Monitoring milk-powder dryers via Bayesian inference in an FPGA

Colin Fox fox@physics.otago.ac.nz
Markus Neumayer, Al Parker, Pat Suggate

Three newish technologies

- Gibbs sampling for capacitance tomography (ECT)
  - and other inverse problems
  - where Hamiltonian is quadratic in field and linear in material properties
- Polynomial acceleration of Gibbs sampling
  - optimal convergence of first and second moments
  - derived for Gaussians
  - learn covariance adaptively for Gaussian-like distributions (ECT)
- Large-scale computation in an FPGA
  - the compiler

Two Paradigms for Imaging

- Signal processing
  - solution is *function* of data
- Model fitting
  - optimization (best solution)
  - statistical inference (summarize all solutions)

Two Paradigms for Imaging

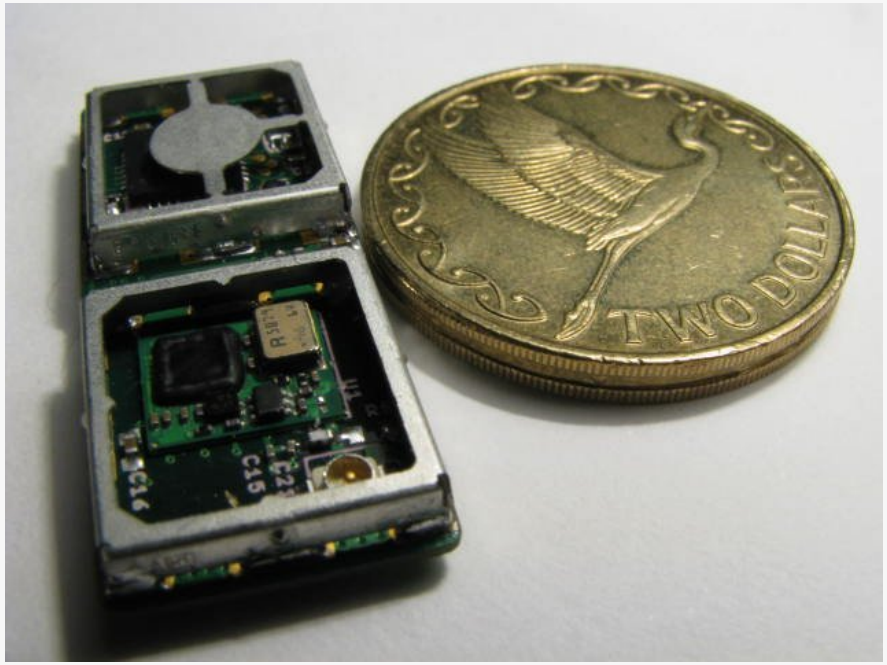
- Signal processing
  - solution is *function* of data
- Model fitting
  - optimization (best solution)
  - statistical inference (summarize all solutions)

Bayesian inference is the 'gold standard' though intensive in computing and modelling

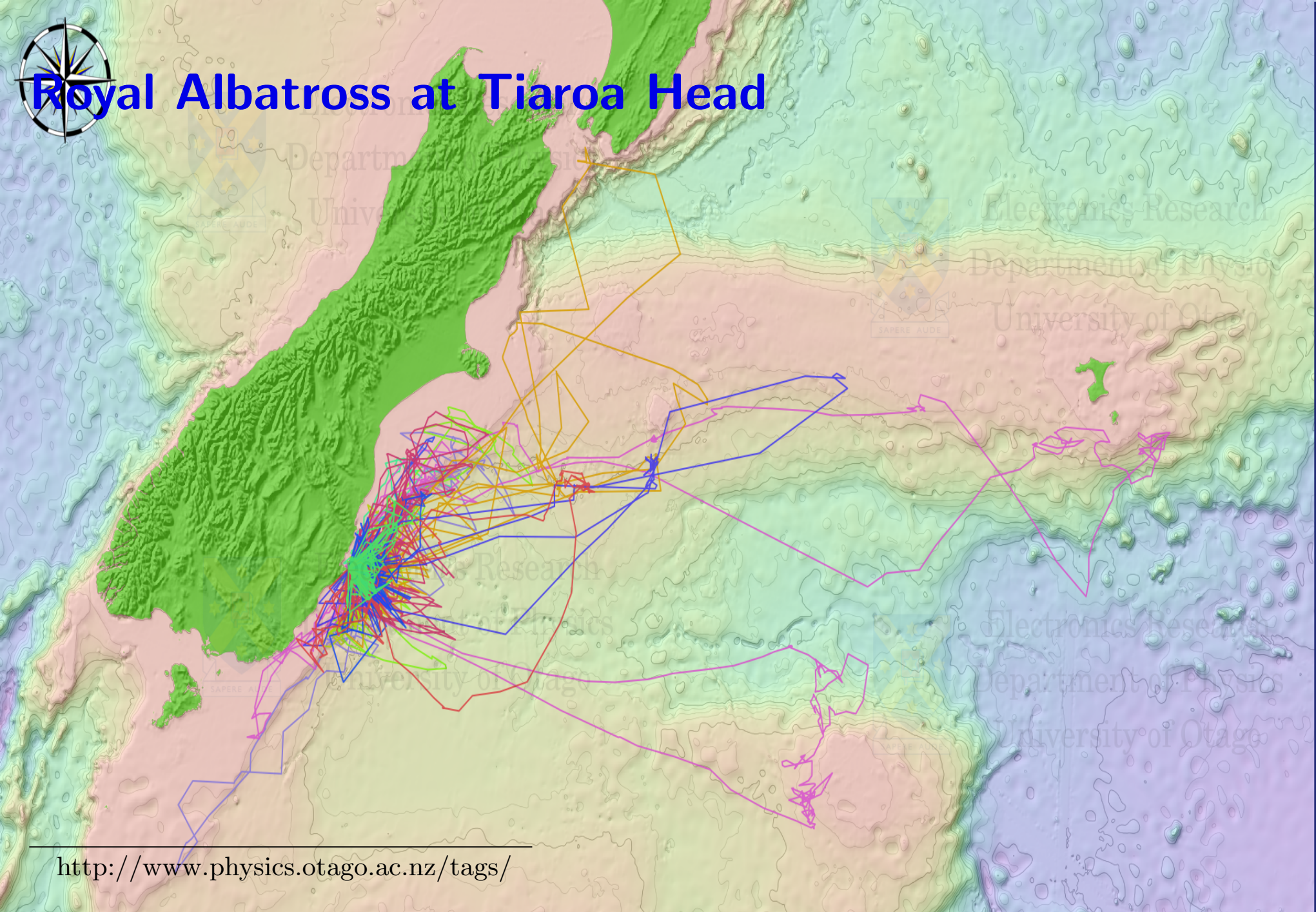
We currently implement statistical inference for:

