

Matlab GUI for Elementary Flows as an Educational Tool

Gabriel A. Heredia Acevedo, Bernardo Restrepo, and Jonathan Holguino

Polytechnic University of Puerto Rico

Abstract

Elementary flows in fluid mechanics are used to describe more complex flow patterns in an incompressible condition assumption. These flows arise from the conservation of mass and eventually leads to Laplace's linear differential equation applied to the fluid's streamline. Typically, these topics are taught without the availability of an advanced technological tool that may aid the teaching/learning process with practical application. These concepts are required to compute pressure distribution, typically used in the design of airfoils and other arbitrary shapes, which are of main interest in the aerospace and naval industries. A Graphic User Interface (GUI) has been developed to assist in the learning process by providing a practical, visual application in which students, teachers, and professionals alike may utilize to properly model the fluid's conditions of speed and shape for any desired design, as well as to learn the fluid mechanics concepts which are presented in these conditions. Furthermore, this GUI allows users to input any desired elementary flow into any arbitrary position and strength, so that the user may see the resulting flow pattern due to its presence, while providing the theoretical information and differential equations that represent said elementary flow. Additionally, the user is able to input or to remove any desired amount of elementary flows into the same pattern, and the system would automatically compute the resulting flow pattern. The tool provides the proper explanations as to how the results are achieved.

Keywords

Elementary flow, Matlab, vortex, doublet, and sink

Introduction

The aerospace and naval industries require professionals who master fluid mechanics so that these may find solutions to their design problems in fluid dynamics. Given this need, university students exposed to these subjects need to fully comprehend the physical phenomena occurring when designing for any given conditions.

Comprehension of elementary flows topic is necessary for applications such as the design of airfoils and hydrofoils. The key idea is that if the flow speed distribution is obtained some distance from the object and afterwards, the pressure distribution around the object's surface may also be obtained. The elementary flows describe such velocity fields in an incompressible and irrotational flow condition, and the use of this theory may also be expanded to flow conditions in which the fluid is considered compressible.

To serve the purpose of aiding the teaching process of elementary flows a Graphic User Interface (GUI) was developed so that students may be exposed to visual examples for different flow

patterns. The interface permits the users to visualize the flow patterns accompanied with the theory that supports it.

Basic Theory for Elementary Flows

An elementary flow, as defined by J. D. Anderson Jr.¹, is any flow that complies with the conditions of being incompressible and irrotational. Now what exactly is an incompressible and irrotational flow?

An incompressible flow is any flow in which the particles' density remains constant. The density is the mass of the particle per unit volume, where ρ (rho) is the density, m is the mass, and V is the volume. If the mass of a particle was fixed and it would be moved through an incompressible flow, then this particle must also have a constant volume (that is $\int \delta V$ is equal to V , where V is constant) is to comply with the constant density condition. Mathematically, this is,

$$m = \int \rho \delta V$$

Now, as defined by Anderson Jr.¹, if the volume of the flow is to remain constant and the volume under question is sufficiently small so that it only contains one particle (that is, $V_{\text{particle}} = \delta V$), then it follows that the time rate of change of volume of a fluid element per unit volume should be as follows,

$$\frac{1}{\delta V} \frac{D(\delta V)}{Dt} = 0$$

It then follows that the divergence of velocity must be equal to the time rate of change of volume of a fluid element per unit volume. We define u and v as the velocity vector components in the x axis and y axis, respectively. Now remember that the divergence of velocity is the derivative of all the velocity components with respect to their own direction. In an x - y plane this is,

$$\nabla \cdot \mathbf{V} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = \frac{1}{\delta V} \frac{D(\delta V)}{Dt} = 0$$

The second condition requires that the flow should be irrotational. This condition is given by computing the curl of the velocity vector field. For this case, the flow's curl should be equal to zero. The curl for the flow's velocity vector field is given by the alternate form of its derivative. This is,

$$\nabla \times \mathbf{V} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = 0$$

If we were to define a function in so that its gradient defines the velocity of a flow field, we may expand our definition for any particle moving through an incompressible and irrotational flow. As Anderson¹ named it, the velocity potential is denoted with the Greek letter ϕ (phi) and shall

serve this purpose. If we were to combine this with the curl and the divergence of a vector field, we obtain the following,

$$\nabla \times (\nabla\phi) = \frac{\partial}{\partial y} \left(\frac{\partial\phi}{\partial x} \right) - \frac{\partial}{\partial x} \left(\frac{\partial\phi}{\partial y} \right) = 0$$

