

Bayesian inference and design space identification for nonlinear chemical kinetics

Dave Woods

Southampton Statistical Sciences Research Institute
University of Southampton

D.Woods@southampton.ac.uk
www.southampton.ac.uk/~davew

UNIVERSITY OF
Southampton

EPSRC
Engineering and Physical Sciences
Research Council

Outline

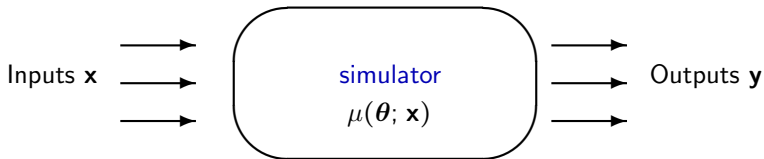
- ▶ Motivation
 - ▶ Computationally expensive simulators (computer models)
 - ▶ Chemical kinetics and “design space”
- ▶ Bayesian inference for expensive simulators
 - ▶ Calibration (parameter estimation)
 - ▶ MCMC via adaptive emulation
- ▶ Prediction and design space generation

Joint work with Antony Overstall (University of St Andrews, UK) and Kieran Martin (Office for National Statistics)

Some underpinning methodology can be found in Overstall and Woods (2013). *Biometrics*, in press (doi:10.1111/biom.12017)

Computer modelling

- ▶ Many physical and social processes can be approximated by computer codes (**simulators**) which encapsulate mathematical descriptions
 - ▶ For example, partial differential equations, solved using e.g. finite element methods ...
 - ▶ ... or “agent-based” simulators, building up the behaviour of the system from that of individual elements



- ▶ Embed the simulator within a statistical model, e.g.

$$\mathbf{y} = \mu(\boldsymbol{\theta}; \mathbf{x}) + \boldsymbol{\varepsilon}$$

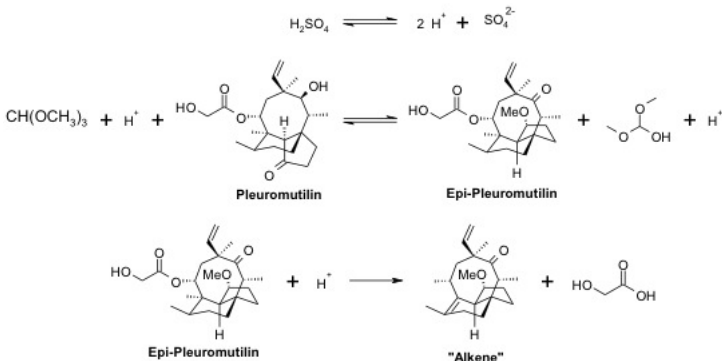
Calibrating simulators

- ▶ Learn about θ from (noisy) data (\mathbf{x}, \mathbf{y})
- ▶ **Problem:** requires very many evaluations of the expensive simulator
 - ▶ e.g. to evaluate the unnormalised log-posterior density
- ▶ **Solution:** build (cheap) approximations to μ as functions of ϕ
 - ▶ $\hat{\mu}$ is an emulator or surrogate - typically a Gaussian process
 - ▶ replace μ by $\hat{\mu}$ in the calibration problem

(Kennedy & O'Hagan, 2001)

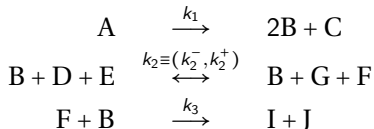
Chemical kinetics example

Pharmaceutical process from GlaxoSmithKline



Acknowledgements: James Wertman, Mohammad Yahyah, John Peterson

Chemical kinetics example



- ▶ Defines a set of differential equations (solve numerically using ode in R package deSolve)
- ▶ Incorporate dependance of rate coefficients on temperature, T , through the Arrhenius equation

$$k_i = k_i^{(r)} \exp \left\{ \frac{E_i^{(a)}}{R} \left(\frac{1}{T} - \frac{1}{T^{(r)}} \right) \right\}$$

Known constants: $T^{(r)}$ - reference temp.; R - gas constant

Chemical kinetics example

- ▶ Control temperature, observation time, volume, starting concentrations $[A]$, $[D]$, $[E]$
- ▶ Measure/model concentrations $[E]$, $[F]$, $[I]$
- ▶ Need to calibrate (estimate) $\theta = (k_2^{(r)-}, k_2^{(r)+}, k_3^{(r)}, E_2^{(a)}, E_3^{(a)})$

Bates & Wates (1988), Ramsey et al. (2007), PK/PD literature, PODE community . . .

