Adaptive Gaussian Process Modeling for Trajectory Simulation with Model Inexactness

Jonathan Cangelosi

Department of Computational Applied Mathematics and Operations Research
Rice University, Houston, Texas
jrc20@rice.edu

November 6, 2023

This work was supported in part by AFOSR Grant FA9550-22-1-0004.

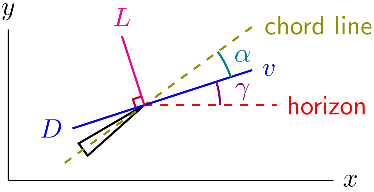
Motivation

- Hypersonic flight modeled using aerodynamic forces (i.e., lift and drag) which depend on altitude, velocity, etc.
- Lift/drag may be computed in high fidelity using CFD simulations $\rightarrow$ very costly
- To reduce cost, construct cheaper surrogate model for lift/drag using small number of high-fidelity samples

Motivation

- Hypersonic flight modeled using aerodynamic forces (i.e., lift and drag) which depend on altitude, velocity, etc.
- Lift/drag may be computed in high fidelity using CFD simulations $\rightarrow$ very costly
- To reduce cost, construct cheaper surrogate model for lift/drag using small number of high-fidelity samples
- Garbage in, garbage out; if surrogate model is poor, trajectory will be inaccurate

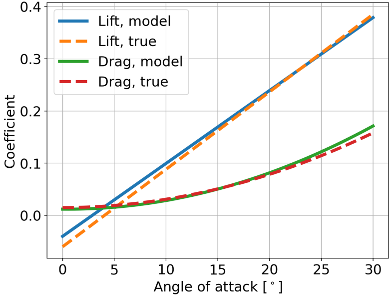
Dynamic model

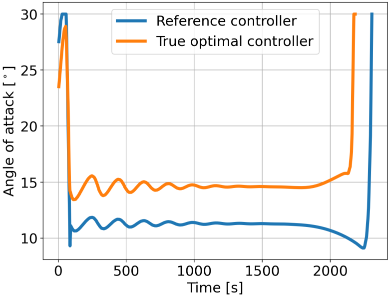


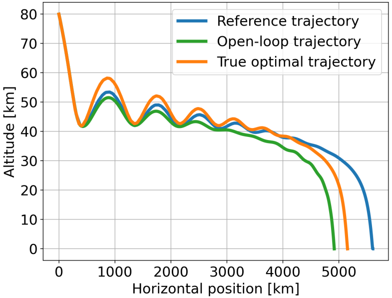
- Trajectory controlled by α through dynamic equations

$$
\begin{aligned}
x' &= v\cos\gamma \\
y' &= v\sin\gamma \\
v' &= -\frac{1}{m}\Big(D(y,v,\alpha) + mg(y)\sin\gamma\Big) \\
\gamma' &= \frac{1}{mv}\Big(L(y,v,\alpha) - mg(y)\cos\gamma + \frac{mv^2\cos\gamma}{R_E + y}\Big)
\end{aligned}
$$

What can go wrong...







- Reference trajectory = solution with lift/drag models
- Open-loop trajectory = actual trajectory resulting from reference controller

Motivation (cont.)

- By computing more high-fidelity lift and drag coefficients, can refine models to get a better solution, but this is costly

Motivation (cont.)

- By computing more high-fidelity lift and drag coefficients, can refine models to get a better solution, but this is costly

- Goal is not to get an accurate model, but one that yields an accurate trajectory with as few samples as possible

Motivation (cont.)

- By computing more high-fidelity lift and drag coefficients, can refine models to get a better solution, but this is costly
- Goal is not to get an accurate model, but one that yields an accurate trajectory with as few samples as possible
- Need to prioritize model refinement at most important locations

Motivation (cont.)

- By computing more high-fidelity lift and drag coefficients, can refine models to get a better solution, but this is costly
- Goal is not to get an accurate model, but one that yields an accurate trajectory with as few samples as possible
- Need to prioritize model refinement at most important locations
- Focus on simulation (ODEs) in this talk, but will be extended to trajectory optimization (OCPs)

Goals for this talk:

- Explain my adaptive model refinement procedure for trajectory simulation

- Demonstrate the method on a simple trajectory simulation problem for a nominal hypersonic vehicle

General formulation (ODE setting)

- Given functions

$$\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^{n_x} \times \mathbb{R}^{n_g}, \mathbb{R}^{n_x}), \qquad \mathbf{g} \in \mathcal{C}^1(\mathbb{R}^{n_x}, \mathbb{R}^{n_g})$$

consider the IVP

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}\Big(\mathbf{x}(t), \mathbf{g}\big(\mathbf{x}(t)\big)\Big), \qquad t \in (t_0, t_f) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \end{aligned}$$

parametrized by model function $\mathbf{g}$, where $\mathbf{f}$ is known

General formulation (ODE setting)

- Given functions

$$\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^{n_x} \times \mathbb{R}^{n_g}, \mathbb{R}^{n_x}), \qquad \mathbf{g} \in \mathcal{C}^1(\mathbb{R}^{n_x}, \mathbb{R}^{n_g})$$

  consider the IVP

$$\mathbf{x}'(t) = \mathbf{f}\Big(\mathbf{x}(t), \mathbf{g}\big(\mathbf{x}(t)\big)\Big), \qquad t \in (t_0, t_f)$$
$$\mathbf{x}(t_0) = \mathbf{x}_0$$

  parametrized by model function $\mathbf{g}$, where $\mathbf{f}$ is known

How would we factor in model uncertainity? Would that just change the GP distribution (Now we have a nontrivial confidence interval around the collocation points?)