And,

$$\nabla \cdot (\nabla\phi) = \nabla^2\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0, \quad \text{in cartesian coordinates}$$

$$\nabla^2\phi = \frac{\partial}{\partial r} \left(r \frac{\partial\phi}{\partial r} \right) - \frac{1}{r} \frac{\partial}{\partial\theta} \left(\frac{\partial\phi}{\partial\theta} \right) = 0, \quad \text{in cylindrical coordinates}$$

Note that the first equation only shows compliance with the irrotational condition and the second equation, which shows the incompressibility condition, is also Laplace's differential equation. Anderson¹ also defined that the streamlines (denoted as ψ) for any flow pattern under these conditions must also satisfy Laplace's equation. That is,

$$\nabla^2\psi = 0$$

After solving Laplace's equation the required boundary conditions evaluated at infinity requires that the velocity along the Uniform Flow direction must be equal to the Uniform Flow's speed and be equal to zero perpendicular to this direction. Let x' and y' be these corresponding directions and u' and v' be their corresponding speeds. Also let V_∞ be the Uniform Flow's speed. Then we have,

$$u' = \frac{\partial\phi}{\partial x'} = \frac{\partial\psi}{\partial y'} = V_\infty$$

$$v' = \frac{\partial\phi}{\partial y'} = -\frac{\partial\psi}{\partial x'} = 0$$

The Interface

To use the program we only need to run the script which holds the GUI. Upon running we will find the following interface.

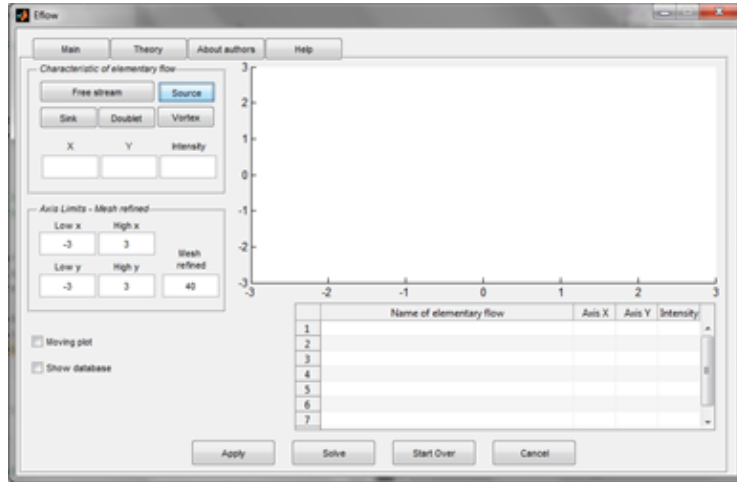


Figure 1 Interface “Main” tab

The program opens directly to the main tab where we can input our desired elementary flows and watch their behavior. The GUI contains four tabs that we may use to gain access to different tools and information. These are the Main tab, Theory tab, About Authors tab, and Help tab. The tabs are described as follows.

Main – This tab is where the user may input the elementary flows and observe their behavior. It contains four major sections, “Characteristics of Elementary Flow,” “Axis Limits – Mesh Refinement,” “Flow Information Table,” and “Flow Vector Plot.” This tab may be used with the Theory tab to assist the student in understanding the concept of elementary flows.

Theory – This tab provides the user with the derivation of each of the elementary flows, as well as the discussion of a few special cases. The information included in each flow also incorporates the required definitions, solution to the Laplace differential equation for velocity potential, the resulting streamline function, and an example. The Theory tab also contains the basic definition of what an elementary flow is and the required conditions for any specific flow to be considered as such. This is intended to be used with the Main tab to aid in the understanding of the course material and general flow behavior.

About Authors – This tab contains information regarding the GUI’s creators, as well as contact information, when available. Additionally, the users may also find information regarding the institution where this GUI was developed.

Help – This tab contains information concerning the GUI structure, tutorials, GUI version information, as well as troubleshooting information.

The Main tab is where the user will input all the required information to observe and understand the flow behavior. Once the user inputs all the necessary information, he/she only needs to press

solve and the program will refresh itself and show a normalized vector plot. The tab is divided into the four major sections previously mentioned. These sections are now described as follows.

