Attitude and position control of a quadcopter in a networked distributed system

Alejandro Alonso García, Amalie V. Petersen, Andrea Victoria Tram Løvemærke, Niels Skov Vestergaard, Noelia Villarmarzo Arruñada - Group 733

Email: [aalons16] [apet13] [alavem13] [nveste12] [nvilla16] @student.aau.dk

AALBORG UNIVERSITY

Introduction

This task is solved by implementing a linear control design. The system is split into an attitude and translational model. These are controlled individually by state space and classical controllers respectively. The prototype gets its attitude and position from a motion tracking system based on infrared cameras, keeping the control in a micro processor on the quadcopter. This layout constitutes a distributed system, where network issues, such as delays and packet losses, are taken into account.

Model

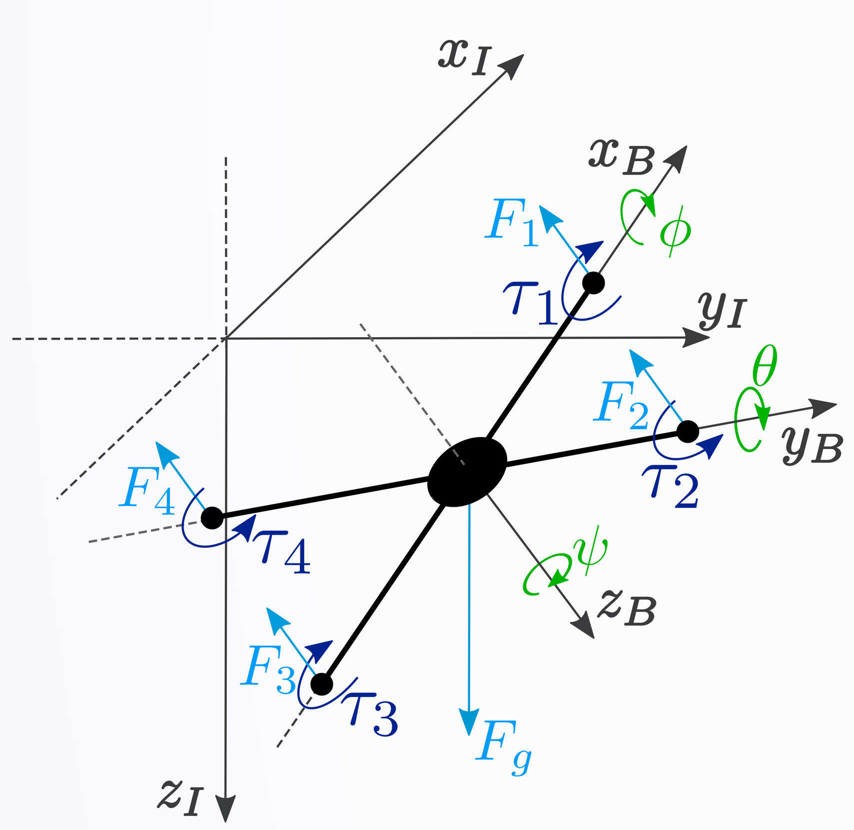

Figure: Free body diagram of the quadcopter along with the inertial and body coordinate systems.

- Attitude model equations

$$J_x\ddot{\phi} = k_{\text{th}}(\omega_4^2 - \omega_2^2)L \tag{1}$$
$$J_y\ddot{\theta} = k_{\text{th}}(\omega_1^2 - \omega_3^2)L \tag{2}$$
$$J_z\ddot{\psi} = k_{\text{d}}(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{3}$$

- Translational model equations

$$m\ddot{x}_{\text{I}} = -k_{\text{th}}({\omega_1}^2 + {\omega_2}^2 + {\omega_3}^2 + {\omega_4}^2) \times (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) \tag{4}$$
$$m\ddot{y}_{\text{I}} = -k_{\text{th}}({\omega_1}^2 + {\omega_2}^2 + {\omega_3}^2 + {\omega_4}^2) \times (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) \tag{5}$$
$$m\ddot{z}_{\text{I}} = F_g - k_{\text{th}}\,({\omega_1}^2 + {\omega_2}^2 + {\omega_3}^2 + {\omega_4}^2) \times \cos\phi\cos\theta \tag{6}$$

