

# Automated Determination of a Package's Center of Mass

Ayaz Hemani  
*University of South Florida*

Advisors:

Gerald Hefley, Mathematics and Statistics  
Ben Lynch, United Parcel Service of America

Problem Suggested By: Ben Lynch

Follow this and additional works at: <https://scholarcommons.usf.edu/ujmm>

 Part of the [Mathematics Commons](#)

UJMM is an open access journal, free to authors and readers, and relies on your support:

[Donate Now](#)

## Recommended Citation

Hemani, Ayaz (2010) "Automated Determination of a Package's Center of Mass," *Undergraduate Journal of Mathematical Modeling: One + Two*: Vol. 3: Iss. 1, Article 3.

DOI: <http://dx.doi.org/10.5038/2326-3652.3.1.15>

Available at: <https://scholarcommons.usf.edu/ujmm/vol3/iss1/15>

---

# Automated Determination of a Package's Center of Mass

## **Abstract**

In order to address the issue of increased efficiency and better planning for parcel shipments, an automated computer program was developed in Microsoft Excel that calculates center of mass and moments of mass with greater speed and reliability than currently implemented systems. This simple program requires only a variable density function and limits of integration for a given object as input within the spreadsheet system. Once the required input has been provided, a series of chain calculations, with the help of a Visual Basic for Applications (VBA) script, is able to process the input, which is done through integration and a Riemann sum. Furthermore, the foundation of the program can also be used for calculating other physical quantities of interest such as the moment of inertia or surface area of an object.

## **Keywords**

Center of Mass, Parcel Packing, Postal Automation

## TABLE OF CONTENTS

Problem Statement.....	3
Motivation.....	3
Mathematical Description and Solution Approach .....	3
Discussion .....	5
Conclusion and Recommendations.....	6
References.....	6
Appendices .....	7

## PROBLEM STATEMENT

Given the density function and general dimensions of an object, provide a fast, reliable, and efficient method for finding its center of mass.

## MOTIVATION

The objective of this project is to assist the United Parcel Service of America in finding a reliable set of data which relates the mass of a system to its location. Due to the practical applications of the concept of comparing the centers of mass of various objects, we have come to enjoy the fruits of modern day shipping and handling. Comparing centers of mass is also effective in product packaging, product handling, and product placement within vehicles of transport.

## MATHEMATICAL DESCRIPTION AND SOLUTION APPROACH

### I. MASS OF A PLANAR LAMINA OF VARIABLE DENSITY.

Given a continuous density function  $\rho(x, y)$  one can integrate over an area  $A$  of region  $R$  to find the mass of an object given by

$$m = \iint_R \rho(x, y) dA. \quad (1)$$

(Larson, Hostetler and Edwards)

### II. MOMENTS OF MASS OF A PLANAR LAMINA OF VARIABLE DENSITY.

Given  $\rho(x, y)$  as the continuous density function on the planar lamina  $R$ , the moments of mass are

$$M_x = \iint_R y \rho(x, y) dA \quad \text{and} \quad M_y = \iint_R x \rho(x, y) dA \quad (2)$$

with respect to the  $x$ -axis and the  $y$ -axis. (Larson, Hostetler and Edwards)

### III. CENTER OF MASS OF A PLANAR LAMINA OF VARIABLE DENSITY.

Given  $m$  as the mass of the lamina, the center of mass is

$$(\bar{x}, \bar{y}) = \left( \frac{M_y}{m}, \frac{M_x}{m} \right). \quad (3)$$

(Larson, Hostetler and Edwards)

### IV. RIEMANN SUMS.

An integral of a function can be estimated by using a Riemann sum  $\sum_{i=1}^n f(c_i) \Delta x$ , where the width of the subinterval,  $\Delta x$ , is multiplied by the sum of the evaluations of  $f(c_i)$ , where  $c_i$  is any point within the subinterval. (Larson, Hostetler and Edwards)