Characteristics of Elementary Flow – This section is where the user adds each flow to obtain the desired flow pattern. Notice that this section contains buttons labeled “Free Stream,” “Source,” “Sink,” “Doublet,” and “Vortex.” The “Source” and “Sink” flows are redundant in behavior, but opposite in vector direction. This redundancy is intended to demonstrate to the user that for any of these two flows, he/she may obtain the opposite flow by simply inputting a negative strength. Additional to the buttons the user will find that for the “Free Stream” button, the available inputs are X Axis Component (Flow speed parallel to X Axis) and Y Axis Component (Flow speed parallel to Y Axis). The rest of the flow buttons contain the following inputs: Intensity (The strength for the desired flow), X Coordinate (The origin for the flow along the X Axis), and Y Coordinate (The origin for the flow along the Y Axis).

Axis Limits – Mesh Refinement – This section is used to set the plot axis limits. The user will find the following inputs: “Low X” (Lower X Axis limit), “High X” (Higher X Axis limit), “Low Y” (Lower Y Axis limit), “High Y” (Higher Y Axis limit), “Mesh Refinement” (How many points are plotted in each axis).

Flow Information Table – This section shows the database that incorporates the specific elementary flows input into the current plot. The user may see the name of the specific flow, its origin, and intensity. If the input is a uniform flow, the intensity will be input into Speed X and Speed Y boxes. The user may also remove any specific flow in this section and then select solve to refresh the normalized vector plot.

Flow Vector Plot – This section is essence of the GUI. This section shows the input flow behavior in a normalized vector plot. By normalized it means that the user may not interpret the magnitude from the vector, but the user may visualize the flow direction correctly. The plot contains contours that better aid in visualizing the flow speed magnitude by establishing lines that define equal values along the entire plot. The color palette is the default established by Matlab.

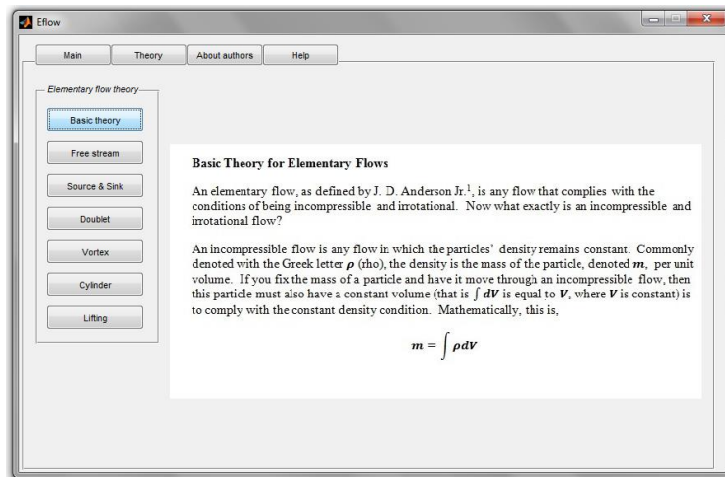


Figure 2 Interface “Theory” tab

As may be seen from figure 2, the Theory tab opens the basic theory by default, although the user may select whichever segment he/she wishes to see. The user only needs to click on the desired segment. The available theory options are “Basic Theory” (defines an elementary flow), “Free Stream,” “Source & Sink,” “Doublet,” “Vortex,” “Cylinder” (defines non-lifting flow over a cylinder), and “Lifting” (defines lifting flow over a cylinder).

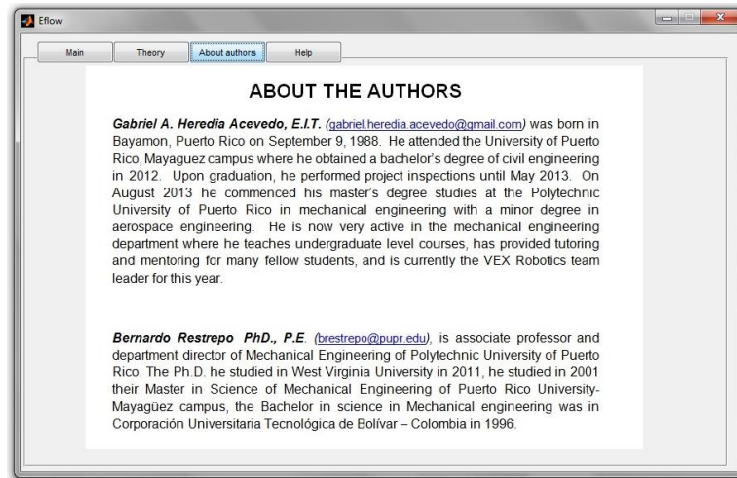


Figure 3 Interface “About the Authors” tab

The “About the Authors” tab shows brief biographical information about the creators of this GUI. It also contains contact information for each of the authors and a brief description about the Academic Institution where this GUI was created.

