Halil İbrahim Uğurlu 1877018

EE587 Introduction to Robotics
Project Report I
Dragonfly kinematics and dynamics model generation

1) Dragonfly analysis and assumptions

In this project a dragonfly insect is analyzed and a robotic structure will be proposed inspired by dragonfly. Dragonfly is a flying insect, shown in figure 1. This insect has some specialties among the other flying animals and insects. Mainly, its four airfoils are independent from each other and controlled by separate muscles. It gives the insect very high maneuverability. Also, its airfoils are very tiny and has a spot close to tip and front of airfoil. This spot is denser than the other parts of air foil. So, although dragonfly can only moves the airfoil up and down, the foil makes a motion on pitch axes. On an aerial robot main convention is that, heading direction is x-axis, left side of robot is y-axis and upside is z-axis. Roll-pitch-yaw are rotation according to x-y-z axises respectively. In this convention, the dragonfly can rotate its airfoil in Roll and also there is an additional Pitch motion comes from its structure.

Figure 1: Dragonfly

Assumptions on Dragonfly robot:

- Robot centered coordinates: heading → x, left → y and upside → z.
- There is a body and tail connected to it with a revolute joint.
- There are four wings connected to body with two dof joints. Roll (x) and Pitch (y) rotations allowed according to robot centered coordinates.
- Airfoils are flat.
- Mass of airfoils are very small with respect to body, neglected when the dynamics of body been considered.
- The flicker of airfoils are very rapid, the average force is used when the dynamics of body been considered.
- Each airfoil can make an average equivalent force acting on body from center of gravity of airfoil at initial position of foil according to Roll and Pitch motion characteristics.
- Airfoil be exposed to air friction proportional to its velocity square and normal surface area which is controlled by roll and pitch angles.

2) D-H parameters

In this section first D-H parameters are generated for the moving leaf, body, airfoils and tail of dragonfly separately. Secondly, the homogeneous matrices are derived from these D-H parameters.

Moving leaf is on an arbitrary point in 3D space. Thanks to our assumption about no rotating leaf it has three degree of freedom which are x, y, z Cartesian coordinates with respect to base. In order to represent this behavior 3 prismatic joints are considered as shown in figure 2. D-H parameters are shown in table 1. Frame-3L is placed at the center of leaf where the dragonfly will be landed safely.

Joint	α_i	a_i	d_i	θ_i
1L	90°	0	d^*_{1L}	90°
2L	90°	0	d^*_{2L}	90°
3L	90°	0	d^*_{3L}	90°

Table 1: D-H parametes of the moving leaf

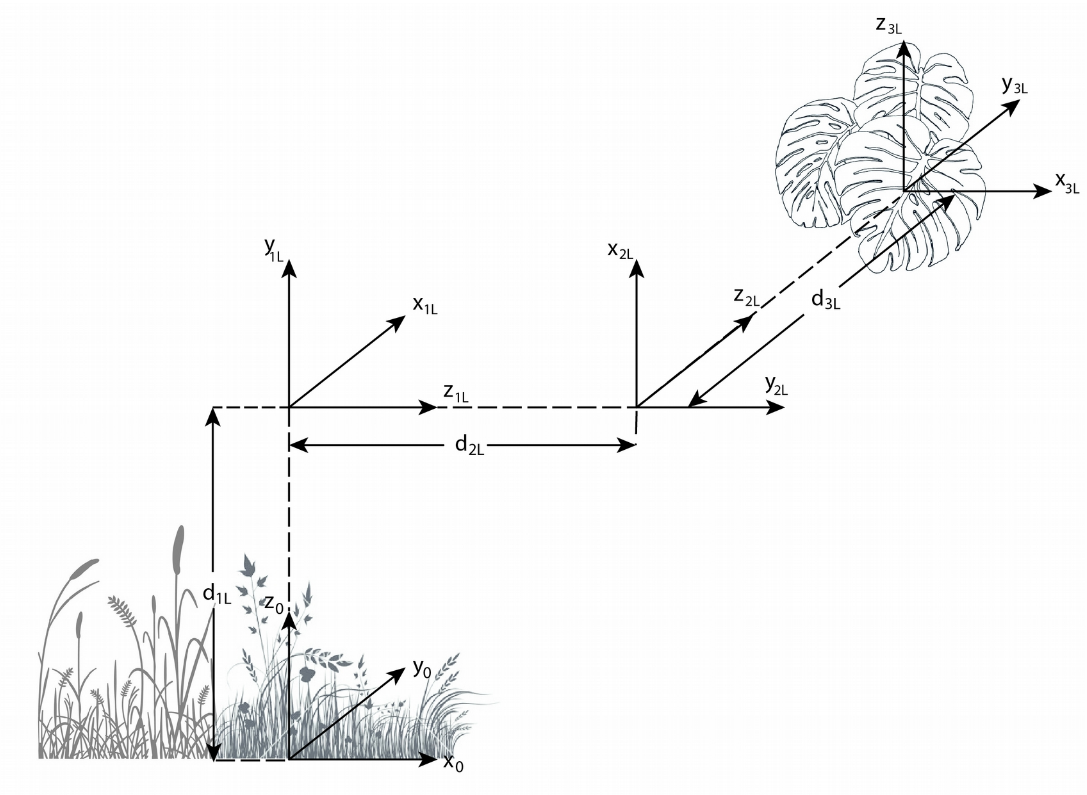

Figure 2: Virtual joints for leaf coordinates

Coordinate of dragonfly body is similar to the leaf. In addition, it has orientation also. First three virtual joints are representing the position same as the leaf and second three joints are representing the orientation of the body. Frame-6 is fixed to the body of dragon fly and as stated in assumptions its x, y, z axises are heading, left and upside of dragonfly respectively. Frame-7 is redundant, it is fixed with respect to frame-6 and no variables defined with it.

