Robust Obstacle Avoidance for Aerial Platforms using Adaptive Model Predictive Control

Gowtham Garimella[1], Matthew Sheckells[2] and Marin Kobilarov[1]

***Abstract*— This work addresses the problem of motion planning among obstacles for quadrotor platforms under external disturbances and with model uncertainty. A novel Nonlinear Model Predictive Control (NMPC) optimization technique is proposed which incorporates specified uncertainties into the planned trajectories. At the core of the procedure lies the propagation of model parameter uncertainty and initial state uncertainty as high-confidence ellipsoids in pose space. The quadrotor trajectories are then computed to avoid obstacles by a required safety margin, expressed as ellipsoid penetration while minimizing control effort and achieving a user-specified goal location. Combining this technique with online model identification results in robust obstacle avoidance behavior. Experiments in outdoor scenarios with virtual obstacles show that the quadrotor can avoid obstacles robustly, even under the influence of external disturbances.**

I. Introduction

Aerial robotic vehicles such as quadrotors are beginning to enable a range of useful capabilities. Current and future applications of quadrotors operating in natural environments include delivery of packages [1], inspection of a building infrastructure and power lines [2], aerial photography, and traffic surveillance [3]. With the increasing use of quadrotors, their safety and reliability are becoming essential. This paper focuses on designing control laws that can predict the vehicle motion for a short time horizon and ensure that it steers away from predicted collisions with approximately high probability.

Several prior works focused on generating reliable controllers for guaranteeing safety and stability of the quadrotor under external disturbances. For instance, the quadrotor dynamics has been estimated online using the autoregressive-moving-average with exogenous inputs (ARMAX) model [4], [5] and has been applied to model and control quadrotor platforms [6]. Model Reference Adaptive Control (MRAC) algorithms that estimate the system parameters while stabilizing the quadrotor under actuator uncertainty have been developed in [7], [8], [9]. These methods, however, do not account for modeling uncertainty and actuator bounds. Robust control techniques, on the other hand, can reject uncertainty in system dynamics [10] and account for actuator limits [11]. A robust backstepping controller for quadrotors which is globally asymptotically stable has been designed in [12]. Robust controllers have been used in practice to track an aggressive figure eight maneuver accurately [13].

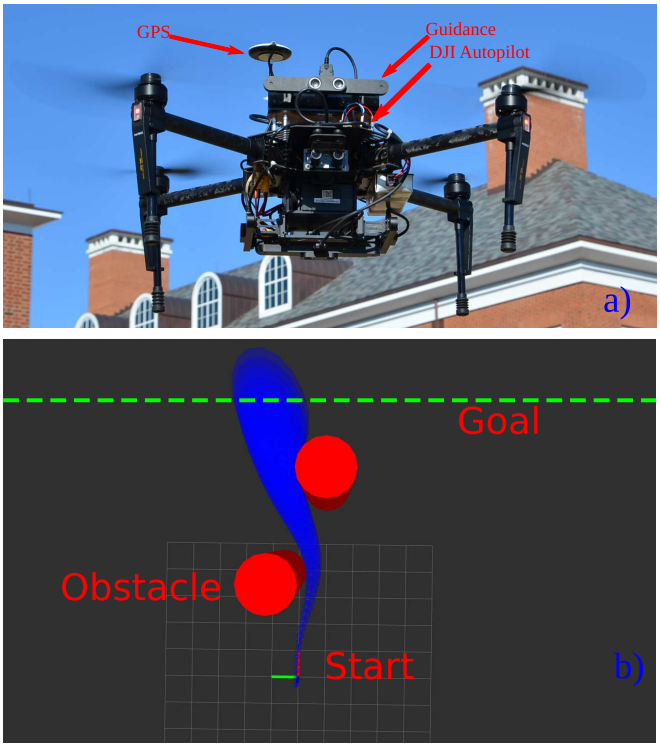

Fig. 1: a) DJI Matrice quadrotor with Guidance sensor suite and an additional short-range stereo used for experiments, b) Simulated safe obstacle avoidance trajectory for the quadrotor. The blue region corresponds to propagated uncertainty denoted by a 2σ standard deviation ellipsoids around the pose of the quadrotor, and red regions show two cylindrical obstacles.

Introducing obstacles into the quadrotor's environment further complicates the safe operation of quadrotors. Potential field methods have been used in [14] to generate an optimal obstacle free path that handles both static and dynamic obstacles. This method uses only the kinematics of the quadrotor model to generate the trajectory. On the contrary, a Linear Quadratic Regulator (LQR) based feedback scheme, combined with a direct collocation-based obstacle avoidance planner has been used to plan dynamically feasible knife-edge maneuvers for fixed wing aircraft in [15]. The feedback system generates a time-varying locally stable feedback control law using LQR optimization. Instead of offline generated feedback controllers, real-time kinodynamic planning using sampling-based motion planning of a quadrotor among

[1]Gowtham Garimella and Marin Kobilarov are with the Department of Mechanical Engineering, Johns Hopkins University, 3400 N Charles Str, Baltimore, MD 21218, USA ggarime1|marin@jhu.edu

[2]Matthew Sheckells is with the Department of Computer Science Johns Hopkins University, 3400 N Charles Str, Baltimore, MD 21218, USA mshecke1@jhu.edu

dynamic obstacles has been demonstrated in [16].