- Wildlife tracking (DoC, Sirtrack, Rakon)
- Dairy processing (TetraPak, Synlait)
- Agritech (Truetest, Silverfern Farms)
- Geothermal electricity generation (Contact Energy, iwi)

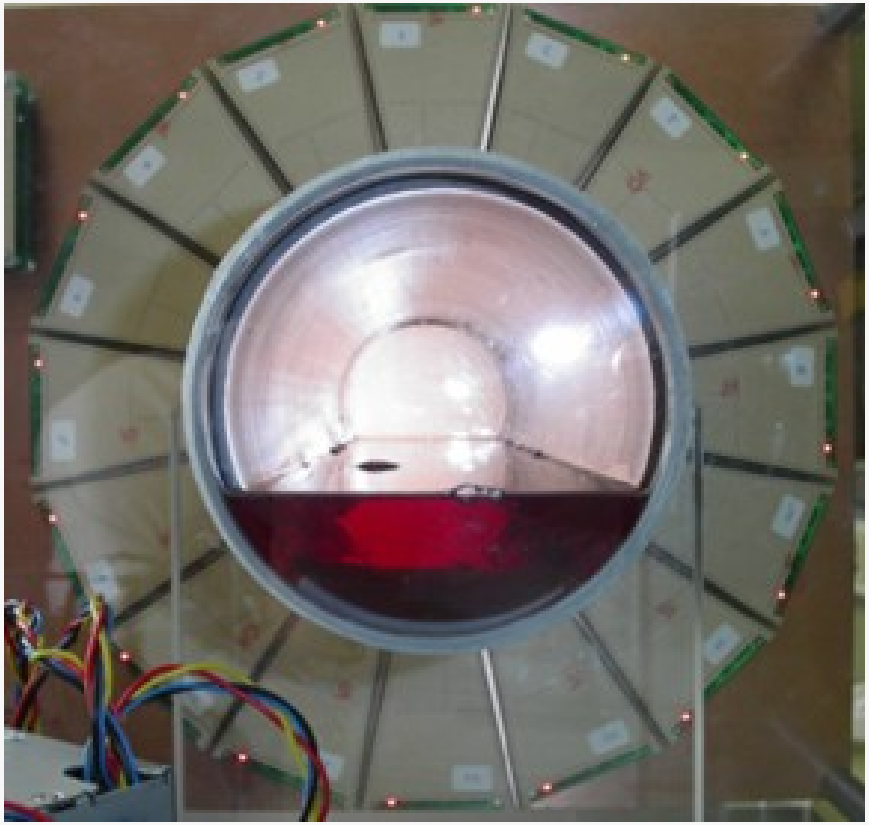
Fast-fix wildlife tags



- inference reduces fix time from 30 seconds to 2 milli-seconds!
- which means less power, less (battery) weight, longer life
- current model weighs 6 gram, runs for 1 year


Royal Albatross at Tiaroa Head
http://www.physics.otago.ac.nz/tags/

Capacitance tomography



- non-contact measurements
- low electric fields (below ambient)
- images permittivity: good contrast for fat, solids, water
- measures bulk properties (total fat, average flow)
- with spatial resolution

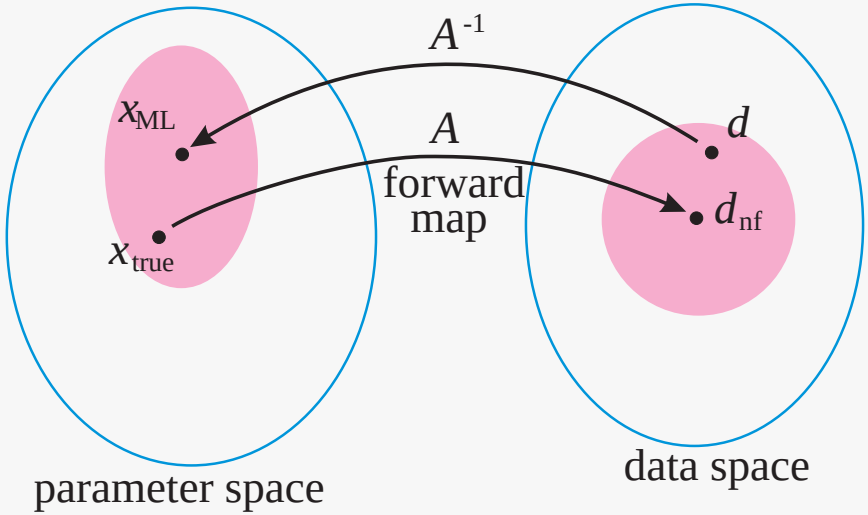
Embed FPGA processing with sensors to perform real-time quantified inference

Edendale plant


Burt Munro
528MO7

Sample-based Bayesian inference

Parameters $x \mapsto d = A(x) + e$, errors e, d is a sample from $l(d|x) = \pi_{\mathrm{e}}(d - A(x))$



Posterior distribution over x given d $\qquad \pi(x|d) = \dfrac{l(d|x)\pi(x)}{\pi(d)}$ Bayes' rule

Posterior estimates

$$\mathsf{E}_{\pi}\left[f\left(x\right)\right] \approx \frac{1}{n}\sum_{i=1}^{n} f\left(x^{(i)}\right)$$

where $x^{(1)}, \ldots, x^{(n)} \sim \pi\left(\cdot|d\right)$ constructed as iterates of an ergodic map

A posterior distribution for ECT

Mathematical model for measurements $\eta : x \mapsto y$ is the Neumann boundary value problem

$$\nabla \cdot x(s) \nabla v(s) = 0, \qquad s \in \Omega$$

$$x(s) u \frac{\partial v(s)}{\partial n(s)} = j(s), \qquad s \in \partial\Omega$$

where $j(s)$ is the current at boundary location s. Voltages v at electrodes gives data y.