However, it is the base for four airfoils of dragonfly. In figure 3 and figure 4, the frames for dragonfly body is shown. D-H parameters are shown in table 2.

Joint	α_i	a_i	d_i	θ_i
1	90°	0	d_1^*	90°
2	90°	0	d_2^*	90°
3	90°	0	d_3^*	90°
4	90°	0	0	90°+ θ_4^*
5	90°	0	0	90°+ θ_5^*
6	90°	0	0	90°+ θ_6^*
7	90°	0	0	90°

Table 2: D-H parameters of center of gravity of Dragonfly body (7 is base for airfoils)

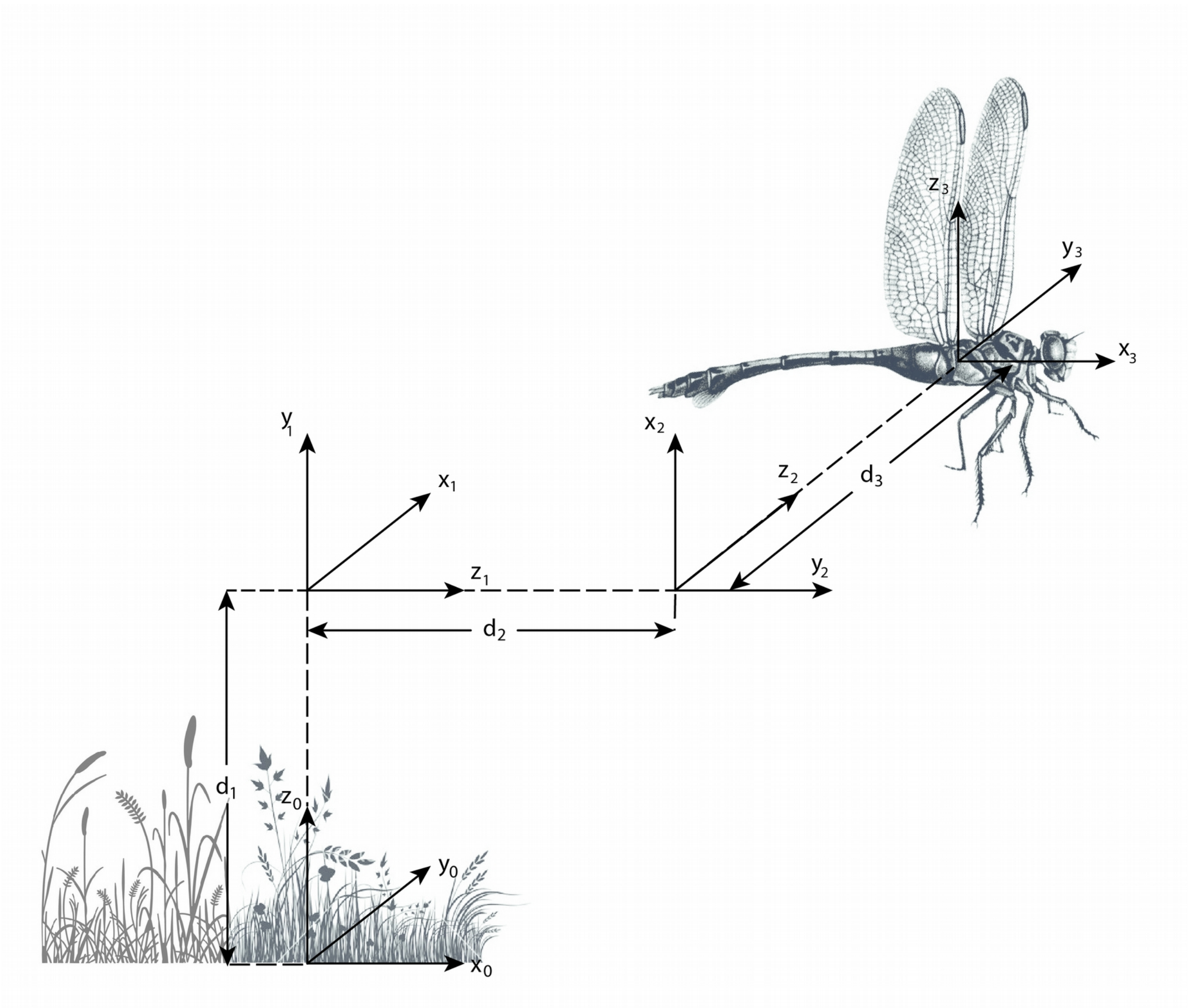

Figure 3: Virtual prismatic joints defining position of dragonfly

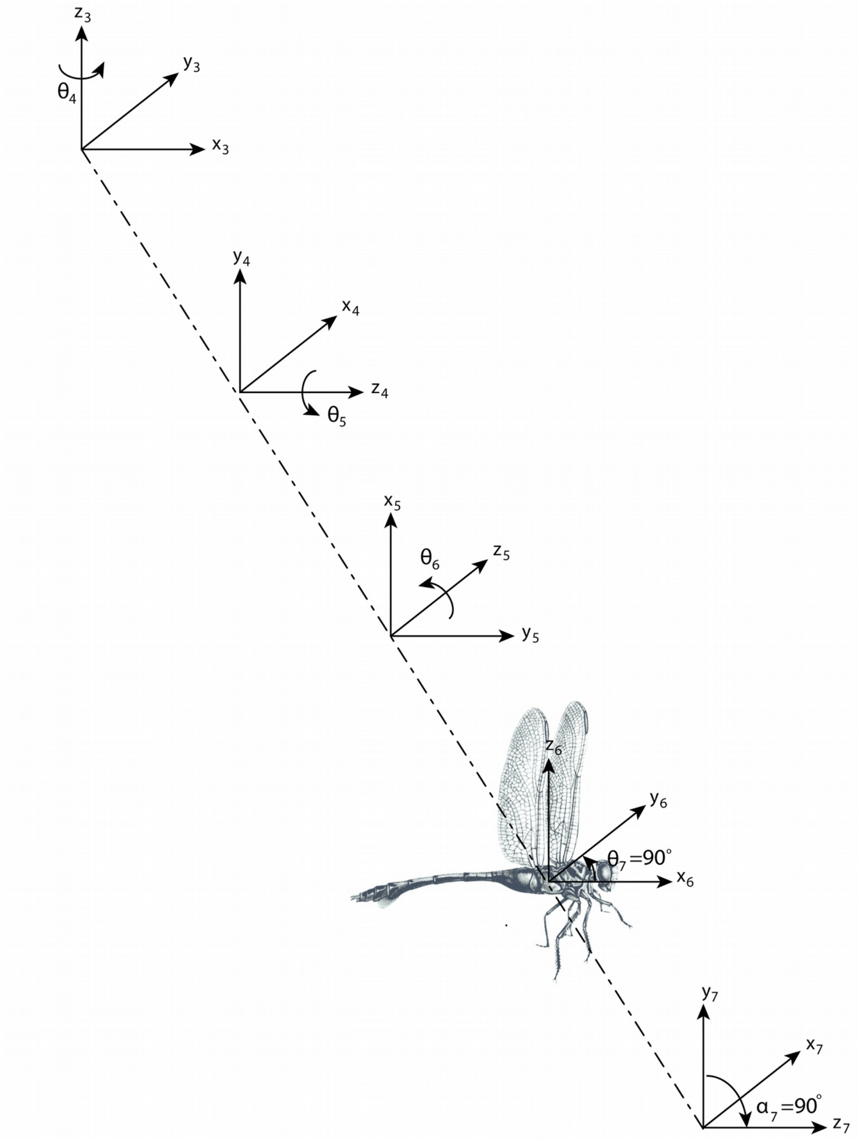

Figure 4: Virtual revolute joints defining orientation of dragonfly

There are four airfoils connected to the body of dragonfly. Each of them is connected with two revolute joint. They are named with first letters of right front(RF), right back(RB), left front(LF) and left back(LB). D-H parameters of each airfoil is shown in tables 3 to 6. In figures 5 to 8, coordinates are visualized.

Joint	α_i	a_i	d_i	θ_i
8RF	-90°	0	d_{8RF}	90°+ $\theta^*_{8\,RF}$
9RF	0°	0	d_{9RF}	$\theta^*_{9\,RF}$

Table 3: D-H parameters of right front airfoil

Joint	α_i	a_i	d_i	θ_i
8RB	-90°	0	$-d_{8RB}$	90°+ $\theta^*_{8\,RB}$
9RB	0°	0	d_{9RB}	$\theta^*_{9\,RB}$

Table 4: D-H parameters of right back airfoil

Joint	α_i	a_i	d_i	θ_i
8LF	-90°	0	d_{8LF}	90°- θ^*_{8LF}
9LF	0°	0	$-d_{9LF}$	θ^*_{9LF}

Table 5: D-H parameters of left front airfoil

Joint	α_i	a_i	d_i	θ_i
8LB	-90°	0	$-d_{8LB}$	90°- θ^*_{8LB}
9LB	0°	0	$-d_{9LB}$	θ^*_{9LB}