Obstacle avoidance in the presence of external disturbances is dealt with using recent extensions in Learning-Based Model Predictive Control (LBMPC) [17], [18]. This method allows for online learning of quadrotor dynamics and the generation of optimal trajectories that guarantee convergence. LBMPC has been used to learn ground effects on quadrotors and to predict the trajectory of balls for the purpose of catching them. LBMPC has also been used in controlling other systems with unknown dynamics such as a 3DOF robot arm [19] and the energy management of air conditioners in a building [20]. This method linearizes the dynamics and solves the MPC problem formulated as a quadratic program in an efficient onboard implementation.

Robust motion planning for fixed wing aircraft and quadrotors under the influence of parametric uncertainty and external disturbances has been studied recently by Majumdar et al. [21]. This work computes an offline library of feedback funnels to provide a reliable system that will remain inside the funnel when the feedback law is executed. The feedback funnels are sequenced together in real time to avoid obstacles reactively based on their positions. This method has the drawback that it does not account for online changes in model parameters and deviations in the uncertainty of external disturbances.

This work focuses on robust obstacle avoidance of quadrotor in an outdoor scenario with varying external disturbances. A nonlinear stochastic quadrotor model incorporating external disturbances is learned online. The learned model has been employed in a novel Nonlinear Model Predictive Control (NMPC) optimization framework that plans quadrotor trajectories to avoid obstacles by a required safety margin. The safety margin is computed based on propagating the model parameter uncertainty and initial state uncertainty as high-confidence ellipsoids in pose space and minimizing the ellipsoid penetration into the obstacles while minimizing control effort, and achieving a user-specified goal location. Figure 1 shows an optimal trajectory for a quadrotor avoiding obstacles. This method propagates the uncertainty due to initial state measurements, model parameters, and external disturbances to uncertainty in state trajectory. Sensor uncertainty is not accounted for in this method.

The proposed method is applied towards reactive avoidance of virtual obstacles in an outdoor scenario. The obstacles are detected online using a virtually-rendered image and a safe trajectory is planned and executed reactively. Additionally, the ability of the learned model to predict the quadrotor's state is verified through multiple experimental trials.

The rest of the paper is organized as follows. In section II, a simplified nonlinear model of quadrotor dynamics that is appropriate for system identification is proposed. Further, the parameters for the nonlinear model are identified using an online setup of a maximum likelihood estimation framework. In section III, an NMPC based optimization scheme is designed that takes into account uncertainty in state space explicitly and plans trajectories that avoid obstacles by a safety margin. Finally, in section IV, the results of avoiding an obstacle using the current framework are demonstrated.

II. System Identification

The quadrotor is modeled as a rigid body attached with four axially aligned rotors. The rotors apply a thrust force along a known body fixed axial direction and torques along three mutually perpendicular body axes. The quadrotor is usually equipped with an autopilot module that converts commanded Euler angles into rotor velocities using a linear Proportional-Integral-Derivative (PID) controller [22]. The goal of this section is to propose a second-order closed-loop model that models both the rigid body dynamics and the autopilot control loop.

The state of the quadrotor system is given by the position p, rotation matrix R, velocity $\dot{p}$ measured with respect to an inertial frame and angular velocity ω in body frame. The rotation matrix is decomposed into body Euler angles as $R(\xi) = e^{\xi_3 \widehat{e}_3} e^{\xi_2 \widehat{e}_2} e^{\xi_1 \widehat{e}_1}$. The inputs for the second order closed-loop model are the commanded rate of body Euler angles as $\dot{\xi}_c$ and the commanded thrust u_t. A quadrotor model emulating a second order rotational dynamics has been proposed in Eq (1). Similar simplified models have been used in system identification of quadrotors [6], [17].

$$\frac{d}{dt}\begin{bmatrix} p \\ R \\ \dot{p} \\ \omega \\ \xi_c \end{bmatrix} = \underbrace{\begin{bmatrix} \dot{p} \\ R\widehat{\omega} \\ g + a_e \\ -k_p(\xi - \xi_c) - k_d\dot{\xi} + \alpha_e \\ 0 \end{bmatrix}}_{\mathbf{f}(\mathbf{x},\theta)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_t R e_3 & 0 \\ 0 & k_d \\ 0 & 1 \end{bmatrix}}_{\mathbf{g}(\mathbf{x},\theta)} \begin{bmatrix} u_t \\ \dot{\xi}_c \end{bmatrix}. \tag{1}$$

The hat operator $\widehat{\cdot}$ maps a vector in $\mathbb{R}^3$ to $se(3)$ as shown in Eq (2).

$$\hat{\omega} := \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \forall \omega \in \mathbb{R}^3. \tag{2}$$

The unknown parameters for the model are the proportional and derivative gains k_p, k_d, thrust gain k_t, and external accelerations a_e, α_e ($\theta = [k_p^T, k_d^T, k_t, a_e^T, \alpha_e^T]^T$). Using position and orientation measurements of the quadrotor and assuming the parameters and the measurements are distributed according to a Gaussian distribution, standard MLE techniques can be applied to find the unknown parameters under sufficient excitation [23]. For the quadrotor system, exciting the quadrotor in roll, pitch, and yaw directions with constant thrust is sufficient to estimate the parameters. Figure 2 compares the predicted Euler angles and body angular velocities with measurements from an Inertial Measurement Unit (IMU) during an experimental flight path.

