

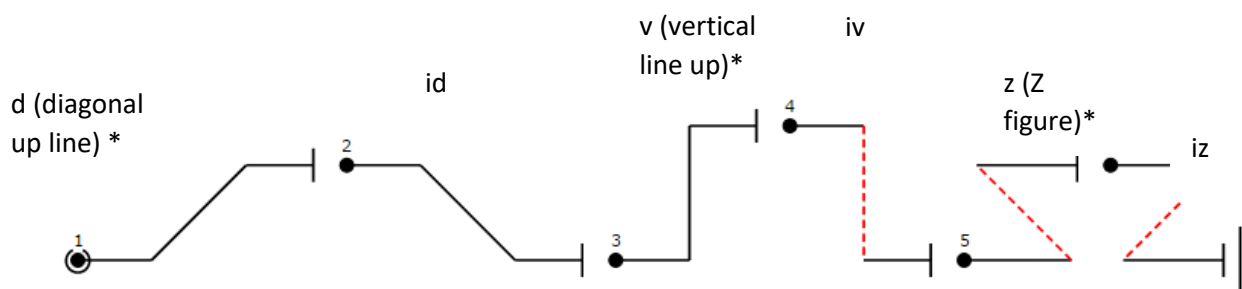
Evaluation of OLAN manoeuvres

This document highlights some of the movements, be that single-element, double-element, loops, double-loops or special and complex moves. Although not all the notated movements presented in the OLAN base figures document are shown here, the most important ones shall be. With each, an explanation of the figure will be displayed, alongside some explanation to the primitive sections of the manoeuvre. This document will serve as a basis of deciding which key line transformations will be required in the simulator, alongside any potential hazardous manoeuvres that may prove difficult.

Note: * Symbols indicate the move can be inverted (using the letter 'i'), ^ indicates the move can be reversed (using the letter 'r').

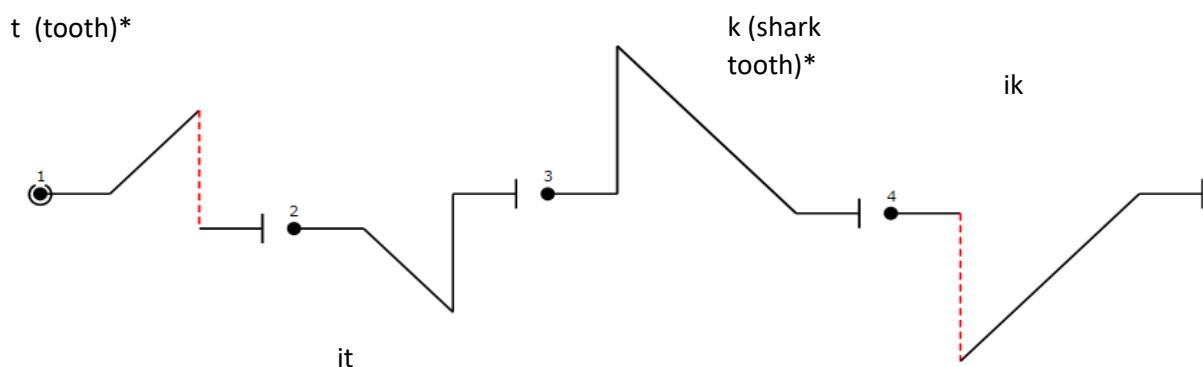
Single-element

Single elements are simply moves that are described as one movement. For example, the first move shown in the figure below is described as a diagonal line up, while move 6 can be simply described as inverted Z figure.



