



SCHOOL OF ENGINEERING
YEAR 3 INDIVIDUAL PROJECT

INTERIM REPORT

3D-printed Origami Solar Sails for Next Generation of CubeSats:
Multi-body folding dynamics

Project no. 201

Raymond Arturo Minarro Escalona
BEng Aerospace Engineering with Year in Industry (H426)
Monday 8th of February, 2021

Supervisor: Soldini, S.

1 Experimental project approach and methods

The purpose of the project is to deploy a set of connected, 3D printed panels that convert into a sail large enough to impulse the vessel in a direction normal to sun. In this report, a Python software was written to study the speed, dampening and restive behaviour of the panels when extended by the solar radiation pressure. This will later on evolve onto a more complex simulation with the aim of studying the folding dynamics of a 3D printable sail. A set of aims are:

- To study the multi-body dynamic behaviour of the deploying panels
- To study the feasibility of more arrangements of unfolding panels to be used in the final design.
- To understand the physical principles governing the attitude control and impulse of the cubesat in deep space missions.
- To study the efficacy of using SRP as source of impulse in deep space without the need for fuel.

1.1 Physics

In this problem, common rotational physical principles are used to rotate reference frames assigned to the different panels and the bus of the cubesat, the dynamic system after which the simulation is modelled is iterated and described in each type of simulation below.

The bus is defined as a prism by common expected dimensions of 0.43m by 0.93m by 0.43m (width, length and depth respectively.) and a panel dimensions of length $l = 0.7m$ and height of $h = 0.7m$

The panel is assumed to be similar to high-end domestic PLA 3D printed layers of $50\mu m$. 3D PLA printers work by heating up PLA filament and setting it on a work surface close to each other to cool and form a solid, layered shape. For a PLA filament of $\rho_{PLA} = 1.24g/cm^3$ (*Properties Table* 2019) then the panel will have an area-density of:

$$\rho_{APLA} = \rho_{PLA}t = 1240kg/m^3 \cdot 50 \cdot 10^{-6}m = 0.062kg/m^2$$

With a moment of inertia, for a rectangle about one of its edges:

$$I_{zz} = \frac{1}{3}m * l = \frac{1}{3}l^2h\rho_{APLA} = 7.09 \cdot 10^{-7}$$

1.2 Solar Radiation Pressure

The pressure on a theoretical flat zone, normal to the sun, reaches a value of:

$$p = G_{sc}/c = 5.97\mu N/m^2$$

Where G_{sc} is the solar constant, and c is the speed of light.

The solar constant is used in astrophysics to identify the energy carried by photos expelled from the sun, measured as flux density at an average Earth-Sun distance nominated as an astronomical unit (AU), 1 AU has a fixed value of 149 597 870 700 meters (International Astronomical Union 2012)

Given that the energy is propagated radially as a sphere away from the sun, the energy decrease is inversely proportional to the distance. As such, the inverse square law is applied to find the equivalent pressure at a distance of R_s , multiplied by the cosine of the angle between the sun frame and the bus frame (McInnes 1999)

$$p_{bus} = \frac{p}{R_s} \cos(\alpha) = \frac{G_{sc}}{cR_s} \cos(\alpha)$$

Afterwards, this equivalent pressure p_{bus} may be translated into a force acting on each panel ("panel" frame of reference) which may be oriented initially at an angle β away from the bus frame, in addition to a state-trigger variable d to model the RCDs (Borggrafe, Heiligers, M. & McInnes 2014), as such:

$$F_{srp} = \cos(\beta) p_{bus} A_{panel} (1 + d) = \frac{G_{sc}}{cR_s} \cos(\alpha) \cos(\beta) A_{panel} (1 + d)$$

At this point, from the force normal to the panel, further mechanical simulation values may be derived.

1.3 Simulation

At the current stage of development, a simulation script was coded in Python utilising libraries like SymPy capable of creating virtual reference frames for the sun, bus and panel components, these panels may be rotated and translated during the simulation calculation.

In the discipline of physics and mechanic simulations there exists different methodology and basis by which different types of simulations arrive to a solution, or lack thereof.

Different methods use different amount of resources and have different solving times depending on the complexity of the problem (Holzer, Wuchner & de Meer 2010). For the routine herein, a time-discrete state-space continuous is used.

- **Time discrete:** The simulation begins at an initial state, which may be configured using the variables defined in 1.2. The variables are refreshed every time-step and handled accordingly. The process is repeated until a desired variable "converges", e.g the number of significant figures desired do not change anymore between time-steps, and any further step taken only increases the precision (not necessarily accuracy).
- **State-space continuous:** The variables calculated therein fall within the spectrum of values the simulation range is working with, giving calculations the liberty of yielding more precise values.