The process of finding an approximation for the mass, moments of mass, and center of mass begins with the continuous density function  $\rho(x, y)$ . Once the user has provided  $\rho(x, y)$  and the limits of integration for the first and second integral in both the Program Interface and Input Sheet (see Appendix B), the program links to data that is stored on the Integral and Riemann Sum Computation Sheet (see Appendix C). From there, depending on the first variable of integration, the respective power for the variable is increased within  $\rho(x, y)$  and divided by the term's coefficient. The respective variable for the first integration is then replaced by the upper and lower bounds of integration as is depicted in the equation  $\int_a^b f(x) dx = F(b) - F(a)$ , where  $F(x)$  is the integrated function of  $f(x)$ . This results in a computation for the first integral of the continuous density function.

In order to obtain an estimate for the second integral, a Riemann sum may be used. An interval size of 64 was chosen for each Riemann sum. Keeping this in mind we obtained  $\Delta x$  by taking the difference in the secondary limits of integration (should be constants) and dividing it by 64. The computations for  $c_1$  were then completed by adding  $\Delta x$  to the lower limit of

integration until 65 values were obtained and  $c_n$  equaled the upper limit of integration. Once the  $c$  values were found, they were put into the integrated uniform density function, by replacing the variable of second integration. These equations were solved, summed, and multiplied by  $\Delta x$  to find the Riemann sum, which gives us a very close approximation for the evaluated integral. This evaluated integral then gives the mass of the object.

To find the moment of mass with respect to the  $y$ -axis for the uniform density function, we must reset the function and repeat the process while also increasing the power of the  $x$ -variables before any computations are made. The process is similar for finding moment of mass with respect to the  $x$ -axis except that the  $y$ -variables are increased. The center of mass is calculated by dividing the moment of mass by the mass.

## DISCUSSION

At the onset of the project, the Java programming language was used, but was ultimately abandoned in favor of the much more transparent Excel environment. Through using Excel it was soon realized that the built-in functions would not suffice because a loop was necessary to find the moments of mass after the mass was found. For this reason, macros needed to be created with Virtual Basic Applications (VBA) in order to loop functions and to store the value for the string functions computed by the Riemann sum (see Appendix A). In order to compute the mass, moments of mass, and center of mass for a different function when the continuous density function was changed, it was required that the program be re-opened in order for the `Workbook_Open()` event to be triggered. To correct this issue, a re-calculate button was created, which is referenced to the `Recalculate()` method in VBA (see Appendix A and Appendix B).

## CONCLUSION AND RECOMMENDATIONS

Using the formulas and processes described above, an approximation of the mass, moments of mass, and center of mass have been successfully made. Although the percent error due to the Riemann sum may be as high as five percent, the program could be made more accurate by adding more intervals within the Riemann sum. Recommendations for variations and improvements of this project include increasing the accuracy of the second integral by increasing the number of intervals (which would decrease interval width), increasing the complexity of the accepted terms for the continuous density function, and increasing efficiency within the VBA code to incorporate PasteSpecial's of mass ranges to avoid code clutter. Other improvements to the program include incorporating the second moments, or moments of inertia, onto the Program Interface and Input Sheet.

## REFERENCES

Larson, Ron, Robert Hostetler and Bruce Edwards. Calculus. 8th Edition. Boston, MA: Houghton Mifflin Company, 2005.

