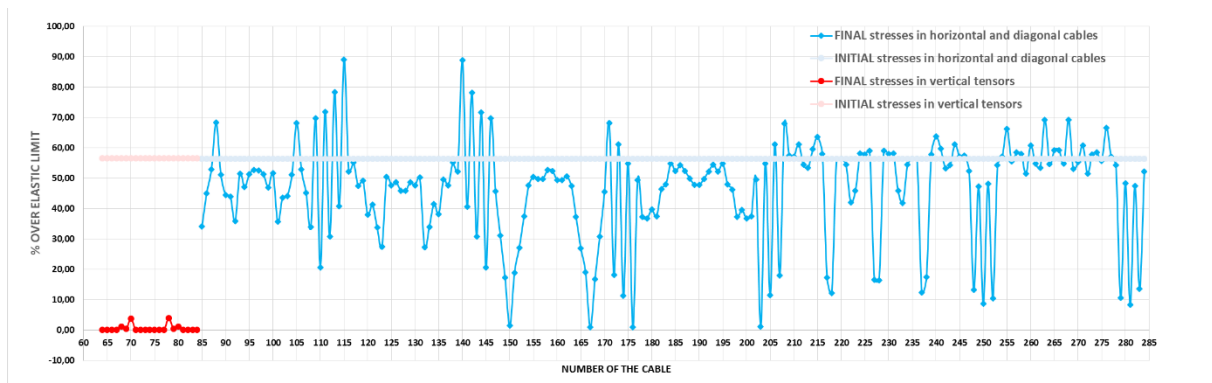


Figure 11. Stress results (% over elastic limit)

ACKNOWLEDGMENTS

The authors are grateful to Landolf Rhode-Barbarigos and Prof. Smith, from IMAC at EPFL for their generosity sharing their useful tools for the analysis of the structures [14], as well as to Dr. Julien Averseng, from LMGC at University of Montpellier, for the valuable help contributed by his software ToyGL [7].

REFERENCES

- [1] Gomez-Jauregui, V. (2015). *CAD/CAE Methods Applied to the Generation of New Double-Layer Tensegrity Grids*. PhD Thesis. Universidad de Cantabria. Santander.
- [2] Juan, S. H., and Mirats Tur, J. M. (2008). "Tensegrity frameworks: Static analysis review." *Mechanism and Machine Theory*, 43(7), 859–881.
- [3] Mirats Tur, J. M., and Juan, S. H. (2009). "Tensegrity frameworks: Dynamic analysis review and open problems." *Mechanism and Machine Theory*, 44(1), 1–18.
- [4] Tibert, A. G., and Pellegrino, S. (2003). "Review of form-finding methods for tensegrity structures." *International Journal of Space Structures*, 18(4), 209–223.
- [5] Zhang, L.-Y., Li, Y., Cao, Y.-P., and Feng, X.-Q. (2014). "Stiffness matrix based form-finding method of tensegrity structures." *Engineering Structures*, 58, 36–48.
- [6] Shekastehband, B., Abedi, K., and Chenaghlou, M. R. (2011). "Sensitivity analysis of tensegrity systems due to member loss." *Journal of Constructional Steel Research*, 67(9), 1325–1340.
- [7] Averseng, J., Quirant, J., and Dubé, J.-F. (2012). "Interactive design and dynamic analysis of tensegrity systems." *International Journal of Space Structures*, 27(Special Issue 2-3).
- [8] Maurin, B., Bagneris, M., and Motro, R. (2008). "Mechanisms of prestressed reticulate systems with unilateral stiffened components." *European Journal of Mechanics, A/Solids*, 27(1), 61–68.
- [9] Pellegrino, S., and Calladine, C. R. (1986). "Matrix analysis of statically and kinematically indeterminate frameworks." *International Journal of Solids and Structures*, 22(4), 409–428.
- [10] Tran, H. C., and Lee, J. (2010a). "Advanced form-finding of tensegrity structures." *Computers and Structures*, 88(3-4), 237–246.
- [11] Tran, H. C., and Lee, J. (2010b). "Initial self-stress design of tensegrity grid structures." *Computers and Structures*, 88(9-10), 558–566.
- [12] Zhang, J. Y., and Ohsaki, M. (2006). "Adaptive force density method for form-finding problem of tensegrity structures." *International Journal of Solids and Structures*, 43(18-19), 5658–5673.
- [13] Zhang, J. Y., and Ohsaki, M. (2007). "Stability conditions for tensegrity structures." *International Journal of Solids and Structures*, 44(11-12), 3875–3886.
- [14] Bel_Hadj_Ali, N., Rhode-Barbarigos, L., and Smith, I. (2011). "Analysis of clustered tensegrity structures using a modified dynamic relaxation algorithm." *International Journal of Solids and Structures*, 48(5), 637–647.
- [15] Gomez-Estrada, G. (2007). "Analytical and numerical investigations of form-finding methods for tensegrity structures." PhD Thesis, Max-Planck-Institut für Metallforschung und Universität Stuttgart, Stuttgart.