Numerically solve for each transmit-receive pattern (32 times) using FEM discretization

Consider a low level pixel representation for $x(s)$ with MRF prior, giving posterior

$$\pi(x|y) \propto \exp\left\{-\frac{1}{2}(y - \eta(x))^{\mathsf{T}} \Sigma_e^{-1} (y - \eta(x))\right\} \exp\left\{\beta \sum_{i \sim j} u(x_i - x_j)\right\}$$

Not Gaussian, but can be evaluated (expensive) so is amenable to MH MCMC

F Nicholls 1997, Moulton F Svyatskiy 2007, Higdon Reese Moulton Vrugt F 2011

Gibbs sampling for ECT

ECT operator is a $\mathbf{W}\mathbf{S}\mathbf{W}^{\mathsf{T}}$ system: $\nabla \cdot x \nabla$

$\mathbf{W}$ is geometry, $\mathbf{S}$ is diagonal matrix of material properties

FEM discretization preserves (or creates) this

$$\mathbf{K}v = j \quad \text{where system matrix} \quad \mathbf{K} = \mathbf{W}\mathbf{S}_x\mathbf{W}^{\mathsf{T}}$$

Maintain Greens functions = columns of $\mathbf{K}^{-1}$ corresponding to electrodes

$$(\mathbf{K} + \Delta\mathbf{K})^{-1} = \mathbf{K}^{-1} - \mathbf{K}^{-1}\mathbf{W}\left(\mathbf{I} + \mathbf{S}_\Delta\mathbf{S}_x^{-1}\tilde{\mathbf{W}}^{-\mathsf{T}}\mathbf{W}\right)^{-1}\mathbf{S}_x\mathbf{W}^{\mathsf{T}}\mathbf{K}^{-1}$$

where $\tilde{\mathbf{W}}^{-\mathsf{T}}$ is a psuedo-inverse of $\mathbf{W}^{\mathsf{T}}$ that can be pre-evaluated

The matrix pencil

$$\left(\mathbf{I} + \gamma\mathbf{S}_\Delta\mathbf{S}_x^{-1}\tilde{\mathbf{W}}^{-\mathsf{T}}\mathbf{W}\right)\mathbf{u} = \mathbf{c}$$

solve for ≈free in co-ordinate directions, cheaply when $\lesssim 20$ components non-zero

Hence we can evaluate the likelihood cheaply in these directions, and perform Gibbs sampling (e.g. by ARS)

Strang *Intro. to App. Math.*, Meyer Cai Perron 2008, Neumayer PhD 2011

Gibbs sampling from normal distributions

Gibbs sampling[a] repeatedly samples from (block) conditional distributions

Normal distributions

$$\pi(\mathbf{x}) = \sqrt{\frac{\det(\mathbf{A})}{2\pi^n}} \exp\left\{-\frac{1}{2}\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} + \mathbf{b}^\mathsf{T}\mathbf{x}\right\}$$

precision matrix $\mathbf{A}$, covariance matrix $\Sigma = \mathbf{A}^{-1}$ (both SPD)

Mean $\bar{\mathbf{x}}$ satisfies

$$\mathbf{A}\bar{\mathbf{x}} = \mathbf{b}$$

Particularly interested in case where $\mathbf{A}$ is sparse (GMRF) and n large

When is $\pi^{(0)}$ is also normal, then so is the n-step distribution

$$\mathbf{A}^{(n)} \to \mathbf{A} \qquad \Sigma^{(n)} \to \Sigma$$

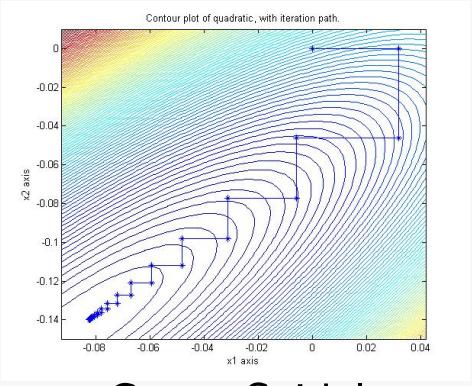
In what sense is "stochastic relaxation" related to "relaxation"?

What decomposition of $\mathbf{A}$ is this performing?

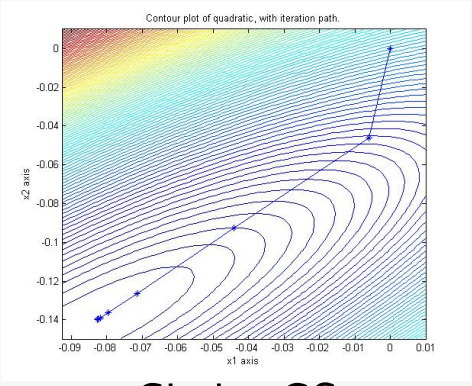
[a] Glauber 1963 (heat-bath algorithm), Turcin 1971, Geman and Geman 1984

Gibbs samplers and equivalent linear solvers

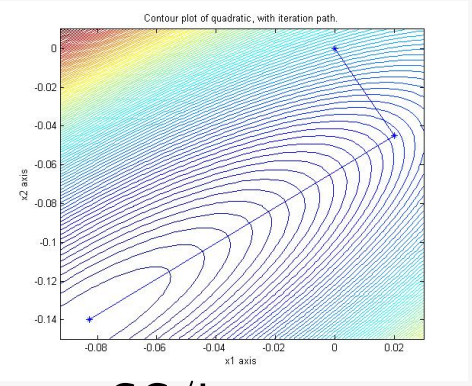
Optimization ...



Gauss-Seidel

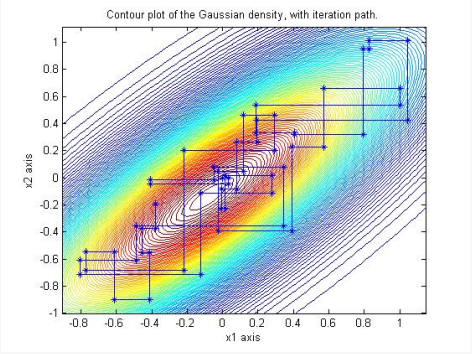


Cheby-GS

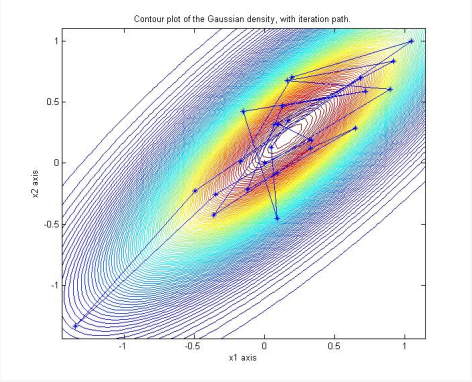


CG/Lanczos

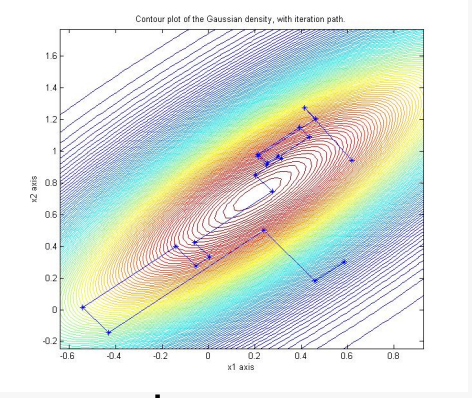
Sampling ...



