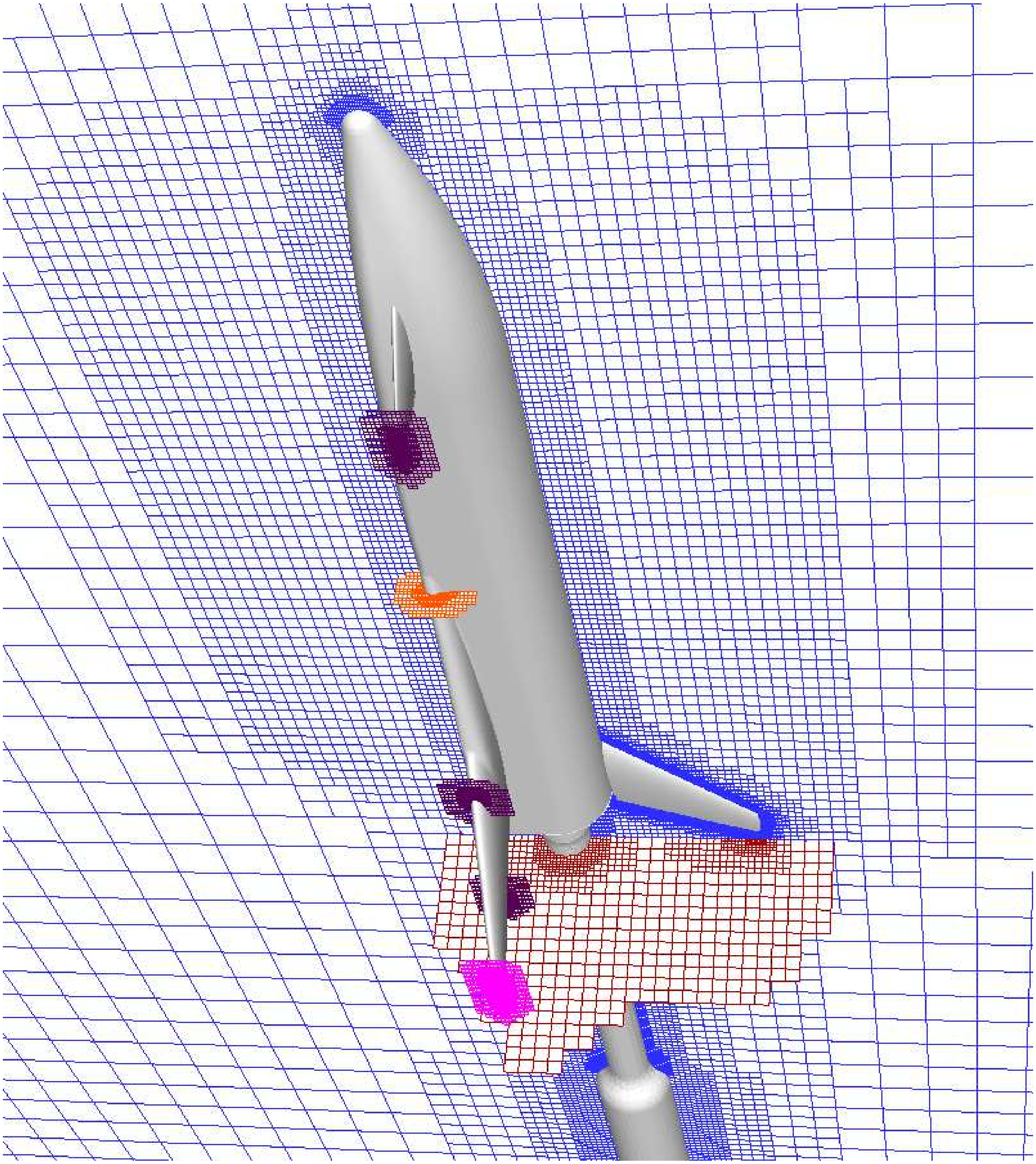
Adaptive Parameter Space Exploration and Nonstationary Modeling Using Gaussian Process Trees

1. Motivating Example
2. Gaussian Process Models
3. Bayesian Treed GP Models
4. Adaptive Sampling
5. Results

Largely from the Ph.D. thesis work of **Robert Gramacy** (UCSC)
in collaboration with **William MacReady** (formerly NASA Ames)



Goal

The primary goal is to develop a **response surface** for a variety of flight conditions.

- Running a standard experiment is infeasible
- Wind tunnel experiments are expensive
- Computing is relatively cheap
- Mathematical sophistication is increasing

Thus a **computer experiment** is used

Rocket Booster Simulations

Inputs

- speed (Mach number)
- angle of attack (alpha)
- side slip angle (beta)

Outputs

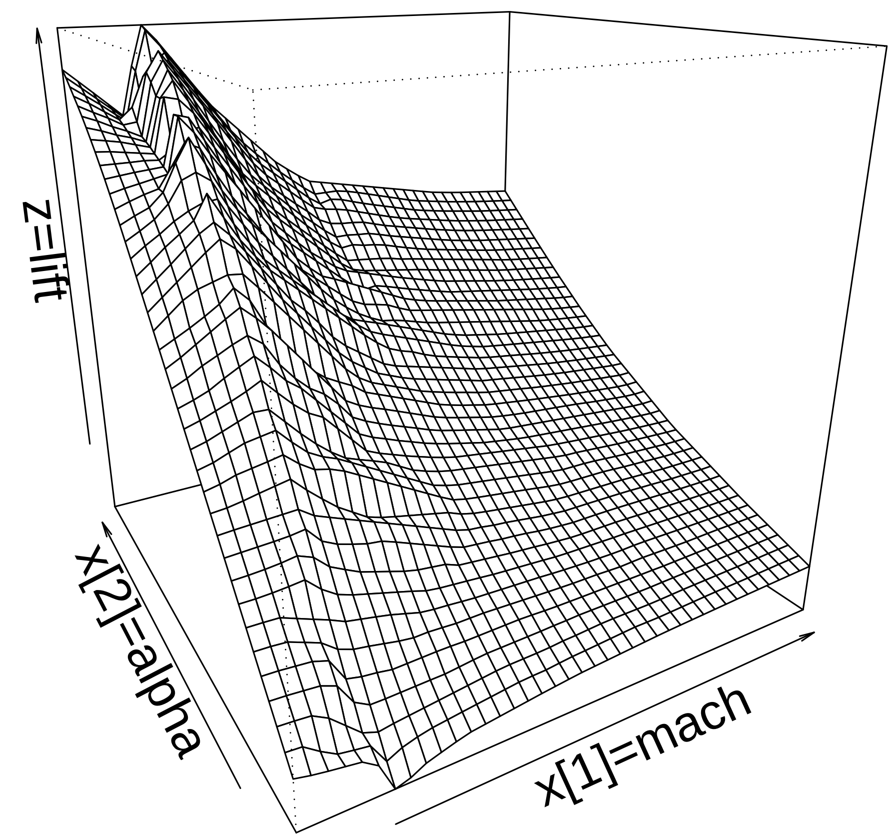
- lift
- drag
- pitch
- side-force
- yaw
- roll

Computational Fluid Dynamics

- Solves a system of differential equations over a fine mesh
- Systems are different for sub-sonic and super-sonic flight
- Takes 5-20 hours on a supercomputer
- May not always converge exactly

Old approach: run over a **full grid** of possible input values, then use the output to fit a response surface.

