# Optimal control for power generating kites

Sanket Diwale

June 11, 2014

## 1 Introduction

The objective of an airborne wind energy system is to exploit the consistent and larger wind speed available at high altitudes that conventional wind turbines cannot reach. This however requires a reliable controller design that can keep the airborne system flying for long durations in varying environmental conditions, while respecting all required operation constraints.

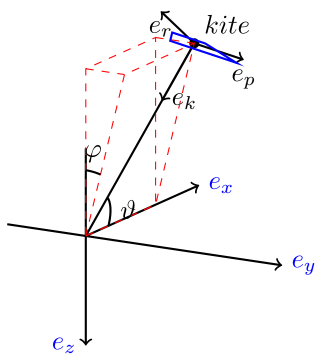
While several competing concepts exist for such systems most systems typically consists of a tethered fixed wing glider or a kite flying cross wind loops to produce power. The power may be generated in two ways. The first is called the drag mode, in which small turbines are placed on-board pointing in the direction of flight, converting the drag into rotational motion in the turbine and thus producing power. The second method is called the lift mode, in which the kite flying cross wind also reels out. The tether during the reel out turns a turbine placed on the ground which generates the power. The kite is then reeled back in so that this power generating cycle can be repeated. During the reel-in, the kite must be manoeuvred in order to minimize the tether forces generated and thus minimize the power consumed. This gives us a net power producing cycle.

Power generating kites provide an interesting paradigm for the application of optimal control as it enables a controller design that can guarantee constraint satisfaction. Even though model nonlinearities and mismatch make it difficult to implement such a controller in practice, numerical optimal control solutions for such systems can still provide useful insights into the required engineering solution and act as a benchmark for comparing different solutions. Further it can also form an important step for a practical MPC controller for the system.

In this article we provide numerical solutions for optimal power generation in the lift mode for a complete power cycle. The problem is essentially an infinite horizon optimal control problem which is then attempted to solve by three methods, finite time horizon approximation, search for the optimal periodic limit cycle and time transformation of the infinite horizon to a finite half-open space. We find the optimal periodic limit cycle formulation to be the most computationally efficient (least time consuming) of the three formulations. While the finite time approximation yields a solution close to the optimal limit cycle, it suffers due to nonlinearities in the system failing to give a consistent limit cycle and also requires large computational time. The time transformation method fails to solve the problem at all as it requires problems to have solutions, exponentially converging to a steady state which is not the case for our system in the time domain.

## 2 System Description

The airborne wind energy system under consideration here is a tethered kite, generating power using the lift mode of generation as described in the introduction. The kite drives the turbine with a main tether, while two additional steering lines are used to apply deflections on the kite for turning. Thus the available actuation for the controller is the reel-out, reel-in control on the main line and the differential steering of the two steering lines.



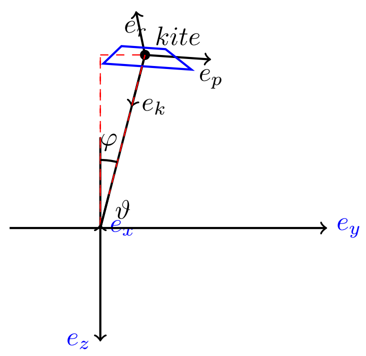




Figure 1: Reference frames and angles

### 2.1 Frame of reference

All the vector variables described below are written with respect to an Earth fixed frame (shown in figure 1) given by the basis vectors $\vec{e}_x, \vec{e}_y, \vec{e}_z$. Additionally a kite fixed coordinate frame is used to derive some relations, whose basis vector are given by $\vec{e}_r, \vec{e}_p, \vec{e}_k$ corresponding to the roll, pitch and yaw axes of the kite respectively.

The kite position is described using "polar-like" coordinates as shown in figure 1, that were introduced in [1]. The angles describe the following series of transformations to get to the kite position and orientation. Starting with kite pointing in the $\vec{e}_z$ direction and the taught tether placed along the $\vec{e}_x$ direction, apply a rotation of $-\psi$ along the $\vec{e}_x$ axis. Then apply a rotation of $\vartheta$ about the $\vec{e}_y$ axis. Then apply a rotation $\varphi$ about the $\vec{e}_x$ axis. This series of transformation defines the Kite fixed frame basis vectors with respect to the Earth fixed frame, as given by equation (2). Further the motion of the kite can be given in terms of the rate of change of these angles and the tether length given by the dynamics of the kite as given by (1). The tether is always assumed to be taught in this description.