Table 6: D-H parameters of left back airfoil

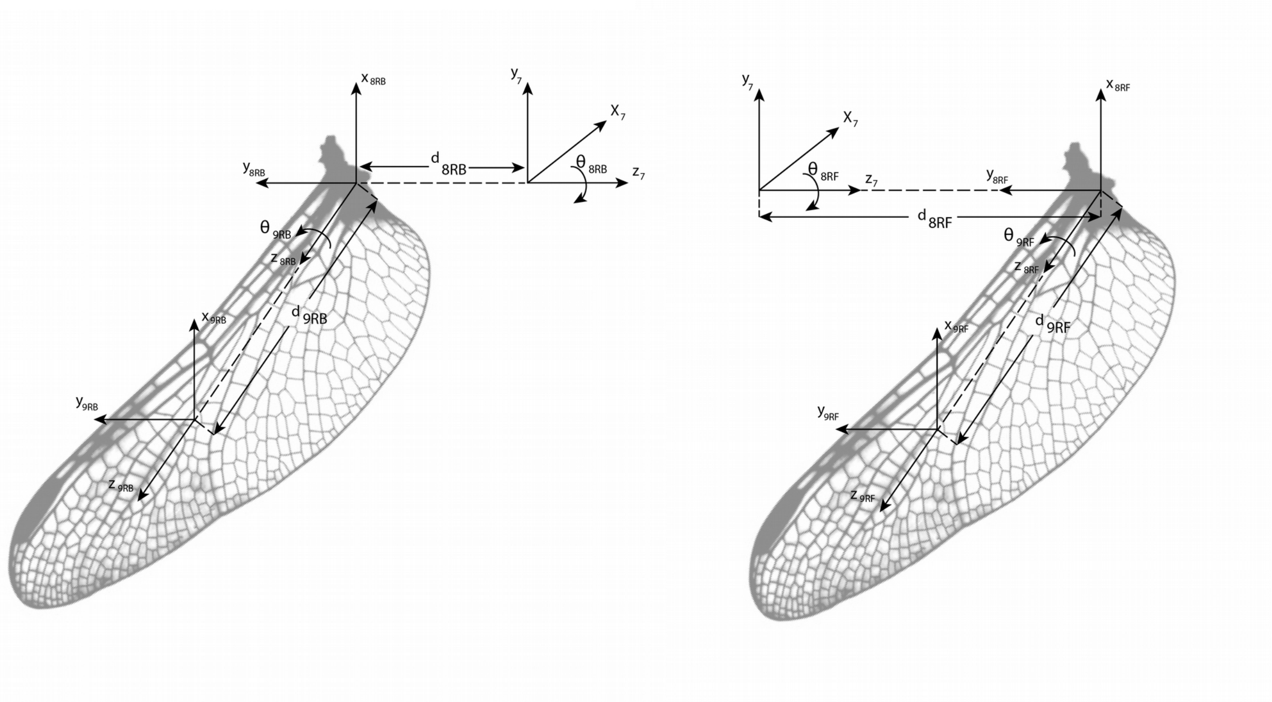

Figure 5 and 6: Coordinates for right wings(back and front)

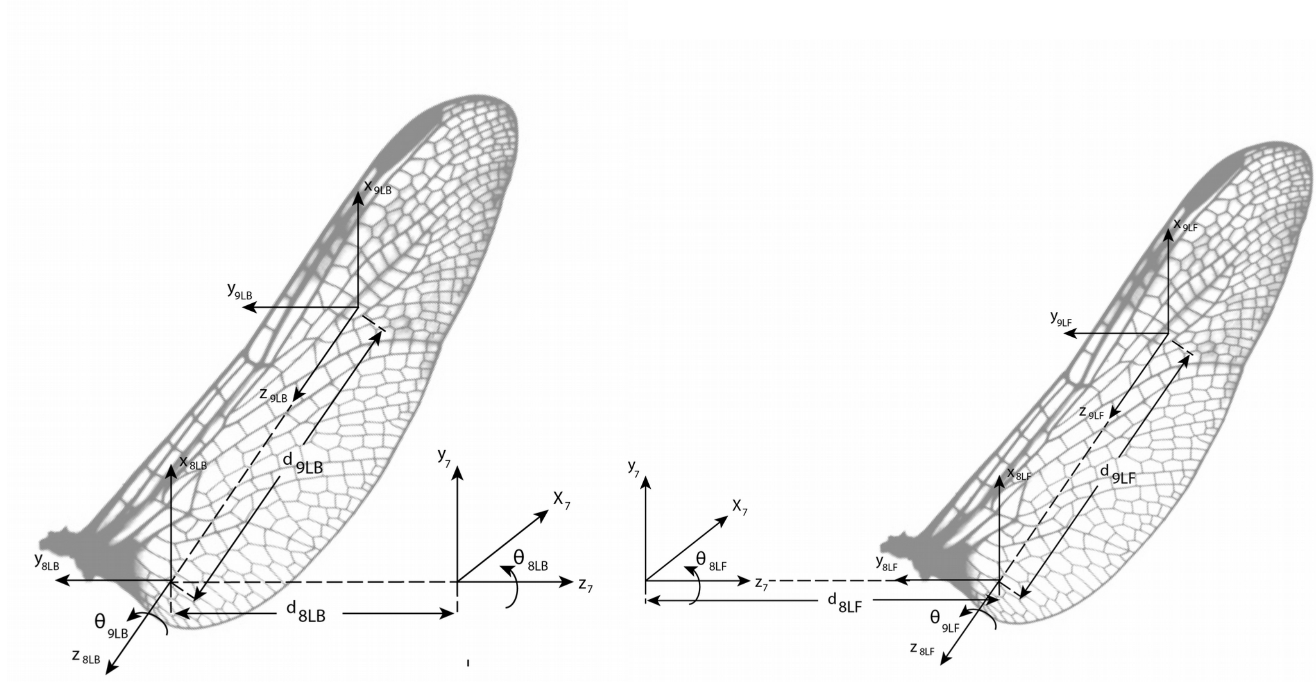

Figure 7 and 8: Coordinates for left wings (back and front)

Lastly, the tail is connected to the body of dragonfly with an only revolute joint. It can move up and down. D-H parameters for tail is shown in table 7. Here frame 7T is redundant there is no variable and it is obtained from frame 6. Frame 8T is centered at the center of mass of the tail.

