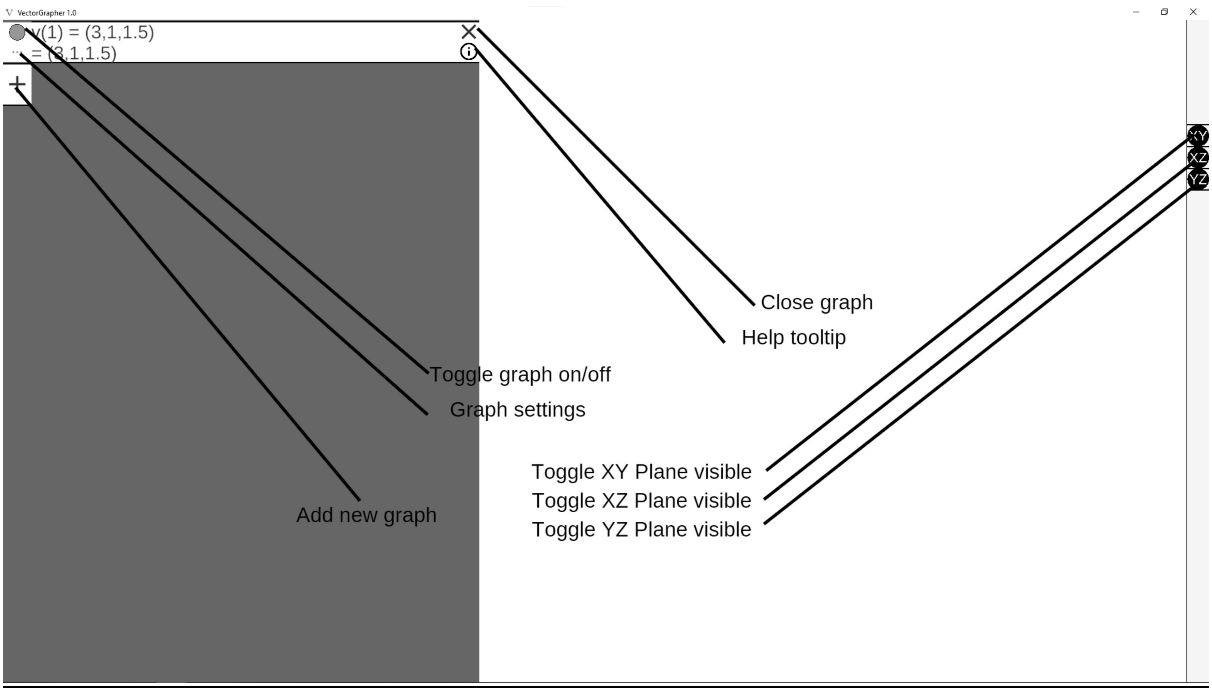
User manual

Navigating the user interface

Below is an annotated diagram of the user interface. It describes the function of each element on the user interface.



Navigating the graph

Zoom in
- To zoom in, the user can use the scroll wheel and scroll up to zoom in.

Zoom out
- To zoom out, the user can use the scroll wheel and scroll down to zoom out.

Pan view
- To pan the user's view, they can left click on the part of the screen where the graph is displayed and drag in a certain direction. The view will pan in the opposite direction.

Rotate view
- To pan the user's view, they can right-click on the part of the screen where the graph is displayed and drag in a certain direction. The view will rotate in the opposite direction.

Hiding planes from view
- Each of the XY, XZ, YZ planes can be individually hidden. This can be done by clicking the buttons labelled "XY", "XZ", and "YZ" respectively on the right of the screen. Clicking them again will re-display them.

