***

Adam Seewald[1], Hector Garcia de Marina, and Ulrik Pagh Schultz

***Abstract*— abstract**
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract
abstract

I. Introduction

*

II. State of the Art

*

III. Model

To evaluate the energy consumption of different flight phases, we use a third-order Fourier series, as the energy evolution in a flying scenario often presents periodic patterns [1]. More specifically, in a survey application, the fixed-wing drone performs ellipsoidal flying trajectories at a fixed height while steering over one of the unconstricted axes. The trajectory is periodic in the sense that the system uncertainty factors, which might impact the predicted energy, such as wind conditions, temperature, and other atmospheric interferences, are alike to those observed performing a similar trajectory under the same conditions.

The third-order Fourier series represents the power in function of time and can be expressed in the following form:

$$f(t) = \sum_{n=0}^{3} a_n \cos\frac{nt}{\xi} + b_n \sin\frac{nt}{\xi}, \tag{1}$$

with ξ the characteristic time, a_n, b_n for $n \in \{0, \dots, 3\}$ the Fourier series coefficient, and:

$$\xi, a_n, b_n, t, f(t) \in \mathbb{R}. \tag{2}$$

[1]Corresponding author; email: ads@mmmi.sdu.dk
Adam Seewald, Ulrik Pagh Schultz are with the SDU UAS Center, Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Odense, Denmark.
Hector Garcia de Marina is with the Faculty of Physics, Department of Computer Architecture and Automatic Control, Universidad Computense de Madrid, Spain.

A. Problem Definition

For the purpose of optimal analysis, the non-linear model expressed by Equation (1) can be expressed using an equivalent linear time-varying state-space model, expressed in the following form:

$$\begin{aligned}\dot{\mathbf{q}}(t) &= A\mathbf{q}(t) + B\mathbf{u}(t) + w(t),\\ y(t) &= C\mathbf{q}(t) + v(t),\end{aligned} \tag{3}$$

where y is the energy evolution of the controlled system, w the process noise (i.e., atmospheric interferences), v the measurement noise, and the control $\mathbf{u}$ along with the input matrix B are defined subsequently. Moreover:

$$\begin{gathered}\mathbf{q}(t) = \begin{bmatrix} a_0(t) & b_0(t) & \cdots & a_3(t) & b_3(t) & | & \xi \end{bmatrix}^T,\\ A_i = \begin{bmatrix} 0 & i \\ -i^2 & 0 \end{bmatrix}, A = \left[\begin{array}{cccc|c} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & A_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & A_3 & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 0 \end{array}\right],\\ C = \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 & | & 0 \end{bmatrix},\end{gathered} \tag{4}$$

with the state $\mathbf{q}$ being the evolution of the Fourier series coefficient in time, A, C the state transition matrix, and the output matrix respectively. Furthermore, the last row and column of the matrix A contain zero row vectors and columns vectors respectively, and:

$$\begin{aligned} A &\in \mathbb{R}^{9\times 9},\\ \mathbf{q}(t), C &\in \mathbb{R}^9.\end{aligned} \tag{5}$$

Motivated by the fact that the system of interest is sampled at discrete times and for the sake of simplicity, we consider the discretized version of the Equation (3) by employing the following expression:

$$\begin{aligned}\mathbf{q}_{k+1} &= A\mathbf{q}_k + B\mathbf{u}_k + w_k,\\ y_k &= C\mathbf{q}_k + v_k.\end{aligned} \tag{6}$$

As the system was observed to behave stochastically, with the process and measurement noise evolving in a normal distribution, a Kalman filter [2], [3] is employed to predict the state $\hat{\mathbf{q}}$. Such a state, along with the predicted output $\hat{y}$, differs from the model state $\mathbf{q}$ and measured output y in Equation (6) due to the presence of uncertainty.

The prediction is done using the following expression:

$$\hat{\mathbf{q}}_{k+1}^- = A\hat{\mathbf{q}}_k + B\mathbf{u}_k, \tag{7a}$$

$$P_{k+1}^- = AP_kA^T + Q, \tag{7b}$$

where $\hat{\mathbf{q}}_k^-, \hat{\mathbf{q}}_k$ depicts the estimate of the state before and after measurement (or simply estimate), and P_k the error covariance matrix (i.e., the variance of the estimate before measurement).

