Gaussian Process Model Predictive Control of Unmanned Quadrotors

2016 The 2nd International Conference on Control, Automation and Robotics

Gang Cao

School of Engineering and Advanced Technology
Massey University, Auckland
New Zealand

April 29, 2016

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

Outline

Abbreviations

- **GP: Gaussian process**
- **MPC: Model predictive control**
- **NTV: Nonlinear time varying**
- **CG: Conjugate gradient**
- **MSE: Mean square errors**

Problem Formulation

Figure: Quadrotor in AUT Lab

Problem Formulation

Quadrotor State-Space Functions:

$$\boldsymbol{\xi}_{k+1} = \mathcal{F}_{\xi}(\boldsymbol{\xi}_k, \mathbf{u}_{\xi,k}) \tag{1a}$$

$$\boldsymbol{\eta}_{k+1} = \mathcal{F}_{\eta}(\boldsymbol{\eta}_k, \mathbf{u}_{\eta,k}) \tag{1b}$$

- $\boldsymbol{\xi} = [x, \dot{x}, y, \dot{y}, z, \dot{z}]^T$, $\mathbf{u}_{\xi} = [U_1, u_x, u_y]^T$,
  $\boldsymbol{\eta} = [\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$, $\mathbf{u}_{\xi} = [U_2, U_2, U_3]^T$,
- $\mathcal{F}_{\xi}$ and $\mathcal{F}_{\eta}$ are time-varying nonlinear functions

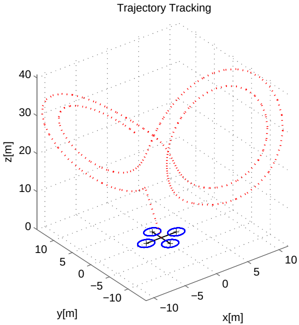
Problem Formulation

Quadrotor State-Space Functions:

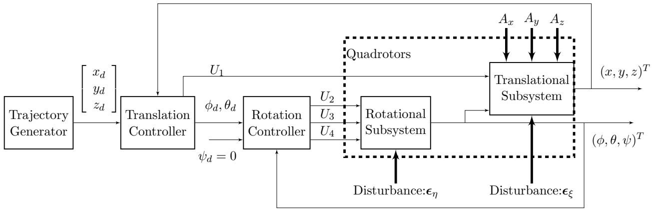
$$\boldsymbol{\xi}_{k+1} = \mathcal{F}_{\xi}(\boldsymbol{\xi}_k, \mathbf{u}_{\xi,k}) \quad (1a)$$

$$\boldsymbol{\eta}_{k+1} = \mathcal{F}_{\eta}(\boldsymbol{\eta}_k, \mathbf{u}_{\eta,k}) \quad (1b)$$

- $\boldsymbol{\xi} = [x, \dot{x}, y, \dot{y}, z, \dot{z}]^T$, $\mathbf{u}_\xi = [U_1, u_x, u_y]^T$, $\boldsymbol{\eta} = [\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$, $\mathbf{u}_\xi = [U_2, U_2, U_3]^T$,
- $\mathcal{F}_\xi$ and $\mathcal{F}_\eta$ are time-varying nonlinear functions



Proposed Control Scheme



Gaussian Process Model Predictive Control of Unmanned Quadrotors

Problem Formulation

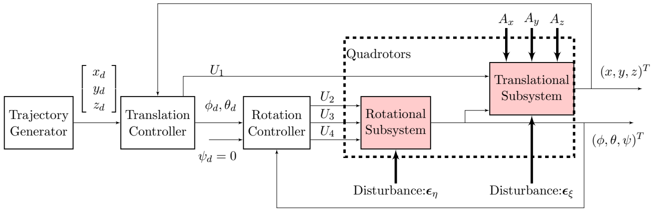
Quadrotor Modelling using GP

GP based MPC

Numerical Simulation

Conclusion

Proposed Control Scheme

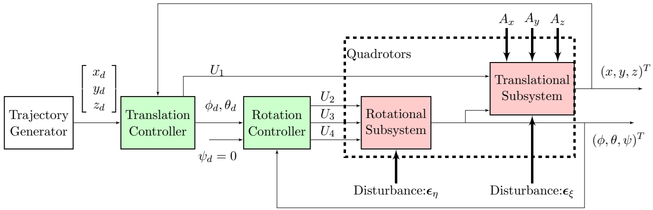


Contributions:

