# Ray tracing

## General description

This program calculates the solar radiation pressure (SRP) resulting from the action of solar rays on the surface of an arbitrarily shaped satellite. The program is written in Matlab. This program takes as input the direction of the sun relative to the satellite (defined in body axes) and the intensity of the solar constant due to the satellite's position relative to the sun. These inputs are precalculated in Matlab with routines that, given an initial time, the sampling time, and a final time, can generate a vector describing the direction of the sun relative to the satellite's position, the magnitude of the solar constant, and any eclipse period. The same routine, not detailed here but referred to in [...], is capable of returning the satellite's attitude in ECI coordinates.

### Data Structure

Once the solar vector in direction and magnitude over time is calculated, this information is used as input to the ray tracing function. The function needs to be initialized by providing the description of the satellite for which SRP (Solar Radiation Pressure) is to be calculated. This information is supplied to Matlab by passing a vector representation of the satellite structure. In an orthogonal XYZ reference, which will represent the satellite's body reference for us, the structure of the satellite's external surfaces is represented with a mesh of points. To simplify the subsequent calculations performed by the ray tracing function, the satellite's surfaces are divided into triangular meshes. This mesh is saved in two matrices. The first matrix, which we'll call *Vertex*, represents the positions in the body reference of all points in the mesh. The second, which we'll call *Faces*, contains, in each row, the 3 row indices of the Vertex vector that describe a triangular face of the mesh.

To give a concrete example, if in row 6 the Faces vector presents three numbers (8,4,5), it means that the triangle 6 of the mesh is defined by the coordinates of the 3 vertices contained in the Vertex matrix at rows 8, 4, and 5 (see Figure).

A satellite mesh with numbers and a grid

Description automatically generated with medium confidence

To calculate radiation pressure, applying the formula..., we need to associate the optical properties of the surface under consideration with each triangular face of the mesh. Therefore, each triangular face of the mesh has been assigned an index called *material\_idx*. This index keeps track of which material (from 0 to n) is associated with the corresponding face.

So, if in row 6 of the *material\_idx* vector we find the value 0, it means that the optical properties of material index 0 should be associated with the triangular mesh described in row 6 of Faces. These optical properties are saved in another matrix (*optical\_par*) that contains the delta rho alpha values for each material. This way, we have a look-up table that will be useful when calculating radiation pressure for the specific face under consideration.

A screenshot of a computer

Description automatically generated

This data structure is finally saved in a specific class-type variable that maintains the same form at each iteration cycle.

A screen shot of a computer screen

Description automatically generatedA screenshot of a computer

Description automatically generated

A graph of a satellite material

Description automatically generated

A screenshot of a computer

Description automatically generated

In relation to the metadata provided by ESA, it should be noted that 2 materials have been added. The 'Material E lato has been added to cover the part that represents the thickness of the solar panels. This area is never illuminated, so this material has no effect. The 'Material Z' has been added to compensate for the fact that in the metadata, the sum of the areas in the Z+ part differs from Z- by approximately 1.4e-2 m^2. In this case as well, the material does not contribute to the SRP, since all optical parameters are set to zero, but it is necessary to avoid leaving a portion of the Box-Wings uncovered.

A green rectangle with red blue and yellow squares

Description automatically generated

A final consideration in the generation of the 3D model (defined as above) is to generate two separate geometries with the same data structure. The first geometry represents the satellite's body, while the second represents only the wings. This is useful because, in the body reference, only the wings are mobile, while the satellite's body will remain fixed during the entire simulation. For each simulation time, considering that the wings can rotate around the Y body axis and must necessarily remain orthogonal to the direction of the sun, the elevation angle that the sun has in the YZ plane is calculated, and the solar panels are rotated by the same angle.

### Iteration Starting

The program, once initialized as described in the previous paragraph, iteratively proceeds as follows:

1. **Solar Panel Rotation**: One of the outputs of the routine that calculates the direction of the sun in the Body reference frame is the vector expressing the normal to the solar panels. From this vector, the rotation angle of the panel in the XZ satellite plane is calculated, and the panel is rotated by the corresponding angle.

2. **Calculation of Normals for Each Face of the Model**: For each integration step, it calculates the outgoing normal vectors for all faces of the meshes in the Body reference frame. Outgoing normals refer to normals pointing outside the solid. These are saved in the *Normals\_mesh* matrix.