As outlined above, the simulations were carried out in discrete time jumps, and a continuous state-space system. Variables like F_{srp} panel torque T_{pn} , angular acceleration α_{pn} , angular speed ω_{pn} were calculated each cycle or time-step of $3.6s$. The time-step was chosen as a compromise between solve time and simulation of the system with real-life values, which are slow-acting due to the nature of the solar radiation pressure. Given the problem summary, the forces acting on a panel assimilate to how forces on a pin-ended beam would behave, adding a torsional spring $-k\theta$ that opposes the torque T_{pn} caused by the force F_{SRP} . An analogue system using a mass m being pushed by F_{SRP} against a spring kx causing a displacement x was used and modelled.

The equivalent governing equation for the translational system is:

$$m\ddot{x} = F_{SRP} - kx$$

and its rotational analogue counterpart:

$$I_{zz}\ddot{\theta} = T_{SRP} - k\theta$$

2 Results to date

2.0.1 Initial simulation system

After the simulation terminated due to the value of θ reaching 90 degrees as seen in figure 1, at which point it would start clipping the bus or folding "backwards", it was seen that the panel was oscillating and never converged.

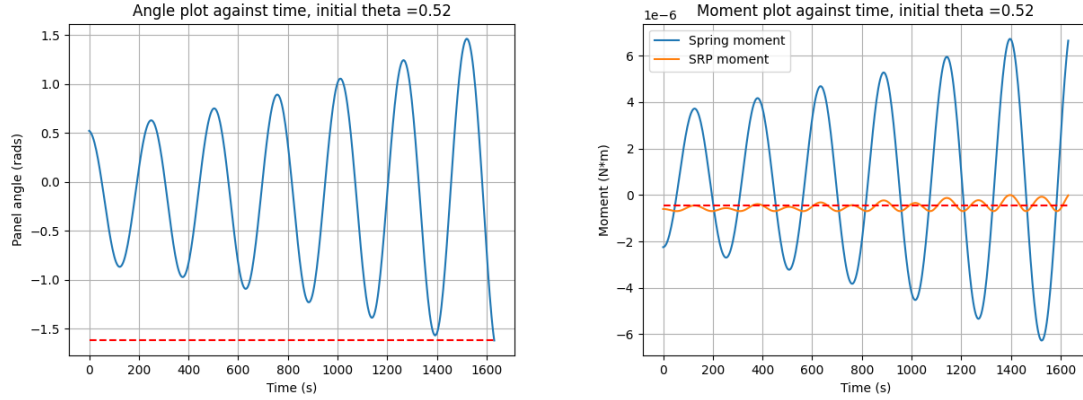


Figure 1: Simulation 1, the panel starts at a $\theta = 0.52 \text{ rads}$ and does not converge.

After further research, the initial system by which the simulation was modelled represented a second-order dynamic system with a governing equation similar to a second-order ordinary differential equation:

$$f''(x) + af'(x) + bf(x) = u$$

Where:

$$a = 2\zeta\omega_n; b = \omega_n^2$$

However, as there is not a $af'(x)$ factor, upon which depends the damping ratio ζ , the present system would oscillate at a frequency of $\sqrt{k/I_{zz}}$

These results indicate that, in real life, the system would have to have a mechanism to stop it from oscillating, be it a damper, or a system that counter-acts moment proportional to angular speed ω or $\dot{\theta}$. The reason why the amplitude of the oscillation increases is related to the limited minimum amount the system can accelerate every time-step, a smaller time-step would yield a system oscillating at a fixed amplitude.

Alternatively, such solution may be achieved by a electrically-controlled spring, which may activate/deactivate at its full spring coefficient k frequently enough to maintain a neutral angular speed and therefore facing angle.

2.0.2 Controllable spring simulation

Instead of the spring being active permanently, an exception to the simulation may be added in such way that:

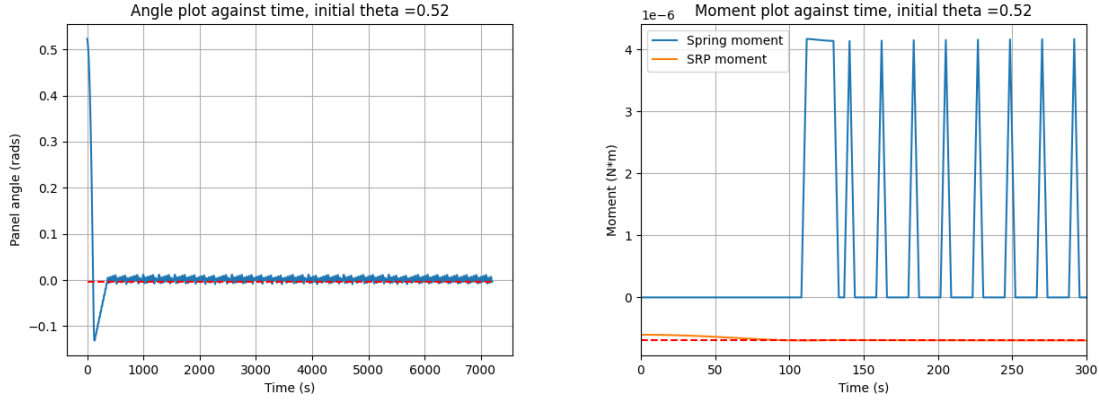


Figure 2: Simulation 2 where in the spring is electrically activated. Resulted in a small converged value around which θ oscillates.

$$\alpha < 0 \implies k = k_s$$

$$\alpha \geq 0 \implies k = 0$$

This means that, when α is negative (accelerating forward towards the sail facing direction), the spring activates and delivers a moment, as it can be seen in figure 2

$$M_{spring} = -k\theta$$

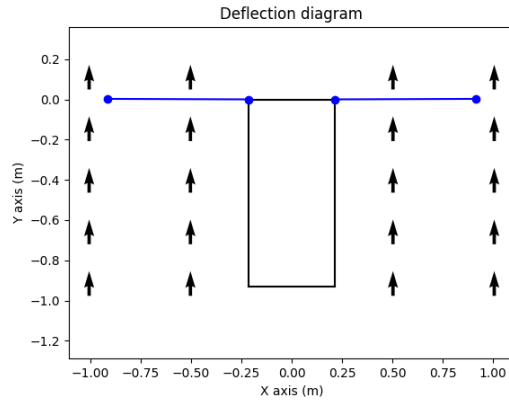


Figure 3: "Approximate" converged diagram for Simulation 2. The panels oscillate close to a value of $\theta = 0$ where F_{srp} is the greatest.

It is worth noting that due to M_{spring} being far larger in magnitude than T_{pn} , a single time-step is enough to cause a $\delta\alpha$ larger than the negative value of α that triggered the spring, driving the panel at a uniform pace until an angle of $\theta = 0$ is reached (See final

"approximate" state in figure 3. This will yield a spring moment of 0 when triggered, self-balancing the system in a constant, small oscillation proportional to the time-step of the simulation.

This resulted on a system that oscillates around the converged value for θ , as with the first simulation, the amplitude of the post-convergence oscillation is proportional to the minimum amount of acceleration in one time-step.

2.0.3 Passively damped simulation

Following the suggestions of adding a dampening mechanism, an arbitrary value of $\zeta = 0.45$ is chosen for the new system (Figure 5), and the simulation tweaked to take this into consideration.

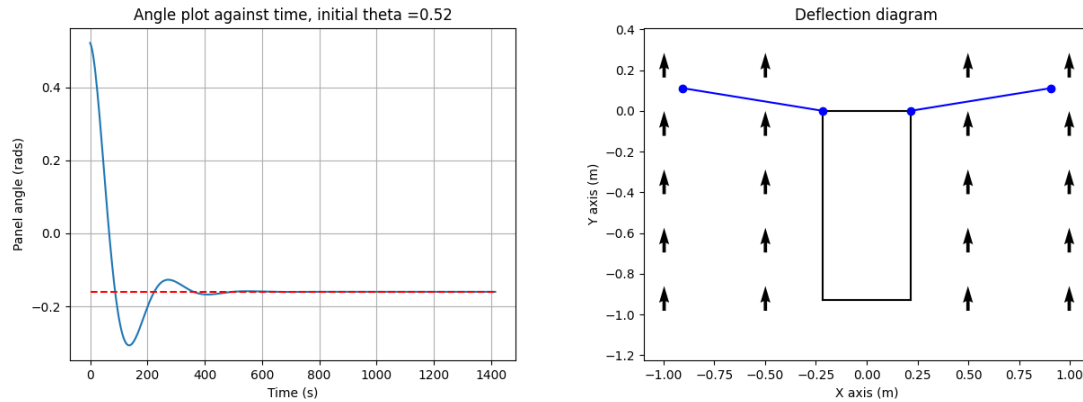


Figure 4: Simulation 3 with passive dampening of $c\omega$, the angle converged faster and definitely where the sum of moments balances out.