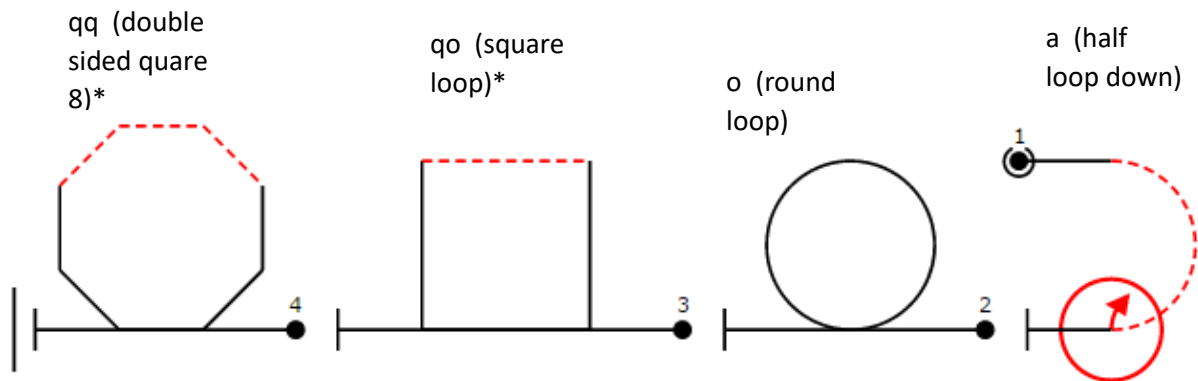
Two-element

A two-element move is a manoeuvre that requires two separate moves to form the entire move. For instance, in the first move, described as a 'tooth' requires firstly a diagonal line up, then an inverted line down. A simple assumption can state that any move in this two-element case can be formed using any of the single-element lines. In move 3, this move known as a 'shark tooth' is comprised of a vertical line up and diagonal line down, though is special in that it can be inverted (by placing an 'i' before the OLAN notation) to mirror the commands.