Estimated Surface



Our Approach

Combine **modeling** and **sequential experimental design**.

- Start with an initial small run
- Fit a model and estimate predictive uncertainty
- Choose new run locations based on uncertainty
- Attempt to put more effort in "interesting" regions
- Bayesian approach allows full uncertainty estimation, which is needed for the design steps

Gaussian Process Models

Traditional approach to modeling computer experiment output is a **Gaussian Process** (**GP**) (Sacks et al., 1989; Santner et al., 2003).

- Nonparametric spatial model
- Fit linear or polynomial trends with spatially correlated deviations
- Correlation between two points depends on their distance
- Result is a smooth yet flexible surface

Gaussian Processes

$$\mathbf{Z}(\mathbf{x}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{W}(\mathbf{x})$$

$\mathbf{Z}$ model outputs

$\mathbf{x}$ an arbitrary input value

$\mathbf{X}$ model inputs at all currently known data points

$\boldsymbol{\beta}$ linear trend coefficients

$\mathbf{W}$ mean zero spatial process

Estimation and Prediction

For observed outputs $\mathbf{t}$, the predictive density at a new point $\mathbf{x}$ is Gaussian with

$$\begin{aligned} \text{mean} \qquad \hat{y}(\mathbf{x}) &= \mathbf{k}^\top(\mathbf{x})\mathbf{K}^{-1}\mathbf{t}, \quad \text{and} \\ \text{variance} \qquad \sigma^2_{\hat{y}}(\mathbf{x}) &= \sigma^2[\mathbf{K}(\mathbf{x},\mathbf{x}) - \mathbf{k}^\top(\mathbf{x})\mathbf{K}^{-1}\mathbf{k}^\top(\mathbf{x})], \end{aligned}$$

where $\mathbf{K} = \frac{1}{\sigma^2}\mathbf{C}$ and $\mathbf{k}^\top(\mathbf{x})$ has elements $\mathbf{K}(\mathbf{x}, \mathbf{x}_i)$

Drawbacks

- Scales poorly — matrix inversions are $O(n^3)$
- Strictly stationary — our data aren't
- Predictive error depends only on aggregate (not nearby) previously observed responses
- Model assumed to be known

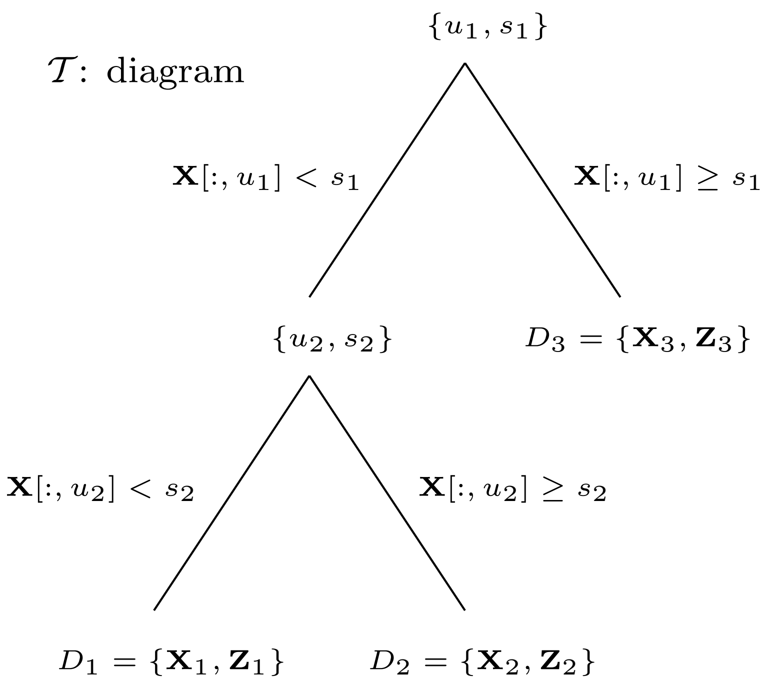
Solution: Partitioning

- Use a binary tree structure to **recursively partition** the space
  - Allow multiple splits per variable
- Fit a separate GP on each partition
- Fitting of tree structure and GPs is done simultaneously through MCMC
- Extension of partitioned linear regression model (Chipman et al., 2002)
- Nonstationarity achieved through partitioning

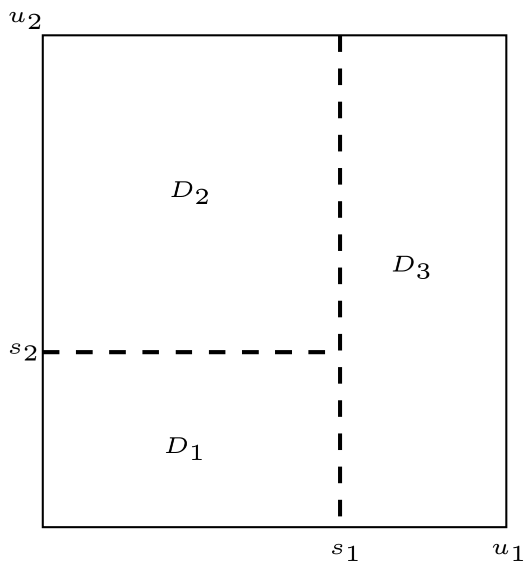
Tree Example

How a tree $\mathcal{T}$ recursively partitions the input space:

$\mathcal{T}$: diagram



$\mathcal{T}$: graphically



Hierarchical Model

Conditioning on tree $\mathcal{T}$, in each of the R regions: $\{r_\nu\}_{\nu=1}^R$

$$\mathbf{Z}_\nu | \boldsymbol{\beta}_\nu, \sigma_\nu^2, \mathbf{K}_\nu \sim N(\mathbf{F}_\nu \boldsymbol{\beta}_\nu, \sigma_\nu^2 \mathbf{K}_\nu) \qquad \sigma_\nu^2 \sim IG(\alpha_\sigma/2, q_\sigma/2)$$

$$\boldsymbol{\beta}_\nu | \sigma_\nu^2, \tau_\nu^2, \mathbf{W}, \boldsymbol{\beta}_0 \sim N(\boldsymbol{\beta}_0, \sigma_\nu^2 \tau_\nu^2 \mathbf{W}) \qquad \tau_\nu^2 \sim IG(\alpha_\tau/2, q_\tau/2)$$

$$\boldsymbol{\beta}_0 \sim N(\boldsymbol{\mu}, \mathbf{B}) \qquad \mathbf{W}^{-1} \sim W((\rho \mathbf{V})^{-1}, \rho)$$

with $\mathbf{F}_\nu = (\mathbf{1}, \mathbf{X}_\nu)$, and $\mathbf{W}$ is a $(m_X + 1) \times (m_X + 1)$ matrix, and

$$K_\nu(\mathbf{x}_j, \mathbf{x}_k) = K_\nu^*(\mathbf{x}_j, \mathbf{x}_k) + g_\nu \delta_{j,k}$$

for nugget g and *true* correlation K^* is separable:

$$K_\nu^*(\mathbf{x}_j, \mathbf{x}_k | \mathbf{d}_\nu) = \exp\{-\textstyle\sum_{i=1}^{m_X} |x_{ij} - x_{ik}|^{p_0} / d_{i\nu}\}$$