III. Obstacle Avoidance using Nonlinear Model Predictive Control

In this section, an optimization scheme is designed to produce NMPC trajectories that avoid obstacles using the quadrotor model identified in (1). Uncertainty in the model

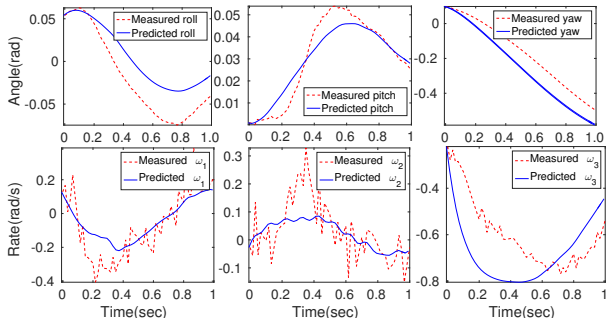

Fig. 2: The graphs show predicted and observed body Euler angles and body angular rates for the quadrotor on a test data set. The predicted Euler angles are close to the measured values using the second order closed-loop model.

is taken into account by planning a trajectory that stays away from obstacles by a safety margin based on the uncertainty propagation from the estimated parameters to trajectory uncertainty.

A. Discrete Dynamics

NMPC optimization requires a discrete version of the dynamics specified in Eq (1). The control inputs to the second order closed-loop model (i.e. the commanded thrust u_t and commanded Euler angle rates $\dot{\xi}_c$) are assumed to be constant during a time step of duration h. The discrete state consisting of position p_i, velocity v_i, rotation matrix R_i, and body angular velocity ω_i is propagated as

$$\underbrace{\begin{bmatrix} p_{i+1} \\ R_{i+1} \\ v_{i+1} \\ \omega_{i+1} \\ \xi_{c_{i+1}} \end{bmatrix}}_{x_{i+1}} = \underbrace{\begin{bmatrix} p_i + \frac{1}{2}(v_i + v_{i+1})h \\ R_i \exp(\frac{1}{2}(\omega_i + \omega_{i+1})h) \\ v_i + h(g + a_0) \\ \omega_i + h(-k_p(\xi_i - \xi_{c_{i+1}}) - k_d\dot{\xi}_i + \alpha_e) \\ \xi_{c_i} \end{bmatrix}}_{f_i} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_t R_i e_3 & 0 \\ 0 & k_d \end{bmatrix}}_{g_i} \underbrace{\begin{bmatrix} u_{t_i} \\ \dot{\xi}_{c_i} \end{bmatrix}}_{u_i}$$

The linear and angular velocities v_{i+1}, ω_{i+1} are propagated first. The average linear and angular velocities in turn are used in the position and orientation updates using a semi-implicit scheme in the above equation.

B. Propagating Uncertainty

The uncertainty of parameters obtained from MLE is propagated to the uncertainty in the states using the unscented transform proposed by Jullier et al. [24]. The uncertainty in states is then used to plan safe trajectories for avoiding obstacles. The unmodeled dynamics are included in the uncertainty through external accelerations and torques a_e, τ_e. The uncertainty in detecting the obstacles is not included in this propagation scheme.

The uncertainty propagation scheme finds the mean and covariance of the states along the trajectory given the estimated mean and covariance of the unknown parameters in dynamics. The mean and covariance of the estimated parameters θ obtained using MLE estimation are denoted by

$$\theta^* = [k_p^*, k_d^*, k_t^*, \bar{a}_e, \bar{\alpha}_e], \quad \Sigma_{\theta^*} = [\Sigma_{k_p^*, k_d^*, k_t^*}, \Sigma_{\bar{a}_e}, \Sigma_{\bar{\alpha}_e}]. \tag{3}$$

The unscented transform perturbs the mean parameters θ^* along the major axis of the uncertainty distribution to obtain sigma point parameters. The controls are propagated using these parameters to obtain sample trajectories. Then, the mean and covariance of the quadrotor along the trajectory are obtained as a weighted sum of the samples. The unscented transform function can be written as

$$(\bar{x}_{0:N-1}, \Sigma_{x_{0:N-1}}) = \textit{UnscentedTransform}(u_{0:N-1}, \theta^*, \Sigma_{\theta^*})$$

where the $\bar{x}_{0:N-1}$ is the mean state along the trajectory, $\Sigma_{x_{0:N-1}}$ is the covariance of the the state along the trajectory, and $u_{0:N-1}$ is the control trajectory.