Gibbs



Cheby-Gibbs



Lanczos

Matrix splitting form of stationary iterative methods

Want to solve

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

The *splitting* $\mathbf{A} = \mathbf{M} - \mathbf{N}$ converts $\mathbf{A}\mathbf{x} = \mathbf{b}$ to $\mathbf{M}\mathbf{x} = \mathbf{N}\mathbf{x} + \mathbf{b}$

If $\mathbf{M}$ is nonsingular

$$\mathbf{x} = \mathbf{M}^{-1}\mathbf{N}\mathbf{x} + \mathbf{M}^{-1}\mathbf{b}$$

Iterative methods compute successively better approximations by

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{M}^{-1}\mathbf{N}\mathbf{x}^{(k)} + \mathbf{M}^{-1}\mathbf{b} \\ &= \mathbf{G}\mathbf{x}^{(k)} + \mathbf{g}\end{aligned}$$

Many splittings use terms in $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$. Gauss-Seidel sets $\mathbf{M} = \mathbf{L} + \mathbf{D}$

$$\mathbf{x}^{(k+1)} = -\mathbf{D}^{-1}\mathbf{L}\mathbf{x}^{(k+1)} - \mathbf{D}^{-1}\mathbf{L}^{\mathsf{T}}\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{b}$$

Matrix formulation of Gibbs sampling from $\mathrm{N}(0, \mathbf{A}^{-1})$

Let $\mathbf{y} = (y_1, y_2, ..., y_n)^T$

Component-wise Gibbs updates each component in sequence from the (normal) conditional distributions

One 'sweep' over all n components can be written

$$\mathbf{y}^{(k+1)} = -\mathbf{D}^{-1}\mathbf{L}\mathbf{y}^{(k+1)} - \mathbf{D}^{-1}\mathbf{L}^T\mathbf{y}^{(k)} + \mathbf{D}^{-1/2}\mathbf{z}^{(k)}$$

where: $\mathbf{D} = \mathrm{diag}(\mathbf{A})$, $\mathbf{L}$ is the strictly lower triangular part of $\mathbf{A}$, $\mathbf{z}^{(k-1)} \sim \mathrm{N}(\mathbf{0}, \mathbf{I})$

$$\mathbf{y}^{(k+1)} = \mathbf{G}\mathbf{y}^{(k)} + \mathbf{c}^{(k)}$$

$\mathbf{c}^{(k)}$ is iid 'noise' with zero mean, finite covariance

---

Goodman & Sokal 1989

Matrix formulation of Gibbs sampling from $\mathrm{N}(0, \mathbf{A}^{-1})$

Let $\mathbf{y} = (y_1, y_2, ..., y_n)^T$

Component-wise Gibbs updates each component in sequence from the (normal) conditional distributions

One 'sweep' over all n components can be written

$$\mathbf{y}^{(k+1)} = -\mathbf{D}^{-1}\mathbf{L}\mathbf{y}^{(k+1)} - \mathbf{D}^{-1}\mathbf{L}^T\mathbf{y}^{(k)} + \mathbf{D}^{-1/2}\mathbf{z}^{(k)}$$

where: $\mathbf{D} = \mathrm{diag}(\mathbf{A})$, $\mathbf{L}$ is the strictly lower triangular part of $\mathbf{A}$, $\mathbf{z}^{(k-1)} \sim \mathrm{N}(\mathbf{0}, \mathbf{I})$

$$\mathbf{y}^{(k+1)} = \mathbf{G}\mathbf{y}^{(k)} + \mathbf{c}^{(k)}$$

$\mathbf{c}^{(k)}$ is iid 'noise' with zero mean, finite covariance

Spot the similarity to Gauss-Seidel iteration for solving $\mathbf{A}\mathbf{x} = \mathbf{b}$

$$\mathbf{x}^{(k+1)} = -\mathbf{D}^{-1}\mathbf{L}\mathbf{x}^{(k+1)} - \mathbf{D}^{-1}\mathbf{L}^{\mathsf{T}}\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{b}$$

Goodman & Sokal 1989; Amit & Grenander 1991

Gibbs converges $\iff$ solver converges

Theorem 1 *Let* $\mathbf{A} = \mathbf{M} - \mathbf{N}$, $\mathbf{M}$ *invertible. The stationary linear solver*

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{M}^{-1}\mathbf{N}\mathbf{x}^{(k)} + \mathbf{M}^{-1}\mathbf{b} \\ &= \mathbf{G}\mathbf{x}^{(k)} + \mathbf{M}^{-1}\mathbf{b}\end{aligned}$$

converges, if and only if the random iteration

$$\begin{aligned}\mathbf{y}^{(k+1)} &= \mathbf{M}^{-1}\mathbf{N}\mathbf{y}^{(k)} + \mathbf{M}^{-1}\mathbf{c}^{(k)} \\ &= \mathbf{G}\mathbf{y}^{(k)} + \mathbf{M}^{-1}\mathbf{c}^{(k)}\end{aligned}$$

converges in distribution. Here $\mathbf{c}^{(k)} \overset{iid}{\sim} \pi_n$ *has zero mean and finite variance*

Proof. Both converge iff $\varrho(\mathbf{G}) < 1$ $\square$

Convergent splittings generate convergent (generalized) Gibbs samplers

Mean converges with asymptotic convergence factor $\varrho(\mathbf{G})$, covariance with $\varrho(\mathbf{G})^2$

Young 1971 Thm 3-5.1, Duflo 1997 Thm 2.3.18-4, Goodman & Sokal, 1989, Galli & Gao 2001
F Parker 2012

Some not so common Gibbs samplers for $\mathrm{N}(0, \mathbf{A}^{-1})$

splitting/sampler	$\mathbf{M}$	$\mathbf{Var}\left(\mathbf{c}^{(k)}\right) = \mathbf{M}^T + \mathbf{N}$	converge if
Richardson	$\frac{1}{\omega}\mathbf{I}$	$\frac{2}{\omega}\mathbf{I} - \mathbf{A}$	$0 < \omega < \frac{2}{\varrho(\mathbf{A})}$
Jacobi	$\mathbf{D}$	$2\mathbf{D} - \mathbf{A}$	$\mathbf{A}$ SDD
GS/Gibbs	$\mathbf{D} + \mathbf{L}$	$\mathbf{D}$	always
SOR/B&F	$\frac{1}{\omega}\mathbf{D} + \mathbf{L}$	$\frac{2-\omega}{\omega}\mathbf{D}$	$0 < \omega < 2$
SSOR/REGS	$\frac{\omega}{2-\omega}\mathbf{M}_{\mathsf{SOR}}\mathbf{D}^{-1}\mathbf{M}_{\mathsf{SOR}}^{\mathsf{T}}$	$\frac{\omega}{2-\omega}\left(\mathbf{M}_{\mathsf{SOR}}\mathbf{D}^{-1}\mathbf{M}_{\mathsf{SOR}}^{T} + \mathbf{N}_{\mathsf{SOR}}^{T}\mathbf{D}^{-1}\mathbf{N}_{\mathsf{SOR}}\right)$	$0 < \omega < 2$