## APPENDICES

### APPENDIX A: VBA CODE

```

Private Sub Workbook_Open()

Call Recalculate

End Sub

Public Sub Recalculate()

Sheets("Sheet1").Range("E3").Value = Sheets("Sheet2").Range("G4").Value
Sheets("Sheet1").Range("E4").Value = Sheets("Sheet2").Range("G5").Value
Sheets("Sheet1").Range("E5").Value = Sheets("Sheet2").Range("G6").Value
Sheets("Sheet1").Range("H3").Value = Sheets("Sheet2").Range("J4").Value
Sheets("Sheet1").Range("H4").Value = Sheets("Sheet2").Range("J5").Value
Sheets("Sheet1").Range("H5").Value = Sheets("Sheet2").Range("J6").Value

Call RefreshData
Sheets("Sheet2").Range("E16").Value = Sheets("Sheet1").Range("E94").Value

Sheets("Sheet1").Range("E3").Value = Sheets("Sheet2").Range("G4").Value
Sheets("Sheet1").Range("E4").Value = Sheets("Sheet2").Range("G5").Value
Sheets("Sheet1").Range("E5").Value = Sheets("Sheet2").Range("G6").Value
Sheets("Sheet1").Range("H3").Value = Sheets("Sheet2").Range("J4").Value
Sheets("Sheet1").Range("H4").Value = Sheets("Sheet2").Range("J5").Value
Sheets("Sheet1").Range("H5").Value = Sheets("Sheet2").Range("J6").Value

Sheets("Sheet1").Range("E3").Value = (1 + Sheets("Sheet2").Range("G4").Value)
Sheets("Sheet1").Range("E4").Value = (1 + Sheets("Sheet2").Range("G5").Value)
Sheets("Sheet1").Range("E5").Value = (1 + Sheets("Sheet2").Range("G6").Value)

Call RefreshData
Sheets("Sheet2").Range("E17").Value = Sheets("Sheet1").Range("E94").Value

Sheets("Sheet1").Range("E3").Value = Sheets("Sheet2").Range("G4").Value
Sheets("Sheet1").Range("E4").Value = Sheets("Sheet2").Range("G5").Value
Sheets("Sheet1").Range("E5").Value = Sheets("Sheet2").Range("G6").Value
Sheets("Sheet1").Range("H3").Value = Sheets("Sheet2").Range("J4").Value
Sheets("Sheet1").Range("H4").Value = Sheets("Sheet2").Range("J5").Value
Sheets("Sheet1").Range("H5").Value = Sheets("Sheet2").Range("J6").Value

Sheets("Sheet1").Range("H3").Value = (1 + Sheets("Sheet2").Range("J4").Value)
Sheets("Sheet1").Range("H4").Value = (1 + Sheets("Sheet2").Range("J5").Value)
Sheets("Sheet1").Range("H5").Value = (1 + Sheets("Sheet2").Range("J6").Value)

Call RefreshData
Sheets("Sheet2").Range("E18").Value = Sheets("Sheet1").Range("E94").Value

Call RefreshData

End Sub

Public Sub RefreshData()

Sheets("Sheet1").Range("G27:G91").Copy
Sheets("Sheet1").Range("E27").PasteSpecial Paste:=xlPasteValuesAndNumberFormats

Application.CutCopyMode = True

```



```
Sheets("Sheet1").[E90].Formula = "=" & Sheets("Sheet1").[E90].Formula
Sheets("Sheet1").[E91].Formula = "=" & Sheets("Sheet1").[E91].Formula
```

```
End Sub
```

## APPENDIX B: PROGRAM INTERFACE AND INPUT SHEET

	A	B	C	D	E	F	G	H	I	J
1	Automated Determination of A Package's Center of Mass									
2										
3					Coefficient		Xpower			Ypower
4		Density Function	=	1	$x^4$	4	*	$y^7$	7	
5			+	2	$x^5$	5	*	$y^8$	8	
6			+	3	$x^6$	6	*	$y^9$	9	
7										
8										
9		1st Integral				2nd Integral				
10										
11		Variable of Integration		y		Variable of Integration		x		
12		Upper Limit of Integration		$4-x^2$		Upper Limit of Integration		2		
13		Lower limit of Integration		0		Lower Limit of Integration		-2		
14										
15										
16			Package Mass =	21884.588					INPUT FIELDS	
17		Moment of Mass with respect to the y-axis =	3676.0927							
18		Moment of Mass with respect to the x-axis =	59838.126						Re-Calculate!	
19		Center of Mass: ( $\bar{x}$ , $\bar{y}$ ) =	0.1679763	2.734259						
20										