Mixture gamma priors on $\mathbf{d}_\nu$ and g_ν encode non-stationarity

Fit with MCMC

Sample from the **joint posterior** of $(\mathcal{T}, \boldsymbol{\theta})$

(Richardson & Green, 1997; Chipman et al., 2002)

- Tree process prior for node η at depth $q_\eta \in \mathcal{T}$:

$$p_{\mathrm{SPLIT}}(\eta, \mathcal{T}) = a(1 + q_\eta)^{-b}$$

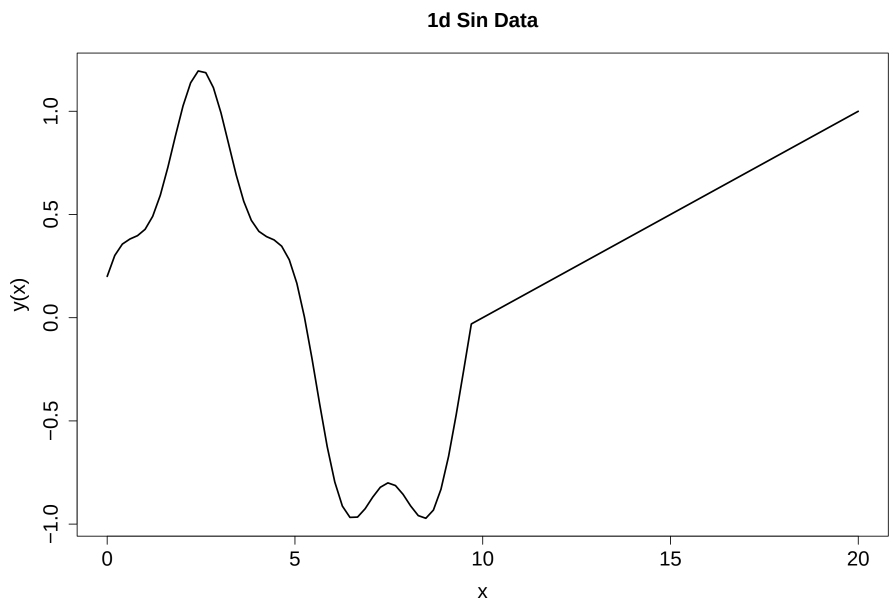
- Average over $\mathcal{T}$ with **reversible-jump MCMC** (RJ-MCMC)
- Tree operations: **grow**, **prune**, **change**, **swap**, and **rotate**

Gibbs sampling for all parameters except the correlation, which requires Metropolis-Hastings

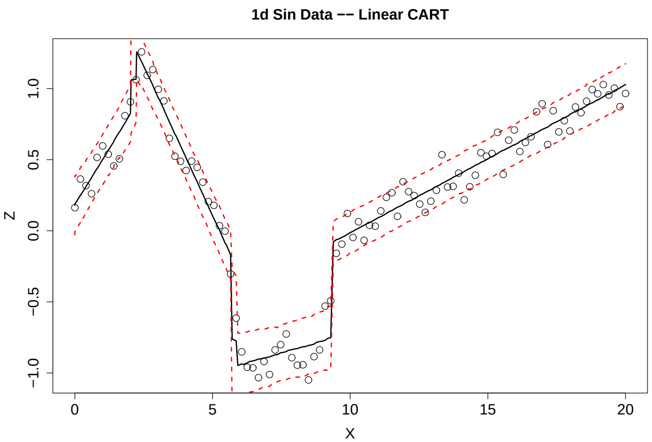
Simulated Example

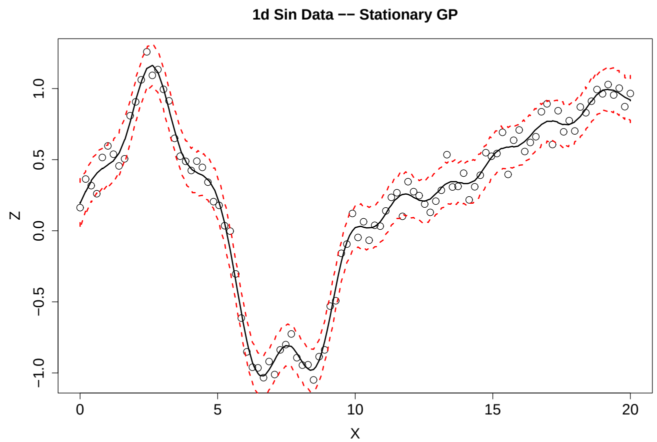
$$y(x) = \begin{cases} \sin\left(\frac{\pi x}{5}\right) + \frac{1}{5}\cos\left(\frac{4\pi x}{5}\right) & x < 10 \\ x/10 - 1 & \text{otherwise} \end{cases}$$

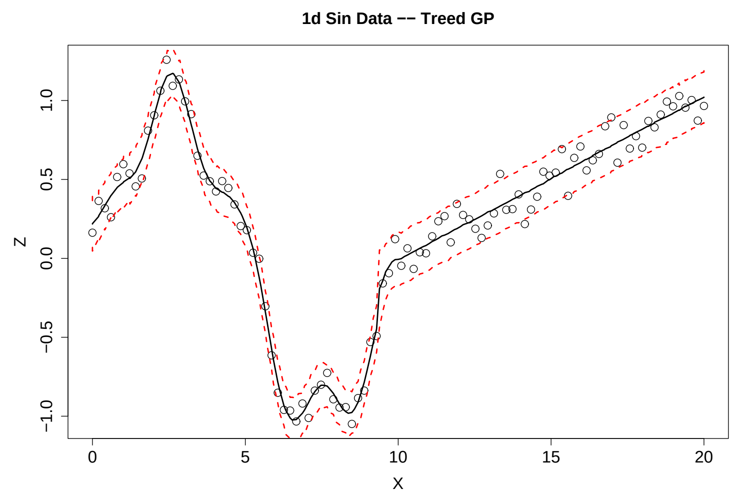
with $N(0, 0.1^2)$ independent noise added



Simulated Example Results





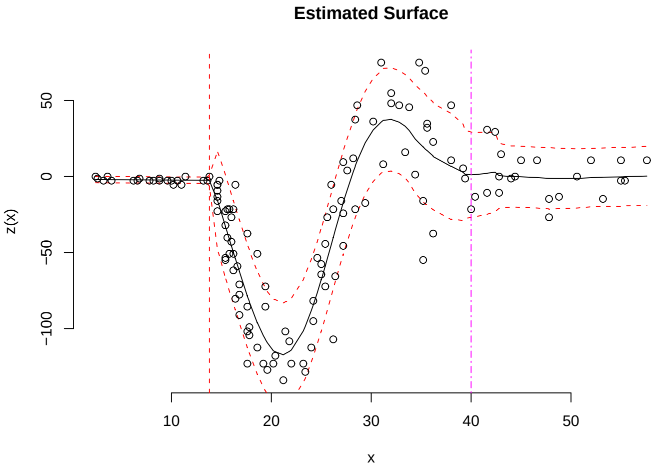


Motorcycle Data

(Silverman, 1985)