### 2.2 Kite dynamics

$$\vec{J} = v_r\vec{e}_r + v_p\vec{e}_p + v_k\vec{e}_k \tag{1a}$$

$$\dot{\vartheta} = -\frac{\vec{J}\cdot\vec{e}_z}{L\cos\vartheta} \tag{1b}$$

$$\dot{\varphi} = \frac{\vec{J}\cdot\vec{e}_y}{L\sin\vartheta\cos\varphi} \tag{1c}$$

$$\dot{\psi} = g v_a u_1 + \dot{\varphi}\cos\vartheta \tag{1d}$$

$$\dot{L} = u_2 \tag{1e}$$

where

$$\vec{e}_r = \begin{pmatrix} -\sin\vartheta\cos\psi \\ -\cos\varphi\sin\psi + \sin\varphi\cos\vartheta\cos\psi \\ -\sin\varphi\sin\psi - \cos\varphi\cos\vartheta\cos\psi \end{pmatrix} \tag{2a}$$

$$\vec{e}_p = \begin{pmatrix} \sin\vartheta\sin\psi \\ -\cos\varphi\cos\psi - \sin\varphi\cos\vartheta\sin\psi \\ -\sin\varphi\cos\psi + \cos\varphi\cos\vartheta\sin\psi \end{pmatrix} \tag{2b}$$

$$\vec{e}_k = \begin{pmatrix} -\cos\vartheta \\ -\sin\varphi\sin\theta \\ \cos\varphi\sin\vartheta \end{pmatrix} \tag{2c}$$

$\vec{e}_r, \vec{e}_p, \vec{e}_k$ are the basis vectors of the Kite fixed frame (written with respect to the Earth fixed frame in (2)) along the roll, pitch and yaw axis of the kite respectively.
$v_r$,$v_p$,$v_k$ are components of the kite's velocity along the $\vec{e}_r, \vec{e}_p, \vec{e}_k$ axes respectively, with

$$v_k = -\dot{L} = -u_2 \tag{3a}$$
$$v_r = v_w^T e_r - E\left(v_w^T e_k - v_k\right) = v_w^T e_r - E v_w^T e_k + E u_2 \tag{3b}$$
$$v_p = v_w^T e_p \tag{3c}$$

$v_a$ is the relative wind speed as faced by the moving kite , given by

$$\begin{aligned} v_a &= -(v_w^T e_r - v_r) \\ &= -E v_w^T e_k + E u_2 \end{aligned} \tag{4a}$$

$\vec{v}_w$ is the free stream wind velocity vector in the $e_x - e_y - e_z$ frame of reference.

And $u_1$,$u_2$ are the control inputs for the main line reel out rate and the steering deflection respectively.
Thus we have a nonlinear system given by,

$$\dot{\vartheta} = -\frac{v_w^T\left((e_r - E e_k) e_{r_z} + e_p e_{p_z}\right)}{L \cos\vartheta} + \begin{pmatrix} 0 & \frac{e_{k_z} - E e_{r_z}}{L\cos\vartheta} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{5a}$$
$$\dot{\varphi} = \frac{v_w^T\left((e_r - E e_k) e_{r_y} + e_p e_{p_y}\right)}{L \sin\vartheta \cos\varphi} + \begin{pmatrix} 0 & \frac{E e_{r_y} - e_{k_y}}{L\sin\vartheta\cos\varphi} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{5b}$$
$$\dot{\psi} = \frac{v_w^T\left((e_r - E e_k) e_{r_y} + e_p e_{p_y}\right)}{L \tan\vartheta \cos\varphi} + \begin{pmatrix} -gE(v_w^T e_k) & \frac{E e_{r_y} - e_{k_y}}{L\tan\vartheta\cos\varphi} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + gEu_1u_2 \tag{5c}$$

Rewriting the above equations in matrix form,

$$\begin{pmatrix} \dot{\vartheta} \\ \dot{\varphi} \end{pmatrix} = A(\vartheta, \varphi, \psi, L) v_w + B(\vartheta, \varphi, \psi, L) \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{6a}$$
$$\dot{\psi} = A_1(\vartheta, \varphi, \psi, L) v_w + B_1(\vartheta, \varphi, \psi, L) \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + gEu_1u_2 \tag{6b}$$
$$\dot{L} = u_2 \tag{6c}$$

Further in order to impose rate limit constraints on the inputs we will treat $u_1, u_2$ as an extended state of the system and instead use $\dot{u}_1, \dot{u}_2$ as inputs for the system.

