

# OpenKITE: User Manual

Petr Listov, Colin Jones

April 25, 2018

# Modelling

OpenKITE [Kite Identification Tracking & Estimation] package contains implementation of several wind energy kite models:

- + Rigid-wing single line kite [reference to future paper]
- + Kinematic model ("tricycle on a sphere") [reference to Sanket]
- + Rigid body 6DoF model

## Rigid-wing single line kite

User is provided with

## C++ API of the simulator

```
1 #include "openkite/kite.h"
2
3 std::string kite_config_file = "path-to-config-file";
4 KiteProperties kite_props = kite_utils::LoadProperties(kite_config_file);
5 AlgorithmProperties algo_props;
6 algo_props.Integrator = RK4;
7
8 KiteDynamics kite(kite_props, algo_props);
```

Listing 1: Instantiation of the kite model

## ROS kite simulator

Kite simulator can be accessed simply by calling the command:

```
1 roslaunch openkite simulator.launch
```

User may edit the launch file to specify the initial point of the simulation:

```
1 <rosparam param="init_state"> [vx,vy,vz, wx,wy,wz, x,y,z, q,qx,qy,qz]</rosparam>
```

Configuration file describing the wing and tether properties:

```
1 <param name="kite_params" value="path-to-configuration-file" />
```

To setup 3D visualisation it is necessary to run RViz from ROS distribution prior to simulator:

```
1 rosrn rviz rviz
```

### Published topics:

- + *kite\_state* (sensor\_msgs::MultiDOFJointState) : navigation data

### Subscribed to topics:

- + *kite\_control* (openkite::aircraft\_controls) : control input