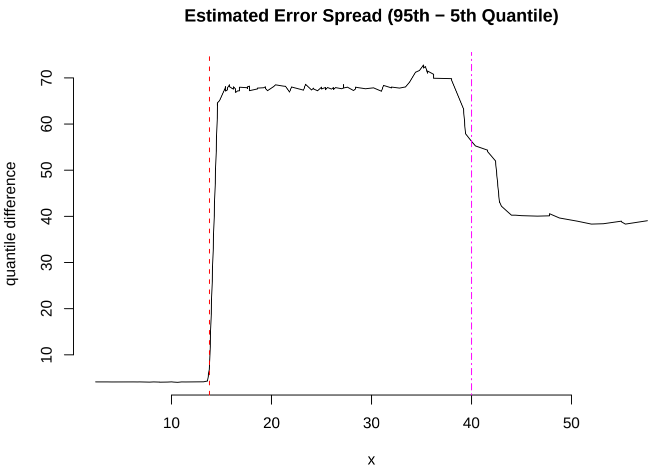
Data features:

- non-stationary
- input-dependent noise
- popular in recent literature (Rasmussen et al., 2002) DP-mix of GP (DPGP)



Treed GP model:

- ~ 3 regions
- notice step-wise transition from middle to right region
- $> 10x$ faster than DPGP



Adaptive Sampling

Active Learning / Sequential Design of Experiments

- select future design sites to improve our knowledge (model)
- maximize some measure of **utility**
  - quadratic loss on parameter estimates (A-optimality)
  - Kullback-Leibler distance between posterior and prior (D-optimality)
  - Kullback-Leibler distance between posterior predictive and prior predictive — equivalent to **minimizing predictive variance**
- points may need to be chosen in **batches**

Issues

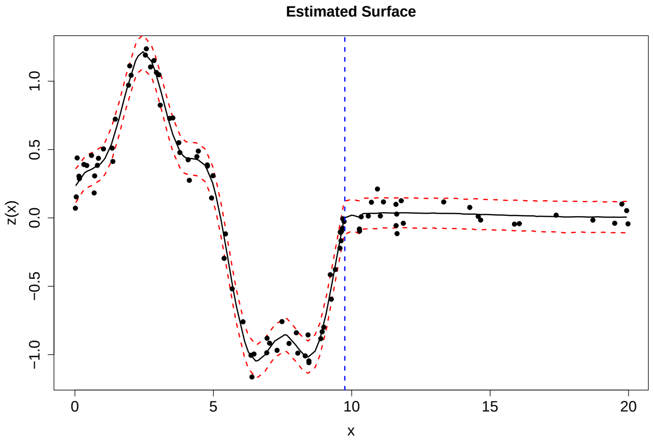
- Design approaches in the literature assume the model and its parameters are known
- Standard designs tend to push points to the boundaries
- Need to select a list of multiple points
- Need to deal with pending data (experiments currently being run)

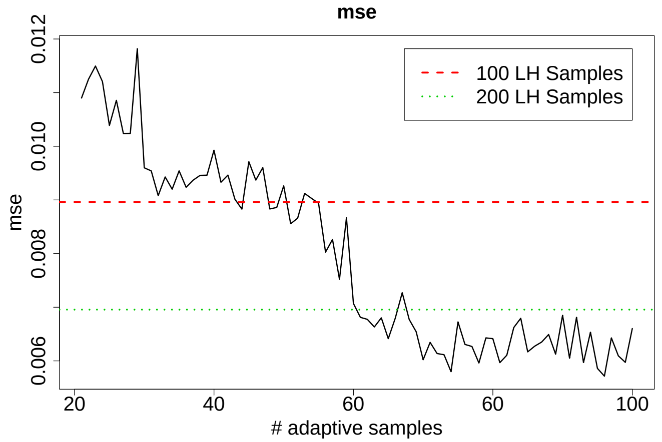
Our Approach

- Select a set of spatially spread candidate points using a Latin Hypercube or D-optimal design
- Prioritize the list based on estimated predictive variance
- Predictive variance estimated with MCMC
- Use currently known data plus fitted values at pending locations
- Iterate as new data becomes available

Adaptive Sampling Demo

$$z(x) = \begin{cases} \sin\left(\frac{\pi x}{5}\right) + \frac{1}{5}\cos\left(\frac{4\pi x}{5}\right) & x < 10 \\ 0 & \text{otherwise} \end{cases}$$