Statistical model

Let \mathbf{y}_{ij} be the 3×1 vector of observed concentrations for treatment \mathbf{x}_i at time t_{ij} , for $i = 1, \dots, n$ and $j = 1, \dots, n_i$. Then we assume the following statistical model

$$\mathbf{y}_{ij} = \boldsymbol{\mu}(\boldsymbol{\theta}; \mathbf{x}_i, t_{ij}) + \boldsymbol{\delta}(\mathbf{x}_{ij}, t_{ij}) + \boldsymbol{\epsilon}(t_{ij}),$$

where

- $\boldsymbol{\delta}$ - Model discrepancy with a Gaussian process prior distribution
- $\boldsymbol{\epsilon}$ - Measurement error following a Gaussian distribution with time dependent correlation

Complete Bayesian model specification with prior distributions for $\boldsymbol{\theta}$ and nuisance parameters defining $\boldsymbol{\delta}$ and $\boldsymbol{\epsilon}$

Gaussian processes

- ▶ A stochastic process for which any finite-dimensional sample has a joint multivariate normal distribution
- ▶ Popular as a **prior distribution over functions** in Bayesian nonparametrics, in machine learning and for building **emulators** in computer experiments

If $Z(\mathbf{v}) \sim \text{GP}(0, \kappa)$, with covariance function

$$\kappa(\mathbf{v}, \mathbf{v}') = \sigma^2 \left\{ \exp \left(- \sum_{i=1}^k \psi_i |v_i - v'_i| \right) + \eta I \right\},$$

then $z(\mathbf{v}_1), \dots, z(\mathbf{v}_n) \sim N(0, \Sigma)$ and $\Sigma_{ij} = \kappa(\mathbf{v}_i, \mathbf{v}_j)$

Design space

The ICH Q8 core definition of design space: “The multidimensional combination and interaction of input variables (e.g., material attributes) and process parameters that have been demonstrated to provide assurance of quality”¹

For illustration, we will define quality through constraints on E, F and I (in moles):

$$E < 3$$

$$F > 20$$

$$I < 3$$

Ultimately, want to identify the set $\{\mathbf{x}, t : P(E < 3, F > 20, I < 3 | \mathbf{y}) > \alpha\}$

¹ICH (2005). ICH Harmonized Tripartite Guideline: Pharma. Develop., Q8.

General problem

- ▶ We have n observations, \mathbf{y} , from the distribution $\mathbf{Y}|\phi, \mathbf{x} \sim F(\boldsymbol{\theta})$
- ▶ Vector $\phi = (\boldsymbol{\theta}, \boldsymbol{\psi})$ holds
 - ▶ $\boldsymbol{\theta}$ - q parameters of interest
 - ▶ $\boldsymbol{\psi}$ - $p - q$ nuisance parameters
- ▶ The distribution F depends on $\boldsymbol{\theta}$ through the simulator, e.g.

$$E(Y_i) \propto \mu(\boldsymbol{\theta}; \mathbf{x}_i),$$

where \mathbf{x}_i is a k -vector of explanatory variables

- ▶ Bayesian inference for ϕ uses the posterior density

$$\pi(\phi|\mathbf{y}) \propto U(\phi) = \pi(\mathbf{y}|\phi)\pi(\phi)$$

Approximating the posterior

- ▶ $U(\phi)$ requires evaluating computationally expensive $\mu(\theta; \mathbf{x})$
- ▶ Instead, construct a statistical approximation or **emulator** using a few hundred simulator runs
- ▶ Replace $u(\phi) = \log U(\phi)$ with the emulator in the MCMC