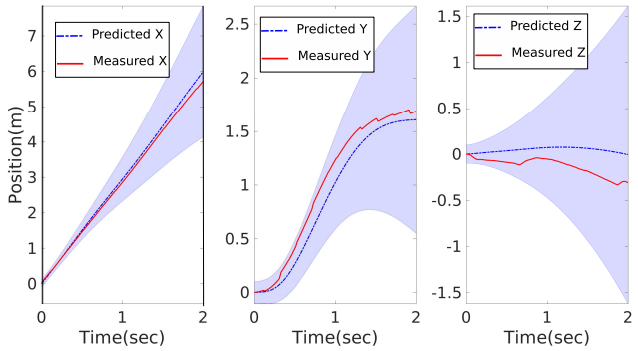

Fig. 3: The predicted mean position and 2σ region of the quadrotor obtained using unscented transforms is shown in blue. The measured position of the quadrotor is shown in red.

Figure 3 shows the plot of a predicted trajectory distribution obtained using unscented transforms based on parameters estimated from real data. It can be observed that the navigated trajectory in red falls within the predicted 2σ region in blue, indicating that the model is a good fit for the dynamics. The uncertainty in the predicted trajectory is used to avoid obstacles using an NMPC formulation explained next.

C. NMPC Formulation

The NMPC problem can be formulated as finding a trajectory for the quadrotor system with states and controls given by $x_{0:N}, u_{0:N-1}$ which minimizes a cost function $L(x_{0:N}, u_{0:N-1})$ while satisfying the system dynamics. The cost function incorporates the specifications of a problem such as minimization of control effort u_i and state velocities v_i, ω_i along the trajectory, while achieving a goal state x_N,

and avoiding obstacles. The NMPC problem for a generic system can be mathematically stated as:

NMPC Framework

Given x_0, θ^*, x_f,

$$u^*_{1:N} = \arg\min \sum_{i=0}^{N-1} L_i + L_N,$$

s.t $x_{i+1} = f_i + g_i u_i$.

The cost function for trajectory tracking is chosen to be a quadratic cost that minimizes the control effort and state velocities along the trajectory. The terminal cost L_N is designed to force the system to achieve a desired terminal state x_f. The cost function is formulated as

$$L_i = \frac{1}{2} x_i^T Q x_i + \frac{1}{2} u_i^T R u_i, \tag{4}$$

$$L_N = \frac{1}{2} (x_N - x_f)^T Q_N (x_N - x_f), \tag{5}$$

where the gain matrix Q_N penalizes the difference between the end of the trajectory x_N and the goal state x_f, while the gains Q and R determine the trade-off between minimizing the state velocities and control effort along the trajectory, respectively. The state of the quadrotor as shown in (1) lives on a manifold and cannot be subtracted in a trivial way. The operator $-$ in the above cost function is overloaded to be the distance between two states on the manifold as explained in [25], [26].

The NMPC formulation is augmented with an additional constraint to ensure obstacle avoidance. We define an inequality constraint for every obstacle at every point along the trajectory as

$$\text{Define: } P_i = \{p : (p - p_i)^T \Sigma_{p_i}^{-1} (p - p_i) \leq k_\sigma^2\} \tag{6}$$

$$d_{i,j} = \text{Dist}(P_i, o_j) \geq 0 \tag{7}$$

The standard deviation ellipsoid P_i consists of all the points which are within k_σ standard deviations from the current position p_i. The signed distance function *Dist* is then defined as the closest distance between the boundary of the ellipsoid and the boundary of an obstacle as shown in Figure 4. The distance between the ellipsoid P_i and a cylindrical obstacle with center Op_j, major axis Oa_j and radius Or_j can be approximated in closed form[1] as shown in Eq (9), where the symbol $\langle , \rangle$ denotes the dot product. When the ellipsoid intersects with the obstacle, the distance function returns a negative value. Therefore, minimization of the constrained NMPC cost function finds a trajectory that avoids obstacles by k_σ standard deviations. In this work, a 2σ standard deviation boundary has been chosen.

$$Dist(P_i, o_j) = \|e_{i,j}\| - \frac{k_\sigma \|e_{i,j}\|}{\sqrt{e_{ij}^T \Sigma_{p_i}^{-1} e_{i,j}}} - Or_j, \tag{8}$$

$$e_{ij} = (Op_j - p_i) - \langle Op_j - p_i, Oa_j \rangle Oa_j. \tag{9}$$

[1]This distance is only an approximation that performs well in practice. The true closest distance cannot be computed in closed form.

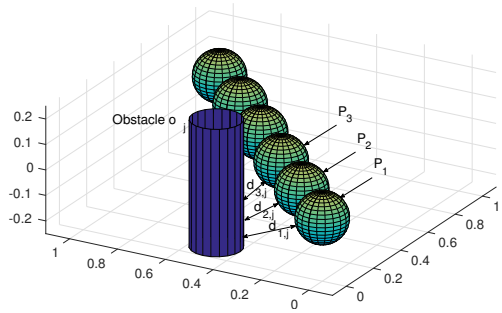

Fig. 4: Plot of a sample trajectory with standard deviation ellipsoids P_i and the distance to obstacle d_i. The ellipsoids P_i consist of points which are closer than k_σ standard deviations from the quadrotor position.