Joint	α_i	a_i	d_i	θ_i
7T	90°	$-a_{7T}$	0	0°
8T	0°	$-a_{8T}$	0	θ_{8T}^{*}

Table7: D-H parameters of tail (7T is base for tail, derived from frame#6)

For each row of D-H parameter tables, homogeneous transformation matrices are derived from the below formula.

$$^{i-1}A_i = Rot_{(z,\theta_i)} Trans_{(z,d_i)} Trans_{(x,a_i)} Rot_{(x,\alpha_i)}$$

$$^{i-1}A_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

All individual homogeneous transforms are listed below:

$$^{0}A_{1L} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{1L} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{1L}A_{2L} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{2L} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{2L}A_{3L} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{3L} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{0}A_{1} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{1}A_{2} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{2}A_{3} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_{3} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{3}A_{4} = \begin{bmatrix} -\sin(\theta_4) & 0 & \cos(\theta_4) & 0 \\ \cos(\theta_4) & 0 & \sin(\theta_4) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{4}A_{5} = \begin{bmatrix} -\sin(\theta_5) & 0 & \cos(\theta_5) & 0 \\ \cos(\theta_5) & 0 & \sin(\theta_5) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{5}A_{6} = \begin{bmatrix} -\sin(\theta_6) & 0 & \cos(\theta_6) & 0 \\ \cos(\theta_6) & 0 & \sin(\theta_6) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{6}A_{7} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{7}A_{8RF} = \begin{bmatrix} -\sin(\theta_{8RF}) & 0 & -\cos(\theta_{8RF}) & 0 \\ \cos(\theta_{8RF}) & 0 & -\sin(\theta_{8RF}) & 0 \\ 0 & -1 & 0 & d_{8RF} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{8RF}A_{9RF} = \begin{bmatrix} \cos(\theta_{9RF}) & -\sin(\theta_{9RF}) & 0 & 0 \\ \sin(\theta_{9RF}) & \cos(\theta_{9RF}) & 0 & 0 \\ 0 & 0 & 1 & d_{9RF} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{7}A_{8RB}=\begin{bmatrix} -\sin(\theta_{8RB}) & 0 & -\cos(\theta_{8RB}) & 0 \\ \cos(\theta_{8RB}) & 0 & -\sin(\theta_{8RB}) & 0 \\ 0 & -1 & 0 & -d_{8RB} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{8RB}A_{9RB}=\begin{bmatrix} \cos(\theta_{9RB}) & -\sin(\theta_{9RB}) & 0 & 0 \\ \sin(\theta_{9RB}) & \cos(\theta_{9RB}) & 0 & 0 \\ 0 & 0 & 1 & d_{9RB} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{7}A_{8LF}=\begin{bmatrix} \sin(\theta_{8LF}) & 0 & -\cos(\theta_{8LF}) & 0 \\ \cos(\theta_{8LF}) & 0 & \sin(\theta_{8LF}) & 0 \\ 0 & -1 & 0 & d_{8LF} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{8LF}A_{9LF}=\begin{bmatrix} \cos(\theta_{9LF}) & -\sin(\theta_{9LF}) & 0 & 0 \\ \sin(\theta_{9LF}) & \cos(\theta_{9LF}) & 0 & 0 \\ 0 & 0 & 1 & -d_{9LF} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{7}A_{8LB}=\begin{bmatrix} \sin(\theta_{8LB}) & 0 & -\cos(\theta_{8LB}) & 0 \\ \cos(\theta_{8LB}) & 0 & \sin(\theta_{8LB}) & 0 \\ 0 & -1 & 0 & -d_{8LB} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{8LB}A_{9LB}=\begin{bmatrix} \cos(\theta_{9LB}) & -\sin(\theta_{9LB}) & 0 & 0 \\ \sin(\theta_{9LB}) & \cos(\theta_{9LB}) & 0 & 0 \\ 0 & 0 & 1 & -d_{9LB} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{6}A_{7T}=\begin{bmatrix} 1 & 0 & 0 & -a_{7T} \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{7T}A_{8T}=\begin{bmatrix} \cos(\theta_{8T}) & -\sin(\theta_{8T}) & 0 & -a_{8T}\cos(\theta_{8T}) \\ \sin(\theta_{8T}) & \cos(\theta_{8T}) & 0 & -a_{8T}\sin(\theta_{8T}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3) Kinematics

Kinematics are observed in two different cases: flight of dragonfly and landing to a leaf.

a) Case 1: Flight

Here, we are interested in the position and orientation of dragonfly mainly. This 6-dof transformation is represented by 6 virtual joints in section 2 and $^{0}A_1, {}^{1}A_2, {}^{2}A_3, {}^{3}A_4, {}^{4}A_5, {}^{5}A_6$ transformations are generated accordingly. Also, I ordered the virtual links such that $^{0}A_1, {}^{1}A_2, {}^{2}A_3$ transforms are representing the position and $^{3}A_4, {}^{4}A_5, {}^{5}A_6$ transforms are representing the orientation with respect to world frame. Thanks to this ordering we can apply kinematic decoupling. The transformation matrices for both couplings are derived below. Now, we can put our dragonfly anywhere by applying inverse kinematics and finding virtual joint parameters.