- Translational subsystem
- Rotational subsystem

} GP Modelling

Proposed Control Scheme



Contributions:

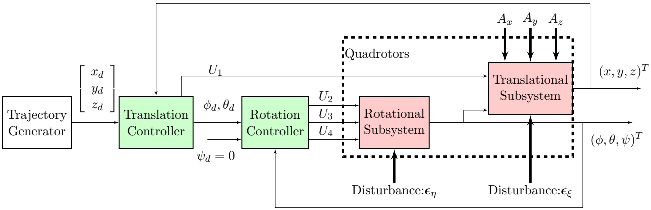
- Translational subsystem
- Rotational subsystem

} GP Modelling

- Translation MPC controller
- Rotational MPC controller

} GP based MPC

Proposed Control Scheme



Contributions:

- Translational subsystem
- Rotational subsystem

} GP Modelling

- Translation MPC controller
- Rotational MPC controller

} GP based MPC

- Gradient based solution to GPMPC

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;

Gaussian Process Model Predictive Control of Unmanned Quadrotors

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;
- Specifying a squared exponential covariance function $\mathbf{K}(\mathbf{x}_{k,i}, \mathbf{x}_{k,j})$;
  $\mathbf{K}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) = \sigma_s^2 \exp(-\frac{1}{2}(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)^T \boldsymbol{\Lambda}(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)) + \sigma_n^2$

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;
- Specifying a squared exponential covariance function $\mathbf{K}(\mathbf{x}_{k,i}, \mathbf{x}_{k,j})$;
- $\mathbf{x}_k^* \sim \mathcal{N}\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$;

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;
- Specifying a squared exponential covariance function $\mathbf{K}(\mathbf{x}_{k,i}, \mathbf{x}_{k,j})$;
- $\mathbf{x}_k^* \sim \mathcal{N}\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$;
- Evaluation of model uncertainty $\rightarrow \boldsymbol{\Sigma}_k$;

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;
- Specifying a squared exponential covariance function $\mathbf{K}(\mathbf{x}_{k,i}, \mathbf{x}_{k,j})$;
- $\mathbf{x}_k^* \sim \mathcal{N}\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$;
- Evaluation of model uncertainty $\rightarrow \boldsymbol{\Sigma}_k$;
- GP Inputs $\tilde{\mathbf{x}}_k = [\mathbf{x}_k, \mathbf{u}_k]^T$;
  GP Outputs $\delta\mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k$;

Quadrotor Modelling using GP

General NTV Form:

$$\mathbf{x}_{k+1} = \mathcal{F}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

Gaussian Process Models:

- Probabilistic data-driven model;
- Specifying a squared exponential covariance function $\mathbf{K}(\mathbf{x}_{k,i}, \mathbf{x}_{k,j})$;
- $\mathbf{x}_k^* \sim \mathcal{N}\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$;
- Evaluation of model uncertainty $\rightarrow \boldsymbol{\Sigma}_k$;
- GP Inputs $\tilde{\mathbf{x}}_k = [\mathbf{x}_k, \mathbf{u}_k]^T$;
  GP Outputs $\delta\mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k$;
- Hyperparameters learning: Minimize the negative log of likelihood (CG);

GP based MPC

$$\underbrace{\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}}$$

GP based MPC

$$\underbrace{\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}} \xrightarrow{\mathbb{E}[\cdot]} \underbrace{\mathbb{E}\Big[\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})\Big]}_{\text{stochastic NMPC}}$$