As shown in figure 4, the angle converged faster thanks to the dampening in the system, yielding a precise offset angle where the moment of the spring is balanced out by the torque of the panel, this offset decreases slightly the total force achievable, as noted by the higher moment peak. The final state converges at a point where $M_{spring} = T_{pn}$, yielding a lower value of F_{srp} than simulation 2.

3 Work programme

All items to this day are done, kick-off meetings, GIT repo set-up and proposal submission achieved. Making exception for some items in the December-January have had their

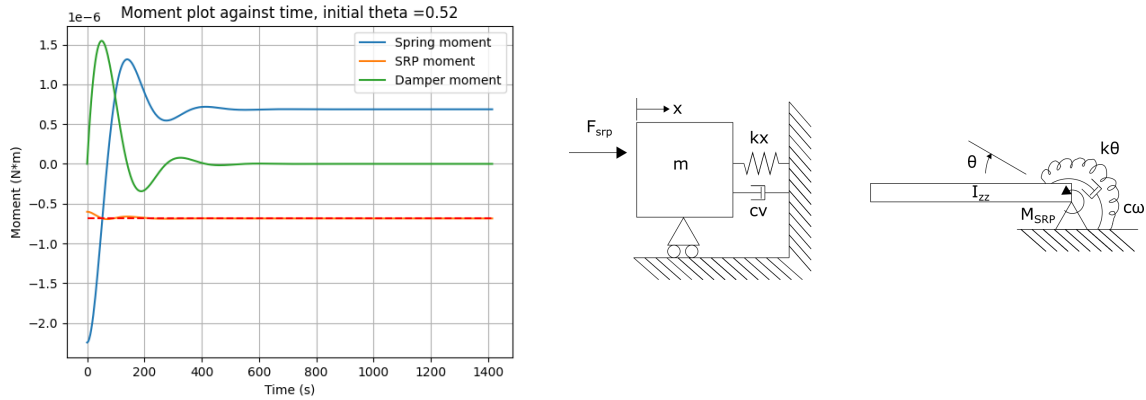


Figure 5: Simulation 3 new dynamic-equivalent system and moment plot obtained, note the spring moment being equally opposite to the SRP moment when the damper moment settles.

deadlines extended. After revision of the Gantt chart, it was decided that items such as "literature review", "principles studies and testing" "liaison for design" "requirement gathering" and "code development" are continuous and concurrent up until the "new code testing and debugging" date. Items "Available code review" and "Available code testing" were moved towards march as it is ideal that the researcher begins the code by understanding the principles from scratch, and then consulting other sources.

4 Future plans

From the results seen from this simulation, it is safe to move forward and extend the code to several connected panels, which may make each cycle have a heavier calculation load, as each cycle and moment calculation will have to consider the motion of the next panel over.

The code, which utilises Reference Frames with three axes, x , y and z , may be adapted to recreate 3D arrangements, for which team liaison must be done to consider the 3D folding possibilities being worked by another member of the project.

Checks need to be made to verify the sizes of the sails needed and the trade with weight are appropriate enough to achieve the orbital transference desired by mission control, which is being worked by another member of the project.

A model can be cloned and simulated into Mathworks' Simscape to validate the Python model and enhance the understanding of the subject.

References

Borggrafe, A., Heiligers, J., M., C. & McInnes, C. (2014), 'Attitude control of large gossamer spacecraft using surface reflectivity modulation', *65th International Astronautical Congress, International Astronautical Federation (IAF)* .

Holzer, R., Wuchner, P. & de Meer, H. (2010), 'Modeling of self-organizing systems: An overview'.