Thus we finally write

$$x = [\vartheta, \varphi, \psi, L]^T, \quad u = [u_1, u_2]^T \tag{7}$$

Control inputs as

$$c = [c_1, c_2]^T \tag{8}$$

And the state dynamics as

$$\dot{x} = f(x) v_w + g_1(x) u_1 + g_2(x) u_2 + gEu_1u_2 \tag{9a}$$
$$\dot{u} = c \tag{9b}$$

## 2.3 Tether forces

Under the assumptions of constant angle of attack, mass less kite, zero tether drag and zero tether sagging, the tether tension can be written as,

$$T = k(\vec{v}_w \cdot \vec{e}_k + u_2)^2 \tag{10}$$

The above equation however is valid only when $\vec{v}_w \cdot \vec{e}_k < 0$ (which implies that the kite must be flying in the wind window) and $-u_2 \gg \vec{v}_w \cdot \vec{e}_k$ (which requires that the reel out must be slow enough so as not to cause tether sagging).

The first inequality is trivially satisfied in the operational conditions in the wind window however the second condition requires us to appropriately bound the reel out rate or the tether tension via a constraint in the problem.

### 2.4 Power generation

The power generated ($P$) is assumed to be a simple non-dynamical process with losses, expressed through a constant efficiency term.

Thus,

$$P = \eta \, T \, u_2 \tag{11}$$

Under the conditions required by (10), we can rewrite the power as,

$$P = \eta' (\vec{v}_w \cdot \vec{e}_k(\vartheta, \varphi, \psi) + u_2)^2 u_2 \tag{12}$$

As expected, power is either generated or consumed depending on the sign of $u_2$, while the dependence on the wind speed, direction and the kites position in the wind window is captured by the $\vec{v}_w \cdot \vec{e}_k$ term.

If the wind is purely in the x-direction of the $e_x - e_y - e_z$ frame with a magnitude $v_o$, then the power becomes

$$\begin{aligned} P &= \eta' (-v_o \cos\vartheta + u_2)^2 u_2 \\ &= \eta' v_o^2 \cos^2\vartheta \, u_2 - 2\eta' v_o \cos\vartheta u_2^2 + \eta' u_2^3 \end{aligned} \tag{13}$$

## 3 Optimal Control Problem Formulation

### 3.1 Problem statement

The objective of our Optimal Control Problem (OCP) is to maximize the average power generated over an infinite time horizon using a finite tether length. Intuition suggests that this would be done by a periodic solution with a reel-out and reel-in cycle forming 1 period of the system.

Numerically, however, we can only solve a finite time horizon OCP, which requires us to reformulate the problem into an equivalent finite time horizon problem. There are multiple ways of doing this and we compare amongst three reformulations for this problem.

The first reformulation is a finite time horizon approximation to the original problem. We solve the problem over a long but finite time horizon and accept this as an approximation to the infinite horizon solution. This formulation is presented in (14). This allows us to discover the optimal limit cycle to a degree of approximation however it requires us to solve the problem over very long time horizons and still yields only an approximate solution.

$$\begin{aligned}
&\max_{u(\cdot)} && \frac{1}{t_1}\int_0^{t_1} \eta' (\vec{v}_w \cdot \vec{e}_k(x) + u_2)^2 u_2 \, dt && && \text{(14a)} \\
&\text{subject to} && (9) && \text{[System dynamics]} && \\
& && x \in [x_{min}, x_{max}],\ u \in [u_{min}, u_{max}] && \text{[State bounds]} && \text{(14b)} \\
& && c \in [c_{min}, c_{max}] && \text{[Input } (u_1, u_2 \text{ rate) bounds]} && \text{(14c)} \\
& && T = k(v_w^T e_k + u_2)^2 \in [T_{min}, T_{max}] && \text{[Bounds on tether force]} && \text{(14d)} \\
& && h = L\sin(\vartheta)\cos(\varphi) \geq h_{min} && \text{[Altitude constraint]} && \text{(14e)} \\
& && x(0) = x_0,\ t_1 = fixed && \text{[Terminal constraints]} && \text{(14f)}
\end{aligned}$$

