LABVIEW APPLICATION FOR THE MOMENT OF INERTIA CALCULUS

CIOCONEA Adina-Mariana[1] and SPÂNU Paulina[2]
[1]Faculty of Industrial Engineering and Robotics, Specialty: Industrial Engineering, 1st year,
e-mail: adina.cioconea@stud.fiir.upb.ro
[2]Faculty of Industrial Engineering and Robotics, Manufacturing Engineering Department,
University POLITEHNICA of Bucharest

ABSTRACT: The moment of inertia is an essential quantity, needed in many engineering areas and also studied, especially on the mechanical and engineering educational institutes. The following LabVIEW Virtual Instrument (VI) is able to compute the moment of inertia, the center of mass and also the area of a specific complex 2D figure with variable dimensions. The VI algorithm uses classical theorems in order to determinate the previous mentioned quantities.

KEYWORDS: LabVIEW, center of mass, moment of inertia, area

1. Introduction

The determination of the moment of inertia (MOI) is of fundamental importance in various engineering domains. It represents a quantity that determines the torque needed for a desired angular acceleration about a rotational axis, akin to how mass determines the force needed for a desired acceleration. The ability to precisely determine the moment of inertia allows the engineer to properly size their components while achieving the high-performance demands, e.g., in the aerospace and defense industries. In other cases, the measurement of MOI can be used to verify that manufacturing and assembly tolerances and processes goals are nominal. In this paper the author conceived a program in LabVIEW software capable to calculate the moment of inertia of a specific complex 2D figure with various dimensions, along with its center of mass and the area. The conception process of the application presented in the following required a fundamental understanding of the algorithm that implies breaking the formulas from a simple, general figure to a composed one.

2. Understanding the problem requirements

The given problem requires the moment of inertia of a 2D figure, with given dimensions. The figure is an irregular one, formed by various basic forms (rectangle, triangle).

The calculus of the moment of inertia requires the center of mass and the area of each individual component of the figure, therefore it is necessary to establish the basic figure components of the main figure and if the basic figure is added or subtracted.

The main figure contain: one rectangle (added), one smaller rectangle(subtracted), two triangles (subtracted)

3. Establishing the inputs and outputs

Resulting by the analyzation of the problem the following inputs are established (see figure 1), followed by their insertion in the VI in the form of a numeric controller (see figure 2):

- Height
- Gap height
- Gap width
- Base gap

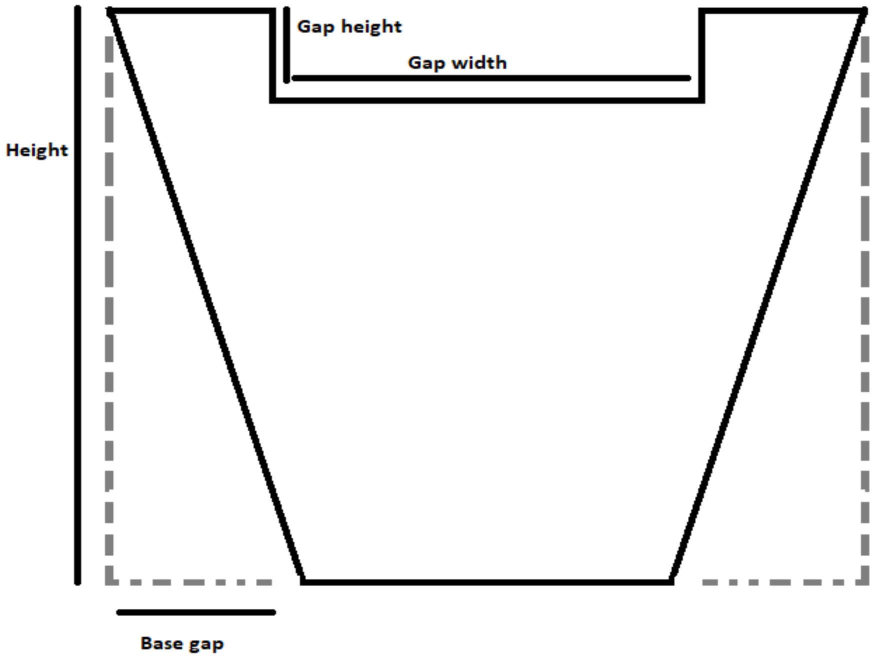

Fig. 1. 2D Figure (inputs notation)