The “Help” tab includes all the information concerning the GUI version number, tutorials on using the program, and troubleshooting guide.

Link to the Classroom

This tool is intended to be used in and out of the classroom as the professor/instructor teaches the concepts concerning elementary flows. Examples may be shown together with the class concepts so that the student may have a visual connection to the topic and further improve comprehension. The professor may use this tool in the classroom to show examples and the student may use this in and out of the classroom to review the class material.

The GUI would be classified as “Computer Based Multimedia” under the article in dynamicflight.com². This means that the student should be fully engaged given that he/she may replicate examples provided by the professor/instructor, improving the understanding of the topic. The student may also access the tool on any computer that contains Matlab 2013 and create arbitrary flow patterns. This aids by alluding to the student’s creativity and permits the student to predict the flow pattern based on the program’s inputs.

The creation of arbitrary flow patterns should aid the student in comprehending the differential equations associated with each elementary flow and will eventually help the student to be able to write and understand the resulting equations due to the inputs. The class notes are critical here because this will provide the link between the class and the tool when the professor is not available to answer the student's questions.

Upon successful use of the tool, the student should be able to recognize complex flow patterns and be able to break them down as a combination of the elementary flows associated to this flow. An exact break down of these flows is not expected by the students, but recognition of the present elementary flows is expected. Additionally, the student will be able to associate the elementary flow pattern to the equations that describe them. Once the stream functions are obtained, the student should also be able to describe the volume flow rate in an area of interest, as well as the pressure differential distribution.

Examples

A series of examples extracted from the program are presented to demonstrate the validity of the flow patterns obtained. We would like to mention that moving plots may be selected to see all the flow patterns, although these will not be appreciated in this text. The first example shows the presence of a uniform flow with speed of 5 units along each of the two axes, x and y . This should yield a uniform speed of 7.071 units. The user will input the x and y flow intensities independently without the need of an origin because of the uniform flow characteristics. The user will obtain a normalized vector plot that demonstrates the correct flow pattern. The plot is now shown as follows.

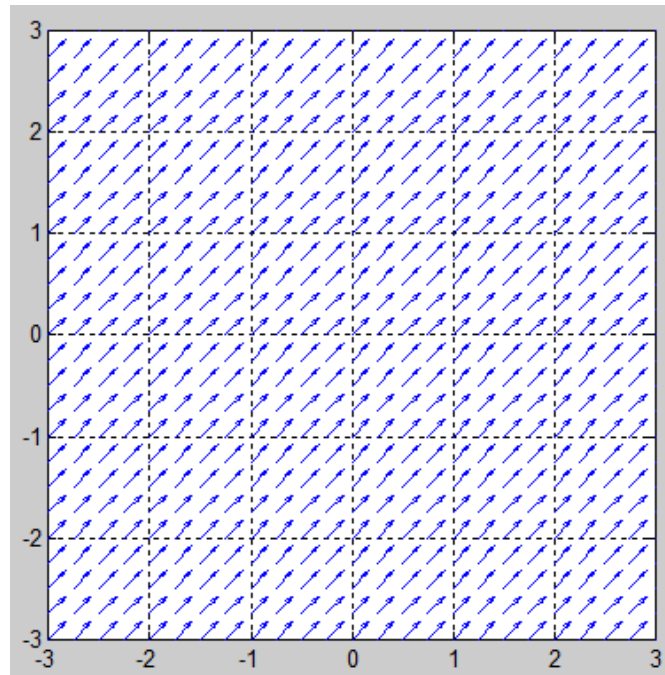


Figure 4 Vector plot of uniform flow speed of 5 units in x and 5 units in y .

As expected, the resulting plot is a uniform flow equal to 7.071 units at an angle of $\pi/4$. Notice that contour lines for this flow are not shown here. This is because the uniform flow does not vary in speed and, thus, will not contain contours. If we desire to see the theory used to be able to obtain this flow pattern, we only need to switch to the theory tab and look under the uniform flow information. The theory section explains the assumptions taken to obtain this flow pattern, as well as solve for the Laplace equation for the velocity potential and the streamline equation.