The estimation of the state and the update of the predicted output is done using the following expression:

$$K_k = (CP_{k+1}^{-}C^T + R)^{-1}(P_{k+1}^{-}C^T), \tag{8a}$$

$$\hat{\mathbf{q}}_{k+1} = \hat{\mathbf{q}}_{k+1}^{-} + K_k(y_k - C\hat{\mathbf{q}}_{k+1}^{-}), \tag{8b}$$

$$P_{k+1} = (I - G_{k+1}C)P_{k+1}^{-}, \tag{8c}$$

$$\hat{y}_k = C\hat{\mathbf{q}}_{k+1}, \tag{8d}$$

where K_k is the gain of the Kalman filter, and I the identity matrix. The noise covariance matrixes Q, R indicates the process noise and sensor noise covariance respectively, and:

$$\begin{gathered} k \in \mathbb{Z}^{\geq 0}, \\ Q, P_k^{-}, P_k \in \mathbb{R}^{9\times 9}, \\ K_k, \hat{\mathbf{q}}_k^{-}, \hat{\mathbf{q}}_k \in \mathbb{R}^9, \\ w_k, v_k, R, \hat{y}_k \in \mathbb{R}. \end{gathered} \tag{9}$$

Equations (7–9) converges to the predicted energy evolution as follows. An initial guess of the values $\hat{\mathbf{q}}_0, P_0$ is derived empirically from collected data. It is worth considering that an appropriate guess of these parameters allows the system to converge to the desired energy evolution in a shorter amount of time. The tuning parameters Q, R are also derived from the collected data and may differ due to i.e., different sensors used to measure the instantaneous energy consumption, or different atmospheric conditions accounting for the process noise.

At time $k = 0$, the initial estimate before measurement of the state and of the error covariance matrix is updated in Equation (7a) and (7b) respectively. The value of $\hat{\mathbf{q}}_1$ is then used in Equation (8b) to estimate the current state along with the measurment from the sensor y_0, where the sensor noise covariance matrix R accounts for the amount of uncertainity in the measurement. The estimated ouput $\hat{y}_0$ is then obtained from Equation (8d). The algorithm is iterative. At time $k = 1$ the values $\hat{\mathbf{q}} = 1, P_1$ computed at previous step are used to estimate the values $\hat{\mathbf{q}}_2, P_2$, and y_1.

Two different components of the overall energy consumption are being modeled in our analysis, an approach that has been extensively reviewed in our previous work [1], [4]–[6]. A mechanical energy model accounts for the energy by reason of the physical system being moved in space, whereas a computational energy model accounts for the computations. The main goal is to derive a control action $\mathbf{u}$ for the current state $\hat{\mathbf{q}}$ from the optimal control law $\kappa(\hat{\mathbf{q}})$. This is achieved by solving online a finite horizon optimal control problem by the hand of a model predictive control (MPC) derived later in this chapter.

—

B. Equations of Motion

For ease of reference, given a point in a 2D Euclidean coordinate system $\mathbf{p} = \begin{bmatrix} x & y \end{bmatrix}^T$, with respect to some inertial frame $\mathcal{O}_N$, we consider a simple elliptical explicit trajectory expression φ defined by the equation:

$$\varphi(\mathbf{p}) = \frac{x - x_e}{r_m^2} + \frac{y - y_e}{r_M^2} - 1, \tag{10}$$

where x_e, y_e are the coordinates of the center of the ellipse with respect to $\mathcal{O}_N$, and r_m, r_M are the minor and major radius respectively. The equation of motion is expressed using a non-holonomic model presented in [7]:

$$\begin{aligned} \dot{\mathbf{p}}(t) &= sm\left(\psi(t)\right) + h, \\ \dot{\psi}(t) &= u_m(t), \end{aligned} \tag{11}$$

where s depicts the airspeed, $m(\psi(t)) = \begin{bmatrix} \cos\psi & \sin\psi \end{bmatrix}^T$ the attitude yaw angle, h a constant which represents the wind, and:

$$\begin{gathered} \psi(t) \in (-\pi, \pi], \\ r_m, r_M, x_e, y_e, \mathbf{p}(t), m(\psi(t)) \in \mathbb{R}^2, \\ u_m(t), h \in \mathbb{R}. \end{gathered} \tag{12}$$