- Assumptions:
  - High-fidelity model $\mathbf{g}_*(\cdot)$ expensive to compute
  - High-fidelity model is true model (disturbances are neglected)
  - For simplicity, $\mathbf{g}_*$ assumed scalar-valued; also assume no control

GP model

- Solving IVP directly with $\mathbf{g}_*(\cdot)$ too expensive
  (e.g., computing $\mathbf{g}_*(\cdot)$ at every timestep of Runge-Kutta)

GP model

- Solving IVP directly with $\mathbf{g}_*(\cdot)$ too expensive (e.g., computing $\mathbf{g}_*(\cdot)$ at every timestep of Runge-Kutta)

- Instead, construct cheaper Gaussian process model

$$\mathbf{g}_c(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_c(\cdot), \boldsymbol{\sigma}_c^2(\cdot)\big)$$

from high-fidelity samples at locations $X \subset \mathbb{R}^{n_x}$ with values $Y = \mathbf{g}_*(X) \subset \mathbb{R}^{n_g}$

GP model

- Solving IVP directly with $\mathbf{g}_*(\cdot)$ too expensive (e.g., computing $\mathbf{g}_*(\cdot)$ at every timestep of Runge-Kutta)

- Instead, construct cheaper Gaussian process model

$$\mathbf{g}_c(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_c(\cdot), \boldsymbol{\sigma}_c^2(\cdot)\big)$$

  from high-fidelity samples at locations $X \subset \mathbb{R}^{n_x}$ with values $Y = \mathbf{g}_*(X) \subset \mathbb{R}^{n_g}$

- For better trajectory, refine GP using additional sample:

$$\mathbf{g}_+(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_+(\cdot), \boldsymbol{\sigma}_+^2(\cdot)\big)$$

  computed from samples $\widetilde{X} = X \cup \{\widetilde{x}\}$, $\widetilde{Y} = \mathbf{g}_*(\widetilde{X})$

GP model

- Solving IVP directly with $\mathbf{g}_*(\cdot)$ too expensive (e.g., computing $\mathbf{g}_*(\cdot)$ at every timestep of Runge-Kutta)

- Instead, construct cheaper Gaussian process model

$$\mathbf{g}_c(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_c(\cdot), \boldsymbol{\sigma}_c^2(\cdot)\big)$$

  from high-fidelity samples at locations $X \subset \mathbb{R}^{n_x}$ with values $Y = \mathbf{g}_*(X) \subset \mathbb{R}^{n_g}$

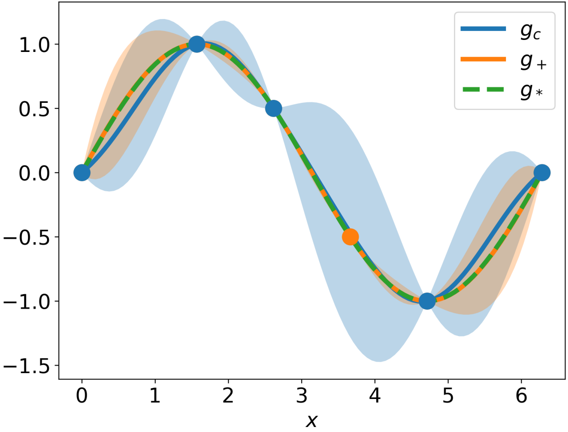
- For better trajectory, refine GP using additional sample:

$$\mathbf{g}_+(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_+(\cdot), \boldsymbol{\sigma}_+^2(\cdot)\big)$$

  computed from samples $\widetilde{X} = X \cup \{\widetilde{x}\}$, $\widetilde{Y} = \mathbf{g}_*(\widetilde{X})$

- Refinement requires one expensive $\mathbf{g}_*$ computation at $\widetilde{x}$; which $\widetilde{x}$ should we choose?

Visual example



GP posterior

And how is kernel determiend? Is there a way to select the "best" or "optimal" kernel?