GP based MPC

$$\underbrace{\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}} \xrightarrow{\mathbb{E}[\cdot]} \underbrace{\mathbb{E}\left[\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})\right]}_{\text{stochastic NMPC}} \xrightarrow{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k} \underbrace{\mathbf{V}_k^*(\boldsymbol{\mu}_k, \color{red}{\boldsymbol{\Sigma}_k}, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}}$$

GP based MPC

$$\underbrace{\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}} \xrightarrow{\mathbb{E}[\cdot]} \underbrace{\mathbb{E}\left[\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})\right]}_{\text{stochastic NMPC}} \xrightarrow{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k} \underbrace{\mathbf{V}_k^*(\boldsymbol{\mu}_k, {\color{red}\boldsymbol{\Sigma}_k}, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}}$$

Implementation Issues:

- Uncertainty propagation → GP predictions at uncertain inputs;

GP based MPC

$$\underbrace{\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}} \xrightarrow{\mathbb{E}[\cdot]} \underbrace{\mathbb{E}\left[\mathbf{V}_k^*(\mathbf{x}_k, \mathbf{r}_k, \mathbf{u}_{k-1})\right]}_{\text{stochastic NMPC}} \xrightarrow{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k} \underbrace{\mathbf{V}_k^*(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \mathbf{r}_k, \mathbf{u}_{k-1})}_{\text{deterministic NMPC}}$$

Implementation Issues:

- Uncertainty propagation → GP predictions at uncertain inputs;
- Effective Solutions → Gradient based optimization;

Gradient based optimization

Unconstrained optimization question:

$$\mathbf{U}_k^* = \arg\min_{\mathbf{U}_k^*} \mathbb{E}\left\{\mathbf{V}_k^*\right\} \tag{3a}$$

$$\text{s.t. } \boldsymbol{\mu}_{k+1} = \mathcal{G}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \mathbf{u}_k) \tag{3b}$$

Gradient based optimization

Unconstrained optimization question:

$$\mathbf{U}_k^* = \arg\min_{\mathbf{U}_k^*} \mathbb{E}\Big\{\mathbf{V}_k^*\Big\} \tag{3a}$$

$$\text{s.t. } \boldsymbol{\mu}_{k+1} = \mathcal{G}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \mathbf{u}_k) \tag{3b}$$

- Gradient based linear search strategy

$$\mathbf{U}_{k+1} = \mathbf{U}_k + \boldsymbol{\alpha} \cdot \underbrace{\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k}}_{\text{Gradient}} \tag{4}$$

$$\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} = \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\mu}_{k+1}} \frac{\partial \boldsymbol{\mu}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\Sigma}_{k+1}} \frac{\partial \boldsymbol{\Sigma}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} \tag{5}$$

Gradient based optimization

Unconstrained optimization question:

$$\mathbf{U}_k^* = \arg\min_{\mathbf{U}_k^*} \mathbb{E}\left\{\mathbf{V}_k^*\right\} \tag{3a}$$

$$\text{s.t. } \boldsymbol{\mu}_{k+1} = \mathcal{G}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \mathbf{u}_k) \tag{3b}$$

- Gradient based linear search strategy

$$\mathbf{U}_{k+1} = \mathbf{U}_k + \boldsymbol{\alpha} \cdot \underbrace{\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k}}_{\text{Gradient}} \tag{4}$$

$$\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} = \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\mu}_{k+1}} \frac{\partial \boldsymbol{\mu}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\Sigma}_{k+1}} \frac{\partial \boldsymbol{\Sigma}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} \tag{5}$$

- End when $\mathbb{E}\{\mathbf{V}_k\}_{\mathbf{U}_k} \leq \epsilon$;

Gradient based optimization

Unconstrained optimization question:

$$\mathbf{U}_k^* = \arg\min_{\mathbf{U}_k^*} \mathbb{E}\left\{\mathbf{V}_k^*\right\} \tag{3a}$$

$$\text{s.t. } \boldsymbol{\mu}_{k+1} = \mathcal{G}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \mathbf{u}_k) \tag{3b}$$