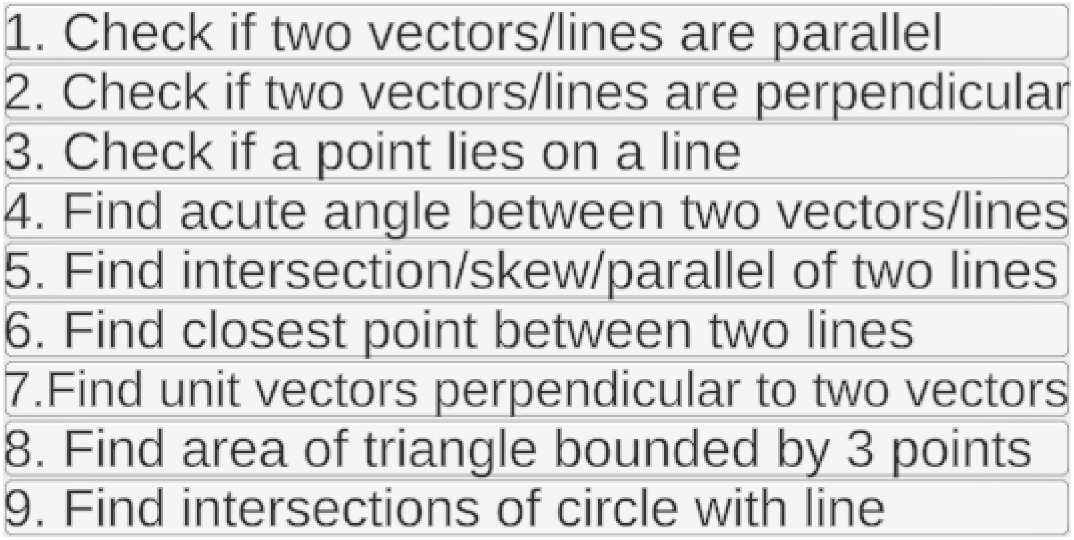
Moving the left-hand side panel
- To move the left-hand side panel left and right, click and drag the black bar at its right side.
- To scroll up and down the list of fields, either left-click drag with the mouse or scroll.

Graphing equations

To graph an equation, first a field must be added. To add a new field (as shown below), left click the "+" button.

To enter an equation, enter it into fields of the above type. Each type of equation such as vector, plane, sphere has specific syntax that must be followed, otherwise the program will not be able to detect it and thus it will not be displayed.

Alternatively, right clicking the "+" button will bring up the following menu:



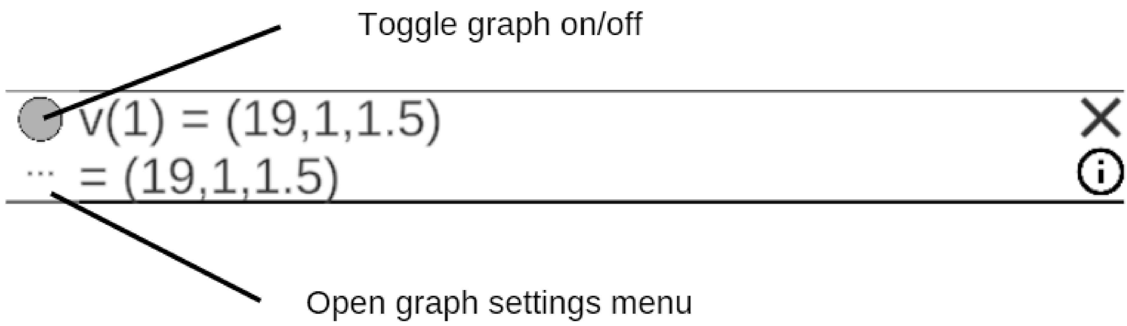
To select one of these functions, click on it. This will display a set of fields that will be labelled with the data that should be inputted, with a placeholder in the fields that provide an example of the input that should be given. To close this menu, click anywhere on the screen that is not on this menu.