Given a desired trajectory expressed in (10), an arbitrary point in space $\mathbf{p}$, and it's ground velocity $\dot{\mathbf{p}}$, considering the equation of motion in (11), the control action is likewise derived in [7]:

$$\begin{aligned} u_m(t) &= u(\mathbf{p}, \dot{\mathbf{p}}, \psi) \\ &= \frac{\|\dot{\mathbf{p}}\|}{s\cos\beta}\left(\dot{\chi}_d(\dot{\mathbf{p}}, \mathbf{p}) + k_d \hat{\dot{\mathbf{p}}}^T R \hat{\dot{\mathbf{p}}}\right), \end{aligned} \tag{13}$$

where $\hat{\dot{\mathbf{p}}} = \dot{\mathbf{p}}/\|\dot{\mathbf{p}}\|$,

C. Computational Energy Model

The energy due to the computational units of the system is obtained with `powprofiler`, an open-source modeling tool that measures empirically a number of software configuration and build an energy model accordingly [4]. A multivariate linear interpolation is derived automatically by `powprofiler`, while being accessed online at the hand of a lookup table in the optimal control algorithm presented later. The robotics system in such an analysis composes a number of computationally expensive ROS nodes, allowing to vary the amount of computations changing node-specific quality of service (QoS) values. For the sake of simplicity and to allow an easy integration in an existing ROS system, the QoS values are utilizing ROS middleware, meaning that defined ROS parameters are intended specifically as QoS ranges.

Let us define $\mathrm{QoS}_i \in \mathbb{Z}^{\geq 0}, \forall i \in \{0, \ldots, r\}$ the i-th QoS range, $g_c(\mathrm{QoS}_i) \in \mathbb{R}^{\geq 0}$ the instantaneous energy value obtained interogating `powprofiler` online, and $\mathbf{u}_c$ the set of r QoS values the system is composed of. The computational energy component can be hence described using the following expression:

$$E_c(\mathbf{u}_c) = \sum_{i=0}^{r} g_c(\mathrm{QoS}_i), \tag{14}$$

TODO

IV. Evaluation

*

V. Conclusion and Future Work

*

Acknowledgment

This work is supported and partly funded by the European Union's Horizon2020 research and innovation program under grant agreement No. 779882 (TeamPlay).

References

[1] A. Seewald, H. Garcia de Marina, A. Hasan, H. S. Midtiby, and U. P. Schultz, "Towards mission-aware energy assessment for autonomous drones," 2020, p. submitted.

[2] R. F. Stengel, *Optimal control and estimation*. Courier Corporation, 1994.

[3] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[4] A. Seewald, U. P. Schultz, E. Ebeid, and H. S. Midtiby, "Coarse-grained computation-oriented energy modeling for heterogeneous parallel embedded systems," *International Journal of Parallel Programming*, pp. 1–22, 2019.

[5] A. Seewald, U. P. Schultz, J. Roeder, B. Rouxel, and C. Grelck, "Component-based computation-energy modeling for embedded systems," in *Proceedings Companion of the 2019 ACM SIGPLAN International Conference on Systems, Programming, Languages, and Applications: Software for Humanity*. ACM, 2019, pp. 5–6.

[6] A. Seewald, H. Garcia de Marina, H. S. Midtiby, and U. P. Schultz, "Mechanical and computational energy estimation of a fixed-wing drone," in *Proceedings of the 2020 Fourth IEEE International Conference on Robotic Computing (IRC)*. IEEE, 2020, p. to appear. [Online]. Available: https://adamseew.bitbucket.io/short/mechanical2020

[7] H. G. De Marina, Y. A. Kapitanyuk, M. Bronz, G. Hattenberger, and M. Cao, "Guidance algorithm for smooth trajectory tracking of a fixed wing uav flying in wind flows," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 5740–5745.