Our second example shows the second elementary flow, the source. The expected characteristic for the source flow is to obtain a flow pattern that spans outward radially from an arbitrary origin and whose speed varies inversely with an increase in radius. For our example we will use a source flow with a strength magnitude of 10 units located at x equals to 1 unit and y equals to 0 units. We will select the source flow, input the strength magnitude and coordinates into the correct boxes. Once again we will obtain a normalized vector plot, but we can extract the correct speed values into the workspace. The user will notice a contour plot overlaid into the normalized vector plot. This contour plot shows the variation of the actual flow speed. The contour's colors should show that black represents the largest values, then goes on to red, and all the way finally to a dark blue, which represents the lowest speed values. We can expect that the contour plot will look as circles around the source flow's origin. The plot is now shown as follows.

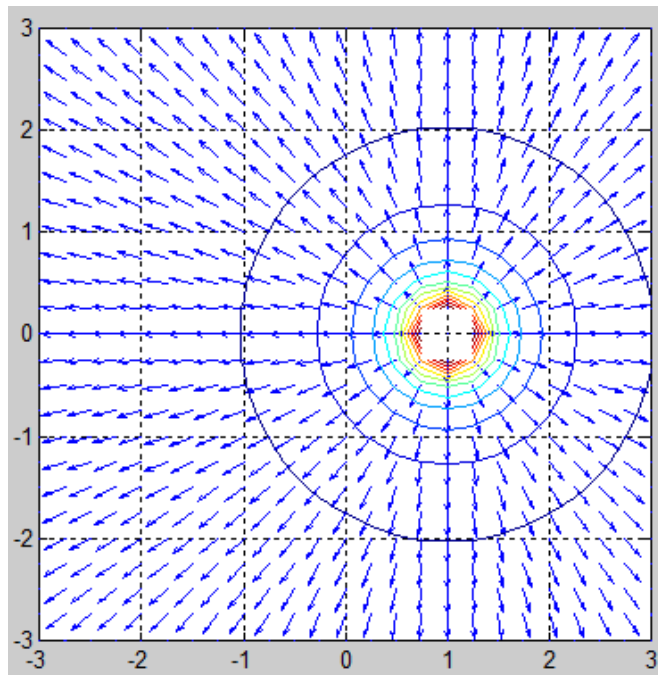


Figure 5 Vector plot with contours showing source flow with strength of 10 units at location of x equals to 1 and y equals to 0 units.

Notice that once again we have obtained a normalized vector plot. The plot shows a vector pattern extending outwards radially from the desired origin of 1 unit in x and 0 units in y , as expected. Notice that the contour lines are simply circles around the flow's origin; and the flow's intensity reduces inversely with an increase in radius. To see the source flow theory, we only need to select the source flow theory under the theory tab.

Our third example is the opposite of the source flow, the sink flow. The characteristics for this type of flow are exactly the same as for the source flow with one exception: the flow pattern tends towards the flow's origin. Our selected example is similar to the previous, the flow's origin is selected to be at x equals to -2 units and y equals to -1 units with a strength equal to 10 units. This time we select the sink flow from the window and place our desired values into the correct boxes. The example plot is now shown as follows.

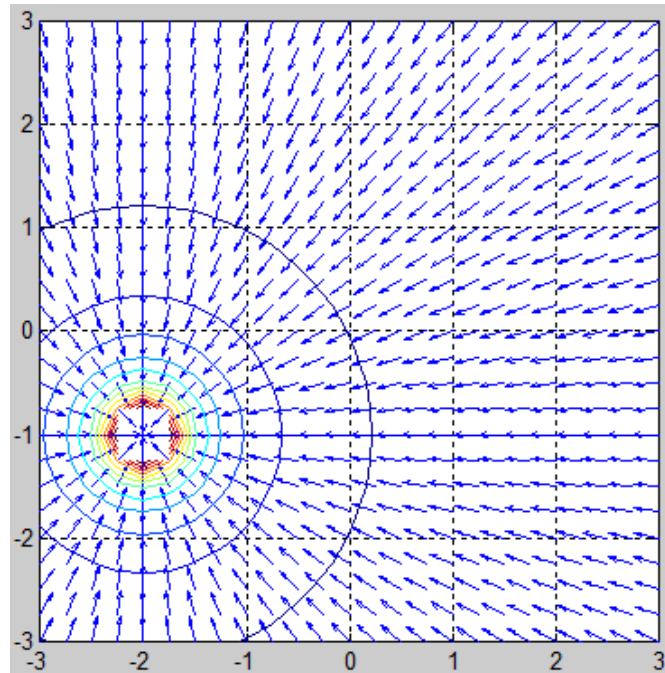


Figure 6 Vector plot with contours showing sink flow with strength of 10 units at location of x equals to -2 and y equals to -1 units.

