Longitudinal Aircraft Dynamics #10– the numerical method

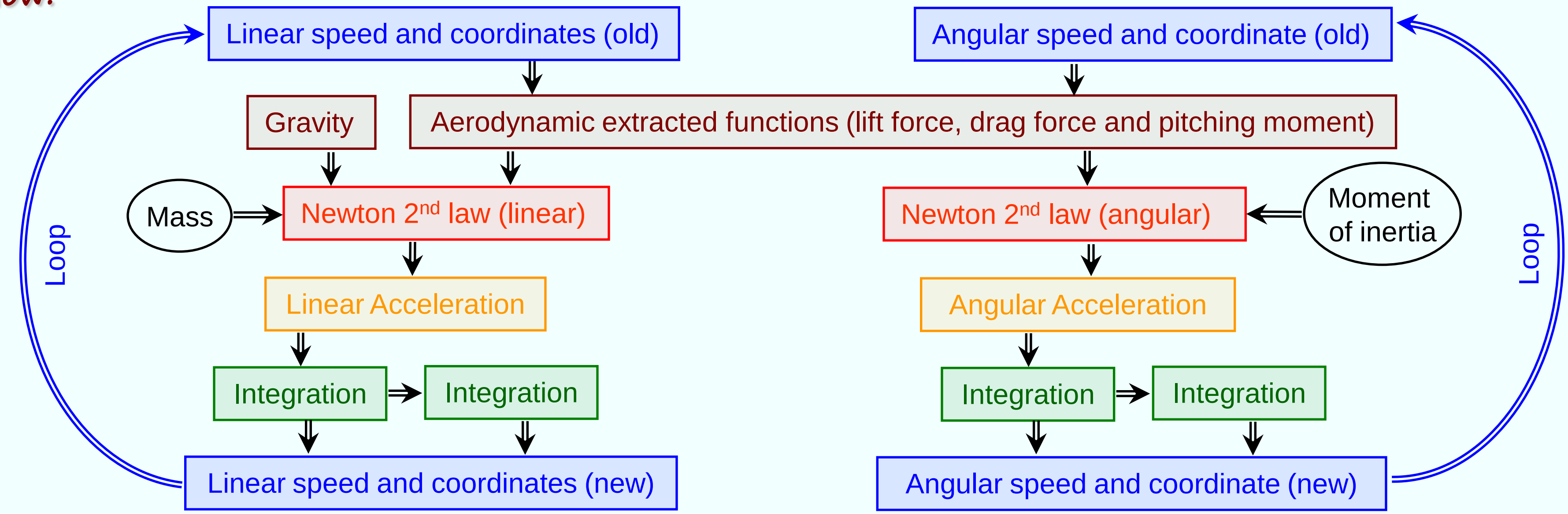
by George Lungu

- This section deals worksheet implementation of the numerical setup for a dynamic modeling of the flight. The model will already be functional by the end of this section.

An review of the method:

-We have both the x component and the y component of the resultant "present" force on the airplane calculated from the "past" speeds and positions. We use the "present" force to calculate the "present" x-y acceleration components (Newton's second law). From acceleration we can calculate the "present" speed components by integration and from speed we calculate the "present" x-y coordinates.

- We avoided calculating something from itself (circular referencing). A diagram of the method is shown below:



Numerical scheme implementation:

- Copy the last worksheet and name the new one "Longitudinal_ Stability_Model_6"
- There are three buttons, the "Reset", the "Run_Pause" and the "CG_visibility" buttons. Make sure to reassign the right macros to them, macros belonging to the current worksheet. By default, the custom buttons (created by the user from a macro assigned shape) keep the macros from the original worksheet assigned to them.
- We moved all the force and momenta calculation range near the input parameter buttons in order to perform some testing. Let's move them back to the original area: select range D19:E36 => Copy => select cell M66 => Paste.
- Copy the x and y force component to the present calculation area: B51: "=R77", E51: "=R78".
- Copy the pitching moment to the present calculation area: H51: "=R79".