D. Optimization Scheme

The Levenberg–Marquardt algorithm is used to find the optimal controls u^* since the cost functions L_i, L_N are of the least-squares form. To reduce the dimension of the optimization, the control inputs u_i are produced using a uniform B-spline of second order with knots given by ψ. The control can be recovered from the knots as

$$u_{0:N-1} = B(t_{0:N-1})\psi. \tag{10}$$

The knot matrix $\psi \in \mathbb{R}^{k \times 4}$ has each row as a knot vector which contains a thrust u_t and commanded body Euler angle rates $\dot{\xi}_c$. The basis matrix $B \in \mathbb{R}^{N \times k}$ is dependent on the time sequence $t_{0:N-1}$ and the order of the B-spline. A second order B-spline has been used in this work.

The residual function and update step for the Levenberg-Marquardt algorithm are then given by

$$e(\psi) = \begin{bmatrix} \sqrt{Q} x_1(\psi) \\ \sqrt{R} u_1(\psi) \\ \sqrt{k_o} D_{1,j}(\psi) \\ \vdots \\ \sqrt{Q_N}(x_N(\psi) - x_f) \end{bmatrix}, \tag{11}$$

$$J = \partial_\psi e(\psi), \tag{12}$$

$$D_{i,j} = \min(d_{i,j}, 0), \tag{13}$$

$$\psi_{i+1} = \psi_i - (J^T J + \lambda I)^{-1} J^T e(\psi).. \tag{14}$$

The obstacle avoidance constraint is enforced as a soft penalty in the residual function ((11)). The residual $\sqrt{k_o} D_{i,j}(\psi)$ minimizes the intersection between the standard deviation ellipsoids P_i and the obstacle o_j. The optimization algorithm is iterated multiple times by increasing the obstacle gain k_o after every run. This procedure smoothly transitions from a trajectory with P_i intersecting the obstacle to a trajectory with P_i slightly grazing the obstacle as the gain k_o becomes very large. The complete optimization procedure is listed as follows:

Obstacle Avoidance using Levenberg-Marquardt

0. Start with initial optimal parameter distribution $(\theta^*, \Sigma_{\theta^*})$, fixed obstacle gain k_o and initial guess for current control knots ψ
1. Compute the standard deviation of the state trajectory at the current control knots using *UnscentedTransform*
2. Evaluate the residual e and Jacobian J of the trajectory at the current control knots
3. Update the knots ψ according to Eq (14); If iteration converged GOTO step 4, else GOTO step 1
4. Increase gain k_o by a fixed factor. If k_o reaches an upper limit **exit** else GOTO step 1

Figure 1 shows an example obstacle avoidance scenario, where the quadrotor avoids two cylinders in front of it while flying at 5m/s. The goal for the quadrotor is to reach 10 meters in front of it while avoiding obstacles. The optimization algorithm finds a trajectory for which the standard deviation ellipsoids in blue do not intersect the obstacles in red while achieving the final goal. Thus, following this open-loop trajectory approximately guarantees that the quadrotor can navigate safely around the obstacles.

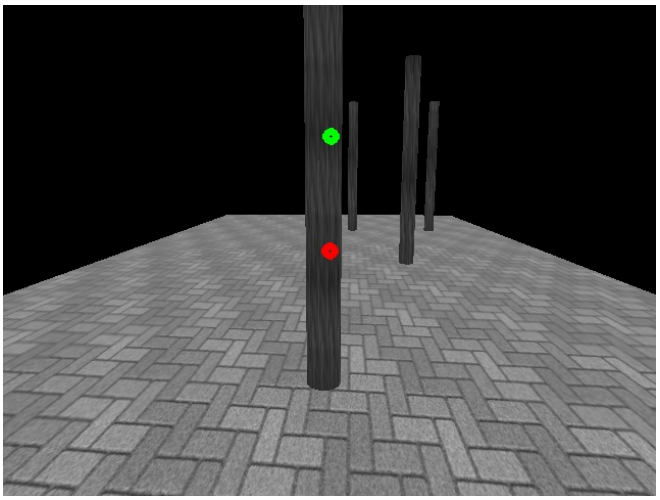

Fig. 5: An onboard image rendered using a virtual camera. The green dot shows the mean of the closest 20 percent points in the foreground which represents the obstacle and the red dot shows the center of the image.

IV. Experiment Setup

The goal of the experiments is to demonstrate safe obstacle avoidance behavior for a quadrotor by following an open-loop trajectory computed using the NMPC optimization technique. The obstacle avoidance experiments are conducted on a real quadrotor in an outdoor scenario, subject to unknown external disturbances, and surrounded by virtual obstacles. The obstacle avoidance behavior has been demonstrated in two ways. First, multiple executions of open-loop trajectories have been performed to verify that the quadrotor model can predict the quadrotor pose well. Next, the quadrotor is set to avoid three consecutive virtual obstacles reactively. The obstacle avoidance experiments require several components such as an obstacle detection algorithm, a position controller, and an online parameter estimation framework to work together as discussed below.

A. Subsystems

1) Hardware: A research grade Matrice quadrotor made by DJI [27] is used in the experiments. The quadrotor is equipped with a Guidance stereo camera sensor [28] whose data is used to produce high-quality position and orientation measurements at 100 Hz. The DJI autopilot is used to achieve desired body Euler angles roll, pitch, yaw and desired thrust. Figure 1 shows the quadrotor setup connected with Guidance and DJI autopilot.

