**5**

**Case Study**

In this chapter, the developed application will be used to design a lattice tower structure that withstands the same loads and has the same geometric constraints as a case study tower provided by Metalogalva.

In the next subchapters, the model will be presented, along with the steps taken to prepare the optimization process.

Finally, the results will be analysed and conclusions about real world implications will be drawn along with future development paths that could be taken to improve the application.

**5.1. Base Model**

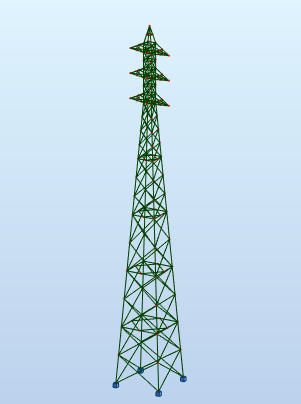


Fig. 5.1 – Base Model

The case study tower is 38 meters high, and carries 7 cables, 3 in each side and one at the tip of the tower. The distance between leg members at the base is 5 meters in both directions and, each arm is 2.25 meters long. The steel used is of type S275. The model provided by Metalogalva has a total weight of 7.616 tonnes, distributed between the different sections as follows:

Table 5.1

|  |  |  |
| --- | --- | --- |
| Sections | Number | Total weight (kg) |
| CAEP 50x5  CAEP 60x6  CAEP 70x7  CAEP 100x10  CAEP 140x13  CAEP 160x15  CAEP 180x6  LP 180x180x15 | 262  44  4  4  4  4  4  8 | 1873  699  123  379  660  870  1046  1967 |
|  |  | 7616 |

Regarding the loads, the base model provided by Metalogalva contained hundreds of load cases. To improve the run time of the optimisation routine, the critical load cases were identified with the help of Metalogalva’s technical department. This interaction with the engineering team allowed the application to run with only 4 critical load cases added to the self-weight of the structure. Given the way the DNA of the structure is generated, this reduction of load cases adds the need of a final inspection of the output model to ensure symmetry. This is a point of further improvement of the program that will be detailed at the end of the chapter.

Figure 5.2 identifies the arm nodes and Table 5.1 shows the forces applied to each of the four critical load cases (LC1 to LC4).

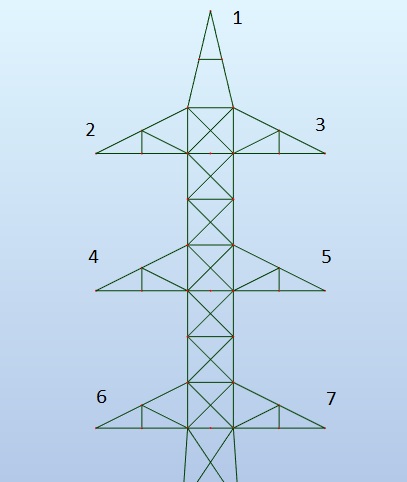


Fig. 5.2 – Arm nodes

Table 5.2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 | Node 6 | Node 7 |
| LC1  (kN) | FX=13.24  FY=1.23  FZ=-3.43 | FX=24.02  FY=0.49  FZ=7.84 | FX=24.02  FY=0.49  FZ=7.84 | FX=24.02  FY=0.49  FZ=7.84 | FX=24.02  FY=0.49  FZ=7.84 | FX=24.02  FY=0.49  FZ=7.84 | FX=24.02  FY=0.49  FZ=7.84 |
| LC2  (kN) | FX=0.0  FY=12.50  FZ=-2.45 | FX=0.0  FY=29.41  FZ=-7.84 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 |
| LC3  (kN) | FX=1.72  FY=12.50  FZ=-2.45 | FX=2.45  FY=22.06  FZ=-4.90 | FX=2.45  FY=22.06  FZ=-4.90 | FX=2.45  FY=22.06  FZ=-4.90 | FX=2.45  FY=22.06  FZ=-4.90 | FX=2.45  FY=22.06  FZ=-4.90 | FX=2.45  FY=22.06  FZ=-4.90 |
| LC4  (kN) | FX=0.0  FY=12.50  FZ=-2.45 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=29.41  FZ=-7.84 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 | FX=0.0  FY=22.06  FZ=-4.90 |

**5.2. Program Setup**

To configure the application to design an optimised structure to withstand the loads listed above and with similar geometrical characteristics, a few steps are needed.

As it is a small tower (38m), the penalty function, which was discussed in section 4., is changed to 0.15 tonnes. A list of sections needs to be provided to the genetic algorithm before starting the optimisation. The sections considered were the following steel equal angles:

* L50x50x5
* L60x60x6
* L70x70x7
* L100x100x10
* L150x150x15
* L180x180x16

The list has 2 fewer sections than the base model. This decision was based on a previous analysis in which it was concluded that there was no need – based on bar utilization factors and the expected optimization ratio – for two intermediate bars between the L 100 and L 180 sections. A single L 150 section was used instead of the L 140 and L160.

This reduction of available sections also allows the program to run faster as the search space is reduced. However, there is a risk that the output structure still has penalties applied, in other terms, some bars need to have larger sections that are not available. To address this issue, a log file containing the list of bars that need to be strengthened (either by secondary bracing or larger sections) is produced for the final structure.

The load cases described in Section 5.1 were added using the Robot UI and for each load case the self-weight of each bar was added. This additional load is automatically updated by Robot in each iteration.

**5.3. Results**

The optimization ran for 14 hours in a computer with 8GB of RAM and 8 cores before the termination criteria of the genetic algorithm was met.