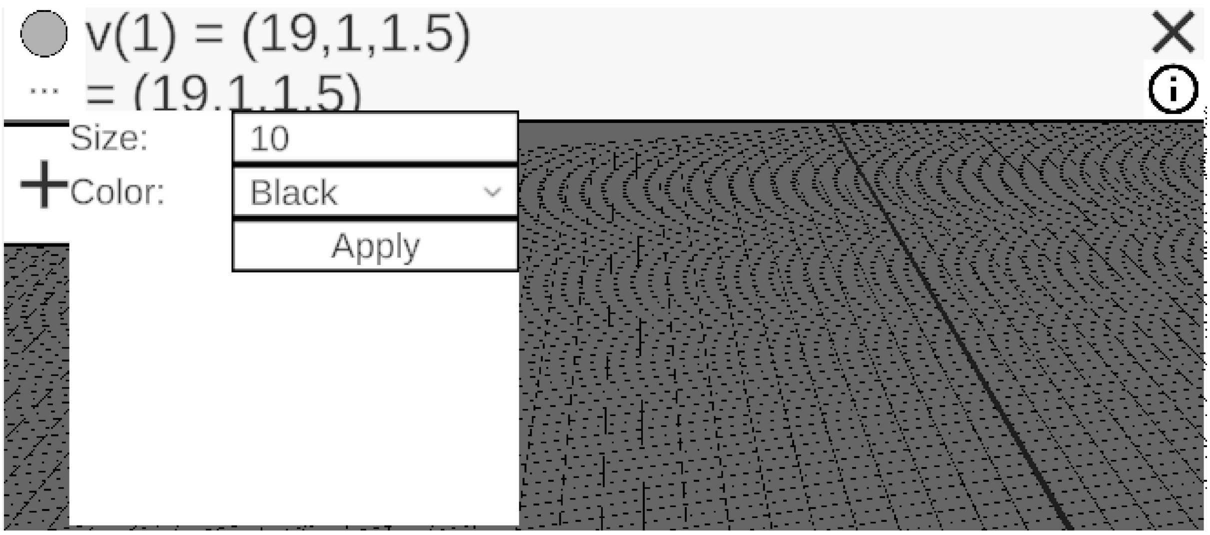
To delete a field, click the "x" button located at the top right of the field. This will close the field and delete the equation entered in the field.

Graph settings

There are settings for each individual graph. There are no program settings that affect the entire program. To open the graph settings, click on the "..." button at the bottom left of each input field. This will open a menu which will allow you to select the size and colour of the graph. The colour setting applies to all graphs, however the size setting only applies to vectors, lines, and planes. For vectors and lines, it changes their thickness. For planes, it changes their width and length. It does not have any effect on spheres.

To hide/show a graph, click on the circle icon at the top left of the input field. A coloured circle indicates an equation being displayed, with the circle having the same colour as the graph. A non-coloured circle indicates that the field is hidden.





Equations

Guidelines

- Each graph of the same type cannot be entered in multiple fields. Vectors, planes, and lines cannot have two graphs defined by the same name, such as v(a) = (1,5,3) and v(a) = (-1,9,9) in two fields. This will result in only one of these being graphed.

Vectors

- The general format for a vector is "V(name) = (vector), {vector}.
- The vector expression is split into 3 main parts - the name, the direction vector, and the tail position. The name and direction vector are required, but the tail position is optional - it can be omitted.
- Name
    - Each vector must be defined with a vector name, which is either "v(name)" or "V(name)", where 'name' contains only alphanumeric characters. For example, "V(a)", "v(vect1)", "v(1)", "v(myVector)" are all valid vector names
- Direction vector
    - The direction vector can consist of either a single vector, vector arithmetic, cross product, projection, dot product, size of a vector or scalar multiple of a vector.
    - A single vector is defined as "(a,b,c)", where a, b, and c are real numbers.
    - Vector arithmetic is defined as an expression such as "(a, b, c) + (d, e, f) - (g, h, i)..." where vectors are added or subtracted together.
    - A cross product is defined as "Cross[(a,b,c),(d,e,f)]"

    - A projection is defined as "Proj[(a,b,c)on(d,e,f)]"
    - A dot product is defined as "Dot[(a,b,c),(d,e,f)]".
    - The size function returns the size of a vector. For example, "Size(a,b,c)".
    - The dot product and size functions return a number, so they cannot be put in a vector expression by themselves, but they can be put as an x, y, z value or scalar multiple of another vector. For example, the expressions "V(a) = (Dot[(1,2,3),(4,5,6)], 0, 0)" and "V(b) = (0, Size(9,9,9), 0)" are valid expressions but "V(a) = Size(3,2,1)" and "V(a) = Dot[(1,2,3),(4,5,6)]" are not.
    - Any of the above direction vectors can be combined together to form a more complex direction vector.
    - Also, the user is able to define vectors and constants elsewhere, and place them within vector expressions.
        - An example of this with user-defined vectors is, if "V(a) = (3,2,1)" is defined by the user, then they can enter another expression in another field - "V(b) = V(a) + (1,1,1)", which will take the value of V(a), which is (3,2,1), making V(b) equivalent to (4,3,2).
        - An example of this with user-defined constants is, if "a = 8.32" is defined by the user, then they can enter another expression in another field - "V(a) = (a,0,3)", which will be equivalent to "V(a) = (8.32,0,3)"
        - If a user-defined constant or vector is changed, then its value in other expressions will change, meaning those other expressions will change. This will be useful for users who wish to graph multiple expressions with different parameters without having to change each individual one.
- Tail vector
    - Any expression that is suitable for a direction vector can be placed in the tail vector, however the entire expression must be enclosed within curly brackets "{}".