3. **Search for Mesh Center Coordinates**: The coordinates of all barycenters of the meshes in the Body reference frame are calculated and saved in the variable *Centers\_mesh.*

4. **Creation of a Virtual Plane Called Source Plane**: At an arbitrary distance from the satellite (5m), a plane is created onto which all mesh centres (*Centers\_mesh*) are projected. The plane changes for each simulation cycle and lies parallel to the direction of the sun.

A diagram of a satellite

Description automatically generated

**Generation of n Source Rays**: From each of these points on the source plane, a vector is generated that has the direction of the solar vector at that moment. This technique ensures the creation of a number of rays equal to the number of meshes that need to be illuminated.

**Conditions for Exclusion** : Not all rays originating from the source plane will illuminate the satellite faces. Some of these faces will be on the shaded side of the satellite, and others might be shadowed by different surfaces.

**A. First Exclusion Condition**:

Faces of the solid whose normal is aligned with the solar vector are excluded. This is determined by checking that the dot product between the solar vector and the normal to the face under consideration is >= 0.

**B. Second Exclusion Condition**:

If the face under consideration is behind the face from which the ray originates, it cannot be illuminated. This condition does not apply to the Box-Wings as there are no faces affected by reflected faces, but it will apply when analyzing more complex satellites.

**Verification of Shaded Faces:** Among all the faces not excluded by the exclusion condition, it is necessary to determine which of these faces are in the shadow of faces already illuminated by a ray. To do this, all face centres of the Box-Wing are sorted in a list according to their distance from the source plane. The first face in this list will definitely be illuminated since it satisfies the exclusion conditions, and there is no other closer face that could cast a shadow on it. The three vertices of this face are projected onto the source plane. The function checks if there are vertices of other faces, projected onto the source plane, falling inside the projection triangle. If this condition is met, it is assumed that the entire face is in shadow and is excluded from the list of illuminated faces.

The function calls itself with the updated list of illuminated faces (as described above) and analyses the next illuminated face in the list closest to the source plane. At the end of this recursive cycle, only faces that are definitely not in the shadow of other faces remain in the list. This function is not necessary for Box-Wings but has been developed in anticipation of more complex satellites.

Once the list of only illuminated and non-shadowed faces is obtained, a vector is generated for each of them that meets the properties of the class `c\_Rays`.

The following information is recorded in this class:

1. The index of the illuminated face
2. Coordinates of the ray's arrival point
3. The direction of the reflected ray
4. The index of the face from which the ray originates (this field remains empty for the first reflection).

A screen shot of a computer program

Description automatically generated

Una volta determinate quali sono le facce realmente illuminati dai raggi del sole, su queste viene calcolato il vettore del raggio riflesso, la posizione di impatto del raggio. Queste informazioni vengono salvate all’interno di una variabile di classe “c\_Rays” che conterrà tutte le informazioni utili necessarie per calcolare l’accelerazione dovuta al raggio incidente. Inoltre questa struttura di dati contiene le informazioni utili come sorgenti dei raggi riflessi (posizione di impatto del raggio e direzione del raggio riflesso). Inoltre , nella stessa struttura dei dati vene salvato l’indice della faccia della mesh da cui proveniva il raggio. Ciò è necessario se si vuole risalire a tutto il percorso che ha fatto il raggio nelle varie riflessione e sfruttare la proditoria dei parametri ottici delle varie superfici colpite per calcolare l’accelerazione generata dalle riflessioni multiple.

Risultati

Per verificare l’accuratezza raggiungibile dal modello di ray-tracing sono stati confrontati i risultati ottenuti da questa sul Box-Wing e quelli ottenuti da un’analisi numerica diretta.

Per il confronto sono stati considerati i metadati forniti da ESA riguardanti il GSAT0201. Il periodo di analisi è stato scelto di 7 giorni con inizio il 22-08-2014 e la frequenza di campionamento posta a 6 minuti.

Il modello numerico precalcolava l’assetto del satellite nel riferimento ECI e la direzione del sole rispetto al satellite durante il periodo in esame. La direzione del vettore solare nel riferimento body così calcolata è stato considerato come input per il ray tracing. A questo è stato aggiunto il valore della costante solare opportunamente modulato in funzione della distanza sole satellite.

La PSD di confronto dei risultati ottenuti è visibile nella figura. Come si vede la differenza tra il modello numerico e il modello di ray-tracing è all’interno dell’erro numerico.

Sfruttando la tecnica di Ray-tracing descritta nella nota tecnica, è stato rea confrontare sulla geometria Box-Wing la SRP calcolata con una integrazione numerica e con la