$$^{0}A_3={}^{0}A_1{}^{1}A_2{}^{2}A_3=\begin{bmatrix} 1 & 0 & 0 & d_2 \\ 0 & 1 & 0 & d_3 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{3}A_6={}^{3}A_4{}^{4}A_5{}^{5}A_6$$

$$^{3}A_6=\begin{bmatrix} c(\theta_4)*c(\theta_6)-s(\theta_4)*s(\theta_5)*s(\theta_6) & -c(\theta_5)*s(\theta_4) & c(\theta_4)*s(\theta_6)+c(\theta_6)*s(\theta_4)*s(\theta_5) & 0 \\ c(\theta_6)*s(\theta_4)+c(\theta_4)*s(\theta_5)*s(\theta_6) & c(\theta_4)*c(\theta_5) & s(\theta_4)*s(\theta_6)-c(\theta_4)*c(\theta_6)*s(\theta_5) & 0 \\ -c(\theta_5)*s(\theta_6) & s(\theta_5) & c(\theta_5)*c(\theta_6) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^0A_6 = {}^0A_3\,{}^3A_6 = \begin{bmatrix} c(\theta_4)*c(\theta_6)-s(\theta_4)*s(\theta_5)*s(\theta_6) & -c(\theta_5)*s(\theta_4) & c(\theta_4)*s(\theta_6)+c(\theta_6)*s(\theta_4)*s(\theta_5) & d_2 \\ c(\theta_6)*s(\theta_4)+c(\theta_4)*s(\theta_5)*s(\theta_6) & c(\theta_4)*c(\theta_5) & s(\theta_4)*s(\theta_6)-c(\theta_4)*c(\theta_6)*s(\theta_5) & d_3 \\ -c(\theta_5)*s(\theta_6) & s(\theta_5) & c(\theta_5)*c(\theta_6) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly, wings and tail are transformed with respect to dragonfly body. These are derived by the formulas;

$^6A_{9RF} = {}^6A_7\,{}^7A_{8RF}\,{}^{8RF}A_{9RF}$ $\quad ^6A_{9RB} = {}^6A_7\,{}^7A_{8RB}\,{}^{8RB}A_{9RB}$

$^6A_{9LF} = {}^6A_7\,{}^7A_{8LF}\,{}^{8LF}A_{9LF}$ $\quad ^6A_{9LB} = {}^6A_7\,{}^7A_{8LB}\,{}^{8LB}A_{9LB}$

$^6A_{8T} = {}^6A_{7T}\,{}^{7T}A_{8T}$

b) Case 2: Landing on a moving leaf

In case of landing first we should construct the leaf to world transformation. It is derived as; $^0A_{3L} = {}^0A_{1L}\,{}^{1L}A_{2L}\,{}^{2L}A_{3L} = \begin{bmatrix} 1 & 0 & 0 & d_{2L} \\ 0 & 1 & 0 & d_{3L} \\ 0 & 0 & 1 & d_{1L} \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Here we want the dragonfly to land on the leaf. Which means both dragonfly and leaf becomes at the same point with respect to world frame. So, landing condition can be formulated as $^0A_{3L} = {}^0A_6$. Solving by inverse kinematics we want to make $\theta_{4,}\theta_{5,}\theta_6 = 0$ and $d_i = d_{iL}\, for\, i = 1,2,3$.

4) Dynamical model

In this section the dynamical model generation procedure is discussed. Dynamical model are solved in two subproblems thanks to the assumption on airfoil mass and speed. First we solve dynamics of each airfoil separately, considering they are connected to a fixed body since their speed is too high and mass is too low with respect to body. Then we formulate a net force on each airfoil and reflect them into dragonfly body dynamics.

a) Airfoil dynamics

Since all of the airfoils are similar and symmetric we only solve for one airfoil's dynamic. As a convention I take $\phi = \theta_{8RB}$ as flapping angle and $\beta = \theta_{9RB}$ as wing inclination. Also, define the maximum and minimum values of flapping angle as ϕ_{top} and ϕ_{bottom} and average angle is $\phi_{av} = (\phi_{top} + \phi_{bottom})/2$. Lastly, we are interested with rate of change of flapping angle. However, we are assumed that wing inclination angle is constant through top to bottom or bottom to top.

Here we cannot apply Lagrange dynamics since the mass is negligibly small. So, the kinetic and potential energies are very low also with their derivatives with respect to friction observed by airfoils. So, we are applying torque commands to change flapping angle and it is projected to a force in 3D which is air friction.

We can decompose this force in x, y and z directions. Let the area of an airfoil becomes A. The averaged friction in one direction is proportional to cross-sectional area and square of velocity. First we consider the force through x direction of body frame(frame 6).

$$F_x = K_x A \sin(\beta)\dot{\phi}^2$$

Here K_x is a constant. Also there is another force perpendicular to airfoil at $\phi=\phi_{av}$. This force is decomposed in y and z directions of body frame according to the formulas below;

$$F_y=-BA\cos(\beta)\sin(\phi_{av})\dot{\phi}^2$$
$$F_z=-BA\cos(\beta)\cos(\phi_{av})\dot{\phi}^2$$

Here B is friction constant. It is different than K_x because the behavior is different in this directions.