Vector plane

- Vector planes are defined as "P(n) = (a,b,c) + λ(d,e,f) + μ(g,h,i)"
- "λ" can be entered by writing "lambda", "LAMBDA", "Lambda" etc. It will be automatically converted to its symbol. It is not case sensitive.
- "μ" can be entered by writing "mu", "MU", "Mu", etc. It will automatically be converted to its symbol. It is not case sensitive.
- For a vector plane, (a,b,c) represents the position vector, and λ(d,e,f) and μ(g,h,i) represent two direction vectors which form the direction of the plane. They must not be parallel and they must not be the zero vector (0,0,0).
- Any of the vectors can be replaced with the names of vectors already defined, although they must be enclosed in brackets. For example, if v(a) = (3,1,5), v(b) = (1,1,1) and v(c) = (9,9,9) are defined, then the equation p(1) = v(a) + λv(b) + μv(c) is valid and will produce the graph of p(1) = (3,1,5) + λ(1,1,1) + μ(9,9,9).

Vector line

- Vector lines are defined as "L(n) = (a,b,c) + λ(d,e,f)".
- "λ" can be entered by writing "lambda", "LAMBDA", "Lambda" etc. It will be automatically converted to its symbol. It is not case sensitive.

- For a vector line, (a,b,c) represents the position vector, and (d,e,f) represents the direction vector which determines which direction the line faces.
- The parameter λ must be used for the direction vector of the line.
- Any of the vectors can be replaced with the names of vectors already defined, although they must be enclosed in brackets. For example, if v(a) = (3,1,5) and v(b) = (1,1,1) are defined,

then the equation l(1) = v(a) + λv(b) is valid and will produce the graph of l(1) = (3,1,5) + λ(1,1,1).

Vector sphere

- A vector sphere is defined as "|v - (a,b,c)| = r^2, or "|v(x) - (a,b,c)| = r^2"
- Here, v does not represent any particular vector, and neither does v(x). (a,b,c) are the coordinates of the centre of the sphere, and r is the radius.

Cartesian sphere

- Cartesian spheres are defined as "$(x-a)^2 + (y-b)^2 + (z-c)^2 = r^2$", where a, b, c, r are real numbers, and r is not 0.

Example equations

For the following examples, if an equation uses another equation within it, it assumes the other equation is defined. For example, v(1) = v(2) + v(3) assumes v(2) and v(3) are already both defined. If they are not,

1. v(a) = (3,1,5)
2. v(1) = (-1,0.3,5)
3. v(2) = 0.3(9,5,1)
4. v(3) = 0.5(0.4 + 9,5-0.3,-1)
5. v(4) = (1+3,5,1) + 0.5(1,3,3) – (3,1,1)
6. v(5) = Cross[(1,5,3),(9,9,9)]
7. v(6) = Dot[(13,33,3),(-5,-5,-5)]. Note – this will not display a vector, but will return a result.
8. v(7) = Proj[(5,3,3)on(1,5,3)].
9. v(8) = v(6) – v(5).
10. v(9) = Proj[v(6)on(Cross[(v(6)), (Proj[(5,3,3)on(1,5,3)])])].
11. l(1) = (3,1,5) + λ(3,3,3)
12. l(2) = v(1) + λv(2).
13. p(1) = (3,1,5) + λ(3,5,-3) + μ(3,3,3)
14. p(2) = v(1) + λv(2) + μv(3)
15. |v-(3,1,5)| = 3
16. |v(x) – (1,1,1)| = 9
17. (x-3)^2 + (y-3)^2 +z^2 = 35
18. (x-1)^2 + y^2 + (z-15)^2 = 3