Good choice has: convenient to solve $\mathbf{Mu} = \mathbf{r}$ and sample from $\mathrm{N}(0, \mathbf{M}^T + \mathbf{N})$

Relaxation parameter ω can accelerate Gibbs

SOR: Adler 1981; Barone & Frigessi 1990, Amit & Grenander 1991, SSOR: Roberts & Sahu 1997

Some not so common Gibbs samplers for $\mathrm{N}(0, \mathbf{A}^{-1})$

splitting/sampler	$\mathbf{M}$	$\mathbf{Var}\left(\mathbf{c}^{(k)}\right) = \mathbf{M}^T + \mathbf{N}$	converge if
Richardson	$\frac{1}{\omega}\mathbf{I}$	$\frac{2}{\omega}\mathbf{I} - \mathbf{A}$	$0 < \omega < \frac{2}{\varrho(\mathbf{A})}$
Jacobi	$\mathbf{D}$	$2\mathbf{D} - \mathbf{A}$	$\mathbf{A}$ SDD
GS/Gibbs	$\mathbf{D} + \mathbf{L}$	$\mathbf{D}$	always
SOR/B&F	$\frac{1}{\omega}\mathbf{D} + \mathbf{L}$	$\frac{2-\omega}{\omega}\mathbf{D}$	$0 < \omega < 2$
SSOR/REGS	$\frac{\omega}{2-\omega}\mathbf{M}_{\mathsf{SOR}}\mathbf{D}^{-1}\mathbf{M}_{\mathsf{SOR}}^{\mathsf{T}}$	$\frac{\omega}{2-\omega}\left(\mathbf{M}_{\mathsf{SOR}}\mathbf{D}^{-1}\mathbf{M}_{\mathsf{SOR}}^{T} + \mathbf{N}_{\mathsf{SOR}}^{T}\mathbf{D}^{-1}\mathbf{N}_{\mathsf{SOR}}\right)$	$0 < \omega < 2$

Good choice has: convenient to solve $\mathbf{Mu} = \mathbf{r}$ and sample from $\mathrm{N}(0, \mathbf{M}^T + \mathbf{N})$

Relaxation parameter ω can accelerate Gibbs

SSOR is a forwards and backwards sweep of SOR to give a *symmetric* splitting

SOR: Adler 1981; Barone & Frigessi 1990, Amit & Grenander 1991, SSOR: Roberts & Sahu 1997

Controlling the error polynomial

The splitting

$$\mathbf{A} = \frac{1}{\tau}\mathbf{M} + \left(1 - \frac{1}{\tau}\right)\mathbf{M} - \mathbf{N}$$

gives the iteration operator

$$\mathbf{G}_\tau = \left(\mathbf{I} - \tau\mathbf{M}^{-1}\mathbf{A}\right)$$

and error polynomial $Q_n(\lambda) = (1 - \tau\lambda)^n$

The *sequence* of parameters $\tau_1, \tau_2, \ldots, \tau_n$ gives the error polynomial

$$Q_n(\lambda) = \prod_{l=1}^{m} (1 - \tau_l \lambda)$$

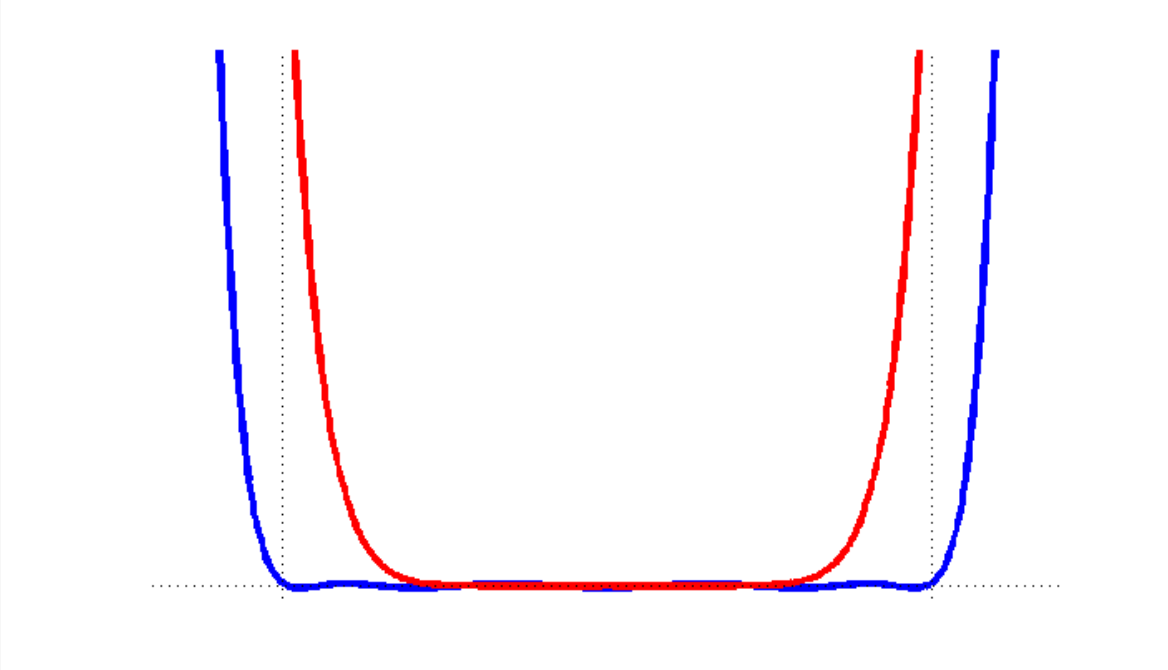
... so we can choose the zeros of Q_n

This gives a non-stationary solver $\equiv$ non-homogeneous Markov chain

---

Golub & Varga 1961, Golub & van Loan 1989, Axelsson 1996, Saad 2003, F & Parker 2012

The best (Chebyshev) polynomial



10 iterations, factor of 300 improvement

Choose

$$\frac{1}{\tau_l} = \frac{\lambda_n + \lambda_1}{2} + \frac{\lambda_n - \lambda_1}{2} \cos\left(\pi \frac{2l+1}{2p}\right) \qquad l = 0, 1, 2, \ldots, p-1$$

where λ_1 λ_n are extreme eigenvalues of $\mathbf{M}^{-1}\mathbf{A}$

Second-order accelerated sampler

First-order accelerated iteration turns out to be unstable

Numerical stability, and optimality at each step, is given by the second-order iteration

$$\mathbf{y}^{(k+1)} = (1-\alpha_k)\mathbf{y}^{(k-1)} + \alpha_k \mathbf{y}^{(k)} + \alpha_k \tau_k \mathbf{M}^{-1}(\mathbf{c}^{(k)} - \mathbf{A}\mathbf{y}^{(k)})$$

with α_k and τ_k chosen so error polynomial satisfies Chebyshev recursion.