Our normalized vector plot is very similar to that of the previous example. The difference in this plot is that the flow pattern is directed towards the flow's origin. The contour plot is just like the one obtained for the source flow. The actual values of the flow speed may be extracted into the workspace.

Our fourth example progresses into the next elementary flow, the doublet flow. Remember that the doublet flow is obtained by evaluation of the limit of the presence of a source and a sink flow as the distance between them tends to zero. The doublet flow pattern is very similar in appearance to that obtained when viewing the pattern obtained in a vertically placed magnet with its poles imposed over each other. Our selected values for this next example are a location of x equals to 0 units and y equals to 0 units and strength of 15 units. Now we select the doublet flow from the interface and input the correct values. The contour plot should show the same pattern for the source and the sink, given that this is merely a superposition of both flow patterns. The normalized vector plot is shown as follows.

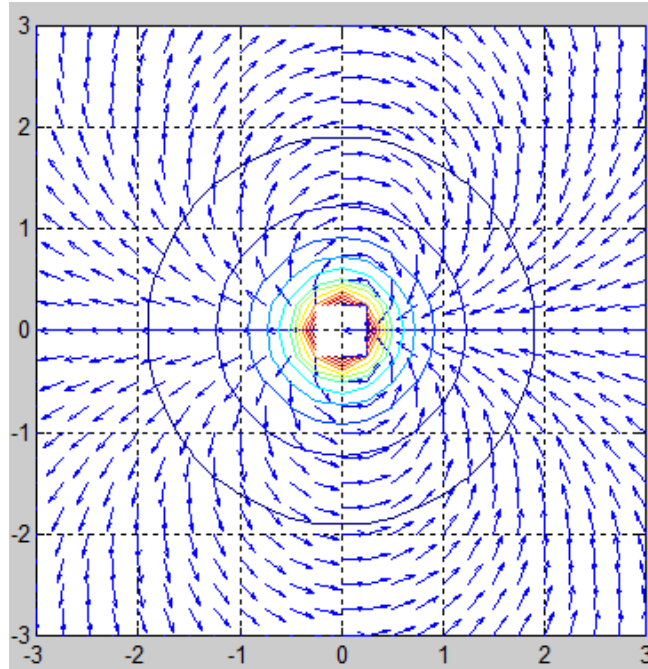


Figure 7 Vector plot with contours showing doublet flow with strength of 15 units located at the origin.

Our vector plot shows the normalized pattern for the doublet flow with our desired conditions. Notice that the contour plot shows circles expanding from the flow's center. Once again this is correct given that the doublet flow's nature is the superposition of the source and sink flow. As expected, the flow's is located at the origin and the flow speed values may be extracted into the workspace. The developed theory for the doublet flow is located under doublet flow in the theory tab.

Now see that the normalized vector plot shows the flow going around the center from the back all the way towards the front and finally coming back into the origin. Given this, the doublet may be used in combination with a source flow to show the flow pattern around a rod or cylinder. Using this sense our next example will show the combination of a uniform flow with a doublet flow, so as to simulate the case of a flow around a cylinder. The values selected for the uniform flow will be 10 units in x direction and 0 units in y direction. The strength of the doublet flow is selected at 15 units and its origin is located at x equals to 0 units and y equals to 0 units. There is no particular order in which the values are to be imposed in the interface. The user will input the uniform flow value in x direction after selecting from the interface and input the doublet values after selecting doublet. The normalized vector plot is shown as follows.