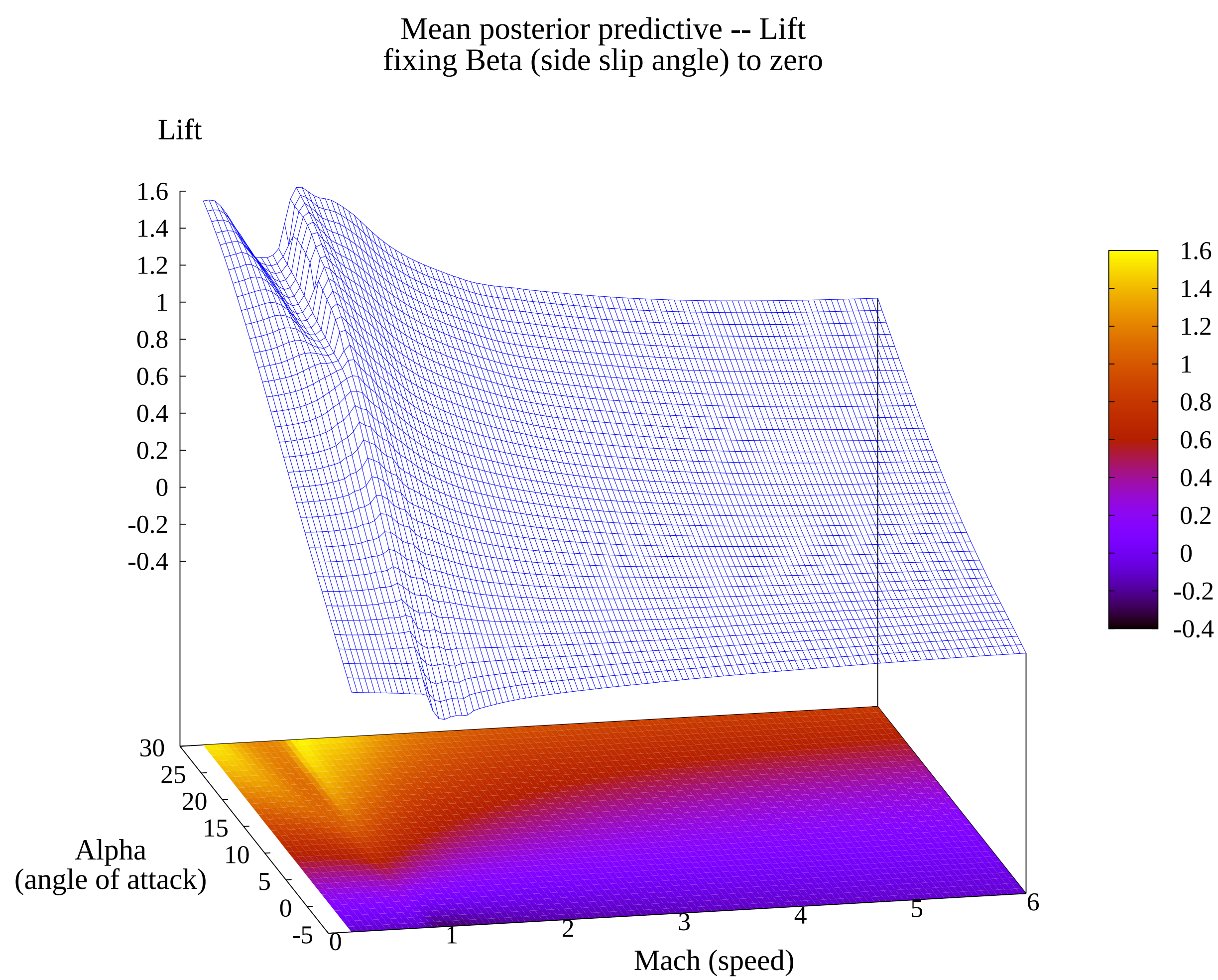


- MSE comparison to LH designs with **100** and **200** samples

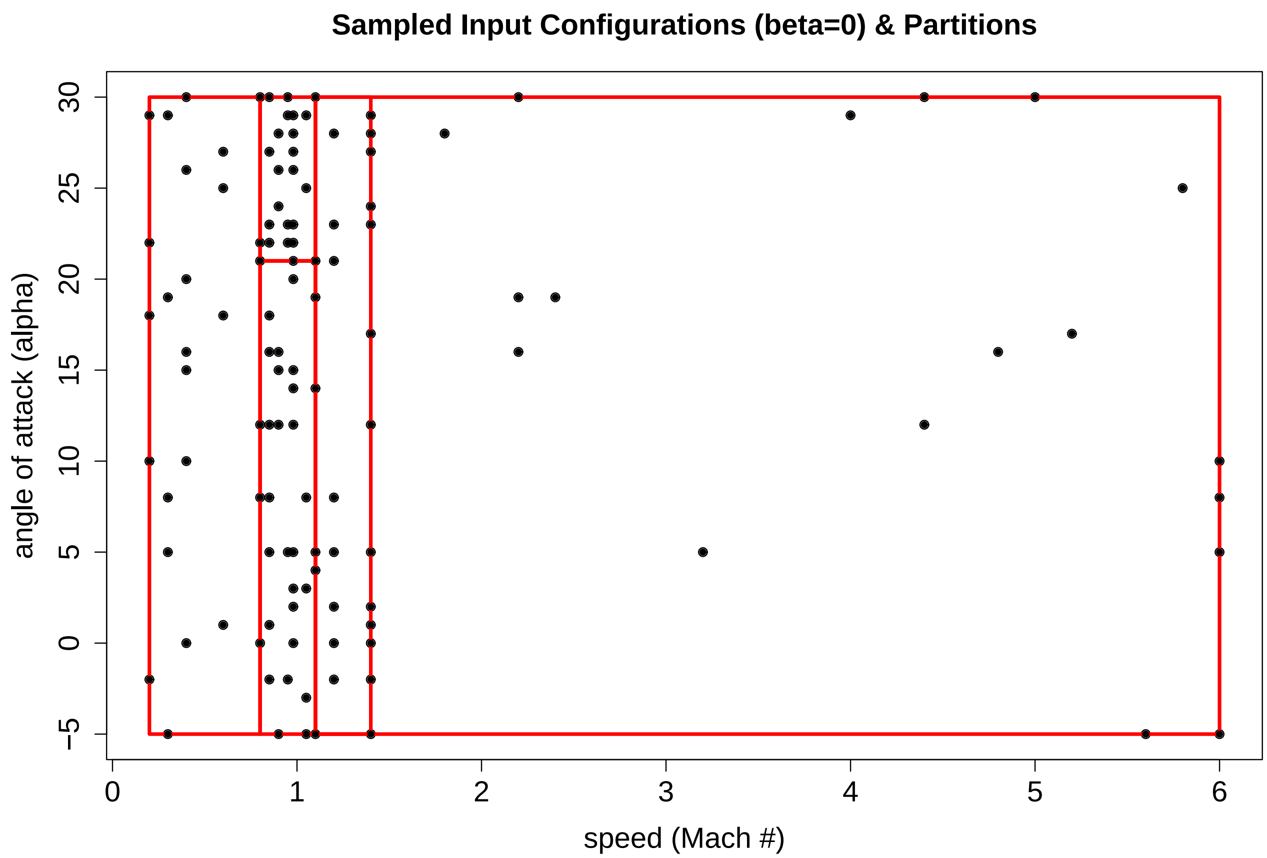
Rocket Booster Example

- 3 inputs, 6 outputs
- Each sample requires 5-20 hours computing time
- Non-stationary
- Fit independent treed GP for each response, use standardized average of predictive variability