2) Obstacle Detection: A virtual camera image is rendered on a georeferenced virtual map using the Open Source 3D Graphics Engine library [29] to provide a depth map for estimating the distance to obstacles. The virtual camera avoids the issue of sensor uncertainty in detecting the obstacles and allows for safe experimental testing of the quadrotor in an outdoor scenario. Figure 5 shows an example virtual image rendered from the quadrotor's position. The obstacle position is determined from the rendered frame by segmenting the depth map into foreground and background based on the global velocity direction of the quadrotor. The pixels inside a tolerance cylinder of 0.5m radius around the global velocity direction and within a maximum depth tolerance of 6m are considered to be foreground. The closest obstacle position is then specified to be the average of the closest 20 percent points in the foreground.

3) Online Parameter Estimation: The obstacle avoidance experiments are conducted in an outdoor environment with unknown external disturbances. The quadrotor is also subject to thrust depreciation over time due to limited onboard power. These effects are compensated using MLE online parameter estimation. The initialization of MLE optimization requires a good prior on the system parameters. Therefore, a manually flown ten-second trajectory is utilized in an initial optimization to produce a tight prior for the system parameters. The system parameters are further refined based on this prior using online optimization running at a frequency of 0.5Hz using the position and orientation measurements collected at 100 Hz. The covariance on the parameters is continuously tracked by the program and is used for fault detection in the case of bad parameter learning.

4) NMPC Trajectory Optimization: The NMPC optimization is performed when an obstacle is within a user specified tolerance distance of 3 meters. The goal of the optimization is to reach 3 meters behind the obstacle within the next two seconds while avoiding the obstacle and achieving a terminal velocity that is equal to the initial velocity. The two second trajectory is broken down into 100 discrete segments (50 Hz) for optimization. In this experiment, the obstacles are avoided one at a time. Hence, a second order B-spline with four knots was sufficient to represent the control trajectory.

The computational resources available onboard are limited and only allow for a 10Hz frequency of the NMPC optimization loop. This restricts the frequency at which the optimal trajectory can be updated. Thus, the quadrotor is run open-loop using the roll, pitch, yaw and thrust inputs to autopilot

until the MPC horizon time is complete.

5) Waypoint Tracker: A linear PID position controller is designed to track user-defined waypoints. The external force parameters and thrust gain obtained from MLE optimization are incorporated into the PID controller.

V. Results

A. Verification of quadrotor model

The ability of the quadrotor model to predict the quadrotor pose is demonstrated by executing open-loop trajectories multiple times to show that the quadrotor stays within the standard deviation funnel almost all the time. The trials require the quadrotor to fly at a speed of 3m/s and execute an open-loop NMPC trajectory to avoid a virtual obstacle assumed to be 3m in front of it. The data samples from the trials are segregated into a bar graph in Figure 6 based on the distance to the ellipsoid surface. Around 89.89% of the 2000 sample data points are within the ellipsoid as observed from the distance metric being negative in the bar graph. These results suggest that by following the NMPC trajectory, the quadrotor is likely to avoid the obstacle with approximately high probability. Figure 7 shows the predicted and measured

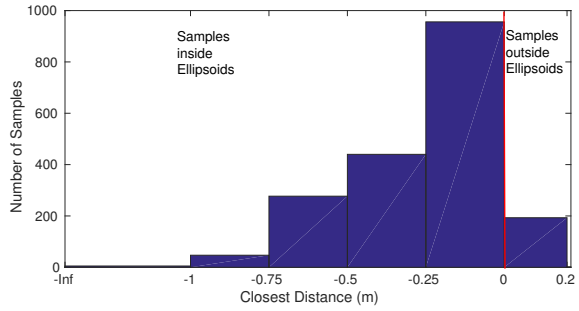

Fig. 6: The bar graph shows the number of data samples at different distances to ellipsoid. Around 89.89% of the samples along the quadrotor trajectories are within the standard deviation ellipsoids and thus are safe for obstacle avoidance

open-loop Euler angles and Figure 3 shows the predicted and measured position of the quadrotor along an experimental trajectory for a single obstacle avoidance trial. It can be observed that the predicted position stays within the standard deviation funnel and predicted Euler angles matched well with the measured Euler angles. These results indicate that the identified model approximates the quadrotor well, and NMPC trajectories are safe for avoiding a real obstacle.

B. Consecutive Virtual Obstacle Avoidance

In this experiment, the quadrotor is tasked to avoid a series of three virtual cylindrical obstacles 0.3m in diameter and spaced 5m apart while following a waypoint path at 3m/s. The quadrotor determines a obstacle avoidance path when it sees a cylinder at 3 meters in front of it. The path is required to reach 3 meters beyond the obstacle within the next two seconds while maintaining the current quadrotor velocity. The combined subsystems of online parameter estimation,