The second reformulation (15) is to solve over one time period by enforcing periodicity constraints for the limit cycle. We however do not know the time period or the initial condition for the limit cycle, which makes this problem difficult. However we can solve the problem for free initial $x$, free end time $t_2$ and introduce explicit phases for reel-in (phase 1) and reel-out (phase 2) to restrict the solution to one period of the limit cycle with a free switching time $t_1$. This reduces the time horizon and the computation time significantly but requires physical insight into the problem to hypothesize that the optimal solution would be such a limit cycle. We also relax the exact periodicity constraint slightly by requiring the final state to be in a small ball neighbourhood of the initial state to reduce the computational time.

$$\begin{aligned}
\max_{x(0),u(\cdot),t_1,t_2} \quad & \frac{1}{t_2}\int_0^{t_2} \eta'(\vec{v}_w \cdot \vec{e}_k(x) + u_2)^2 u_2 \, dt && (15a)\\
\text{subject to} \quad & (9) && \text{[System dynamics]}\\
& x \in [x_{min}, x_{max}] && \text{[Bounds on } x\text{]} \quad (15b)\\
& u \in [u_{min}^{(1)}, 0] \forall t \in [0, t_1] && \text{[Bounds on } u \text{ in phase 1]} \quad (15c)\\
& u \in [0, u_{max}^{(2)}] \forall t \in (t_1, t_2] && \text{[Bounds on } u \text{ in phase 2]} \quad (15d)\\
& c \in [c_{min}, c_{max}] && \text{[Input } (u_1, u_2 \text{ rate) bounds]} \quad (15e)\\
& T = k(v_w^T e_k + u_2)^2 \in [T_{min}, T_{max}] && \text{[Bounds on tether force]} \quad (15f)\\
& h = L\sin(\vartheta)\cos(\varphi) \geq h_{min} && \text{[Altitude constraint]} \quad (15g)\\
& x(t_2) \in \mathcal{B}(x(0), r_x), u(t_2) \in \mathcal{B}(u(0), r_u) && \text{[Periodicity constraint]} \quad (15h)
\end{aligned}$$

$\mathcal{B}(\vec{x}, \vec{r})$ is an ellipsoid centered at position $\vec{x}$ with principle axes lengths given by elements of $\vec{r}$. Also here we have chosen norm-1 to measure the lengths for $\mathcal{B}$.

The final method (16) is to transform the time scale, mapping a finite half open interval to the infinite horizon and rescale the dynamics in this time frame. This provides another method to approximate the infinite horizon problem as a finite horizon problem however this approximation allows for discretization of the complete horizon with collocation points going arbitrarily close to infinity. This however requires a large number of collocation points to reduce the state approximation error towards the later part of the horizon and works only if the states and inputs are exponentially converging to a steady state. This is however not the case for our system and thus is not applicable to our system. With this method we saw large state approximation errors from the collocation and the method did not converge to any solution within acceptable tolerance levels. Also we cannot put static rate bounds on $u_1, u_2$ due to the non stationary time transformation in this formulation.

$$\begin{aligned}
& \max_{u(\cdot)} \quad \frac{1}{2}\int_{-1}^{1} \left(\frac{d\phi}{d\tau}\right) \eta'(\vec{v}_w \cdot \vec{e}_k(x) + u_2)^2 u_2 \, d\tau && (16a)\\
& \text{subject to} \\
& \frac{dx}{d\tau} = \left(\frac{d\phi}{d\tau}\right)(f(x)v_w + g_1(x)u_1 + g_2(x)u_2 + gEu_1u_2), && (16b)\\
& \frac{du}{d\tau} = \left(\frac{d\phi}{d\tau}\right) c, && \text{[Transformed (9))]} \quad (16c)\\
& x \in [x_{min}, x_{max}], \quad u \in [u_{min}, u_{max}] && \text{[State bounds]} \quad (16d)\\
& T = k(v_w^T e_k + u_2)^2 \in [T_{min}, T_{max}] && \text{[tether force bound]} \quad (16e)\\
& h = L\sin(\vartheta)\cos(\varphi) \geq h_{min} && \text{[Alt. constraint]} \quad (16f)\\
& x(-1) = x_0 && \text{[Initial condition]} \quad (16g)\\
& t = \phi(\tau), && \phi(\tau) : [-1, 1) \to [0, \infty) \quad (16h)
\end{aligned}$$