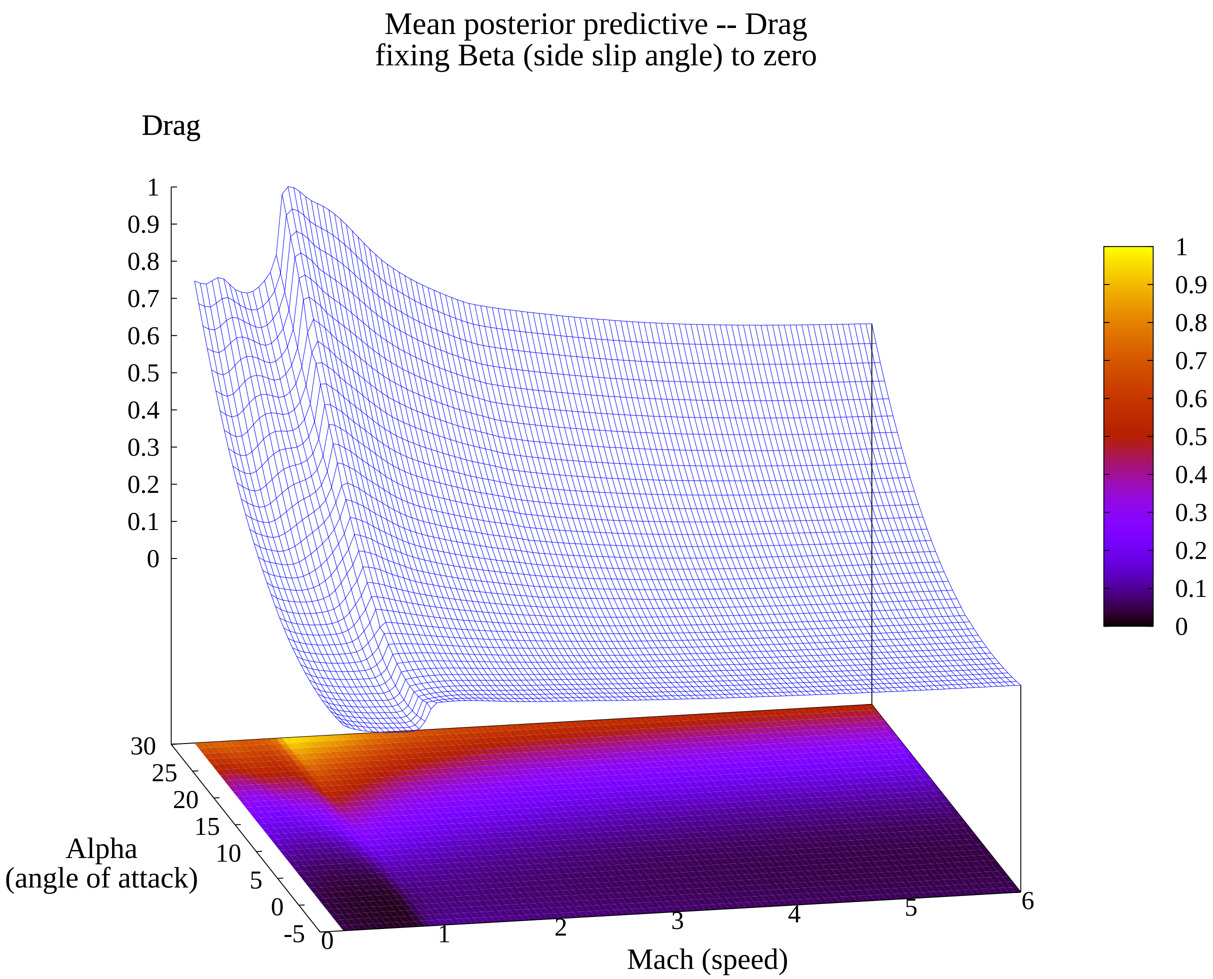
Adaptive Sampling on LGBB: Lift



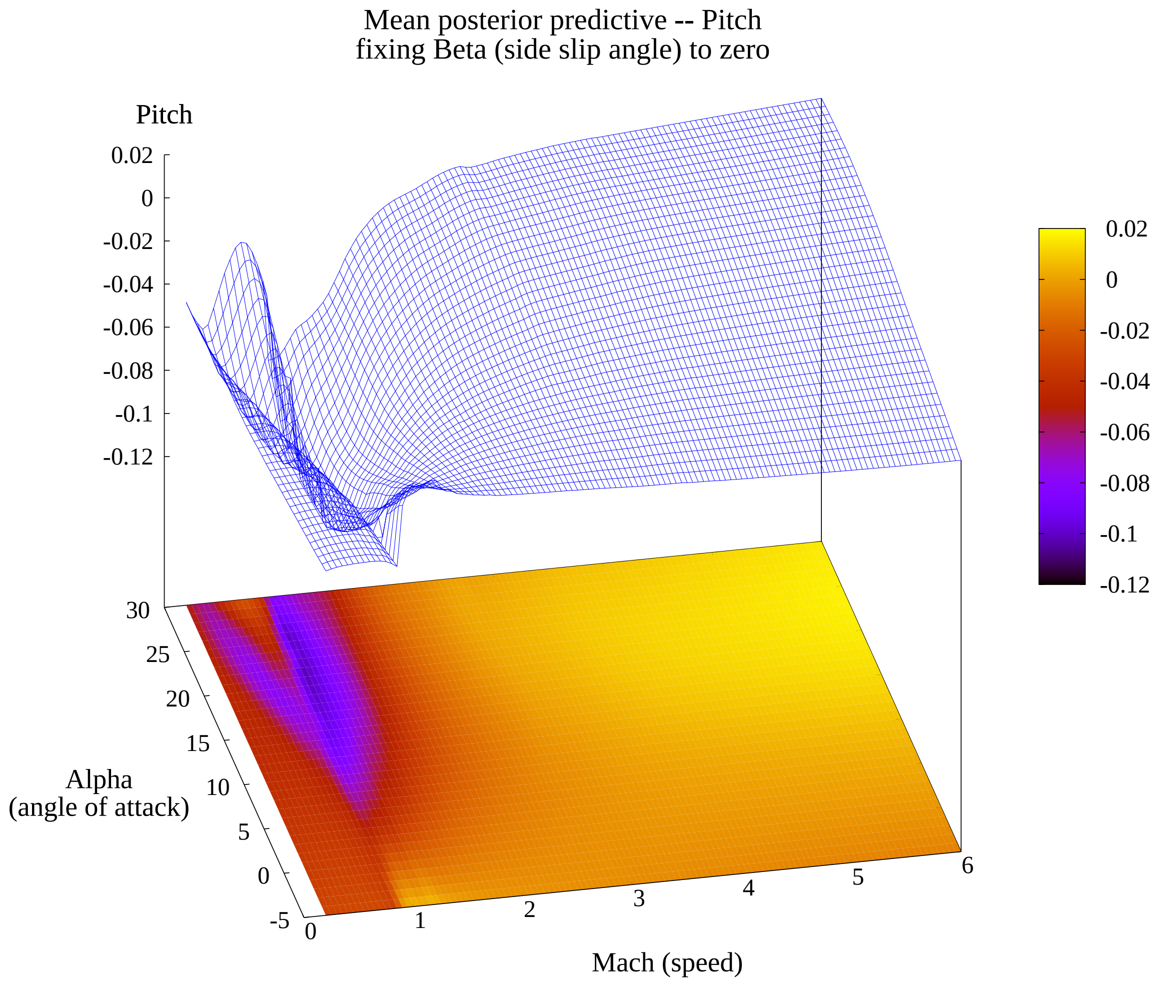
Adaptive Sampling on LGBB: Lift



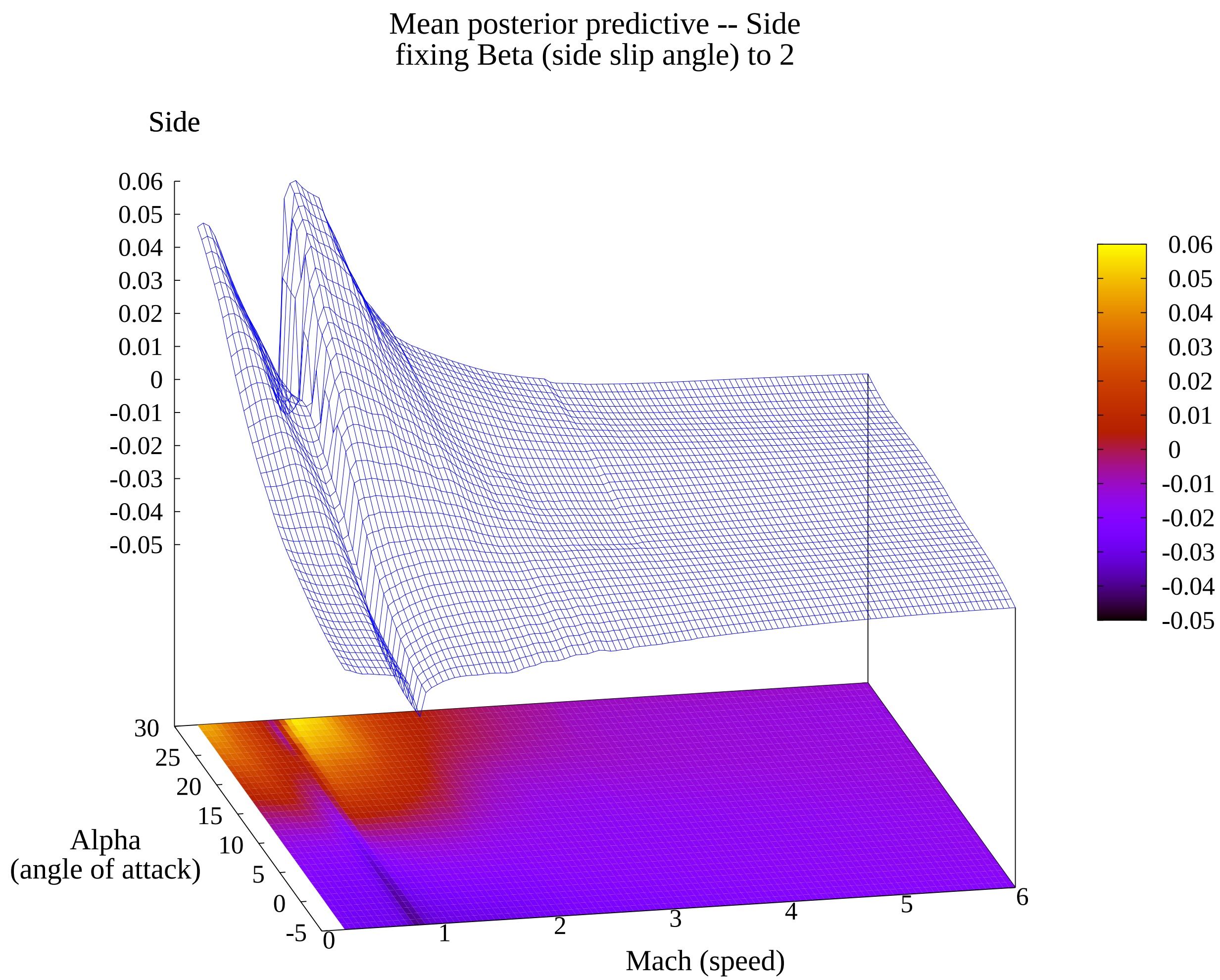
Adaptive Sampling on LGBB: Drag



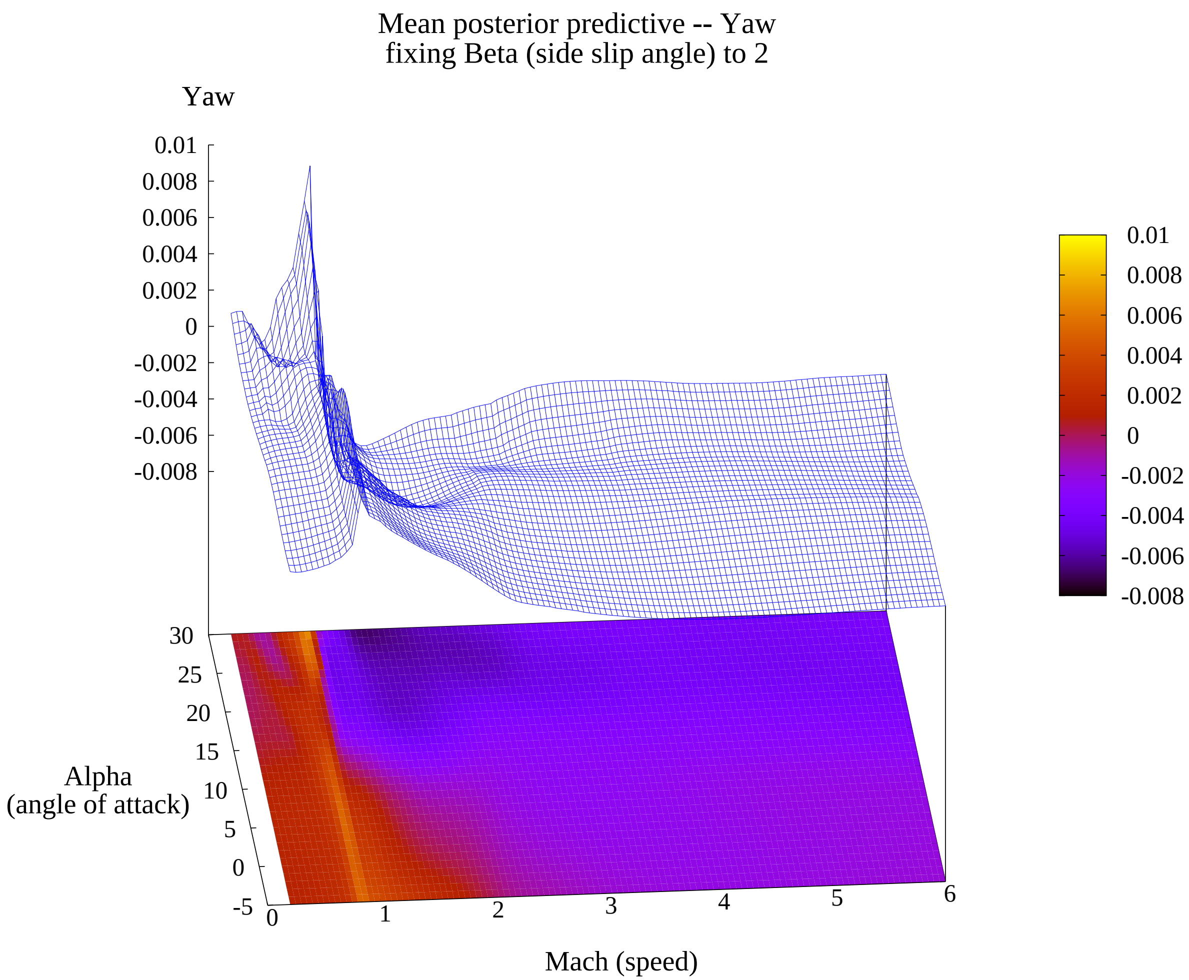
Adaptive Sampling on LGBB: Pitch



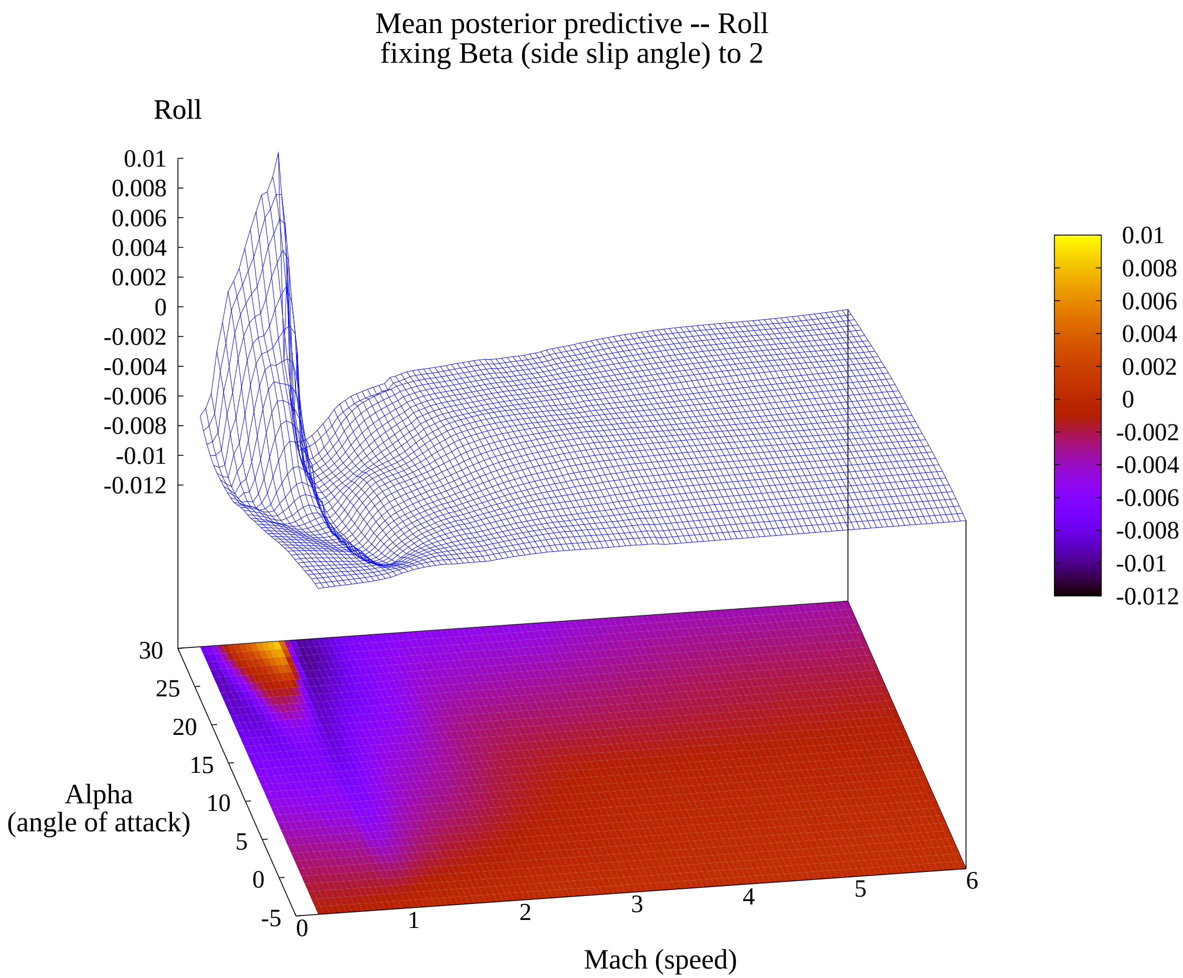
Adaptive Sampling on LGBB: Side Force



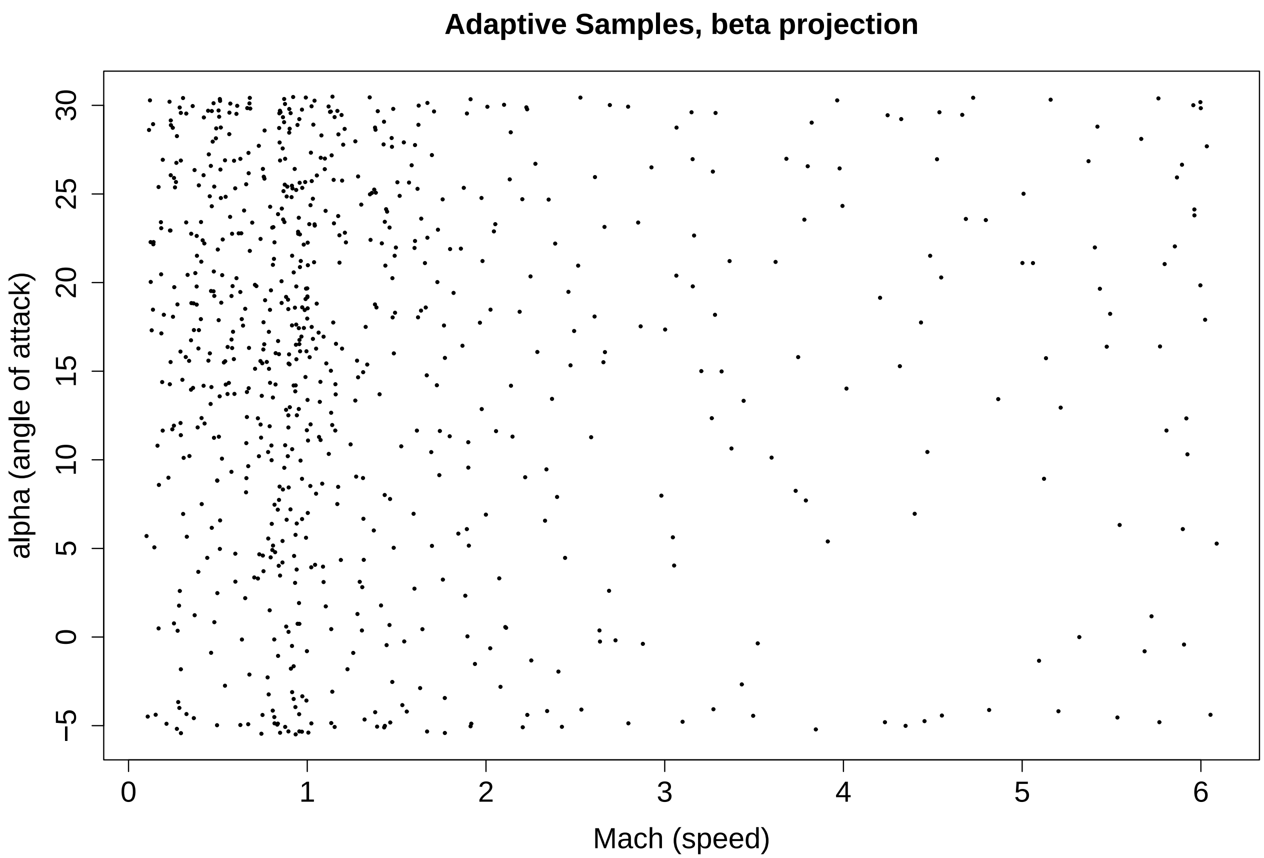
Adaptive Sampling on LGBB: Yaw



Adaptive Sampling on LGBB: Roll



Adaptive Sampling on LGBB



750 adaptive samples, compared to more than **3250** at NASA

Conclusions

- Can create a surrogate model during sequential experimental design
- Can model nonstationarity
- Can greatly reduce necessary computing time
- R package available on CRAN