Speed calculation by the numerical integration of the linear acceleration:

- We will use the simplest approximation, a linear variation of the speed with time, which means we assume that acceleration (hence force) is almost constant during the duration of a simulation time step (the smaller the time step the better the approximation):

$$a = \frac{dv}{dt}$$

$$a_{present} = \frac{v_{present} - v_{past}}{dt} \Rightarrow \quad v_{present} = v_{past} + a_{present} \cdot dt$$

using Newton's second law we write:

$$v_{present} = v_{past} + \frac{F_{present}}{m} \cdot dt$$

We will model the problem separately along the axes of a Cartesian system of coordinates:

$$\begin{cases} v_{present\_x} = v_{past\_x} + \dfrac{F_{present\_x}}{m} \cdot dt \\ v_{present\_y} = v_{past\_y} + \dfrac{F_{present\_y}}{m} \cdot dt \end{cases}$$

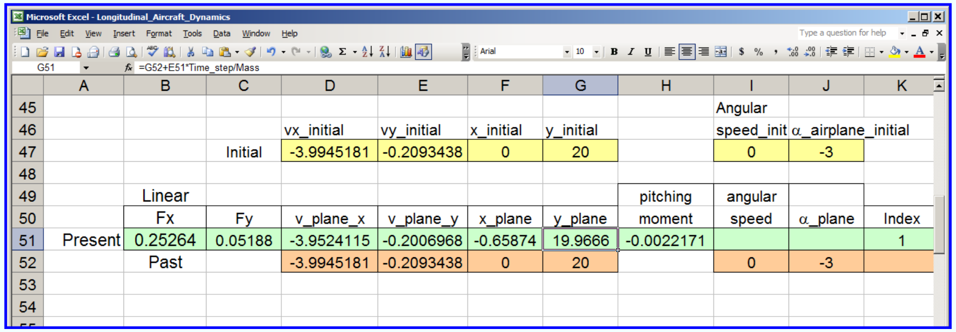
Coordinate calculation by the numerical integration of speed:

- Similarly we will use the simplest approximation, a linear variation of the coordinates with respect to time and just like in the case of the speed approximation we can write the numerical approximation of the linear coordinates:

$$\begin{cases} x_{present} = x_{past} + v_{present\_x} \cdot dt \\ y_{present} = y_{past} + v_{present\_y} \cdot dt \end{cases}$$

Worksheet implementation:

- We can write the following formulas:
  - => D51: "=D52+B51*Time_step/Mass"
  - => E51: "=E52+C51*Time_step/Mass"
  - => F51: "=F52+D51*Time_step"
  - => G51: "=G52+E51*Time_step"



Microsoft Excel - Longitudinal_Aircraft_Dynamics

G51 =G52+E51*Time_step/Mass

	A	B	C	D	E	F	G	H	I	J	K
45									Angular		
46				vx_initial	vy_initial	x_initial	y_initial		speed_init	α_airplane_initial	
47			Initial	-3.9945181	-0.2093438	0	20		0	-3	
48											
49		Linear						pitching	angular		
50		Fx	Fy	v_plane_x	v_plane_y	x_plane	y_plane	moment	speed	α_plane	Index
51	Present	0.25264	0.05188	-3.9524115	-0.2006968	-0.65874	19.9666	-0.0022171			1
52		Past		-3.9945181	-0.2093438	0	20		0	-3	
53											
54											

- In the snapshot to the right, the active (present) formulas are on a green background and the "past" constant data is on an orange background.

Angular speed calculation by the numerical integration of the angular acceleration:

- Formally, the angular treatment is very similar to the linear dynamics treatment. The linear acceleration (a) becomes angular acceleration (α), the linear speed (v) becomes angular speed (ϖ) and linear coordinate (x) becomes angle or phase (ϑ)

- There is even an angular form of Newton's second law which says that the angular acceleration is equal to the product between the moment of inertia and the moment of force

$$\alpha = \frac{d\omega}{dt} = \frac{d^2\theta}{d^2t}$$

This says that the angular acceleration is equal to the first derivative of angular speed with respect to time and also equal to the second derivative of the angle with respect to time.

moment of inertia

moment

$$\vec{M} = I \cdot \vec{\alpha}$$

angular acceleration

Newton's second law in angular form

<www.excelunusual.com>

- The formula for the moment of inertia of a uniform rod of mass "m" and length "L" about its middle (center of mass - CM) is:

$$I_{CM} = \frac{1}{12} \cdot m \cdot L^2$$

- Using the same procedure used for the derivation of the linear speeds and coordinates you can derive the numerical approximation formulas for calculating the angular speed and the angular coordinate (airplane angle) using the moment, the moment of inertia and the previous angular speeds and coordinates. The proof is not given here, however you can see a side by side comparison of the linear and angular formulas:

linear speeds

force

$$\begin{cases} v_{present\_x} = v_{past\_x} + \dfrac{F_{present\_x}}{m} \cdot dt \\[2ex] x_{present} = x_{past} + v_{present\_x} \cdot dt \end{cases}$$