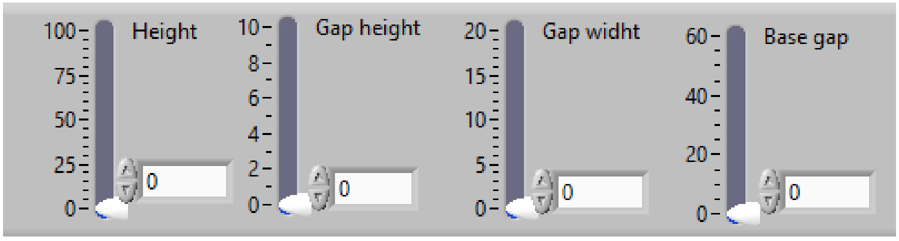

Fig. 2. Inputs in form of numeric controllers

Also, from the requirements of the problem we can detect the main output moment of inertia, but also the intermediary outputs (table 1), and represented as numeric indicators in the VI (see figure 3):

Table 1. Outputs

Figure name	Area	Center of gravity
2D figure	Total area	Center of gravity
Big rectangle	Area rectangle	CGV rectangle
Gap rectangle	Area top-gap	CGV top-gap
Side triangle	Area side-gap	CGV side-gap

Fig. 3. Outputs in form of numeric indicators

4. Equations and algorithm applied in the VI

The necessary formulas to determine the moment of inertia are [1] :

- Center of gravity (1)
- Steiner's formula for moment of inertia (2)

$$Z_c = \frac{\sum A_i Z_i}{\sum A_i} \tag{1}$$

Z_c= center of gravity for the entire figure [mm];
Z_i= center of gravity for one basic figure [mm];
A_i=The area of one basic figure [mm^2]

$$I_c = \sum I_i + \sum (Z_i - Z_c)^2 A_i \tag{2}$$

I_c= moment of inertia for the entire figure [mm^4];
I_i= moment of inertia for one basic figure [mm^4];

The algorithm of the VI implies the following steps:

1. Breaking down the figure in basic figures: one big rectangle and one small rectangle and two triangles that will be subtracted from the big rectangle.
2. Computing the Center of gravity
   a. Areas for each figure
   b. Center of gravity for each figure
3. Computing the moment of inertia

The LabVIEW permits different ways of implementation, one of them is the one using the basic mathematical functions of the LabVIEW (see figure 4), but considering the amount of work and the difficulty on debugging, a more optimal implementation was provided, using Formula Node (see figure 5)