- Gradient based linear search strategy

$$\mathbf{U}_{k+1} = \mathbf{U}_k + \boldsymbol{\alpha} \cdot \underbrace{\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k}}_{\text{Gradient}} \tag{4}$$

$$\frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} = \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\mu}_{k+1}} \frac{\partial \boldsymbol{\mu}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \boldsymbol{\Sigma}_{k+1}} \frac{\partial \boldsymbol{\Sigma}_{k+1}}{\partial \mathbf{U}_k} + \frac{\partial \mathbb{E}\{\mathbf{V}_k\}}{\partial \mathbf{U}_k} \tag{5}$$

- End when $\mathbb{E}\{\mathbf{V}_k\}_{\mathbf{U}_k} \leq \epsilon$;
- Multi-Start with different initial values;

Numerical Simulation

$$\boldsymbol{\xi}_{k+1} = \mathcal{F}_{\xi}(\boldsymbol{\xi}_k, \mathbf{u}_{\xi,k}; \boldsymbol{\theta}_{\xi}) + \boldsymbol{\epsilon}_{\xi} \tag{6a}$$

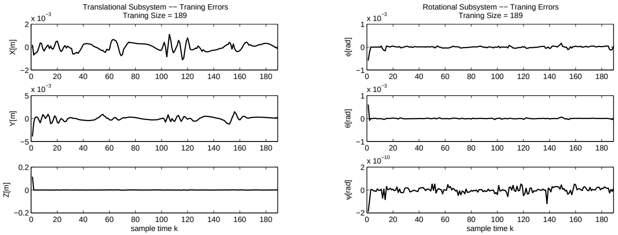
$$\boldsymbol{\eta}_{k+1} = \mathcal{F}_{\eta}(\boldsymbol{\eta}_k, \mathbf{u}_{\eta,k}; \boldsymbol{\theta}_{\eta}) + \boldsymbol{\epsilon}_{\eta} \tag{6b}$$

- Specify the parameters $\boldsymbol{\theta}_{\xi}$ and $\boldsymbol{\theta}_{\eta}$
- Specify two Gaussian noises $\boldsymbol{\epsilon}_{\xi}$ and $\boldsymbol{\epsilon}_{\eta}$
- Observations are generated by using a NMPC
- All observations are used to train the GP model
- 50 repeats
- Compare to the Nelder-Mead method (derivative-free)

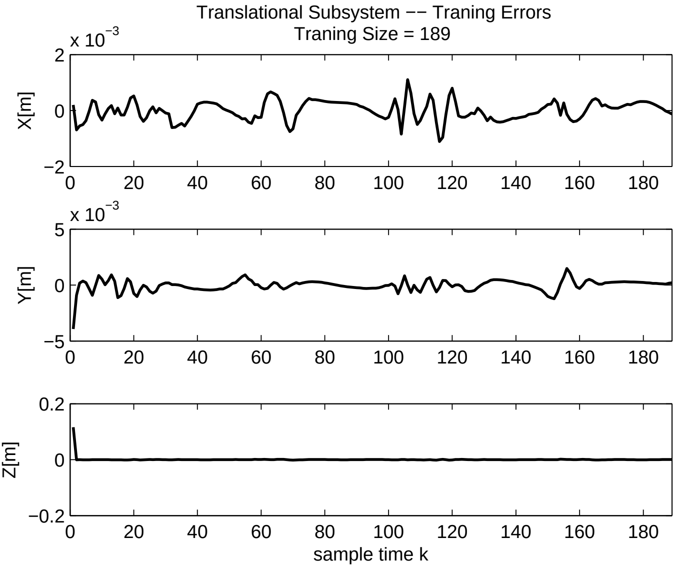
Numerical Simulation

Modelling Results

	Training MSE
Translational Subsystem	1.8693×10^{-7}
Rotational Subsystem	8.5238×10^{-9}