We use the transform,

$$t = \phi(\tau) = \frac{1+\tau}{1-\tau} \tag{17}$$

## 3.2 Solution method

The results presented here are using a collocation method for solving the optimal control problem using the GPOPS-II software ([2]). The time horizon is discretized at specific intervals with collocation points. The states and inputs are then interpolated for values between these collocation points. Thus the first order necessary conditions of optimality is reduced to a NLP in finite dimensional search space along with algebraic constraints for the original differential equation constraints.

## 3.3 Results

### 3.3.1 Finite time approximation

| Horizon length | Objective | Collocation points | Maximum mesh error | Solution time (minutes) |
|---|---|---|---|---|
| 300s | 1621.1 | 1424 | 0.0033 | 16.3 |
| 500s | 1473.7 | 2442 | 0.0059 | 16.2 |
| 700s | 1392.8 | 3419 | 0.0109 | 28.5 |
| 1000s | 1400.9 | 4559 | 0.0130 | 49.1 |
| 1500s | 1330.8 | 6110 | 0.0086 | 52.9 |
| 1700s | 1344.2 | 6903 | 0.0234 | 102.68 |

Table 1: Solution metrics after 11 mesh refinement iterations and NLP tolerance of 10e-7

The finite time approximation formulation provides a periodic limit cycle solution without explicitly enforcing periodicity. However due to the finite time horizon and fixed intial condition the solutions have a phase difference (as seen from figure 4). Also due to nonlinearity the solution diverges on some cycles away from the optimal limit cycle. Also since the formulation does not use the knowledge of periodicity in the system and it suffers from extremely long computational time.

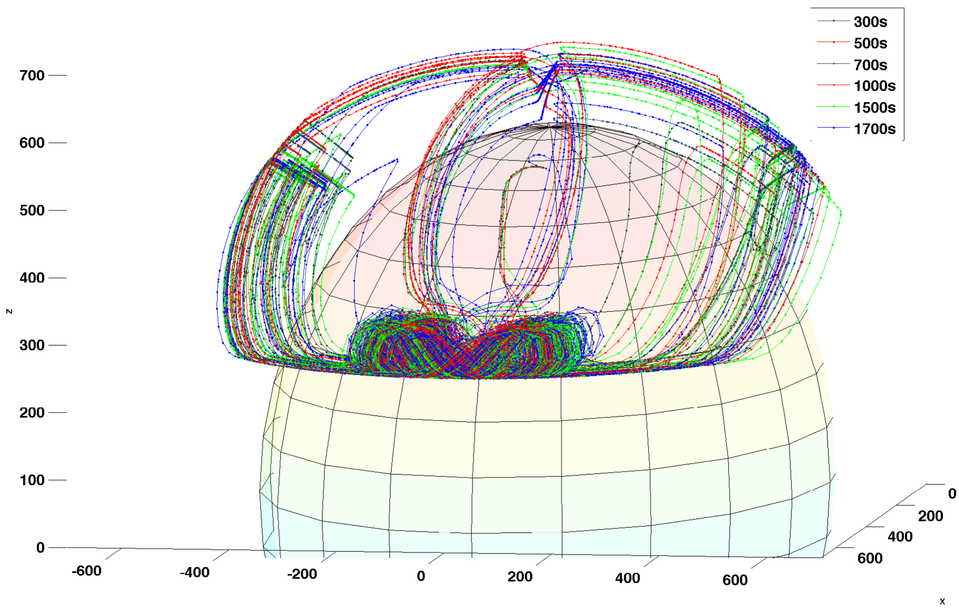



Figure 2: Superimposed optimal trajectories for varying finite time approximations