mass

linear coordinates

LINEAR

angular speeds

moment of force

$$\begin{cases} \varpi_{present} = \varpi_{past} + \dfrac{M_{present}}{I} \cdot dt \\[2ex] \theta_{present} = \theta_{past} + \varpi_{present} \cdot dt \end{cases}$$

moment of inertia

angles

ANGULAR

- It is interesting to notice that in our case we used "α_plane" to denote the angle of the airplane which in the above formula becomes theta (ϑ).
- Another very important observation is that we have chosen as an input parameter the fractional moment of inertia, which is the ratio of the moment of inertia of the plane divided with the moment of inertia of a uniform rod having the same length and same mass as our airplane. For a glider I guess a reasonable number would be around 0.3-0.5, since most of its mass is concentrated around the pilot and the main wing.
- With the last notice in mind our final numerical equations become (see next page):

$$
\begin{cases}
\varpi_{present} = \varpi_{past} + \dfrac{12 \cdot M_{present}}{I_{fractional} \cdot Mass \cdot Length^2_{fuselage}} \cdot dt \\
\alpha\_plane_{present} = \alpha\_plane_{past} + \varpi_{present} \cdot dt \cdot \dfrac{180}{\pi}
\end{cases}
$$

For ease of usage we expressed our angles all over the model in degrees. The second formula includes a radians to degrees conversion factor.

Conversion from radians to degrees

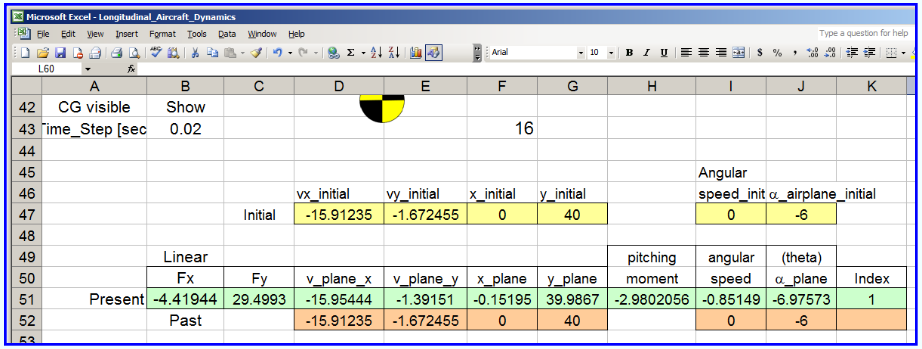
Worksheet implementation:

- We can write the following formulas:

=> I51: "=I52+(12*H51*Time_step)/(Momentum_inertia*Mass*Length_fuselage^2)"

=> J51: "=J52+180*I51*Time_step/PI()"

- In the snapshot to the right, the active (present) formulas are on a green background and the "past" constant data is on an orange background.



	A	B	C	D	E	F	G	H	I	J	K
42	CG visible	Show									
43	Time_Step [sec	0.02				16					
44											
45									Angular		
46				vx_initial	vy_initial	x_initial	y_initial		speed_init	α_airplane_initial	
47			Initial	-15.91235	-1.672455	0	40		0	-6	
48											
49		Linear						pitching	angular	(theta)	
50		Fx	Fy	v_plane_x	v_plane_y	x_plane	y_plane	moment	speed	α_plane	Index
51	Present	-4.41944	29.4993	-15.95444	-1.39151	-0.15195	39.9867	-2.9802056	-0.85149	-6.97573	1
52		Past		-15.91235	-1.672455	0	40		0	-6	

Increasing the size of the historical trajectory data:

- We provided a 1000 point long historical trajectory data (within the "Run_Pause()" and "Reset()" macros). We would like to increase that length let's say to 3000.
- In order to do that let's change the "Run_Pause()" and the "Reset()" macros as seen to the right.
- We can see that the macros are the same except for the size of the copied/pasted/cleared range.