- Posterior for initial/current GP model:

How is the prior determined?

$$\boldsymbol{\mu}_c(\cdot) = \boldsymbol{\mu}_0(\cdot) + \mathbf{k}_0(\cdot, X)\left(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\right)^{-1}\left(Y - \boldsymbol{\mu}_0(X)\right)$$

$$\boldsymbol{\sigma}_c^2(\cdot) = \mathbf{k}_0(\cdot, \cdot) - \mathbf{k}_0(\cdot, X)\left(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\right)^{-1}\mathbf{k}_0(X, \cdot)$$

GP posterior

- Posterior for initial/current GP model:

$$\boldsymbol{\mu}_c(\cdot) = \boldsymbol{\mu}_0(\cdot) + \mathbf{k}_0(\cdot, X)\big(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\big)^{-1}\big(Y - \boldsymbol{\mu}_0(X)\big)$$
$$\boldsymbol{\sigma}_c^2(\cdot) = \mathbf{k}_0(\cdot, \cdot) - \mathbf{k}_0(\cdot, X)\big(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\big)^{-1}\mathbf{k}_0(X, \cdot)$$

- Model will be refined by adding one new sample to X:

$$\boldsymbol{\mu}_+(\cdot) = \boldsymbol{\mu}_0(\cdot) + \mathbf{k}_0(\cdot, \widetilde{X})\big(\mathbf{k}_0(\widetilde{X}, \widetilde{X}) + \epsilon^2\mathbf{I}\big)^{-1}\big({\color{red}\widetilde{Y}} - \boldsymbol{\mu}_0(\widetilde{X})\big)$$
$$\boldsymbol{\sigma}_+^2(\cdot) = \mathbf{k}_0(\cdot, \cdot) - \mathbf{k}_0(\cdot, \widetilde{X})\big(\mathbf{k}_0(\widetilde{X}, \widetilde{X}) + \epsilon^2\mathbf{I}\big)^{-1}\mathbf{k}_0(\widetilde{X}, \cdot)$$

where $\widetilde{X} = X \cup \{\widetilde{x}\}$, ${\color{red}\widetilde{Y} = \mathbf{g}_*(\widetilde{X})}$

GP posterior

- Posterior for initial/current GP model:

$$\boldsymbol{\mu}_c(\cdot) = \boldsymbol{\mu}_0(\cdot) + \mathbf{k}_0(\cdot, X)\big(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\big)^{-1}\big(Y - \boldsymbol{\mu}_0(X)\big)$$
$$\boldsymbol{\sigma}_c^2(\cdot) = \mathbf{k}_0(\cdot, \cdot) - \mathbf{k}_0(\cdot, X)\big(\mathbf{k}_0(X, X) + \epsilon^2\mathbf{I}\big)^{-1}\mathbf{k}_0(X, \cdot)$$

- Model will be refined by adding one new sample to X:

$$\boldsymbol{\mu}_+(\cdot) = \boldsymbol{\mu}_0(\cdot) + \mathbf{k}_0(\cdot, \widetilde{X})\big(\mathbf{k}_0(\widetilde{X}, \widetilde{X}) + \epsilon^2\mathbf{I}\big)^{-1}\big(\widetilde{Y} - \boldsymbol{\mu}_0(\widetilde{X})\big)$$
$$\boldsymbol{\sigma}_+^2(\cdot) = \mathbf{k}_0(\cdot, \cdot) - \mathbf{k}_0(\cdot, \widetilde{X})\big(\mathbf{k}_0(\widetilde{X}, \widetilde{X}) + \epsilon^2\mathbf{I}\big)^{-1}\mathbf{k}_0(\widetilde{X}, \cdot)$$

where $\widetilde{X} = X \cup \{\widetilde{x}\}$, $\widetilde{Y} = \mathbf{g}_*(\widetilde{X})$

- Note: $\boldsymbol{\mu}_+(\cdot)$ requires computation of $\mathbf{g}_*(\widetilde{x})$, while $\boldsymbol{\sigma}_+^2(\cdot)$ does not; this will be important later

Reinterpreting the problem

- Directly plugging GP model $\mathbf{g}_c$ into IVP $\rightarrow$ unclear how to interpret, likely too expensive to solve

Reinterpreting the problem

- Directly plugging GP model $\mathbf{g}_c$ into IVP $\rightarrow$ unclear how to interpret, likely too expensive to solve
- To remain deterministic, solve IVP using GP mean as the model:

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}\Big(\mathbf{x}(t), \boldsymbol{\mu}_c\big(\mathbf{x}(t)\big)\Big), \qquad && t \in (t_0, t_f) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \end{aligned}$$

Reinterpreting the problem

- Directly plugging GP model $\mathbf{g}_c$ into IVP $\rightarrow$ unclear how to interpret, likely too expensive to solve

- To remain deterministic, solve IVP using GP mean as the model:

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}\Big(\mathbf{x}(t), \boldsymbol{\mu}_c\big(\mathbf{x}(t)\big)\Big), \qquad && t \in (t_0, t_f) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \end{aligned}$$

- My proposed approach: use sensitivity analysis and GP uncertainty to select "optimal" $\mathbf{g}_*$ sample and construct new GP

$$\mathbf{g}_+(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_+(\cdot), \boldsymbol{\sigma}_+^2(\cdot)\big)$$

  then solve IVP with new mean $\boldsymbol{\mu}_+$ to obtain better solution

Reinterpreting the problem

- Directly plugging GP model $\mathbf{g}_c$ into IVP $\rightarrow$ unclear how to interpret, likely too expensive to solve

- To remain deterministic, solve IVP using GP mean as the model:

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}\Big(\mathbf{x}(t), \boldsymbol{\mu}_c\big(\mathbf{x}(t)\big)\Big), \qquad && t \in (t_0, t_f) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \end{aligned}$$

- My proposed approach: use sensitivity analysis and GP uncertainty to select "optimal" $\mathbf{g}_*$ sample and construct new GP

$$\mathbf{g}_+(\cdot) \sim \mathcal{GP}\big(\boldsymbol{\mu}_+(\cdot), \boldsymbol{\sigma}_+^2(\cdot)\big)$$

  then solve IVP with new mean $\boldsymbol{\mu}_+$ to obtain better solution

- This uses both the mean and variance of the GP, taking full advantage of its features

Hypersonic IVP

- Goal: solve

$$
\begin{aligned}
x' &= v\cos\gamma \\
y' &= v\sin\gamma \\
v' &= -\frac{1}{m}\Big(D(y,v,\alpha) + mg(y)\sin\gamma\Big) \\
\gamma' &= \frac{1}{mv}\Big(L(y,v,\alpha) - mg(y)\cos\gamma + \frac{mv^2\cos\gamma}{R_E + y}\Big)
\end{aligned}
$$

$$x(0) = 0 \text{ km}, \quad y(0) = 80 \text{ km}, \quad v(0) = 5 \text{ km/s}, \quad \gamma(0) = -5^\circ$$

with test control

$$\alpha(t) = \begin{cases} 20^\circ, & t < 70 \\ 8^\circ, & t \geq 70 \end{cases}$$

Hypersonic IVP

- Goal: solve

$$\begin{aligned}
x' &= v\cos\gamma \\
y' &= v\sin\gamma \\
v' &= -\frac{1}{m}\Big(D(y,v,\alpha) + mg(y)\sin\gamma\Big) \\
\gamma' &= \frac{1}{mv}\Big(L(y,v,\alpha) - mg(y)\cos\gamma + \frac{mv^2\cos\gamma}{R_E + y}\Big)
\end{aligned}$$

$$x(0) = 0\text{ km},\quad y(0) = 80\text{ km},\quad v(0) = 5\text{ km/s},\quad \gamma(0) = -5^\circ$$

  with test control

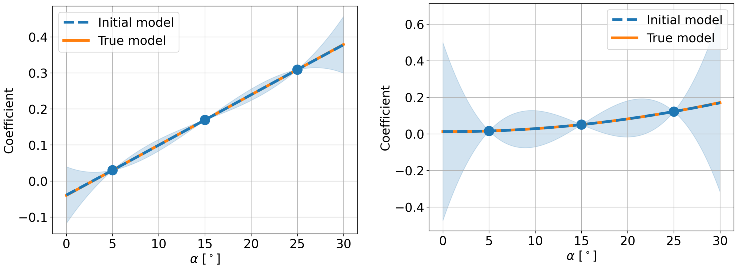
$$\alpha(t) = \begin{cases} 20^\circ, & t < 70 \\ 8^\circ, & t \geq 70 \end{cases}$$

- Lift and drag coefficients expensive to compute in high fidelity; model using GPs

$$L(y,v,\alpha) = q(y,v)c_L(\alpha)A_w, \qquad D(y,v,\alpha) = q(y,v)c_D(\alpha)A_w$$

Aero coefficients

- Initial models constructed as GPs using samples at $\alpha = 5°, 15°, 25°$

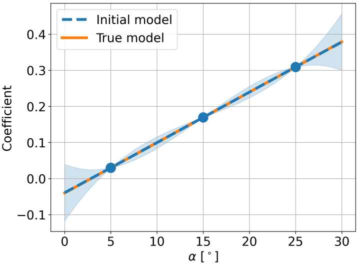


Lift

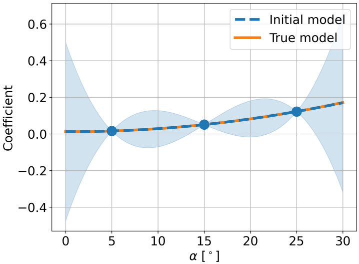
Drag

Aero coefficients

- Initial models constructed as GPs using samples at $\alpha = 5°, 15°, 25°$



Lift



Drag

- Goal: find the best $\widetilde{\alpha} \in \{0°, 10°, 20°, 30°\}$ at which to compute c_L and c_D in high fidelity to improve trajectory

Sensitivity analysis

- Solution of IVP depends on model $\mathbf{g}$; denote $\mathbf{x}(\mathbf{g})$ for $\mathbf{g} \in \mathcal{C}^1(\mathbb{R}^{n_x}, \mathbb{R}^{n_g})$, and let $\mathbf{x}_c := \mathbf{x}(\boldsymbol{\mu}_c)$

Sensitivity analysis

- Solution of IVP depends on model $\mathbf{g}$; denote $\mathbf{x}(\mathbf{g})$ for $\mathbf{g} \in \mathcal{C}^1(\mathbb{R}^{n_x}, \mathbb{R}^{n_g})$, and let $\mathbf{x}_c := \mathbf{x}(\boldsymbol{\mu}_c)$
- How sensitive is $\mathbf{x}_c$ to perturbations in $\mathbf{g}$?