  `http://cran.r-project.org/src/contrib/Descriptions/tgp.html`

References

Chipman, H. A., George, E. I., and McCulloch, R. E. (2002). "Bayesian Treed Models." *Machine Learning*, 48, 303–324.

Cohn, D. A. (1996). "Neural network exploration using optimal experimental design." In *Advances in Neural Information Processing Systems*, eds. J. D. Cowan, G. Tesauro, and J. Alspector, vol. 6, 679–686. Morgan Kaufmann Publishers.

Gramacy, R. B., Lee, H. K. H., and Macready, W. (2004). "Parameter Space Exploration With Gaussian Process Trees." In *ICML*, 353–360. Omnipress & ACM Digital Library.

MacKay, D. J. C. (1992). "Information-based objective functions for active data selection." *Neural Computation*, 4, 4, 589–603.

Rasmussen, C. E. and Ghahramani, Z. (2002). "Infinite mixtures of Gaussian process experts." In *Advances in Neural Information Processing Systems*, vol. 14, 881–888. MIT Press.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). "Design and Analysis of Computer Experiments." *Statistical Science*, 4, 409–435.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. New York, NY: Springer-Verlag.

Seo, S., Wallat, M., Graepel, T., and Obermayer, K. (2000). "Gaussian Process Regression: Active Data Selection and Test Point Rejection." In *Proceedings of the International Joint Conference on Neural Networks*, vol. III, 241–246. IEEE.

Silverman, B. W. (1985). "Some Aspects of the Spline Smoothing Approach to Non-Parametric Curve Fitting." *Journal of the Royal Statistical Society Series B*, 47, 1–52.