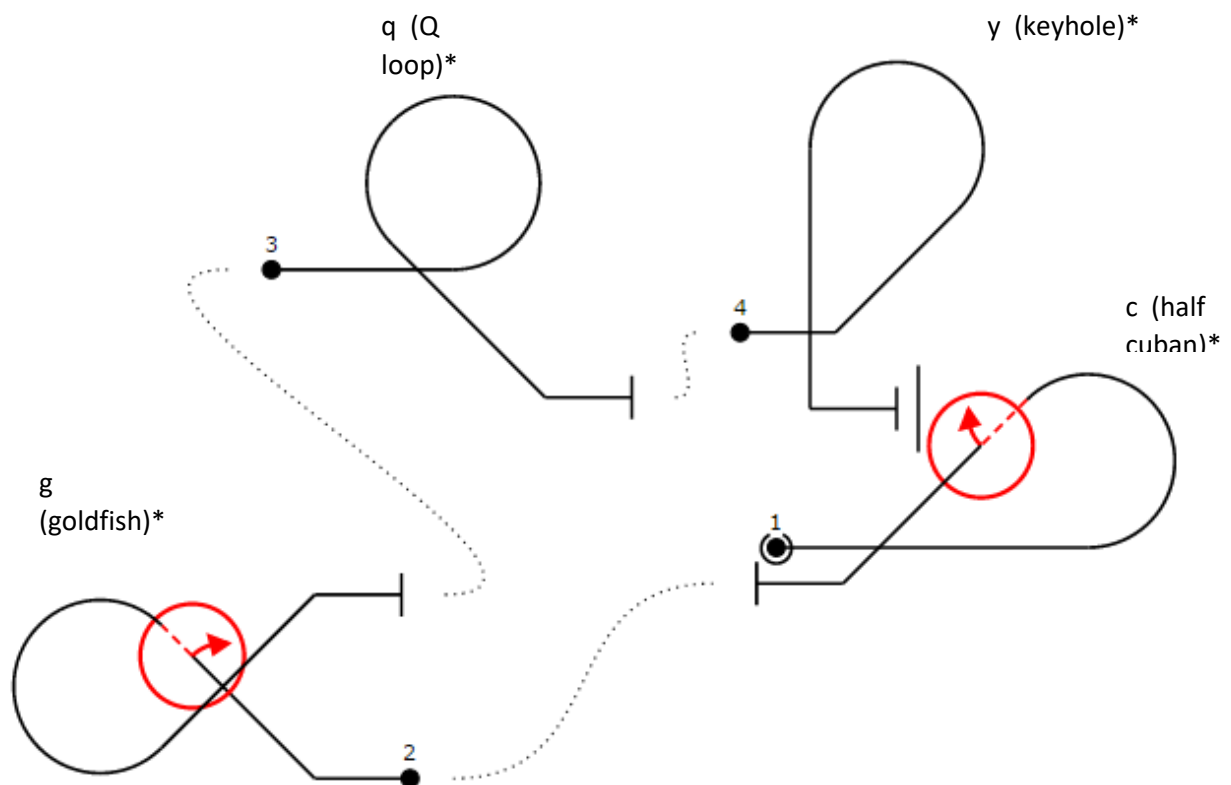
Loops

Loops can be deduced to be single element moves, where the flight consists of one simple move, for instance in move 1, these can be understood to be simply one half loop, containing rolls at the top and bottom of the arc. For each of these loops, they all work by multiples of 45 again, though with examples 4 and 3, rather than a smooth curve, a more square or sudden turn is expected to achieve the desired result. This is one thing that will need to be considered.



Loop-line combinations

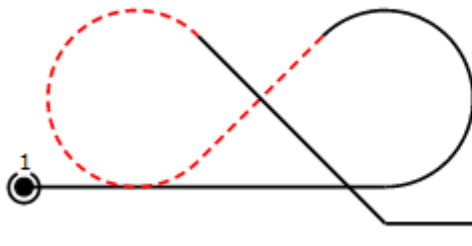
A loop line combination defines moves where both single and two-element lines are combined with loops as shown in the previous section. The first example, which shows a half-Cuban requires a line through to a 5/8 loop (again this is a multiple of 45 degrees) followed by a roll and diagonal line down. Another example shown in move 3, is comprised of a full loop followed by a diagonal line down.



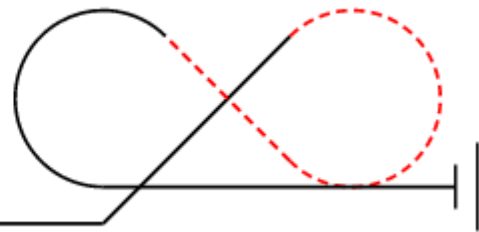
Double-Loops

Double loops are cases where two loops are used after one another to achieve the desired result. The two examples here reflect each other because of the letter 'r' meaning reverse in the second diagram. As you can see, this should be easily achieved by simply swapping over the roll from one loop to the other. Again with these, it is a matter of the angle of turn creating the loop that must be correct to be able to run a diagonal line down to create the next.

tcc (full
cuban)*^



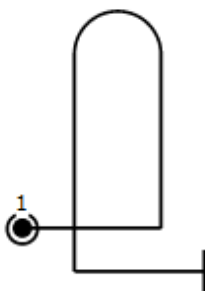
rcc



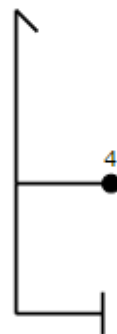
Humpty-Dumpties, Hammerheads and Tailslides

A humpty dumpty (1) consists of a bump, followed by a 180 degree turn to come back down vertically. Meanwhile the hammerhead manoeuvre described in move 4 consists of flying straight up, and then rotating on the wing so to fall back down. In diagram 5, the tailslide move also looks a tricky one to implement, as it requires falling back on itself after slowing down. This change of direction in this and in the hammerhead moves could be an issue as simply giving the plane a constant speed in the WebGL program would not work.