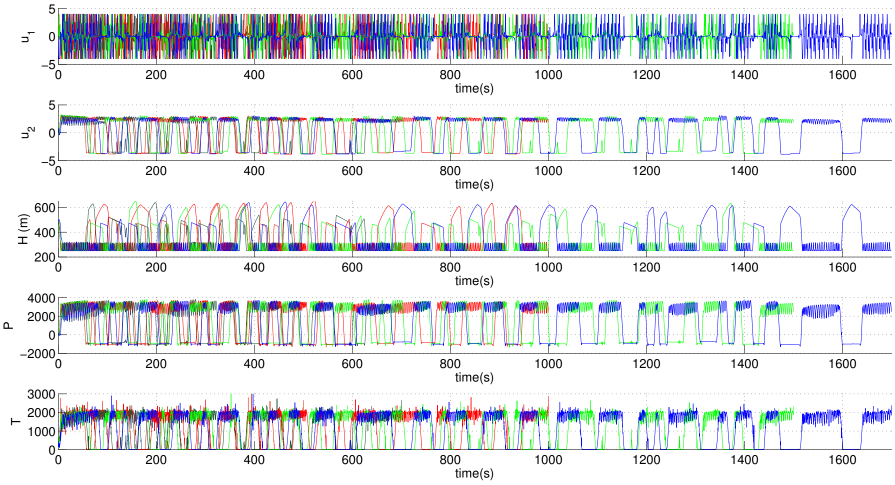



Figure 3: Control, Height, Power & Tether force

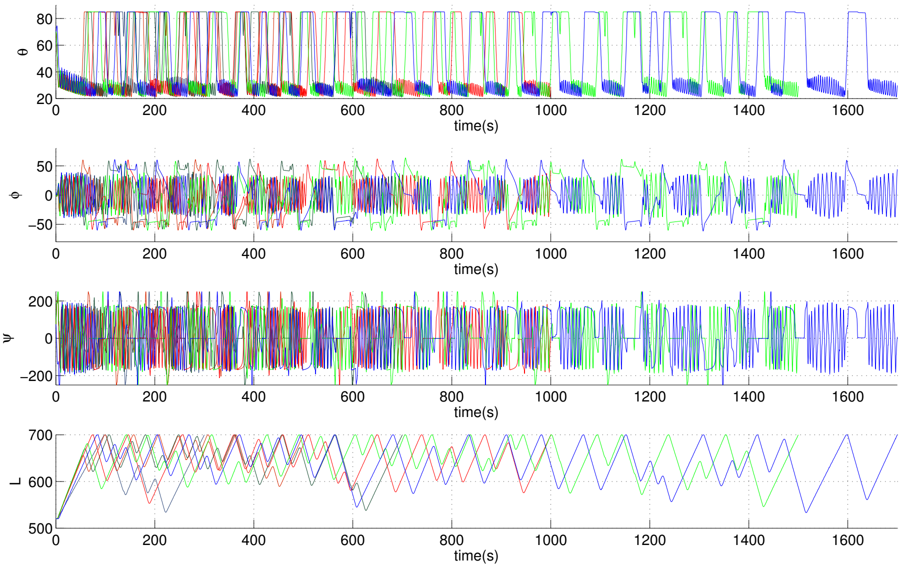



Figure 4: States $(\vartheta, \varphi, \psi, L)$ as a function of time

### 3.3.2 Periodic solution

| Time period | Objective | Collocation points (per phase) | Maximum mesh error | Solution time (seconds) | Duty cycle | Initial state $(x, u)$ |
|---|---|---|---|---|---|---|
| 138.0s | 1247.5 | 395, 146 | 2.6e-3 | 157.22 | 0.63 | (60,10,0,520,0,-1) |
| 130.4s | 1278.3 | 100, 392 | 4e-3 | 135.6 | 0.61 | (70,10,0,700,0,-1) |
| 87.81s | 1307.1 | 76, 292 | 9.5e-4 | 192.1 | 0.62 | (84,55,0,7000,-1) |

Table 2: Solutions with fixed initial conditions and periodic constraint with $r_x = [1, 2, 10, 0]$, $r_u = [\infty, \infty]$

| Time period | Objective | Collocation points (per phase) | Maximum mesh error | Solution time (seconds) | Duty cycle | Initial state $x$,$u$ | $r_x$,$r_u$ |
|---|---|---|---|---|---|---|---|
| 96.19s | 1328.9 | 172, 319 | 7e-3 | 209.5 | 0.56 | free,(0,-1) | $1^*$ |
| 93.45s | 1352.0 | 162, 301 | 1.7e-3 | 3049.3 | 0.59 | free,free | $2^*$ |