Theorem 2 *2^{nd}-order solver converges $\Rightarrow$ 2^{nd}-order sampler converges (given correct noise distribution)*
Error polynomial is optimal, at each step, for both mean and covariance

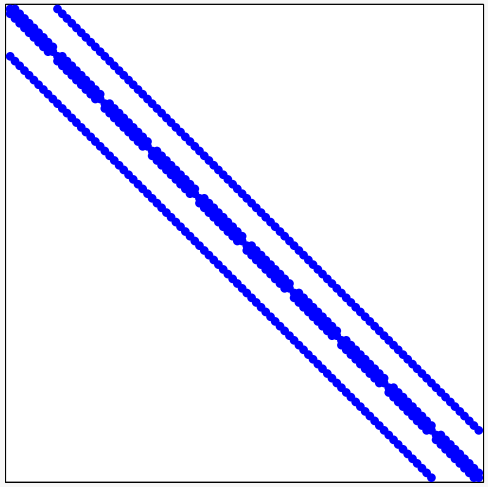
Asymptotic average reduction factor (Axelsson 1996) is

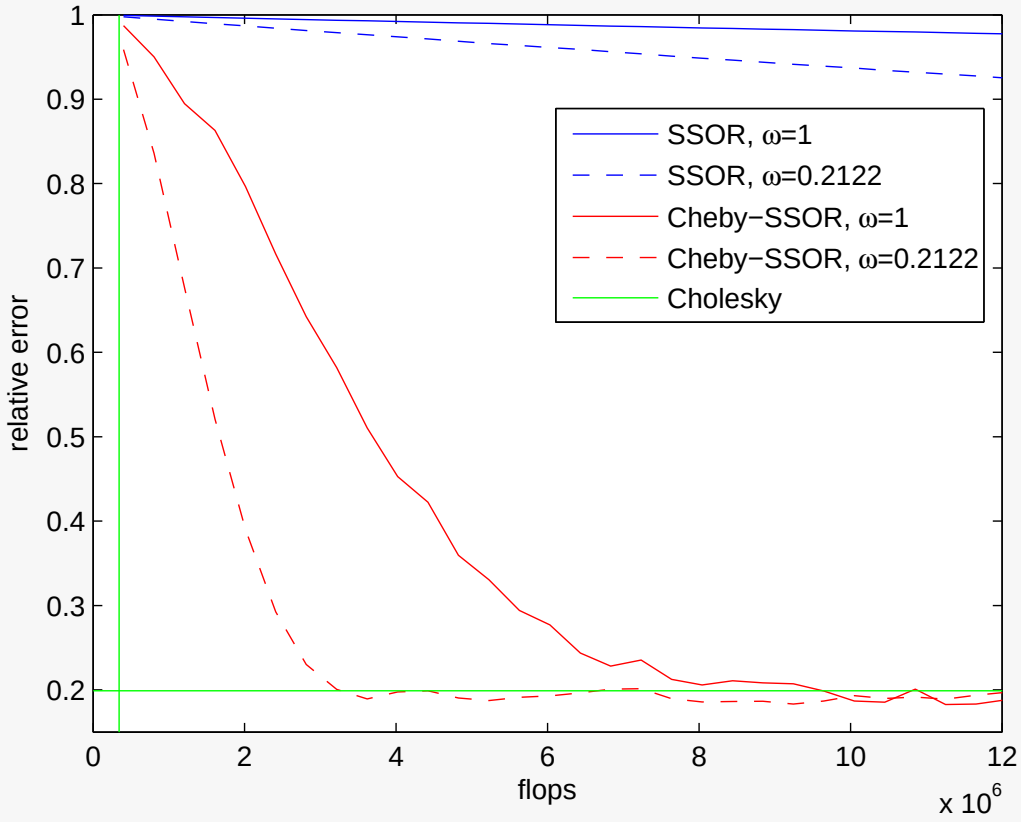
$$\sigma = \frac{1 - \sqrt{\lambda_1 / \lambda_n}}{1 + \sqrt{\lambda_1 / \lambda_n}}$$

Axelsson 1996, F & Parker 2012

10×10 lattice ($d = 100$) sparse precision matrix

$$[\mathbf{A}]_{ij} = 10^{-4}\delta_{ij} + \begin{cases} n_i & \text{if} \quad i = j \\ -1 & \text{if} \quad i \neq j \quad \text{and} \quad ||s_i - s_j||_2 \leq 1 \\ 0 & \text{otherwise} \end{cases} .$$





$\approx 10^4$ times faster

Polynomial acceleration of parameter estmation in ECT

Second-order Chebyshev acceleration of Gibbs give optimal convergence of first and second moments – given mean and inverse of covariance matrix $\mathbf{A} = \Sigma^{-1}$ where $\Sigma = \mathrm{cov}(\pi(x|y))$
We don't have $\mathbf{A}$ *so we adapt to it*.

Initialize $\mu = x_{\mathsf{MAP}}$ and $\mathbf{A} =$ Hessian of $-\log \pi$ at x_{MAP}

Algorithm 1 *At state x^l with values for τ and α:*

1. *Simulate x' via generalised scaled Gibbs sweep with parameter τ from x^l*
2. *Set $x^{l+1} = \alpha x' + (1-\alpha)x^{l-1}$*
3. *Evaluate recursion on α and τ*
4. *Update μ and $\mathbf{A}$ using empirical estimates (as AM)*

IACT for Gibbs was ≈ 3 sweeps bit slower than optimization
'IACT' after acceleration is ~ 1 sweep bit faster than optimization
passes all numerical tests, but no proof of convergence