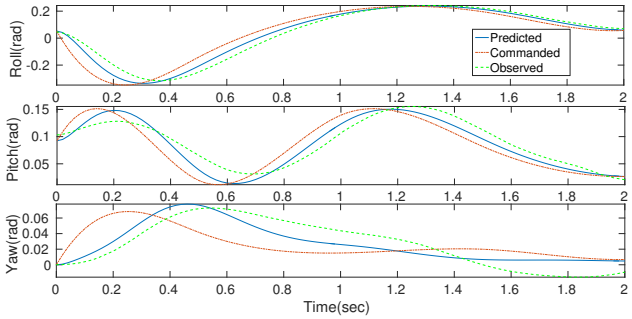

Fig. 7: The graphs show the predicted, commanded and measured body Euler angles for a quadrotor during a single obstacle avoidance run. It can be observed the measured angles are very close to the predicted Euler angles.

obstacle detection, NMPC trajectory optimization, and waypoint tracking are tested through this experiment. The upper half of Figure 8 shows the quadrotor positions in the virtual map overlayed with detected obstacles in red and computed NMPC trajectories with covariance ellipsoids in blue. The obstacles are inflated by 0.3m to include the envelope of the quadrotor. The series of pictures in the lower half of the figure show the quadrotor flying through the virtual obstacle course. The figures show that the quadrotor avoids the obstacles successfully while staying in the predicted standard deviation funnel when executing the NMPC trajectory.

The following steps are performed during an obstacle avoidance experiment:

1) Identify model parameters online by running MLE-based parameter estimation at 0.5Hz
2) Follow a waypoint path using PID position controller
3) When an obstacle is detected in the path within a user specified tolerance, compute and execute NMPC trajectory
4) Resume following waypoint path
5) A human operator takes over the controls if any of the above steps go wrong

VI. Conclusions

This work proposes an NMPC trajectory generation technique that combines online parameter estimation with uncertainty propagation to generate approximately safe obstacle avoidance trajectories. The ability of the nonlinear stochastic quadrotor model to predict the quadrotor state has been demonstrated through multiple open-loop trajectories. Further, the quadrotor is able to reactively avoid consecutive virtual obstacles while staying in the standard deviation funnel while following the NMPC trajectories. Future work will enable obstacle avoidance in a real cluttered environment by performing closed-loop NMPC trajectory optimization.

References

[1] "Prime air." http://www.amazon.com/b?node=8037720011.
[2] "Airobots." http://airobots.ing.unibo.it.
[3] S. Gupte, P. I. T. Mohandas, and J. M. Conrad, "A survey of quadrotor unmanned aerial vehicles," in *2012 Proceedings of IEEE Southeastcon*, pp. 1–6, IEEE, March 2012.

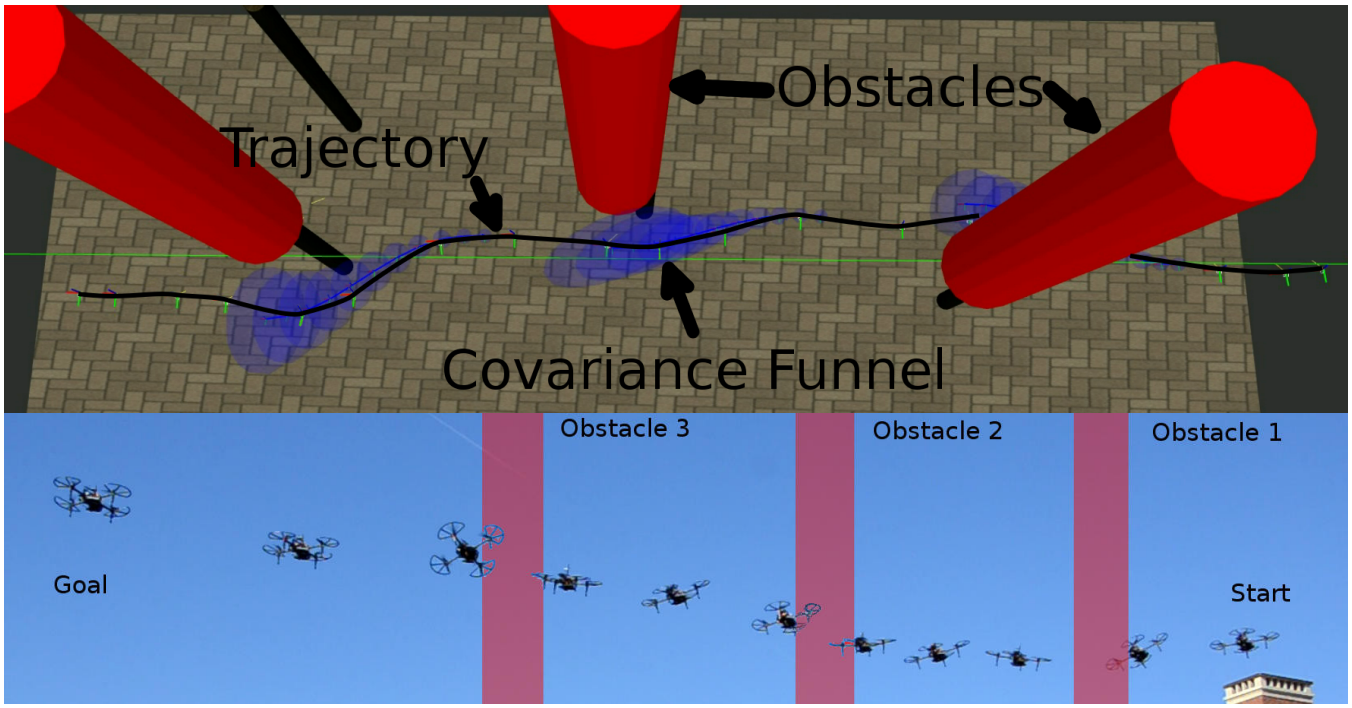

Fig. 8: Pictures show a quadrotor moving at 3m/s evading three virtual cylinders by following an open-loop NMPC trajectory. The quadrotor stays within the funnel while avoiding the obstacles as shown in the figure in the top row. The figure in the bottom row shows the physical quadrotor flying through the obstacle course.