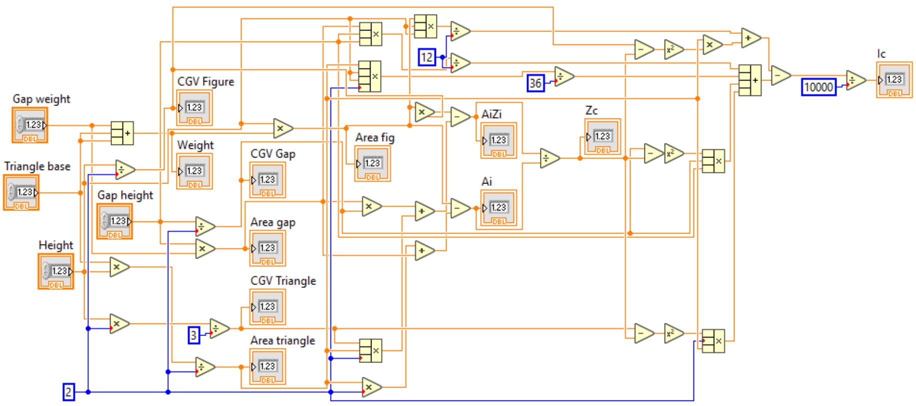

Fig. 4. Formulas implementation using the basic numerical functions

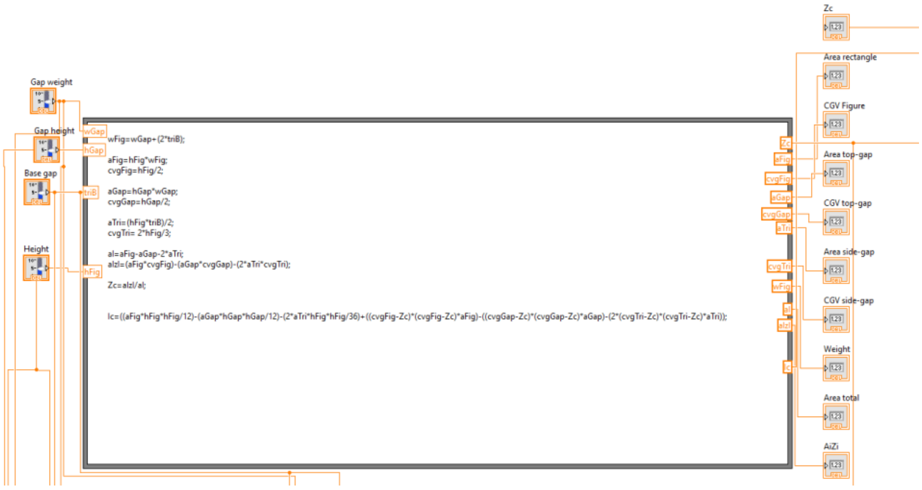

Fig. 5. Formulas implementation using Formula Node

5.Converting and displaying the result

The moment of inertia is calculated in mm^4 by default, but the in order to change the measurement unit another controller was added.

The results are presented in a table, created by a two-dimensional array, including the moment of inertia with the required unit and the center of gravity in mm (see figure 6), the array contain t=data of type string.

The table was created by converting the results from the numeric form in a formatted string concatenated to another given string adjusted to the unit required and adding the specific values to the array in the respective form (see figure 7)

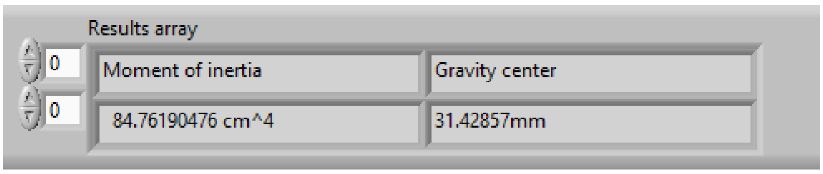

Fig. 6. Results table

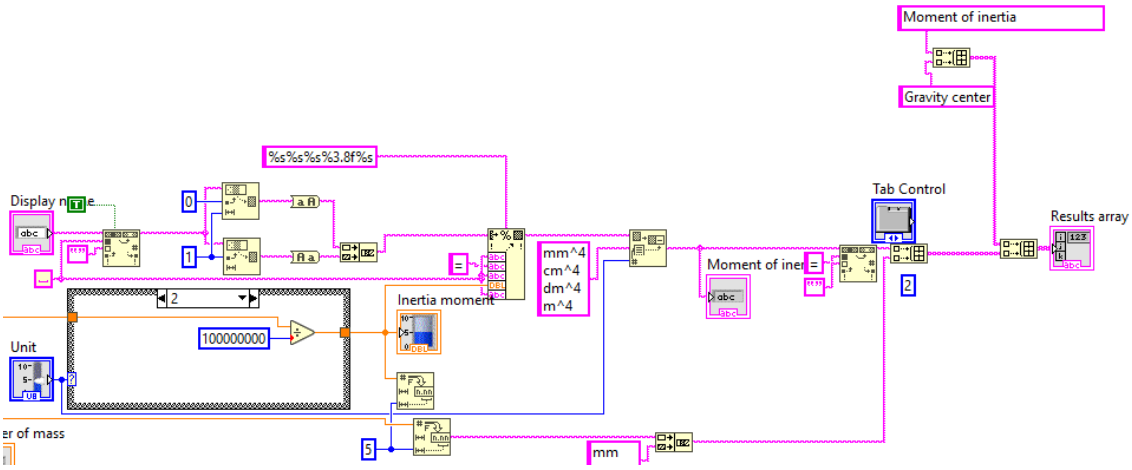

Fig. 7. Converting and formatting the results

6. Designing the figure using 2D picture and XY Graph

LabVIEW perming the visual representation in multiple ways. For this project were used 2 methos of visual representation:

a) 2D image, that also permit a more dynamic approach for the visualization (see figure 8)
   The process of constructing this image implies:
   - Calculating the coordinates of each point
   - Grouping each point coordinates by the Bundle function
   - All point were transmitted to the image using the function Build Array

Also, the center of gravity was added together with the text and in order to give dynamic to the image, there were used a While loop structure, that was the center of migrates from the top of the image to its place