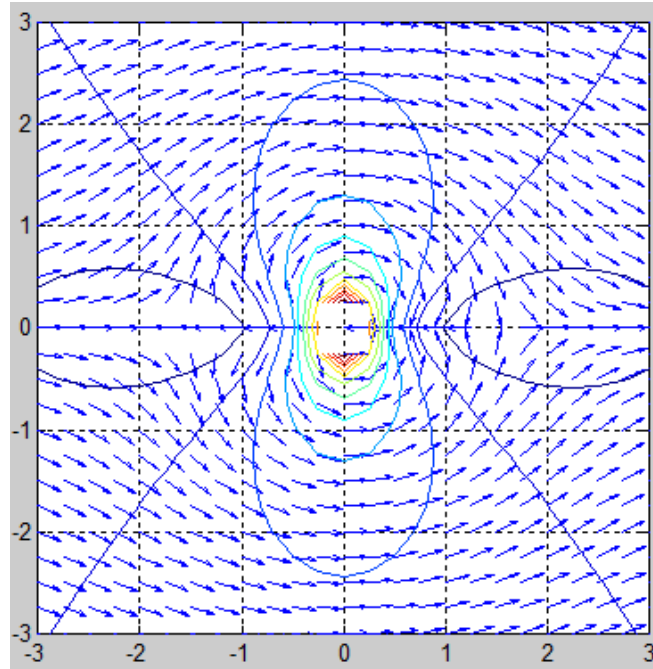


Figure 8 Vector plot with contours showing uniform flow with strength of 10 units in x and 0 units in y and a doublet of strength equal to 15 located at the origin.

Notice that mathematically, there is a flow value inside the cylinder, although, the physical sense is that there should be no speed inside this area. The speed inside the cylinder may be discarded and will not affect the flow conditions elsewhere. The radius of the cylinder may be computed using the equation provided in the theory section for this case. An additional detail to notice is that the contour lines shown in the plot are increasing in complexity as the flows are added.

Now we see the next flow pattern, the vortex. The vortex flow pattern shows the flow moving radially around the flow's origin and the speed varies inversely with an increase in radius. The developed theory will be located under vortex flow in the theory tab. The selected values for this example are strength of 10 units going clockwise around the flow's origin and a location of x equals to 0 units and y equals to 0 units. The normalized vector plot is shown as follows.

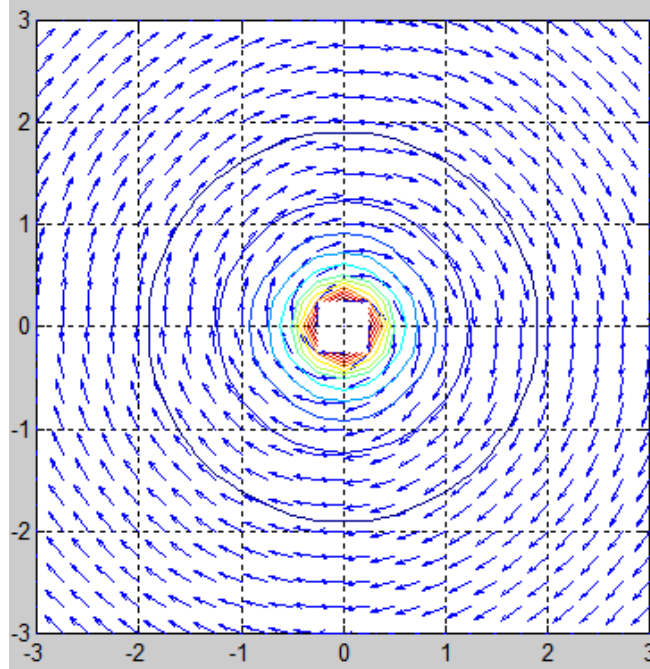


Figure 9 Vector plot with contours showing vortex flow with strength of 10 units located at the origin.

As expected the flow pattern shows constant speed for a given radius and moves tangent along the circle produced by this same radius. Notice that for this case the contour lines are just circles around the flow's origin and the flow intensity varies inversely with an increase in radius. Once again the actual flow speed values may be extracted into the workspace.

Now that we have seen the four elementary flows, the final example will show the flow pattern of a lifting flow over a cylinder. This may be achieved by adding a uniform flow and a doublet and vortex located at the same origin. For our example, we first select a uniform flow with strength of 5 units in x and 0 units in y directions. We then select a doublet of 7 units of strength located at x equals to 0 units and y equals to 0 units. Finally, we add a clockwise vortex with strength of 30 units, also located at the origin. The normalized vector plot is shown as follows.