Also, assume that when the airfoils goes from bottom to top, it arranges β = 90 degrees such that there is no force applied.

b) Body dynamics

The body is a single body and it is obeying Newton-Euler law. Suppose that the forces generated by air foils represented with F as a combined force and torques at center of mass of body: $F=\begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix}$. This force is applied to our body with mass m and moment of inertia tensor I. By Newton-Euler formulation $F+F_{gravity}=Ma$ where $F_{gravity}$ is the gravitational force effected by the body. $M=\begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{bmatrix}$ and $a=\begin{bmatrix} a_x \\ a_y \\ a_z \\ \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix}$ are inertia matrix combination of m and the tensor I and the acceleration vector, respectively.

5) Jacobian

In this section Jacobian matrix of Dragonfly will be obtained. Jacobian of the body with respect to world and the Jacobians of each airfoil is obtained separately. First start with body Jacobian.

J_{body} is a 6x6 matrix that transforms time derivative of first six generalized coordinates into the velocity vector of body with respect to world frame. J_{body} Is calculated as below;

$$J_{body}=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos(\theta_4) & -\cos(\theta_5)\sin(\theta_4) \\ 0 & 0 & 0 & 0 & \sin(\theta_4) & \cos(\theta_4)\cos(\theta_5) \\ 0 & 0 & 0 & 1 & 0 & \sin(\theta_5) \end{bmatrix}$$

Secondly, the airfoil Jacobians are calculated with respect to frame-6 which is the body fixed frame. So, velocities of each airfoil could be controlled by controlling velocities of revolute joints of airfoils. These Jacobians are 6x2 matrices because we have two generalized coordinates for each airfoil. Transition from frame-6 to frame-7 is a constant, so there is no generalized coordinate. However, as stated before the Jacobians are generate the velocities with respect to frame 6;

$$J_{RF}=\begin{bmatrix} 0 & 0 \\ d_{9RF}\sin(\theta_{8RF}) & 0 \\ -d_{9RF}\cos(\theta_{8RF}) & 0 \\ 1 & 0 \\ 0 & -\cos(\theta_{8RF}) \\ 0 & -\sin(\theta_{8RF}) \end{bmatrix} \quad J_{LF}=\begin{bmatrix} 0 & 0 \\ d_{9LF}\sin(\theta_{8LF}) & 0 \\ d_{9LF}\cos(\theta_{8LF}) & 0 \\ 1 & 0 \\ 0 & -\cos(\theta_{8LF}) \\ 0 & \sin(\theta_{8LF}) \end{bmatrix}$$

$$J_{RB}=\begin{bmatrix} 0 & 0 \\ d_{9RB}\sin(\theta_{8RB}) & 0 \\ -d_{9RB}\cos(\theta_{8RB}) & 0 \\ 1 & 0 \\ 0 & -\cos(\theta_{8RB}) \\ 0 & -\sin(\theta_{8RB}) \end{bmatrix} \quad J_{LB}=\begin{bmatrix} 0 & 0 \\ d_{9LB}\sin(\theta_{8LB}) & 0 \\ d_{9LB}\cos(\theta_{8LB}) & 0 \\ 1 & 0 \\ 0 & -\cos(\theta_{8LB}) \\ 0 & \sin(\theta_{8LB}) \end{bmatrix}$$

The calculations of transformation matrices and Jacobian matrices up to here is done by MATLAB. Their codes are shared at the github link provided at the end of this report.

6) Control

In this section control strategy of our Dragonfly robot is examined for flight maneuvers and landing on a leaf.

a) Flight maneuvers

In this section control of dragonfly in 3D environment is examined. We know from the dynamical analysis that a force is generated by each airfoil. These four forces are acted on the body as a combined forces and torques at the center of mass. Thus, we can control the body by generating a desired force. A PD controller can be applied here with an additive gravitational force. The gravitational term is applied in order to decrease the steady state error to zero.

After obtaining a desired force and torque set, we map this to equivalent airfoil force and equivalent airfoil joint position and velocities. However, there are constraints on these joint position and velocities. This constraints are limit our controller performance. We know from airfoil dynamics that higher forces in z-direction about dragonfly can be more easily generated. So, when dragonfly makes vertical movements it keeps itself rotationally aligned about world frame. However, when it makes horizontal movement, it gives the heading direction to the movement direction. Next, it goes to desired position with a pitch angle such that body frames z-axis be coupled with movement direction. So, it can move faster in horizontal motions also.

b) Landing on a leaf

Landing on a leaf is similar to flight maneuver when considering the desired destination is on top of the leaf. However now there is also a leaf in the environment which can be an obstacle for our path. We can design a switching like controller for this purpose.

First we should ensure from being directly under the leaf, in this case the target is going left side to a safe region. Then, switch the controller to next step. The aim of this step is gaining the required altitude such that Dragonfly be above the leaf. In the next step the horizontal alignment is controlled. When we ensure about being vertically upwards of the leaf now go to downwards and sit on leaf.