Table 3: Periodic solutions with free initial $x$ and varying tolerance of periodicity constraint: $1^* : r_x = [1, 2, 10, 0]$, $r_u = [\infty, \infty]$, $2^* : r_x = [0, 0, 0, 0]$, $r_u = [\infty, \infty]$

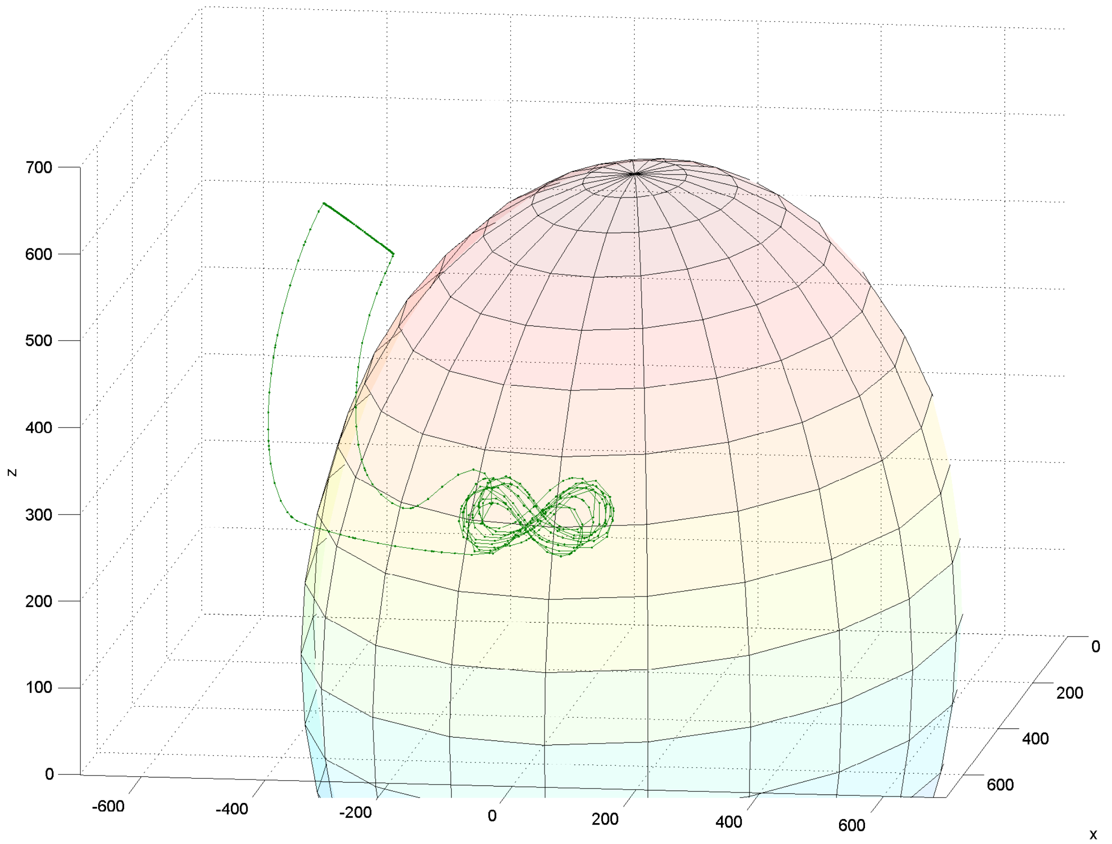



Figure 5: Periodic optimal trajectory for free initial $x, u$ and tolerance $2^*$

As we see from tables 2 and 3, while it is possible to solve the formulation with exact periodicity constraint in $x, u$ and free initial conditions, it also takes a large time. Providing some reasonable initial conditions and relaxing the periodicity constraint tolerance $r_x, r_u$ significantly reduces the computation time as it makes solving the NLP easier.