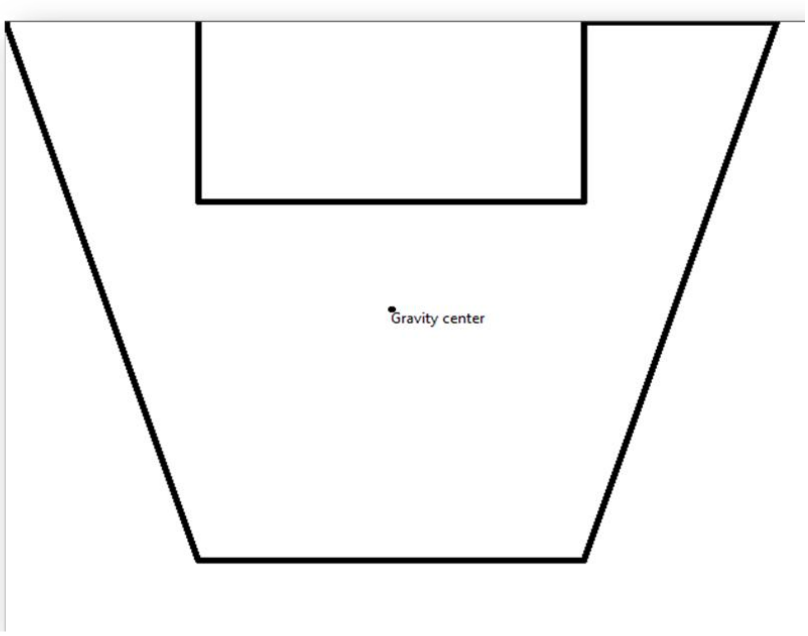

Fig. 8. 2D Image representation

b) a XY Graph representation, that permits the visualization of the system of axis and the scaling (see figure 9)
   The process of constructing this graph implies:
   - Computing all x coordinates for each point and grouping them in an array, using the function Build Array
   - The same procedure for the y coordinates
   - The coordinates were transmitted to the XY Graph using the Bundle function

   Also, the arrows for the graph were added using a SubVI.

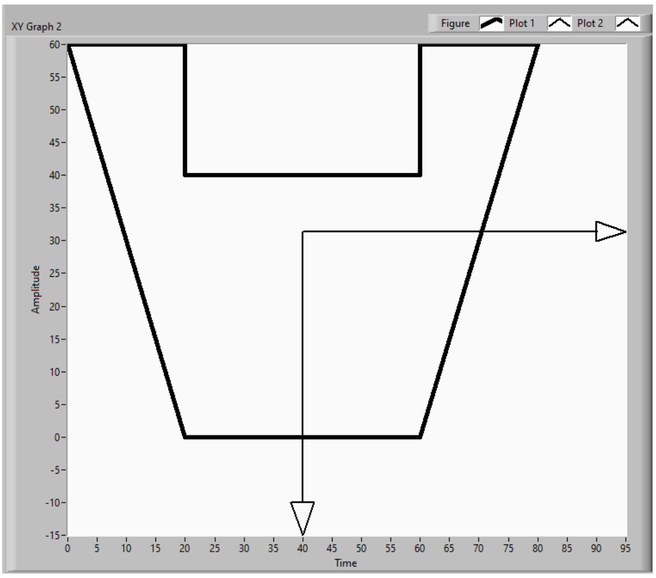

Fig. 9. XY Graph representation

7. Current state

In the present the application has the following characteristics:

- Receiving the data from the user, representing the dimensions of the respective figure
- Computing the moment of inertia, along with the intermediary necessary quantities as: areas of the individual parts of the figure, the general area, the center of mass of each individual parts of the figure, the general center of gravity on the Z axis
- Displaying the moment of inertia in the required unit measure and in a specific format.
- Displaying the figure in a XOY Graph along with the center of mass, along with the scale for each axis according with the dimension provided by the user.
- Displaying the figure in a 2D image form and mobile center of mass according with the dimension provided by the user.

8. Conclusion

In conclusion the presented VI represents a tool that compute mainly the moment of inertia and consequently, the area and the center of mass of a specific complex figure. Also, it permits visualization of the figure in two different forms using the dimensions given by the user.

As future resolutions it is intended to add the function to export the results into a text file in a specific folder and to read the inputs from a text file.

9. Bibliography

[1]. Jiga, G.(2014) Strength of Materials,Volume 1, Ed. PRINTECH, Bucureşti, ISBN 978-606-23-0168-2

[2]. Savu T., Spânu P., Abaza B.(2014), Reprezentări grafice – îndrumar de laborator, Ed. PRINTECH, Bucureşti , ISBN 978-606-23-0230-6

[3]. Savu T., Spânu P., Abaza B.(2014), Algoritmi – îndrumar de laborator, Ed. PRINTECH, Bucureşti , ISBN 978-606-23-0229-0

[4]. Savu T., Spânu P., Abaza B.(2018)Bilingual Laboratory Guide (Romanian -English)Computer Programming, Ed. Printech Bucuresti , ISBN 978-606-23-0819-3