Sensitivity analysis

How is a change delta g defined for a continuous function?

- Solution of IVP depends on model $\mathbf{g}$; denote $\mathbf{x}(\mathbf{g})$ for $\mathbf{g} \in \mathcal{C}^1(\mathbb{R}^{n_x}, \mathbb{R}^{n_g})$, and let $\mathbf{x}_c := \mathbf{x}(\boldsymbol{\mu}_c)$
- How sensitive is $\mathbf{x}_c$ to perturbations in $\mathbf{g}$?
- Using implicit function theorem, can obtain

So this is a measure of how sensitive
the solution x is to fluctuations in g at time t?

$$\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_c)\delta\mathbf{g} =: \mathbf{S}(t)$$

Is this a derivative with respect to the function g? how is that defined?

Is this a Frechet derivative

which solves the IVP

$$\begin{aligned} \mathbf{S}'(t) &= \mathbf{A}(t)\mathbf{S}(t) + \mathbf{B}(t)\delta\mathbf{g}\big(\mathbf{x}_c(t)\big), \qquad t \in (t_0, t_f), \\ \mathbf{S}(t_0) &= \mathbf{0}, \end{aligned}$$

where

$$\begin{aligned} \mathbf{A}(t) &= \mathbf{f}_x[t] + \mathbf{f}_g[t]\boldsymbol{\mu}_c'\big(\mathbf{x}_c(t)\big), \quad && \mathbf{B}(t) = \mathbf{f}_g[t], \\ \mathbf{f}_x[t] &:= \mathbf{f}_x\Big(\mathbf{x}_c(t), \boldsymbol{\mu}_c\big(\mathbf{x}_c(t)\big)\Big), \quad && \mathbf{f}_g[t] := \mathbf{f}_g\Big(\mathbf{x}_c(t), \boldsymbol{\mu}_c\big(\mathbf{x}_c(t)\big)\Big) \end{aligned}$$

GP variance heuristic

- Approximate solution error by applying sensitivity operator in direction of model error:

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_c; t) \approx \left[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_c)(\mathbf{g}_* - \boldsymbol{\mu}_c)\right](t) =: \mathbf{S}_c(t), \quad t \in [t_0, t_f]$$

So is this basically a linear approximation?

GP variance heuristic

- Approximate solution error by applying sensitivity operator in direction of model error:

$$\mathbf{x}(\mathbf{g}_*;t) - \mathbf{x}(\boldsymbol{\mu}_c;t) \approx \big[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_c)(\mathbf{g}_* - \boldsymbol{\mu}_c)\big](t) =: \mathbf{S}_c(t), \quad t \in [t_0, t_f]$$

- Problem: computing $\mathbf{S}_c(\cdot)$ requires solving IVP involving $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)$; still prohibitively expensive

GP variance heuristic

- Approximate solution error by applying sensitivity operator in direction of model error:

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_c; t) \approx \left[\mathbf{x}_\mathbf{g}(\boldsymbol{\mu}_c)(\mathbf{g}_* - \boldsymbol{\mu}_c)\right](t) =: \mathbf{S}_c(t), \quad t \in [t_0, t_f]$$

- Problem: computing $\mathbf{S}_c(\cdot)$ requires solving IVP involving $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)$; still prohibitively expensive

This is an unbiased estimator for the variance of the GP, if we interpret g* as a random var. from GP? Since var is actually the expected value of (X-mu)^2 where X is the space of all functions in the GP space?

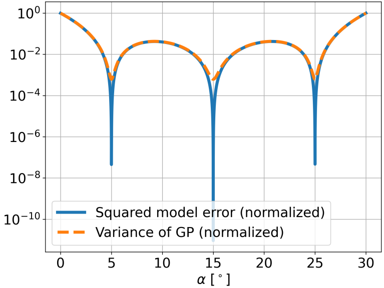
- Idea: approximate (up to a constant)

$$\boldsymbol{\sigma}_c^2(\cdot) \approx \left(\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)\right)^2$$

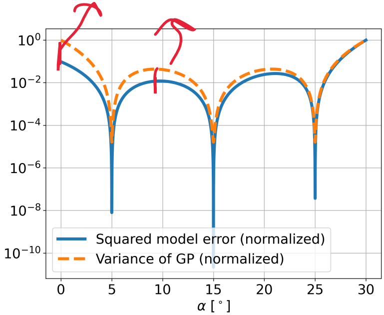
- Intuition: interpret $\mathbf{g}_*(x)$ as realization of Gaussian RV $\mathbf{g}_c(x)$ with mean $\boldsymbol{\mu}_c(x)$ and variance $\boldsymbol{\sigma}_c^2(x)$

GP variance heuristic

What creates the bias here? Is it possible to improve the heuristic? Why/why not?



Lift



Drag

- This heuristic works really well!