Numerical Simulation



Gaussian Process Model Predictive Control of Unmanned Quadrotors

Problem Formulation

Quadrotor Modelling using GP

GP based MPC

Numerical Simulation

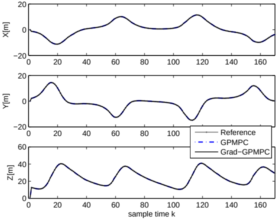
Conclusion

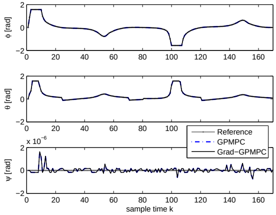
Numerical Simulation

Control Results

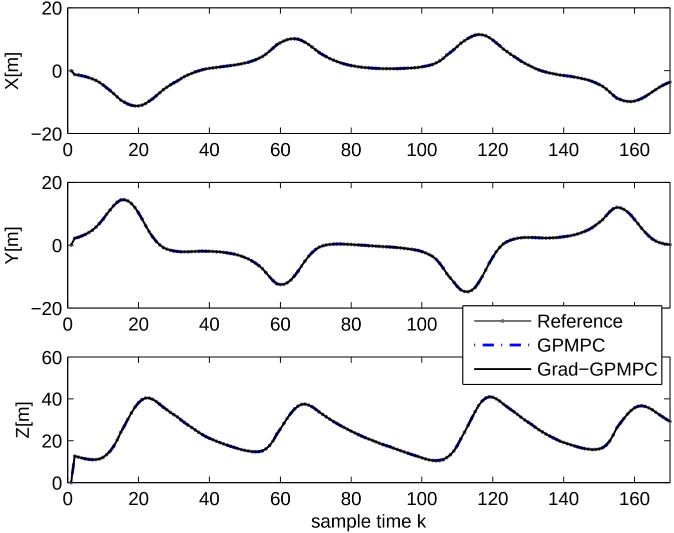
- "Grad-GPMPC" only requires approximately **30%** time of "GPMPC" to solve the MPC problem

Positions	Tracking MSE	Attitudes	Tracking MSE
X	3.3418×10^{-4}	ϕ	1.5743×10^{-8}
Y	5.1399×10^{-5}	θ	5.5213×10^{-9}
Z	0.0010	ψ	6.4365×10^{-14}



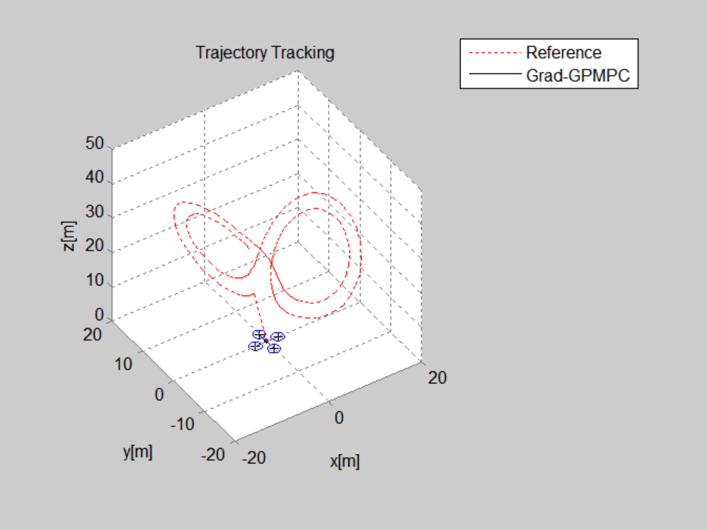


Numerical Simulation



Tracking Results

Viewing angle: az=-52, el=14



Conclusion

Conclusions:

- Proposed GPMPC performs well in Quadrotor tracking problems
- Proposed gradient based method solve the GPMPC problem more efficiently

Conclusion

Conclusions:

- Proposed GPMPC performs well in Quadrotor tracking problems
- Proposed gradient based method solve the GPMPC problem more efficiently

Future Works:

- GPMPC to control Quadrotor with constraints
- Guaranteed stability and robustness against uncertainties

Thanks!

Questions?