[4] I. M. Salameh, E. M. Ammar, and T. A. Tutunji, "Identification of quadcopter hovering using experimental data," in *2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pp. 1–6, IEEE, 2015.

[5] T. Ryan and H. J. Kim, "LMI-Based Gain Synthesis for Simple Robust Quadrotor Control," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 4, pp. 1173–1178, 2013.

[6] I. Sa and P. Corke, "System identification, estimation and control for a cost effective open-source quadcopter," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2202–2209, IEEE, 2012.

[7] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor uavs: A design trade study with flight evaluations," *IEEE Transactions on control systems technology*, vol. 21, no. 4, pp. 1400–1406, 2013.

[8] C. Nicol, C. J. B. Macnab, and A. Ramirez-Serrano, "Robust adaptive control of a quadrotor helicopter," *Mechatronics*, vol. 21, no. 6, pp. 927–938, 2011.

[9] E. N. Johnson and S. K. Kannan, "Adaptive Trajectory Control for Autonomous Helicopters," *Journal of Guidance, Control, and Dynamics*, vol. 28, pp. 524–538, may 2005.

[10] R. Xu and Ü. Özgüner, "Sliding mode control of a quadrotor helicopter," in *2006 45th IEEE Conference on Decision and Control*, pp. 4957–4962, IEEE, 2006.

[11] R. Cunha, D. Cabecinhas, and C. Silvestre, "Nonlinear trajectory tracking control of a quadrotor vehicle," in *2009 European Control Conference (ECC)*, pp. 2763–2768, IEEE, 2009.

[12] M. Kobilarov, "Trajectory tracking of a class of underactuated systems with external disturbances," in *American Control Conference (ACC)*, pp. 1044–1049, 2013.

[13] Y. Liu, J. M. Montenbruck, P. Stegagno, F. Allgower, and A. Zell, "A robust nonlinear controller for nontrivial quadrotor maneuvers: Approach and verification," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5410–5416, IEEE, 2015.

[14] A. Budiyanto, A. Cahyadi, T. B. Adji, and O. Wahyunggoro, "Uav obstacle avoidance using potential field under dynamic environment," in *2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, pp. 187–192, 2015. ID: 1.

[15] A. J. Barry, T. Jenks, A. Majumdar, H. T. Lin, I. G. Ros, A. A. Biewener, and R. Tedrake, "Flying between obstacles with an autonomous knife-edge maneuver," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2559–2559, IEEE, May 2014.

[16] R. Allen and M. Pavone, "A real-time framework for kinodynamic planning with application to quadrotor obstacle avoidance," in *AIAA Conf. on Guidance, Navigation and Control, San Diego, CA*, 2016.

[17] P. Bouffard, A. Aswani, and C. Tomlin, "Learning-based model predictive control on a quadrotor: Onboard implementation and experimental results," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 279–284, IEEE, 2012.

[18] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *Automatica*, vol. 49, no. 5, pp. 1216–1226, 2013.

[19] C. Lehnert and G. Wyeth, "Locally Weighted Learning Model Predictive Control for nonlinear and time varying dynamics," in *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2619–2625, IEEE, 2013.

[20] A. Aswani, N. Master, J. Taneja, D. Culler, and C. Tomlin, "Reducing transient and steady state electricity consumption in hvac using learning-based model-predictive control," *Proceedings of the IEEE*, vol. 100, pp. 240–253, Jan 2012.

[21] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *arXiv preprint arXiv:1601.04037*, 2016.

[22] F. Goodarzi, D. Lee, and T. Lee, "Geometric nonlinear pid control of a quadrotor uav on se (3)," *arXiv preprint arXiv:1304.6765*, 2013.

[23] M. H. DeGroot, *Optimal statistical decisions*, vol. 82. John Wiley & Sons, 2005.

[24] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *AeroSense'97*, pp. 182–193, International Society for Optics and Photonics, 1997.

[25] M. Kobilarov and J. Marsden, "Discrete geometric optimal control on Lie groups," *IEEE Transactions on Robotics,*, vol. 27, pp. 641–655, August 2011.

[26] G. Garimella and M. Kobilarov, "Towards model-predictive control for aerial pick-and-place," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4692–4697, May 2015.

[27] "Dji matrice quadrotor experimental platform." http://www.dji.com/product/matrice100, 2015.

[28] "Dji guidance sensor." http://www.dji.com/product/guidance, 2015.

[29] "Ogre - open source 3d graphics engine." http://www.ogre3d.org/, 2015.