Bounding the solution error estimate

- Postulate that $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)$ lies between $-\boldsymbol{\sigma}_c(\cdot)$ and $\boldsymbol{\sigma}_c(\cdot)$, motivated by GP variance heuristic

$$\boldsymbol{\sigma}_c^2(\cdot) \approx \big(\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)\big)^2$$

Bounding the solution error estimate

- Postulate that $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)$ lies between $-\boldsymbol{\sigma}_c(\cdot)$ and $\boldsymbol{\sigma}_c(\cdot)$, motivated by GP variance heuristic

$$\boldsymbol{\sigma}_c^2(\cdot) \approx \left(\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)\right)^2$$

- Given this assumption, can bound solution error estimate $\mathbf{S}_c(\cdot)$ by solution of linear quadratic optimal control problem

$$\begin{aligned}
\max_{\mathbf{s}(\cdot),\boldsymbol{\delta}(\cdot)} \quad & \int_{t_0}^{t_f} \mathbf{s}(t)^T \mathbf{s}(t)\, dt \\
\text{s.t.} \quad & \mathbf{s}'(t) = \mathbf{A}(t)\mathbf{s}(t) + \mathbf{B}(t)\boldsymbol{\delta}(t), && t \in (t_0, t_f) \\
& \mathbf{s}(t_0) = \mathbf{0} \\
& -\boldsymbol{\sigma}_c\big(\mathbf{x}_c(t)\big) \le \boldsymbol{\delta}(t) \le \boldsymbol{\sigma}_c\big(\mathbf{x}_c(t)\big), && t \in [t_0, t_f]
\end{aligned}$$

Is this the right idea of whats happening here: we're trying to find the model error function delta(t) that results in the smallest possible max error in the solution to the IVP (minmax), and we are assuming that this delta is bounded within 1 SD everywhere, based o the previously derived heuristic

How valid is the 1 SD assumption? Is there a way to show that this is a reasonable assumption?

Bounding the solution error estimate

- Postulate that $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)$ lies between $-\boldsymbol{\sigma}_c(\cdot)$ and $\boldsymbol{\sigma}_c(\cdot)$, motivated by GP variance heuristic

$$\boldsymbol{\sigma}_c^2(\cdot) \approx \big(\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_c(\cdot)\big)^2$$

- Given this assumption, can bound solution error estimate $\mathbf{S}_c(\cdot)$ by solution of linear quadratic optimal control problem

$$\begin{aligned}
\max_{\mathbf{s}(\cdot),\boldsymbol{\delta}(\cdot)} \quad & \int_{t_0}^{t_f} \mathbf{s}(t)^T \mathbf{s}(t)\, dt \\
\text{s.t.} \quad & \mathbf{s}'(t) = \mathbf{A}(t)\mathbf{s}(t) + \mathbf{B}(t)\boldsymbol{\delta}(t), && t \in (t_0, t_f) \\
& \mathbf{s}(t_0) = \mathbf{0} \\
& -\boldsymbol{\sigma}_c\big(\mathbf{x}_c(t)\big) \le \boldsymbol{\delta}(t) \le \boldsymbol{\sigma}_c\big(\mathbf{x}_c(t)\big), && t \in [t_0, t_f]
\end{aligned}$$

- Allows to estimate solution error, but also provides means of selecting new samples (discussed next)

Selecting the new sample

- Approximate solution error with new model as

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_+; t) \approx \left[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t), \qquad t \in [t_0, t_f]$$

Selecting the new sample

- Approximate solution error with new model as

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_+; t) \approx \left[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t), \qquad t \in [t_0, t_f]$$

- Want to choose sample $\widetilde{x}$ that minimizes this error and then compute $\mathbf{g}_*(\widetilde{x})$ to obtain new model

Selecting the new sample

- Approximate solution error with new model as

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_+; t) \approx \left[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t), \qquad t \in [t_0, t_f]$$

- Want to choose sample $\widetilde{x}$ that minimizes this error and then compute $\mathbf{g}_*(\widetilde{x})$ to obtain new model
- Recall: can't compute $\boldsymbol{\mu}_+(\cdot)$ without computing $\mathbf{g}_*(\widetilde{x})$ (putting the cart before the horse)

Selecting the new sample

- Approximate solution error with new model as

$$\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_+; t) \approx \big[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\big](t), \qquad t \in [t_0, t_f]$$

- Want to choose sample $\widetilde{x}$ that minimizes this error and then compute $\mathbf{g}_*(\widetilde{x})$ to obtain new model
- Recall: can't compute $\boldsymbol{\mu}_+(\cdot)$ without computing $\mathbf{g}_*(\widetilde{x})$ (putting the cart before the horse)
- Since $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_c$ differ by only one sample, approximate

$$\begin{aligned}\mathbf{x}(\mathbf{g}_*; t) - \mathbf{x}(\boldsymbol{\mu}_+; t) &\approx \big[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\big](t) \\ &\approx \big[\mathbf{x}_{\mathbf{g}}(\boldsymbol{\mu}_c)(\mathbf{g}_* - \boldsymbol{\mu}_+)\big](t) =: \mathbf{S}_+(t), \quad t \in [t_0, t_f]\end{aligned}$$