	Output dimension	Input dimension
$u(\phi)$	1	p^*
$\mu(\theta; \mathbf{x}_i)$	1	$q + k^{**}$
$\mu(\theta)$	n	q ($< p$) ^{***}

- ▶ Emulating the simulator allows fast prediction and model checking

* Rasmussen (2003), Bliznyuk et al. (2008), Fielding et al. (2011)

** Kennedy & O'Hagan (2001)

*** Higdon et al. (2008), Overstall & Woods (2013)

Adaptive construction of an emulator

Rasmussen (2003) proposed the following algorithm:

- (1) Evaluate the simulator for an initial “design” in the input space
 - (2) Construct an emulator for $u(\phi)$
 - (3) Augment the design by points sampled from an MCMC method with target distribution based on the current approximation
 - (4) Evaluate the simulator for these new points
 - (5) Return to (2) and repeat until the emulator is sufficiently accurate or the computational budget is exhausted
- ▶ At each iteration of (3), the adequacy of the constructed emulator is checked through comparison with the newly evaluated points
 - ▶ The emulator can then replace $u(\phi)$ in subsequent MCMC inference

Adaptive construction of an emulator

(2) Construct an emulator for $u(\phi)$

Emulate the n -dimensional $\boldsymbol{\mu}(\boldsymbol{\theta})$ using a singular value decomposition and Gaussian Process models

- ▶ Model $\boldsymbol{\mu}(\boldsymbol{\theta})$ via a principal component basis

$$\boldsymbol{\mu}(\boldsymbol{\theta}) \approx \sum_{i=1}^m \mathbf{b}_i w_i(\boldsymbol{\theta})$$

- ▶ Fit m GP models to the first (most important) principal component weights, w_i
- ▶ m is chosen so that the PC basis explains 100d% of the variation

Adaptive construction of an emulator

(3) Augment the design by points sampled from an MCMC method with target distribution based on the current approximation

- ▶ Rasmussen (2003) used Hybrid (Hamiltonian) Monte Carlo
 - ▶ Proposals from the solution to a set of Hamiltonian equations, discretised with step size δ
 - ▶ Low autocorrelation, high effective sample size (good space-filling)
 - ▶ Per iteration, requires one evaluation of $u(\phi)$ and L evaluations of $\partial u(\phi)/\partial \phi$
- ▶ Fielding et al. (2011) extended the method to include parallel tempering
 - ▶ Explores multi-modal distributions using parallel tempered chains
 - ▶ Sample from densities $U(\phi)^{1/T_k}$, $k = 1, \dots, K$

Adaptive construction of an emulator

- (3) Augment the design by points sampled from an MCMC method with target distribution based on the current approximation

HMC and parallel tempering require selection of tuning parameters

- ▶ If $\mu(\boldsymbol{\theta}; \mathbf{x})$ is expensive, this can be computationally inefficient
1. Replace HMC with Gibbs sampling with ARMS (Gilks et al., 1995)
 - ▶ Adaptive Rejection Metropolis Sampling
 - ▶ No tuning parameters
 - ▶ Use thinning to reduce autocorrelation
 2. Adaptive parallel tempering (Miasojedow et al., 2012) chooses temperatures to achieve a target acceptance rate

Synthetic Examples

We compare our proposed method (ARMS and adaptive parallel tempering) with Fielding et al. (2011) using two synthetic examples:

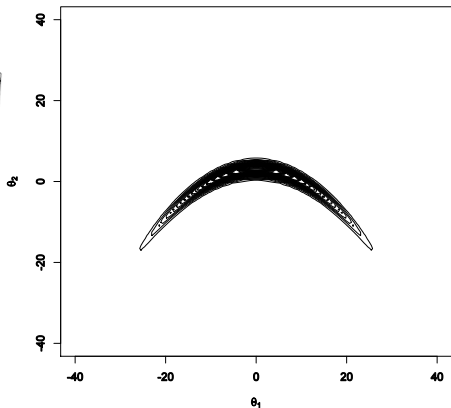
1. A two-dimensional distribution with non-elliptical contours
2. A seven-dimensional distribution with two modes
 - ▶ Approximate $u(\theta)$ directly as there is no simulator here
 - ▶ Compare true density and approximation using Hellinger distance