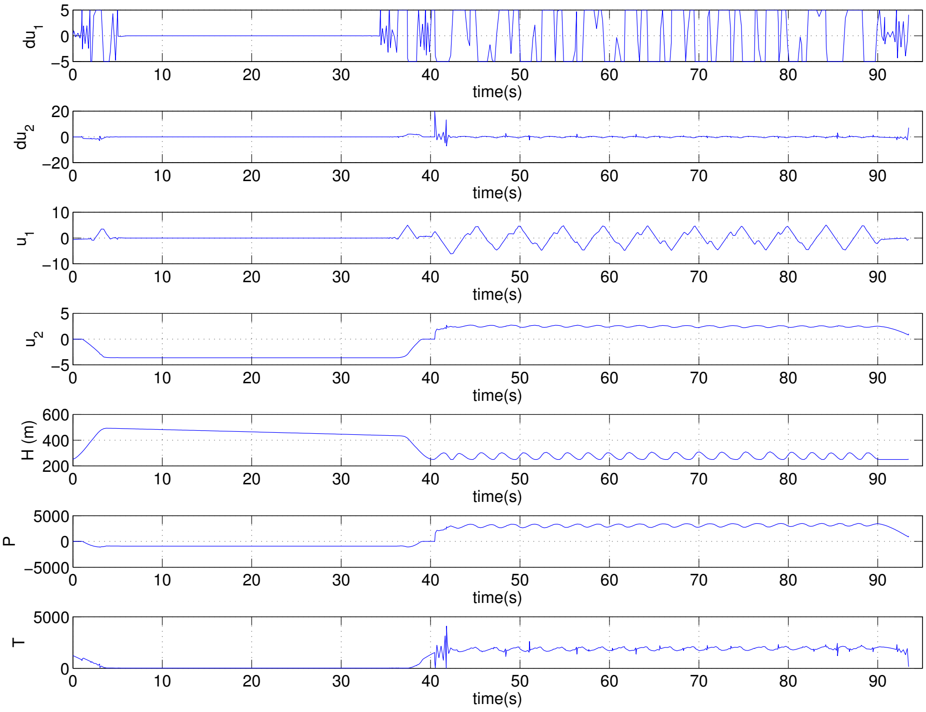



Figure 6: Control, Height, Power & Tether force (free initial $x, u$ and tolerance $2^*$)

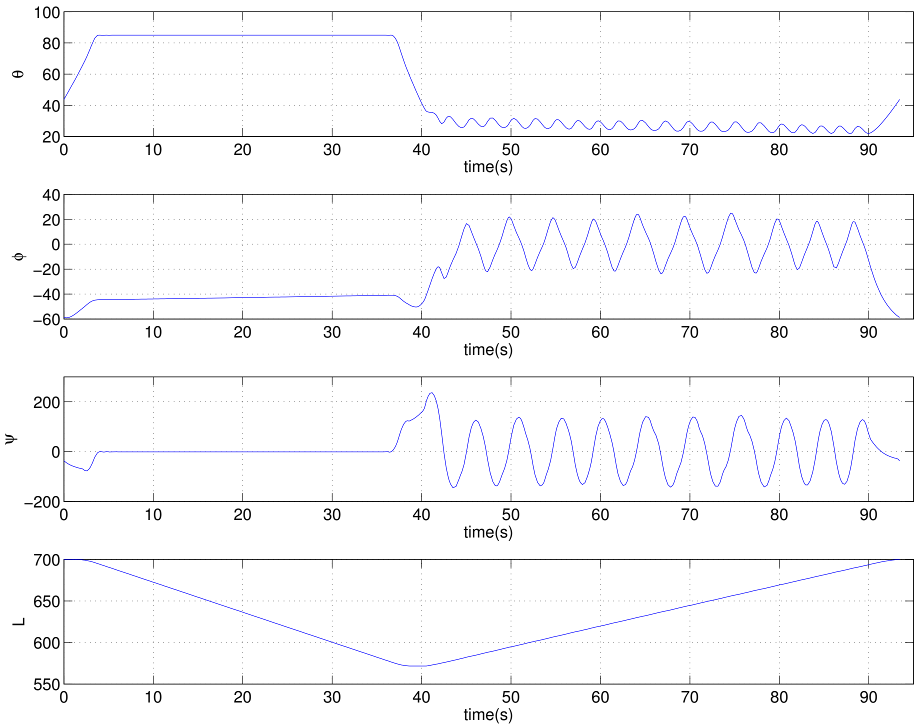



Figure 7: States $(\vartheta, \varphi, \psi, L)$ as a function of time (free initial $x, u$ and tolerance $2^*$)

# References

[1] M. Erhard and H. Strauch. Control of towing kites for seagoing vessels. 21:1629–1640, November 2012.

[2] Michael A Patterson and Anil V Rao. GPOPS- II: A matlab software for solving multiple-phase optimal control problems using hp–adaptive gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software*, 39(3), 2013.