Selecting the new sample

- Approximate solution error with new model as

$$\mathbf{x}(\mathbf{g}_*;t) - \mathbf{x}(\boldsymbol{\mu}_+;t) \approx \left[\mathbf{x}_\mathbf{g}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t), \qquad t \in [t_0, t_f]$$

- Want to choose sample $\widetilde{x}$ that minimizes this error and then compute $\mathbf{g}_*(\widetilde{x})$ to obtain new model

- Recall: can't compute $\boldsymbol{\mu}_+(\cdot)$ without computing $\mathbf{g}_*(\widetilde{x})$ (putting the cart before the horse)

- Since $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_c$ differ by only one sample, approximate

$$\begin{aligned}\mathbf{x}(\mathbf{g}_*;t) - \mathbf{x}(\boldsymbol{\mu}_+;t) &\approx \left[\mathbf{x}_\mathbf{g}(\boldsymbol{\mu}_+)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t) \\ &\approx \left[\mathbf{x}_\mathbf{g}(\boldsymbol{\mu}_c)(\mathbf{g}_* - \boldsymbol{\mu}_+)\right](t) =: \mathbf{S}_+(t), \quad t \in [t_0, t_f]\end{aligned}$$

So were assuming the derivative of the solution x with respect to variations in g stays basically the same with the addition of one more collocation point. But, aren't we trying to explicitly choose new point x* to minimize this sensitivity, so isn't it smaller?

- Same sensitivity operator as before, but applied in direction of new model error

Is there a way to do a 2nd order approximation here?

Selecting the new sample

- Postulate that new model error $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_+(\cdot)$ lies between $-\boldsymbol{\sigma}_+(\cdot)$ and $\boldsymbol{\sigma}_+(\cdot)$

Selecting the new sample

- Postulate that new model error $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_+(\cdot)$ lies between $-\boldsymbol{\sigma}_+(\cdot)$ and $\boldsymbol{\sigma}_+(\cdot)$
- Remember that $\boldsymbol{\sigma}_+^2(\cdot)$ does not require computation of $\mathbf{g}_*(\widetilde{x})$; can compute $\boldsymbol{\sigma}_+^2(\cdot)$ before refining model!

Selecting the new sample

- Postulate that new model error $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_+(\cdot)$ lies between $-\boldsymbol{\sigma}_+(\cdot)$ and $\boldsymbol{\sigma}_+(\cdot)$

If we are assuming g* works like a random var. selected from the GP function space, then this is only a 68% probability right? Would changing this interval (to, say, 2 sigma), change the heuristic in significant ways? Or would the LQOCP minmax problem still result in roughly the same x+ point?

- Remember that $\boldsymbol{\sigma}_+^2(\cdot)$ does not require computation of $\mathbf{g}_*(\widetilde{x})$; can compute $\boldsymbol{\sigma}_+^2(\cdot)$ before refining model!

- Can bound new solution error estimate $\mathbf{S}_+(\cdot)$ by solving LQOCP

$$\begin{aligned}
\max_{\mathbf{s}(\cdot),\boldsymbol{\delta}(\cdot)} \quad & \int_{t_0}^{t_f} \mathbf{s}(t)^T \mathbf{s}(t)\, dt \\
\text{s.t.} \quad & \mathbf{s}'(t) = \mathbf{A}(t)\mathbf{s}(t) + \mathbf{B}(t)\boldsymbol{\delta}(t), && t \in (t_0, t_f) \\
& \mathbf{s}(t_0) = \mathbf{0} \\
& -\boldsymbol{\sigma}_+\big(\mathbf{x}_c(t)\big) \le \boldsymbol{\delta}(t) \le \boldsymbol{\sigma}_+\big(\mathbf{x}_c(t)\big), && t \in [t_0, t_f]
\end{aligned}$$

Selecting the new sample

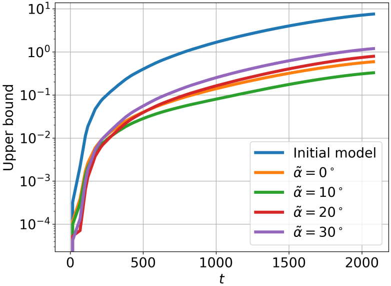
- Postulate that new model error $\mathbf{g}_*(\cdot) - \boldsymbol{\mu}_+(\cdot)$ lies between $-\boldsymbol{\sigma}_+(\cdot)$ and $\boldsymbol{\sigma}_+(\cdot)$
- Remember that $\boldsymbol{\sigma}_+^2(\cdot)$ does not require computation of $\mathbf{g}_*(\widetilde{x})$; can compute $\boldsymbol{\sigma}_+^2(\cdot)$ before refining model!
- Can bound new solution error estimate $\mathbf{S}_+(\cdot)$ by solving LQOCP

$$\begin{aligned}
\max_{\mathbf{s}(\cdot),\boldsymbol{\delta}(\cdot)} \quad & \int_{t_0}^{t_f} \mathbf{s}(t)^T \mathbf{s}(t)\, dt \\
\text{s.t.} \quad & \mathbf{s}'(t) = \mathbf{A}(t)\mathbf{s}(t) + \mathbf{B}(t)\boldsymbol{\delta}(t), && t \in (t_0, t_f) \\
& \mathbf{s}(t_0) = \mathbf{0} \\
& -\boldsymbol{\sigma}_+\big(\mathbf{x}_c(t)\big) \leq \boldsymbol{\delta}(t) \leq \boldsymbol{\sigma}_+\big(\mathbf{x}_c(t)\big), && t \in [t_0, t_f]
\end{aligned}$$

What method do you actually implement to solve?