Synthetic Examples

For each example:

- ▶ Initial design has 50 points
- ▶ Number of temperatures is $K = 5$
- ▶ Add an additional 150 points (i.e. 30 iterations of the algorithm)
- ▶ 20 implementations of each method using different values of
 - ▶ HMC step size, δ , for Fielding et al. (2011)
 - ▶ the number, Q , of MCMC scans discarded between iterations (thinning)
- ▶ Repeat each implementation for each method 100 times

Synthetic Example 1 - non-elliptical contours



The pdf here is:

$$U(\theta) \propto \exp\left(-\frac{1}{2}\mathbf{u}^T \mathbf{C}^{-1} \mathbf{u}\right)$$

with

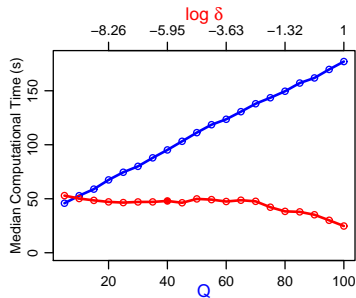
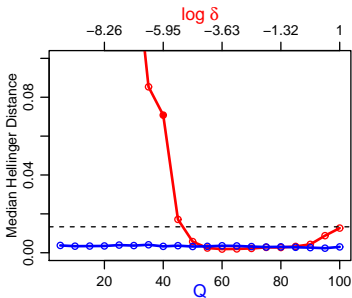
$$u_1 = \theta_1$$

$$u_2 = \theta_2 + \kappa\theta_1^2 - 100\kappa$$

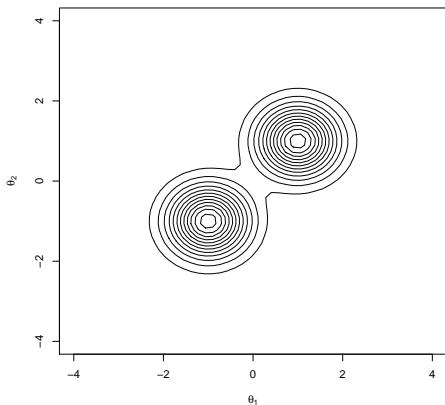
$$\mathbf{C} = \text{diag}(100, 1)$$

$$\kappa = 0.03$$

Synthetic Example 1 - non-elliptical contours



Synthetic Example 2 - bimodal



The distribution here
is

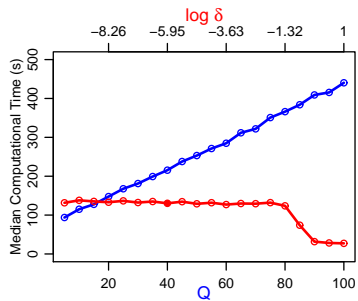
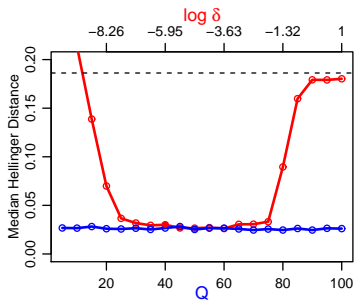
$$\frac{1}{2}N(\mathbf{m}_1, 0.35\mathbf{I}) + \frac{1}{2}N(\mathbf{m}_2, 0.35\mathbf{I})$$

with

$$\mathbf{m}_1 = (1, 1, 0, 0, 0, 0, 0)$$

$$\mathbf{m}_2 = -\mathbf{m}_1$$

Synthetic Example 2 - bimodal



Conclusions from Synthetic Examples

- ▶ The proposed method is robust to the degree of thinning
- ▶ The Fielding et al. (2011) method is very sensitive to the HMC tuning parameter δ
- ▶ When successfully-tuned, the Fielding method performs similarly to our method and is typically faster
- ▶ However when poorly-tuned, the Fielding method is a poor approximation, and the computational advantage decreases as the expense of evaluating the simulator increases