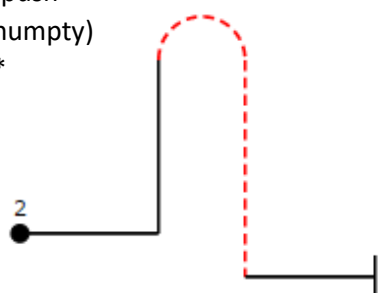
b (humpty
bump)*



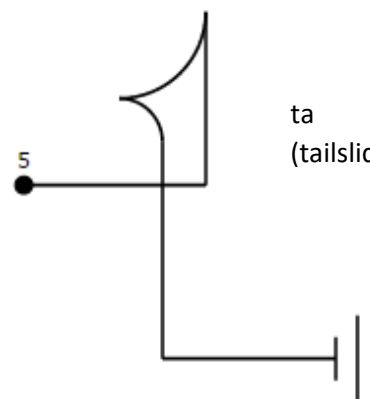
h
(hammerhead)



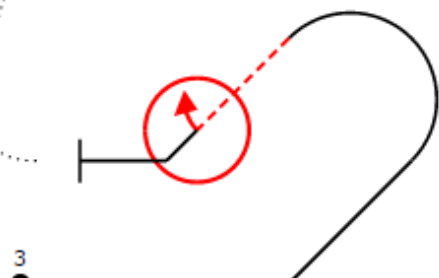
pb
(push
humpty)
*



ta
(tailslide)*

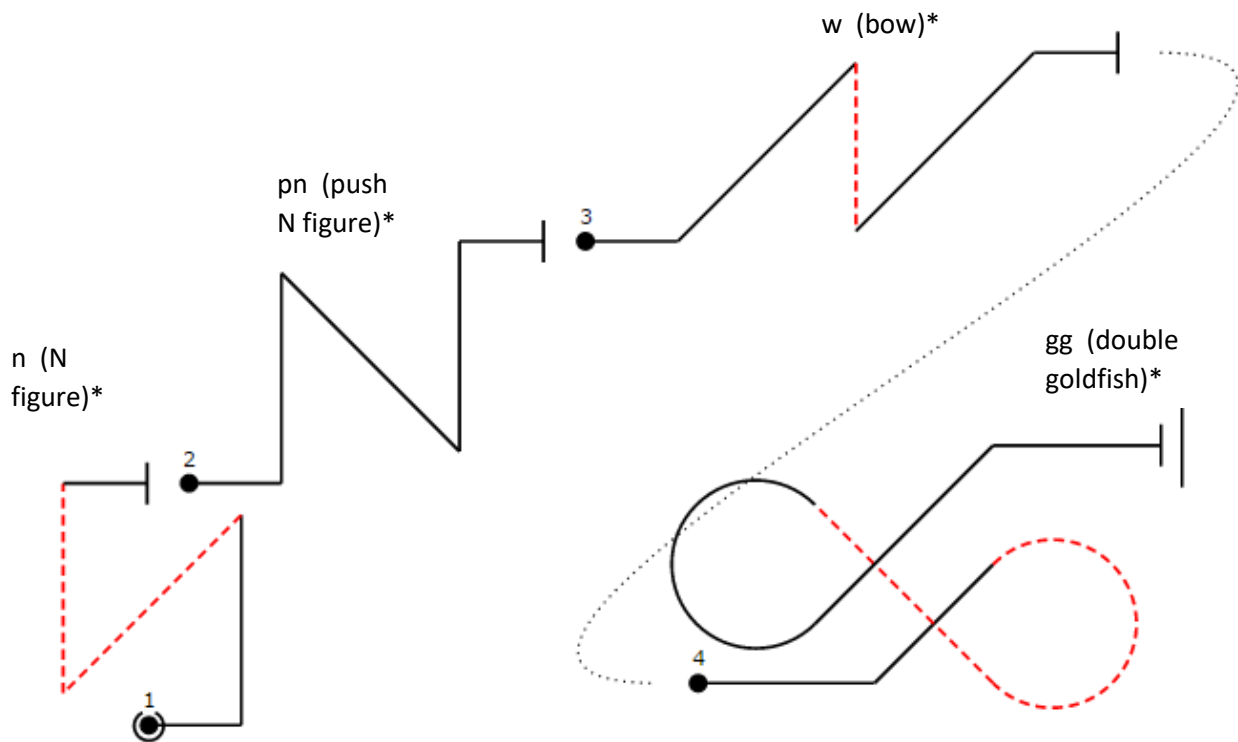


db (sasuage)*^



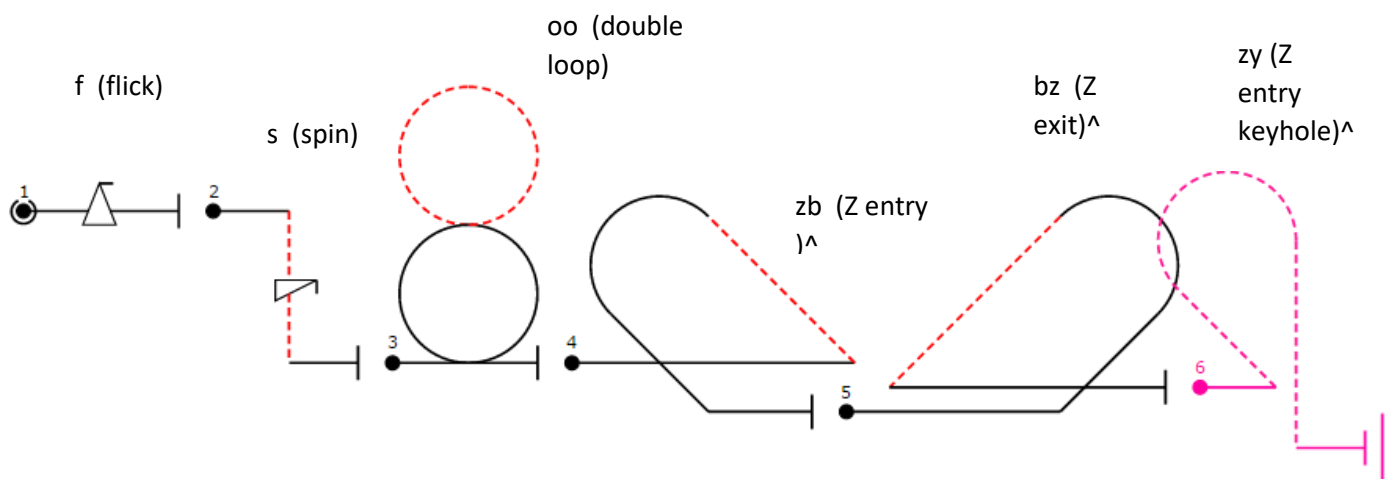
Complex 3-Rolling elements

This set of maneuverers contain a total of three different elements, so in diagram one an N figure, which itself is comprised of two moves, followed by a pull back to the diagonal. These moves should not be too difficult to recreate, as each of which should be easily modified previous moves. An example of this is the double goldfish move shown in diagram 4, which is simple a double loop but with more of a tighter loop on either side (this could be defined by size of radius).