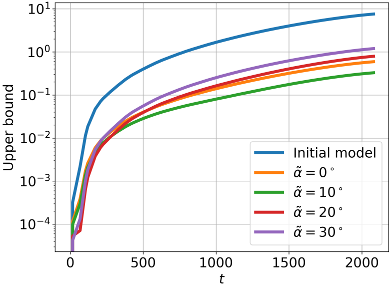
- Select $\widetilde{x}$ that minimizes this bound (minmax problem) as new sample

Model refinement for hypersonic ODE

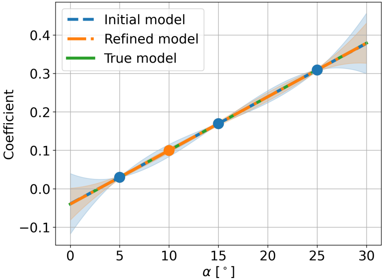


- Sensitivity bound is smallest for $\widetilde{\alpha} = 10^\circ$, so model gets refined there
- This makes sense, as angle of attack is 8° for most of trajectory

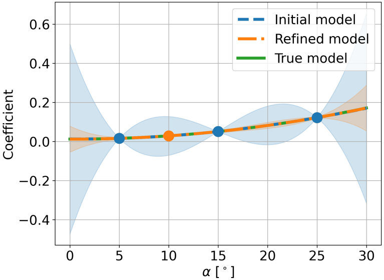
Model refinement for hypersonic ODE



- Sensitivity bound is smallest for $\widetilde{\alpha} = 10^\circ$, so model gets refined there
- This makes sense, as angle of attack is 8° for most of trajectory



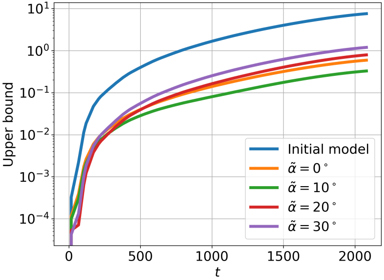
Lift



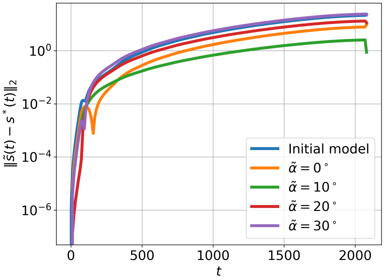
Drag

Trajectory errors

- Sensitivity curves obtained by solving LQOCP for each $\boldsymbol{\sigma}_+$
- Trajectory errors obtained by refining lift/drag at $\widetilde{\alpha}$ and solving original IVP with new GP mean $\boldsymbol{\mu}_+$



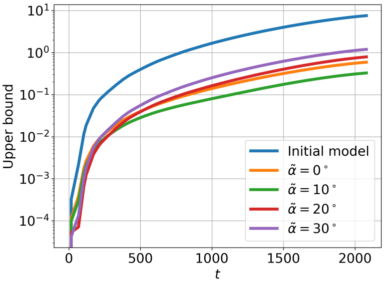
Sensitivity curves



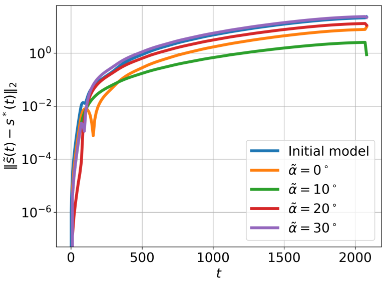
Trajectory errors

Trajectory errors

- Sensitivity curves obtained by solving LQOCP for each $\boldsymbol{\sigma}_+$
- Trajectory errors obtained by refining lift/drag at $\widetilde{\alpha}$ and solving original IVP with new GP mean $\boldsymbol{\mu}_+$



Sensitivity curves



Trajectory errors

- Chosen refinement point ($\widetilde{\alpha} = 10^{\circ}$) minimizes solution error!
- Better yet, sensitivity curves closely resemble actual trajectory errors!

Conclusions and Future Work

- Model refinement procedure combines sensitivities and model error heuristics to select new samples
- Performs well on hypersonic ODE with manufactured lift/drag models

Conclusions and Future Work

- Model refinement procedure combines sensitivities and model error heuristics to select new samples
- Performs well on hypersonic ODE with manufactured lift/drag models
- Ongoing work: extend this model refinement approach to trajectory optimization, not just simulation; main difference is computation of sensitivities
- Can also obtain different samples for different GPs