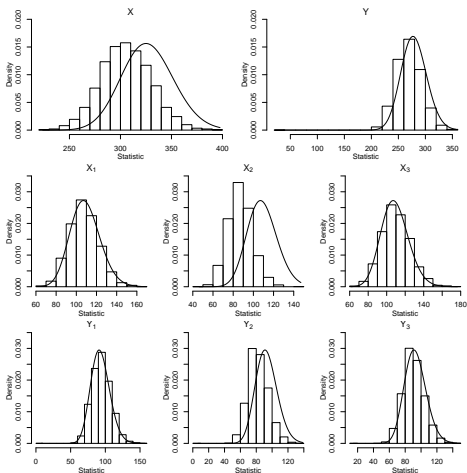
Stages in constructing a design space

1. Generate a sample from the posterior distribution of the model parameters (including θ) using MCMC methods
 - ▶ Gibbs sampling and Metropolis-Adjusted Langevin Algorithm (MALA; Girolami & Calderhead, 2011)
 - ▶ Emulate $\mu(\theta) = [\mu(\theta; \mathbf{x}_1, t_{11}), \dots, \mu(\theta; \mathbf{x}_n, t_{nn})]$ as a function of θ
2. For each MCMC-sampled parameter vector ($b = 1, \dots, B$), generate a prediction, $\mathbf{y}^{(b)}$, from the statistical model, for a setting of the factors
 - ▶ Emulate $\mu(\theta; \mathbf{x}, t)$ as a function of θ, \mathbf{x}, t (Conti & O'Hagan, 2010)
3. Calculating the proportion of the predictions $\{\mathbf{y}^{(1)}, \dots, \mathbf{y}^{(B)}\}$ that satisfy the constraints.

Chemical kinetics - data

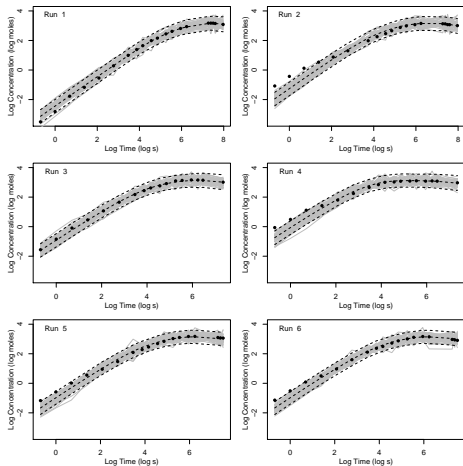
	Temperature (°C)	Volume (L)	A (moles)	Initial amounts	
				D (moles)	E (moles)
1	25.00	31.31	22.50	91.59	26.47
2	25.00	32.56	45.00	91.59	26.47
3	40.00	31.28	22.50	91.50	26.45
4	40.00	32.53	45.00	91.50	26.45
5	32.50	31.88	33.75	91.40	26.42
6	32.50	31.88	33.75	91.40	26.42

Chemical kinetics - diagnostics

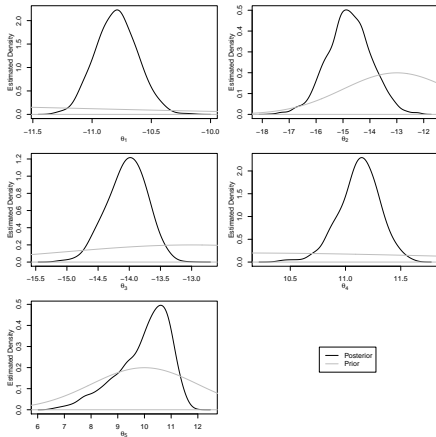


Chi-squared statistics
for the sum of
standardised squared
errors for the mean
response (X) and
individual
observations (Y)