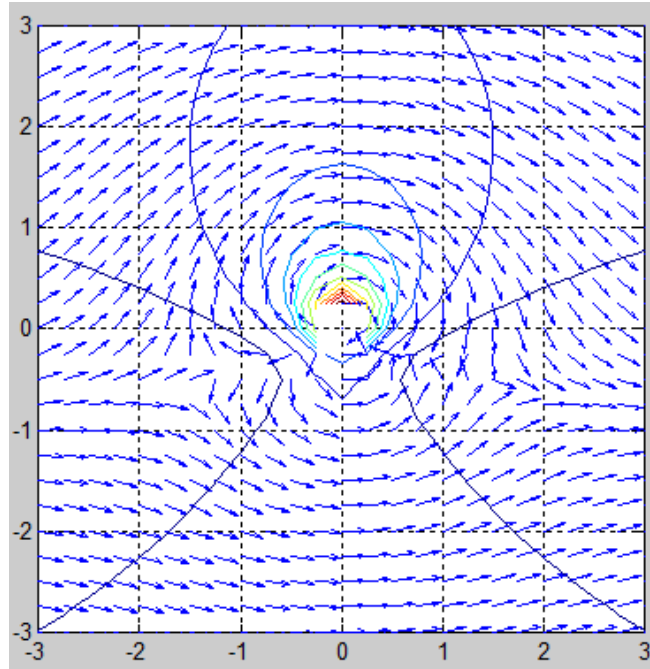


Figure 10 Vector plot with contours showing the combination of a uniform flow with speed of 5 units along x axis, doublet with 7 units of strength located at the origin, and a vortex flow with strength of 30 units located at the origin.

Notice that this plot has an implicit cylinder with a radius that may be computed using the theory developed under lifting flow over a cylinder in the theory tab. It may be demonstrated from the theory that this case shows two particular stagnation points located on the underside of the implicit cylinder. Extracting the speed values in the workspace also shows that the speed under the cylinder is lower than that above it, a necessary condition for there to be lift around the cylinder.

Future Work

Additional options to add to the GUI have already been identified for future development. The upcoming work will show the plotted streamlines in the normalized vector plot. Additionally, the user will be able to switch from a velocity vector plot to a pressure differential contour plot. Other developments shall include three dimensional vector plots for the elementary flows, as well as an extension to include airfoil/hydrofoil analysis. Surveys will be incorporated to assess national usage success rate, as well as feedback.

References

- 1 Anderson, John, Fundamentals of Aerodynamics, McGraw Hill Education, New York, 2010, 105-106, 169-172, 221-252.
- 2 Instructional Aids & Training Technologies, Dynamic Flight, Inc., 1999 – 2007
Retrieved from
http://www.dynamicflight.com/avcfibook/inst_aids/

Gabriel A. Heredia Acevedo, E.I.T.

Gabriel A. Heredia Acevedo (gabriel.heredia.acevedo@gmail.com) was born in Bayamon, Puerto Rico on September 9, 1988. He attended the University of Puerto Rico, Mayaguez campus where he obtained a bachelor's degree of civil engineering in 2012. Upon graduation, he performed project inspections until May 2013. On August 2013 he commenced his master's degree studies at the Polytechnic University of Puerto Rico in mechanical engineering with a minor degree in aerospace engineering. He is now very active in the mechanical engineering department where he teaches undergraduate level courses, has provided tutoring and mentoring for many fellow students, and is currently the VEX Robotics team leader for this year.

Dr. Bernardo Restrepo Torres, PhD., P.E.

Dr. Bernardo Restrepo (brestrepo@upr.edu) is an associate professor and Department Head of the Mechanical Engineering Department at the Polytechnic University of Puerto Rico. He obtained his PhD. at West Virginia University, his M.S. at the University of Puerto Rico, Mayaguez campus and his B.S. at the Corporacion Universitaria Tecnologica de Bolivar. Main research interest is focused in system integration of energy sources in order to make hybrid power systems. Secondary interests lie in the area of controls applied to aerospace systems. He has been teaching class for undergraduate and graduate students and always working in improving teaching techniques in engineering education.

Jonathan Holguino, M.E.M.E.

Jonathan Holguino (holbor@hotmail.com) studied his master's degree in Mechanical Engineering at the Polytechnic University of Puerto Rico with a minor degree in aerospace engineering in 2014. He performed his undergraduate studies in Mechanical Engineering at the San Antonio Abad del Cusco University, Peru in 2009. He was also a teaching assistant in undergraduate courses such as Mechanism Design, Mechanical of Materials, Fluid Mechanics, and other topics. Research interests include Aerodynamics, Control Systems Engineering, Vibrations Engineering, Mechanical Design, and Finite Element Analysis.