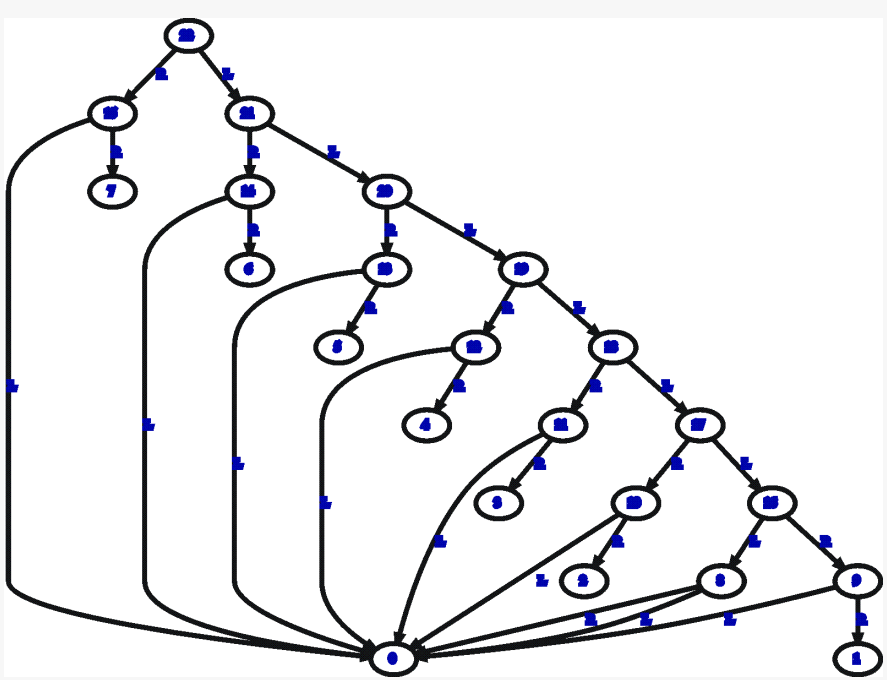
Why FPGAs ?

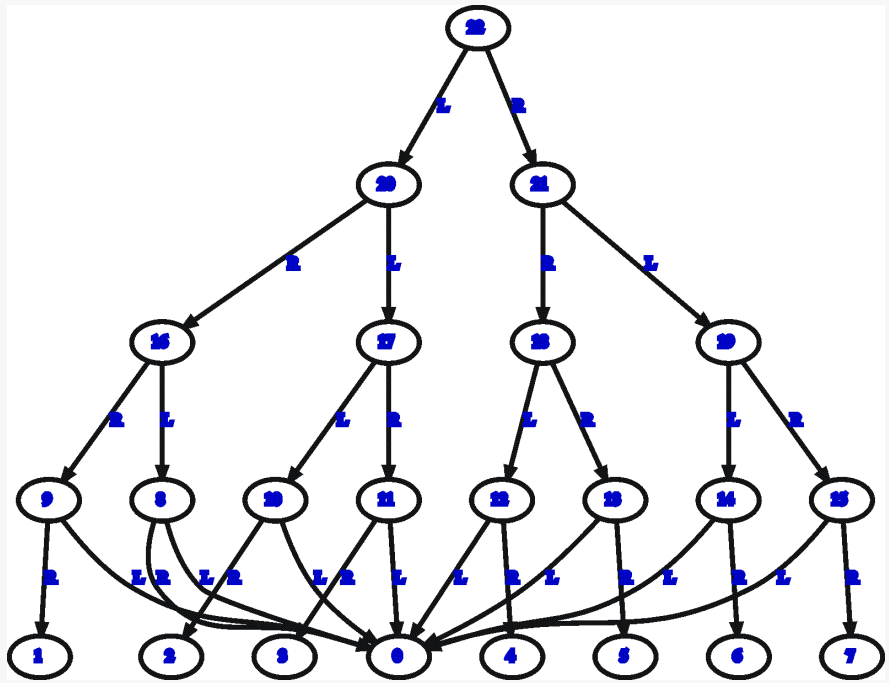
	CPU	GPU	FPGA
			
peak	50 Giga flops	500 Giga flops	100 Giga flops
sustained (FEM)	10%	1-5%	30-50%
power	150 W	300-400 W	30-50 W
tools	Fortran, C, ...	CUDA, OpenCL	no standard
% die	10 %	80%	60%

- GPU good for dense independent calculations
- FPGA compiles to silicon, cookie-cutter gives massive parallelization

Numerics in an FPGA

- Linux kernel in FPGA (10%) for i/o
- several groups active (MIT, USC, Otago, Accelogic, Drexel, Penn State, Microsoft, ...)
- developing tools for sparse linear algebra





- floor plan from high-level language (LAVA)

Optimizing compiler

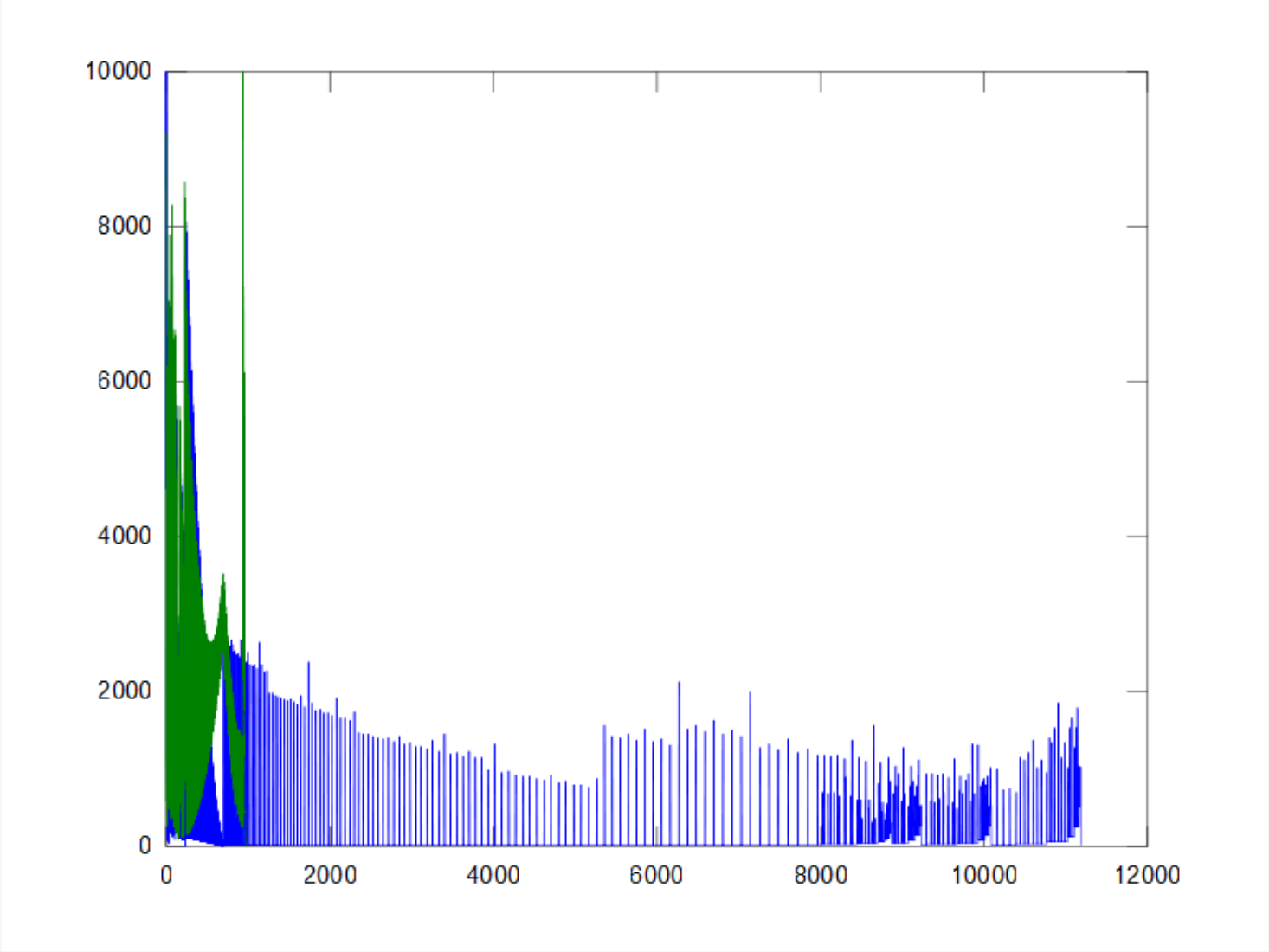
- partial evaluation at compile time
- extremely-aggressive code-unrolling
- multiple transformation and optimisation stages
  - unused-calculation elimination
  - constant propagation
  - strength reduction (subtract $\rightarrow$ add, fusion of negates, etc)
  - critical-path length reduction
  - data-locality optimisations
  - subtree pattern matching and operator fusion
- compiler back-end (currently generates C)