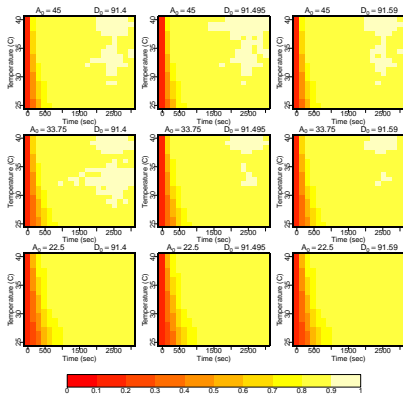
Chemical kinetics - diagnostics



Chemical kinetics - posterior densities

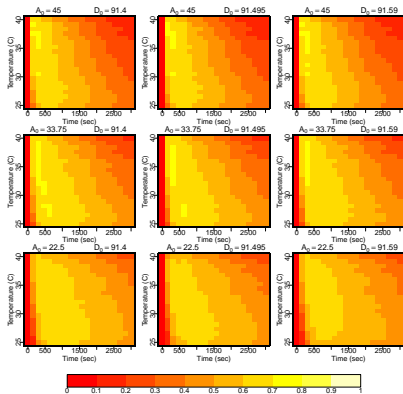


Design space - $E < 3$



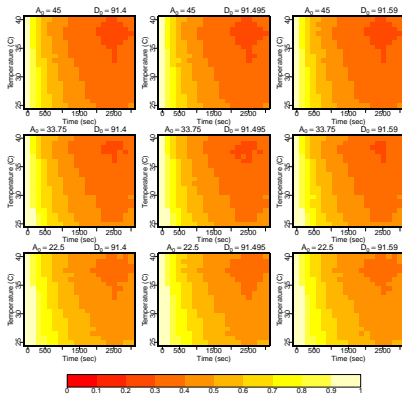
Vol = 31.92L, $E = 26.445$ mol

Design space - $F > 20$



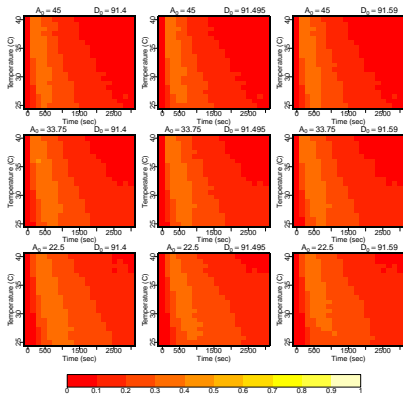
Vol = 31.92L, E = 26.445 mol

Design space - $I < 3$



Vol = 31.92L, E = 26.445 mol

Overall design space



Vol = 31.92L, E = 26.445 mol

Discussion

- ▶ MCMC inference for computationally expensive simulators via adaptive construction of a Gaussian Process emulator
 - ▶ Gibbs sampling with ARMS reduces tuning
 - ▶ Emulating the simulator, rather than the unnormalised log-posterior, allows fast prediction and model checking
- ▶ Chemical kinetics and design space
 - ▶ Fast generation of design spaces
 - ▶ Extend to population models and random effects: separate emulators for each run
- ▶ The proposed Bayesian inference methods are implemented in an R package `bayxcomp`

References

- ▶ Overstall, A.M. and Woods, D.C. (2013). A Strategy for Bayesian inference for computationally expensive models with application to the estimation of stem cell properties. *Biometrics*, in press (doi:10.1111/biom.12017).
- ▶ Bliznyuk, N., Ruppert, D., Shoemaker, C., Regis, R., Wild, S. and Mugunthan, P. (2008). *JCGS*, 17, 270-294.
- ▶ Fielding, M., Nott, D.J. and Liong, S-Y. (2011). *Technometrics*, 53, 16-28.
- ▶ Gilks, W.R., Best, N.G. and Tan, K.K.C. (1995). *JRSSC*, 44, 455-472.
- ▶ Higdon, D., Gattiker, J., Williams, B. and Rightley, M. (2008). Computer model calibration using high-dimensional output. *JASA*, 103, 570-583.
- ▶ Kennedy, M. C. and OHagan, A. (2001). *JRSSB*, 63, 425-464.
- ▶ Miasojedow, B., Moulines, E. and Vihola, M. (2012). Technical Report. Institut Mines-Telecom/Telecom Paris-Tech. arXiv:1205.1076v1.
- ▶ Perney, N.M.B., Horak, P., Hanley, N. and Melvin, T. (2012). *Physical Biology*, 9, 24001-24008.
- ▶ Rasmussen, C.E. (2003). *Bayesian Statistics 7*, p. 651-659. Oxford University Press.