Control Solution

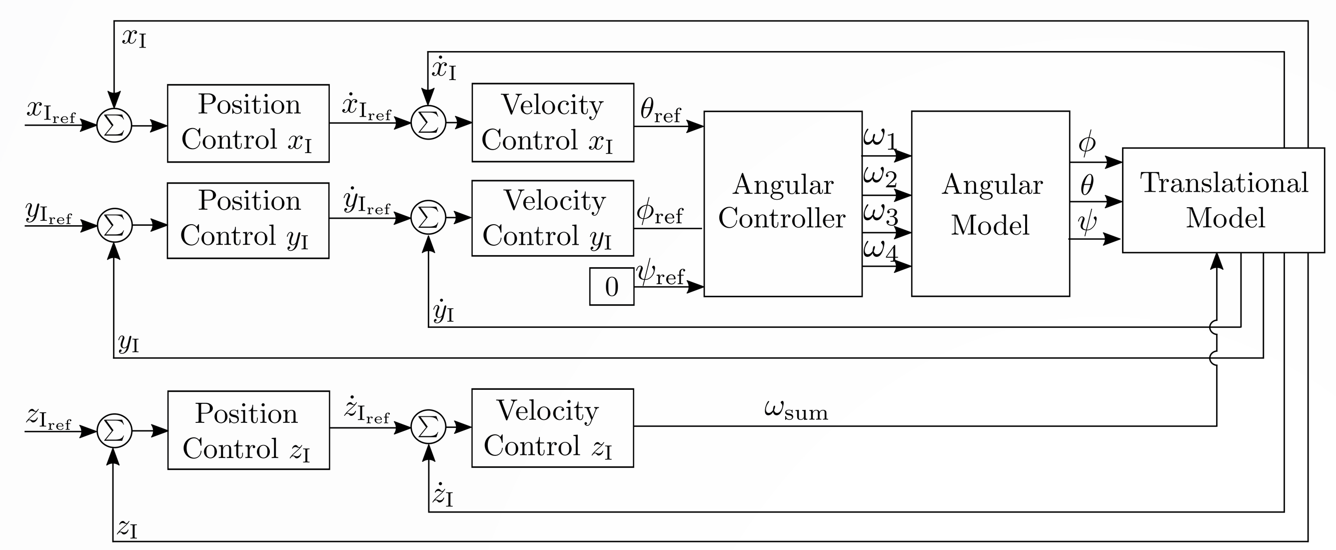

Figure: Figure caption

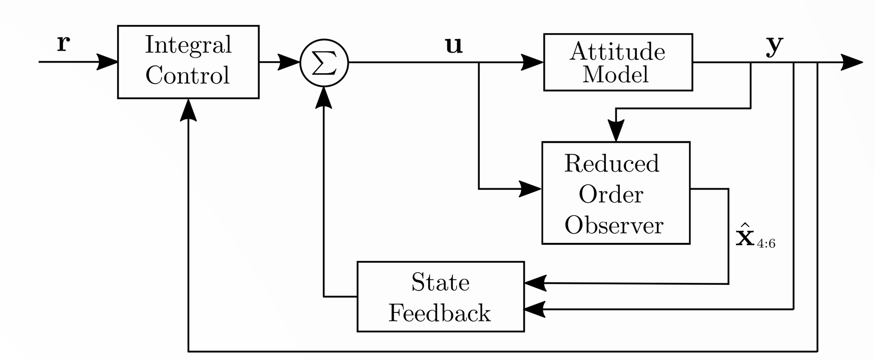

Figure: Figure caption

Results

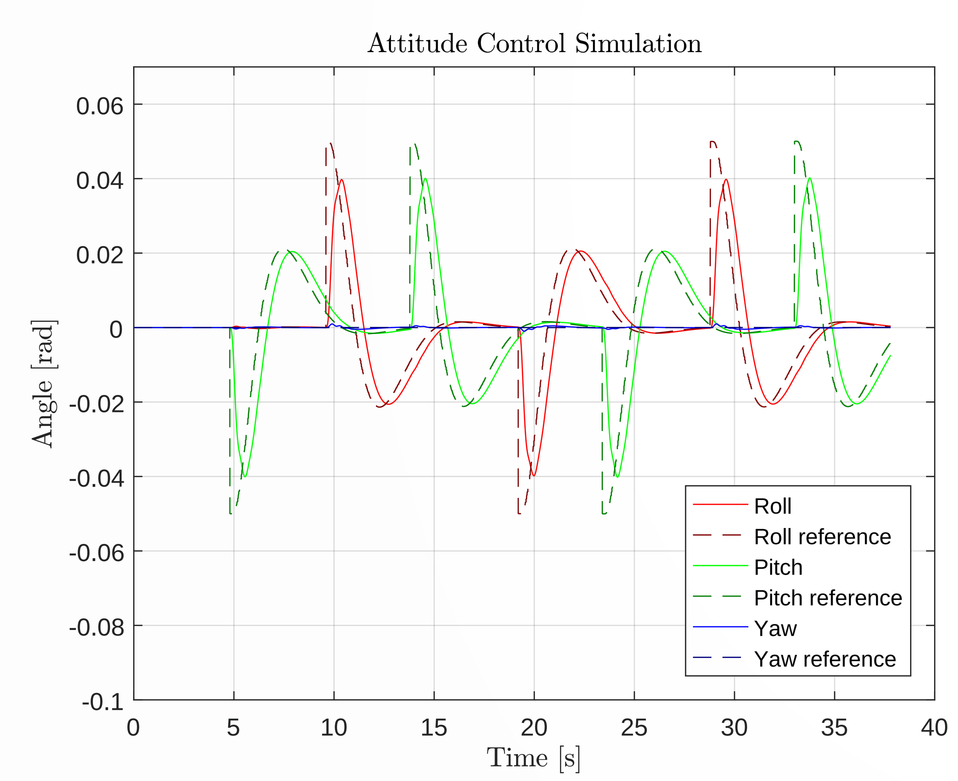

Figure: Figure caption

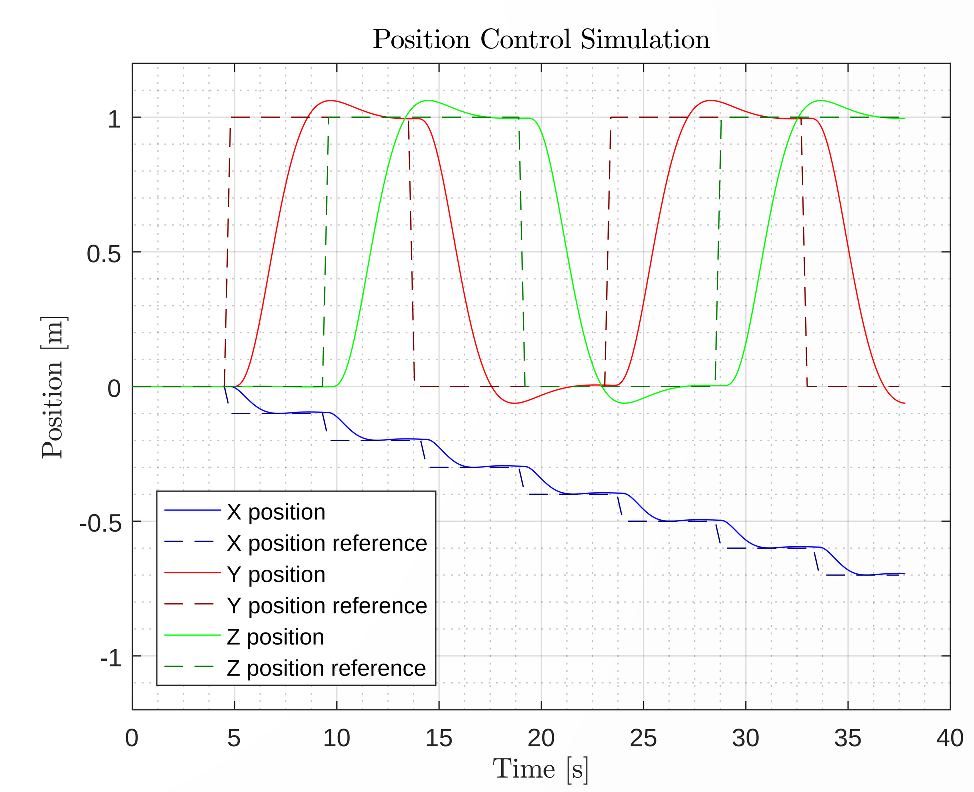

Figure: Figure caption

The attitude controller is defined by the chosen feedback and integral poles, $[-6.0, -6.2, -6.4, -6.6, -6.8, -7.0, -7.2, -7.4, -7.6]$, and the observer poles, $[-20, -25, -30]$. The translation velocity controllers for x and y are $C_{\dot{x}}(s) = -0.1$, $C_{\dot{y}}(s) = 0.1$ and the position ones are $C_x(s) = 0.5$, $C_y(s) = 0.5$. The PI-controller for the z translational velocity is $C_{\dot{z}}(s) = -201\frac{s+0.8}{s}$ and the outer loop P-controller is $C_z = 0.9$.

Discussion

It is seen that the controllers achieve the desired reference even though the network delay and the sampling rate affect the performance. The main network effect is the designed bandwidth of the controllers. This occurs due to the limited frequency in which the sensor data is obtained from the motion tracking system through the wireless connection.

Conclusion

The control system has been split into an attitude and a translational controller. The former has been designed using a state space approach, including state feedback with integral control and a reduced order observer. The translational control system has been designed with a classical control approach and result in three cascade loops, including proportional and PI controllers. The results obtained from the design show that both the attitude and the translational behavior of the quadcopter has been successfully controlled.

References

CONTROL BOOK.

Acknowledgements

Henrik Shøiler, Associated Professor, Aalborg University.
Christoffer Sloth, Associated Professor, Aalborg University.