Comparisons for 'toy' problem

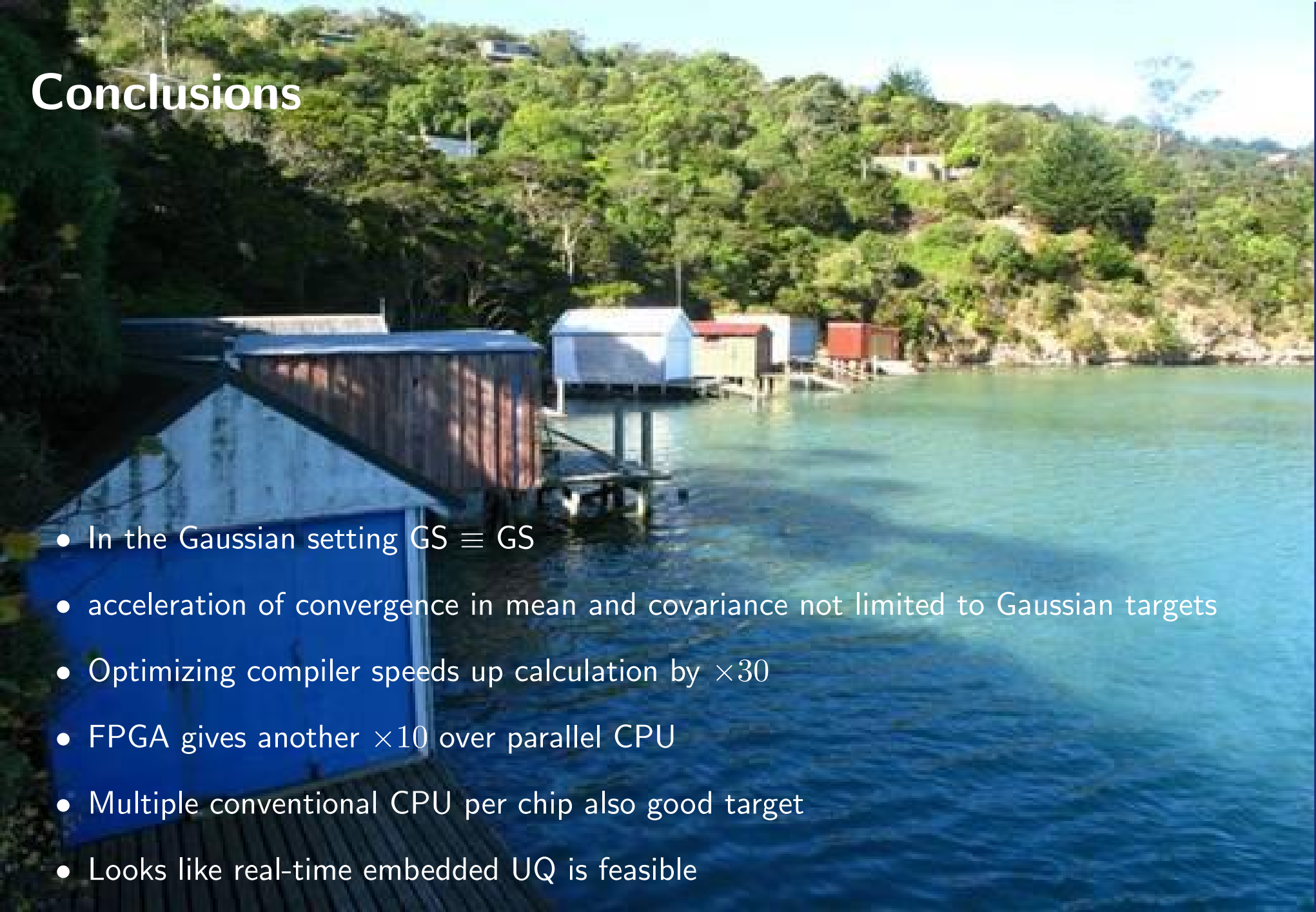
	FLOPS	Critical path
non-perm, no-opts	2.20 M	48 k
perm, no opts	1.90 M	11 k
perm, opt. for FLOPs	1.71 M	1.4 k
perm, opt. for crit-path	2.20 M	1 k

Fill-reducing permutation is amazing for parallelism!

Optimized critical path



Conclusions



- In the Gaussian setting GS $\equiv$ GS
- acceleration of convergence in mean and covariance not limited to Gaussian targets
- Optimizing compiler speeds up calculation by $\times 30$
- FPGA gives another $\times 10$ over parallel CPU
- Multiple conventional CPU per chip also good target
- Looks like real-time embedded UQ is feasible