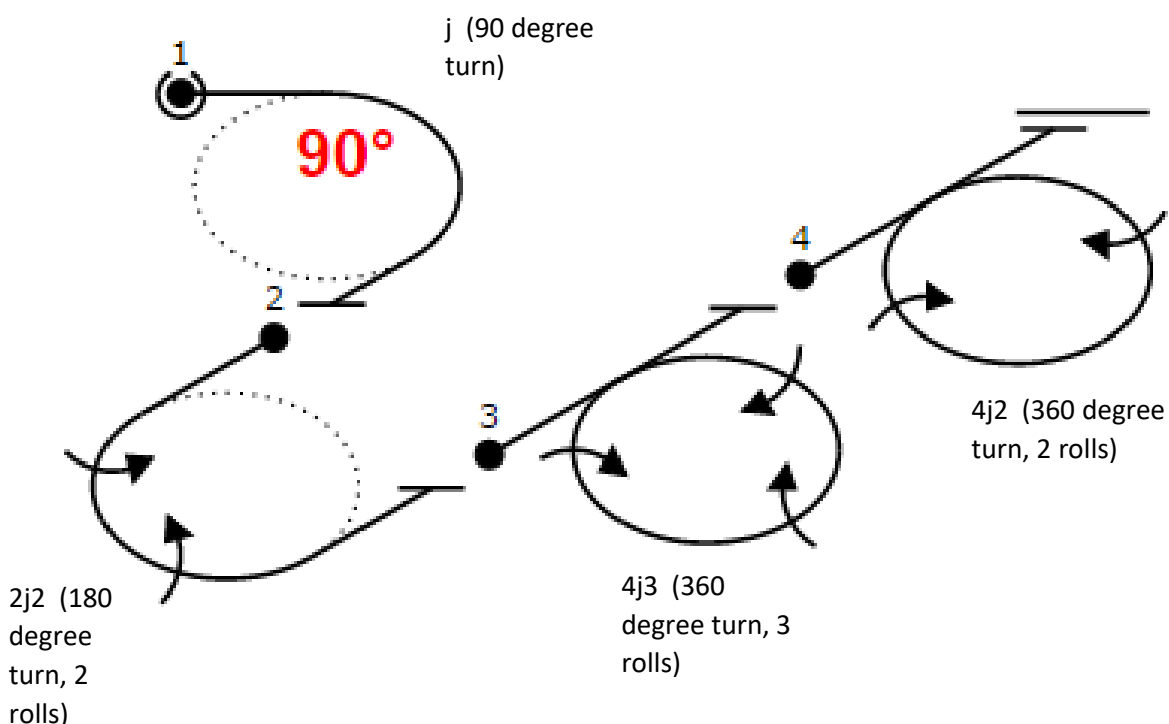


Oddball and special figures

The moves shown in this sections are ones that are more complex is the way that they require special manoeuvres. Starting with diagrams one and two, they both have a triangle shape places on the line, indicating a flick roll. This means rolling the plane a total 360 on the horizontal line to return it to its original rotation. Other moves shown below could be recreated again with a use of different single and two element moves. Moves such as diagram 3, would be comprised of two full loops, one inverted, and these loops can be broken down again into 45 degree segments. As the plane leaves the first loop, it should be easy to enter the second already inverted.



The final set of Aresti moves illustrated here look at turns and rolling turns. The angle in the first move shows a simple 90 degree turn, whilst in the second, because '2' has been places both sides of the 'j', this signifies 2 90 degree turns, and 2 rolls. This parameter set can be shown in the other two examples, with the last showing 4 90 degree turns (creating a loop), containing 2 rolls.



Overall assumptions and comments

- Some moves can be inverted, so flipped/ mirrors according to their commands e.g. line up, diagonal down -> line down (inverted), diagonal up.
- All moves can be crated with a set of 45 degree turns and rolls. By doing these straight after one another should mean being able to achieve a whole loop if done 8 times.
- In order to do different kinds of loops (round, square or double sided), it will mean that there will have to be some means of parameters to each turn, defining sharpness. This could be done by defining:
 - Radius of loop
 - Height of loop
- Moves that can be inverted or reversed should be easily implemented, as these should be able to use the same methods, just in reversed order, or giving reversed parameters (10 = -10).
- There should be a function that allows parameters to be passed to signify number of rolls in a turn or loop too, like in the '2j2' examples.
- Moves such as the tailslide and hammerhead will be difficult to implement, mainly because of the need to slow down the plane to a standing position, and then allow it to fall back at a certain angle. This though might be solvable by manipulating methods that allow it travel in a 45 degree based curve, by allowing the defining of the planes orientation.
- There should be a total of three different primary moves, each of which will have their own required smaller movements. These are:
 - Diagonal and straight lines
 - These should allow different angles, in multiples of 45
 - Length needs to be defined
 - Allow for rolls of 180 to invert
 - Curved lines
 - These should have radius and height inputs
 - Can add 45 degree curves to each other to make larger curves, loops and figure eights etc.
 - Turns and rolls
 - These should have parameters similar to OLAN; the amount of rolls, and 45 degree increments to create quarter, half and full loops
- These primary elements can be seen on the next page contained within a move hierarchy.

Move hierarchy