In this last step there is a collision happens that we should consider. In order to have a smooth connection we should adjust the PD gains of last step such that we have a overdamped controller.

7) Simulation

Final section of this project report is on the simulation of Dragonfly robot. The simulation platform used is Gazebo. Gazebo is a Multi Robot open source physics simulator. It can take 3D mesh models for rigid bodies. Generate robotic models by its model editor. The model editor is capable to defining links and joints. Figure 9 and 10 shows the simulation environment.

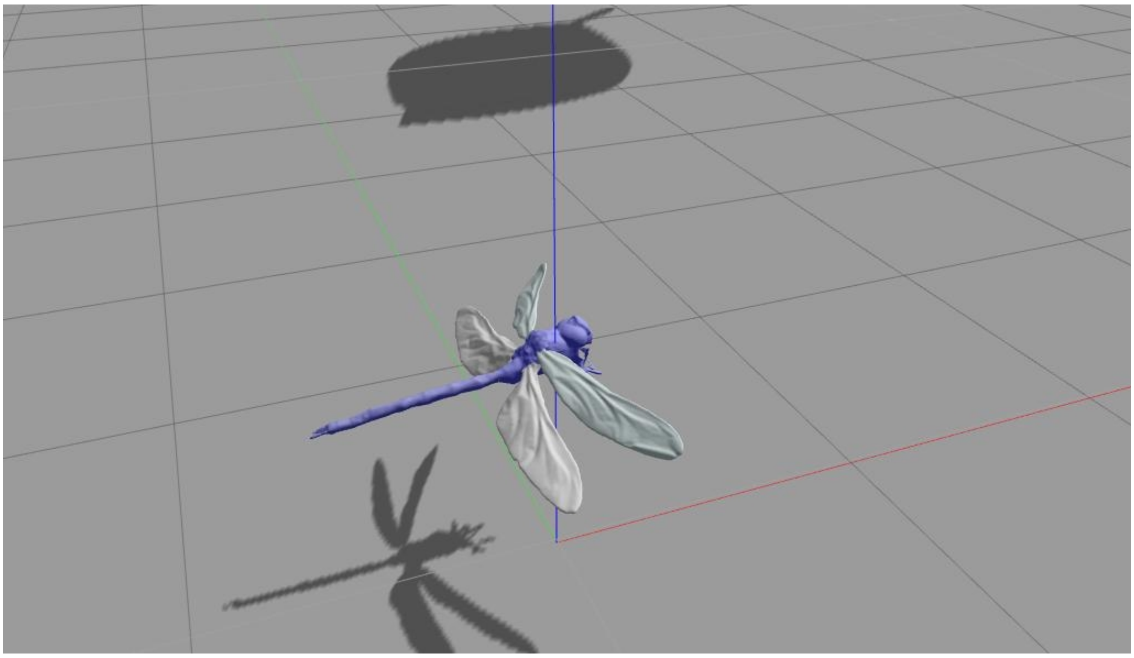

Figure 9: Dragonfly model in Gazebo Simulator

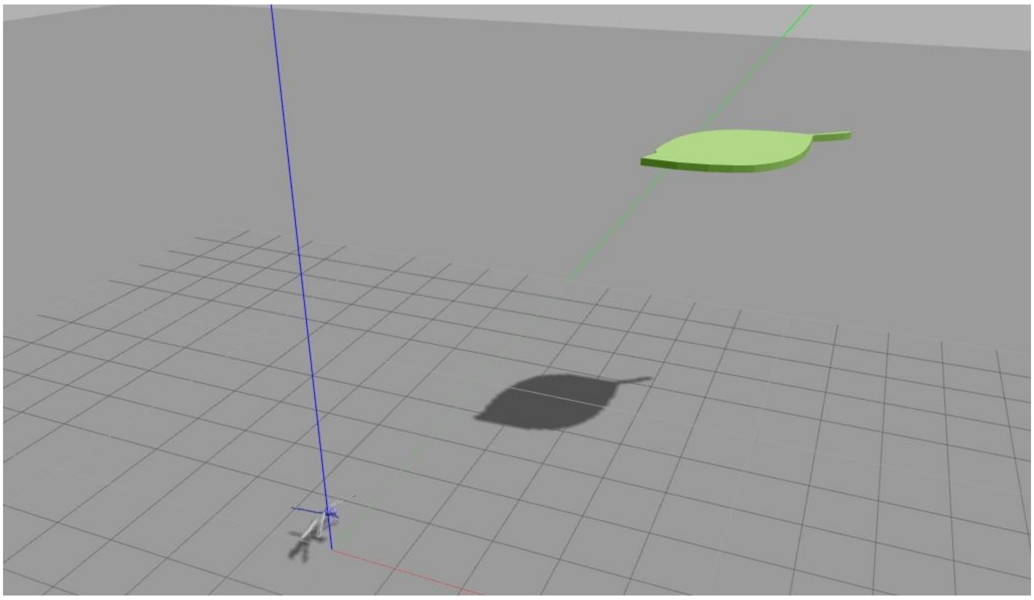

Figure 10: Dragonfly together with the leaf in Gazebo Simulator

All documents of this project is open at my github page: https://github.com/halil93ibrahim/dragonfly