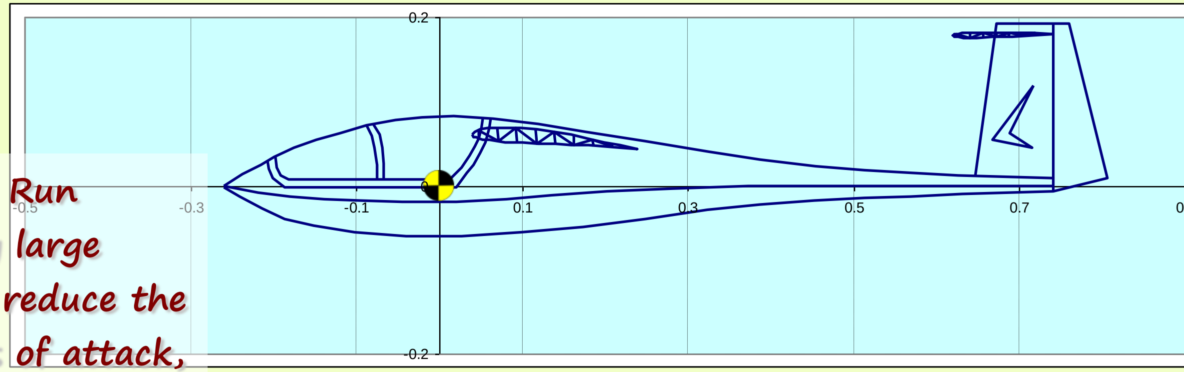
```
Sub Run_Pause()
runpause = Not (runpause)
Do While runpause = True
DoEvents
[D52:K3052] = [D51:K3051].Value
Loop
End Sub

Sub Reset()
[D52:K3052].ClearContents
[D52:K52] = [D47:K47].Value
End Sub
```

Plotting the trajectory of the glider:

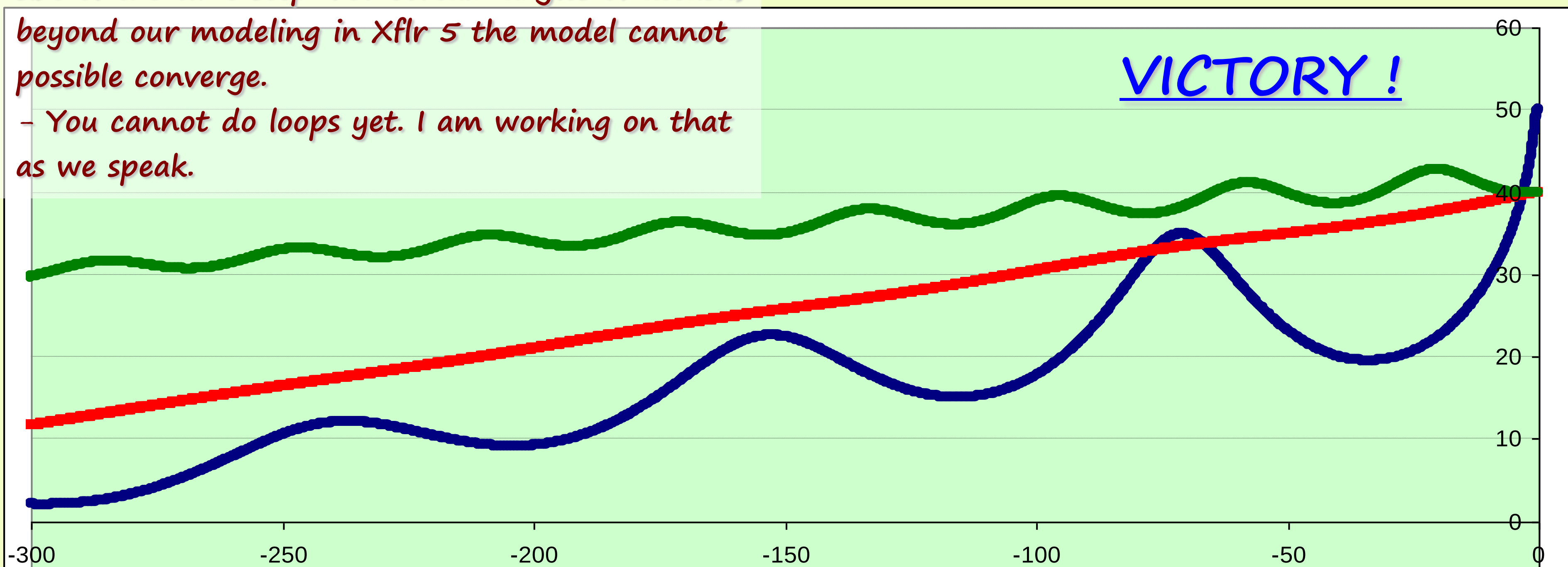
- Insert a 2D scatter chart with the range F51:F3051 as x-data and the range G51:G3051 as the y-data. Also create a speed gauge:

M43: “=SQRT(D52^2+E52^2)”



Run the model:

- Adjust parameters, then => Reset => Run
- If you see a lack of convergence (very large numbers and the chart going haywire) reduce the size of the time step. For certain angles of attack, beyond our modeling in Xflr 5 the model cannot possible converge.
- You cannot do loops yet. I am working on that as we speak.



Three different glider trajectories for various launch conditions and CG positions.

To be continued...