Hybrid (Hamiltonian) Monte Carlo

- ▶ HMC evolves an artificial dynamic system on an augmented parameter space according to a set of Hamiltonian equations whose numerical solutions form the next proposed value in a chain
- ▶ HMC is typically superior to Metropolis-Hastings (e.g. random-walk or Gibbs sampling) in terms of autocorrelation and effective sample sizes
- ▶ However you need to specify two tuning parameters:
 - ▶ δ - step size
 - ▶ L - number of steps
- ▶ Typically, fix L and vary δ

Parallel tempering

- ▶ Parallel tempering sets up K MCMC chains where the primary chain samples from the posterior and the other chains sample from increasingly diffuse versions of the posterior given by

$$U_k(\boldsymbol{\theta}) \propto U(\boldsymbol{\theta})^{1/T_k} = \exp(u(\boldsymbol{\theta})/T_k),$$

where $1 = T_1 < \dots < T_K$ are the temperatures

- ▶ There are then swap moves between the different chains
- ▶ This allows large moves performed at the higher chains to be passed down to the primary chain

Adapting the parallel tempering

- ▶ To specify the temperatures, T_1, \dots, T_K , in the parallel tempering we use the adaptive parallel tempering scheme proposed by Miasojedow et al. (2012)
- ▶ This adapts the temperatures so that the acceptance rates between the chains reaches a target value, $\alpha^* = 0.5$
- ▶ The amount of adaption diminishes as the number of iterations increases.

Gibbs sampling with ARMS

- ▶ As an alternative to HMC we propose the use of *Gibbs sampling with adaptive rejection Metropolis sampling* (ARMS, Gilks et al., 1995) as an alternative MCMC method to HMC for selecting the next design points
- ▶ A scan of the Gibbs sampling algorithm involves updating each element of θ
- ▶ We generate θ_i from the full conditional distribution, i.e. $\theta_j | \theta_{\setminus j}, \mathbf{y}$
- ▶ The full conditionals will not be known distributions but we can use ARMS to sample from them.
- ▶ ARMS is an extension of rejection sampling, with an adaptive proposal distribution
- ▶ It has no tuning parameters.

Gibbs sampling with ARMS

- ▶ However in a cheap- $u(\theta)$ problem the advantage of HMC over Gibbs sampling is that it will have lower autocorrelation and higher effective sample size
- ▶ To counteract this, we use the equivalent of thinning, i.e. we repeat the scan Q times
- ▶ Q might now be considered a tuning parameter
- ▶ Synthetic examples show that our proposed method is largely insensitive to the choice of Q

Assessing emulator adequacy

- ▶ Extend univariate measure of Bastos & O'Hagan (2008, Tech.) to multivariate output
- ▶ Assess the predictive performance of the emulator after iteration j using nK simulator runs from iteration $j + 1$

$$R_{j+1} = \sum_{k=1}^K \sum_{l=1}^n \frac{(\mu(\phi_k^{j+1}; \mathbf{x}_l) - \hat{\mu}(\phi_k^{j+1}; \mathbf{x}_l))^2}{\hat{v}(\phi_k^{j+1}; \mathbf{x}_l)},$$

- ▶ \hat{v} is the estimated variance of $\hat{\mu}$
- ▶ R_{j+1} approximately follows a χ_{kn}^2 distribution

Bayesian model checking

1. Use MCMC sample $\theta^1, \dots, \theta^B$ to generate predictions $\tilde{\mathbf{y}}^1, \dots, \tilde{\mathbf{y}}^B$ from $\pi(\mathbf{y}|\theta^j)$
 - ▶ Compare observations and predictions graphically
2. Pivotal Discrepancy Measures (PDM; Yuan & Johnson, 2012, Biometrics)
 - ▶ For each of the B MCMC iterations, partition residuals according to value of $\mu(\phi^j)$
 - ▶ Within each partition, bin into cells according to the distribution of the standardised residuals
 - ▶ Calculate a χ^2 statistic for each partition, and sum across the partitions to obtain the PDM
 - ▶ Obtain “ p -value” from the sample of B PDMs; < 0.25 provides evidence of model inadequacy