During runtime, several debug logs were collected and they revealed a bottleneck present in Robot API. The function used to update bar properties for each individual evaluation was responsible for nearly 70% of the runtime of the entire optimization routine. Such delay in this operation points to a possible limitation in the API to handle fast updates in models with several bars. This is a problem that will need to be addressed if the present software is to be used to optimize larger structures.

5.3.1. Postprocessing

Given the steps taken to reduce the number of load cases, symmetrical load cases were removed, only loads critical to the upper right quadrant (highlighted in figure 5.3) of the tower were kept. This meant the output would be a non-symmetrical structure where that upper right quadrant would need to be reproduced on the remaining three corners of the structure. In figure 5.4 the critical quadrant is displayed in isolation, next to the final symmetrical structure before any required strengthening work was performed.

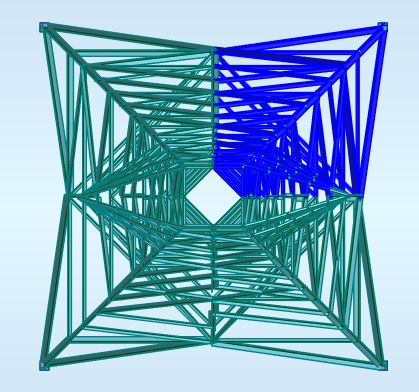


Fig. 5.3 – Plan view

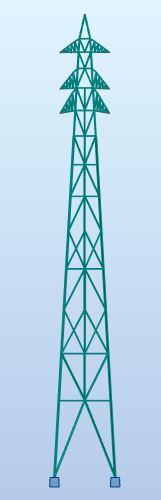
  

Fig. 5.4 – Critical quadrant, front and side planes

After the symmetry operations, the log file was opened to see if any bars needed additional strengthening. In this case, all the leg members were listed as well as a few bars in the middle of the structure with U/f slightly higher than 1.0. Given the locations of the elements to strengthen, secondary bracing – with section L 40x40x5 – was used instead of larger sections.

Figure 5.5 details how the leg members were strengthened. A similar triangulation method was used for the other bars on the list.

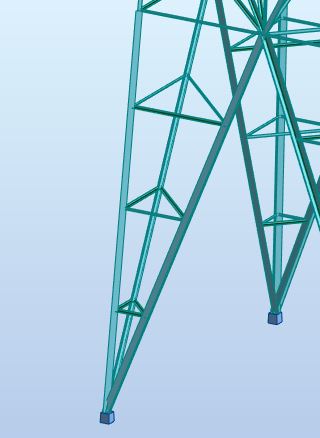


Fig. 5.5 – Legs with secondary bracing added

5.3.2. Analysis

The final structure (Fig 5.6) has a total weight 6.8 tonnes, which corresponds to a 10.5% material reduction. The weight is distributed between the different sections as follows:

[INSERIR QUADRO] Table 5.3

Adding to the material savings, the list of sections used is also smaller, allowing for more efficient manufacturing.

Given the current concerns about sustainability, savings will be analysed in terms of reduction of CO2 emissions by using less steel. The national grid of the United Kingdom will be used as the data needed for the analysis is publicly available.

According to the European Strategic Energy Technologies Information System, in 2012 the European steel industry emitted 2.3 tonnes of CO2 per tonne of steel. The report also identifies paths for improvement that could reduce this value by 70% to 0.7 tonnes of CO2 per tonne of steel. For this analysis, the best-case scenario (the improvements were successfully applied to the European steel industry) will be used.

Data from the National Grid (UK) website points that there are 88000 electricity pylons in the UK and their average weight is 30 tonnes.

Using this information if a 10% reduction in material usage was applied to every pylon, 2 376 000 tonnes of CO2 could be saved on the entire grid.

To make some sense of these numbers, data from the Environment Protection Agency (US) was used to compare these emissions with other activities that equate to similar emission levels, 2 376 000 is equivalent to:

* 501 891 Passenger vehicles driven for one year;
* 11920 round trips to the moon in a passenger vehicle;
* 754 038 Tons of waste recycled instead of landfilled;
* 1 010 million litres of petrol burned;
* 600 installed wind turbines;
* Energy to power 250 000 homes for a year;
* 70 % of the yearly emissions of a coal-fired power plant.

**5.4. Future Work**

This first version of the program validated the use of genetic algorithms to automate to a certain degree the design of optimised lattice tower structures, returning gains in productivity and material efficiency. During development and testing, some points to improve in a future iteration of the software were also identified.

The Genome class, responsible for the DNA generation should be updated to ensure symmetry of the structure even when loads are reduced, as they were in the case study, to its most critical loads for a specific quadrant. This would shorten the execution time of the optimisation by reducing the number of load cases applied to the structure (by removing the symmetrical LCs). This avoid the need for human operations to re-establish symmetry of the solution. This could be implemented by defining a master quadrant, from which every bar in the remaining three corners of the structure would inherit its properties. The same approach should also be applied to the nodes.

When scaling up to optimise larger structures a constraint in the Robot API was identified. The calls needed to update bar properties between each individual evaluation take too much time when compared to other calls such as the ones responsible for running and retrieving results from the structural analysis. In fact, when the bar count increases above a certain number, the communication between the developed program and Robot are slow enough to make the application stop responding. In a future version, changing the Robot\_call class to work with another structural analysis package such as OpenSees or Oasys GSA will deliver higher performance and will enable more complex optimization problems to run.

Finally, to address the need for human intervention to read the log file and add secondary bracing to the model, a future academic work could study the implementation of a neural network to read input from that file and the structure, and automatically apply the needed bracing elements.

<https://setis.ec.europa.eu/related-jrc-activities/jrc-setis-reports/energy-efficiency-iron-and-steel-industry-technology> (06/06/2017)