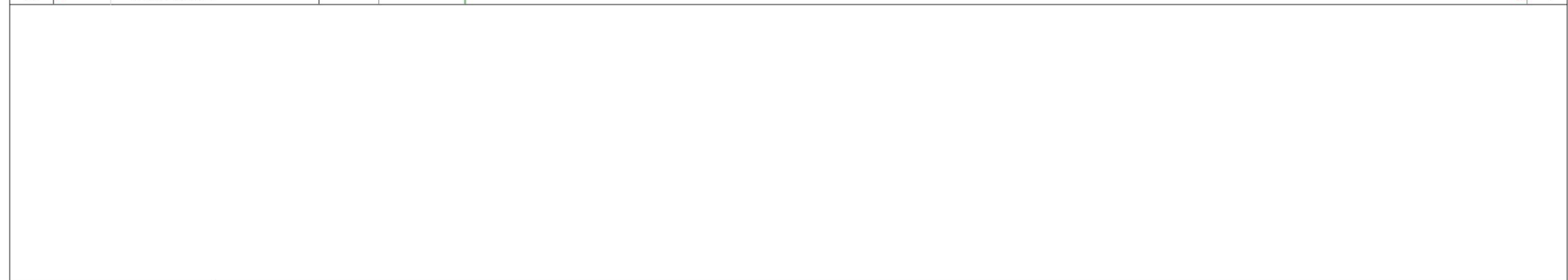
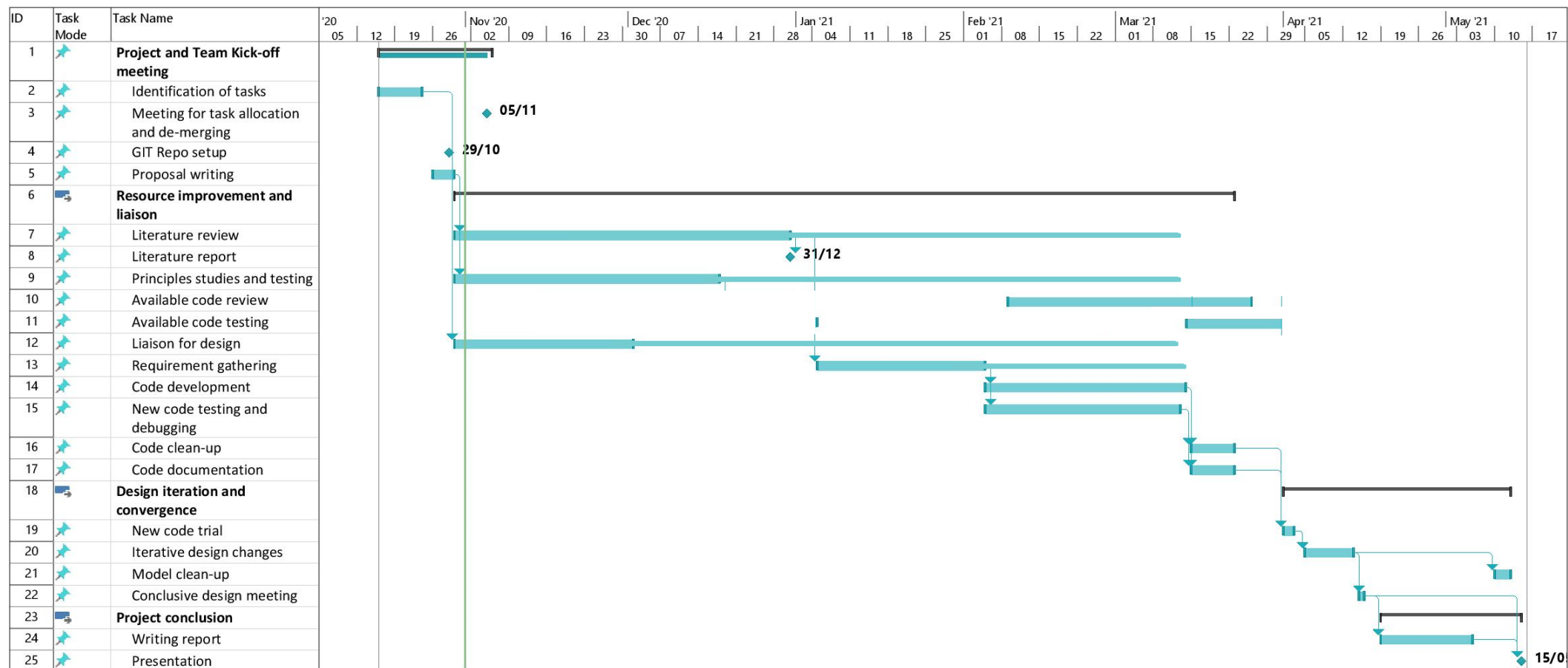
International Astronomical Union (2012), General Assembly of the International Astronomical Union.

McInnes, C. R. (1999), *Solar Sailing: Technology, Dynamics and Mission Applications*, Springer, Verlag, Berlin.

Properties Table (2019).

URL: <https://www.simplify3d.com/support/materials-guide/properties-table/>

A Revised Gantt Chart



Project: 3D printed origami sola
Date: Sun 01/11/20

Task		Project Summary		Manual Task		Start-only		Deadline	
Split		Inactive Task		Duration-only		Finish-only		Progress	
Milestone		Inactive Milestone		Manual Summary Rollup		External Tasks		Manual Progress	
Summary		Inactive Summary		Manual Summary		External Milestone			

Page 1