Optimization and Water Resources Policy

C. T. Kelley
NC State University
tim_kelley@ncsu.edu
Joint with Karen Dillard, David Mokrauer, Greg Characklis,
Brian Kirsch

Supported by NSF, ARO, and the state of North Carolina

Fields Institute, May 2008

Outline

Why does North Carolina Care?

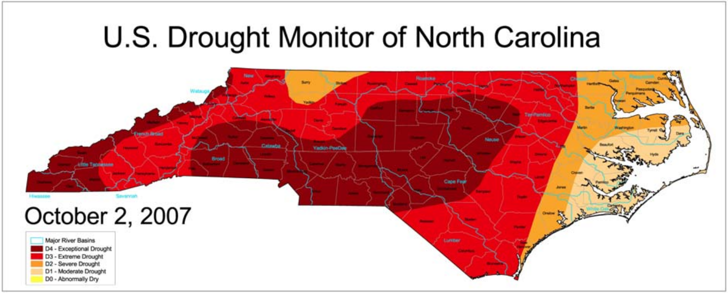
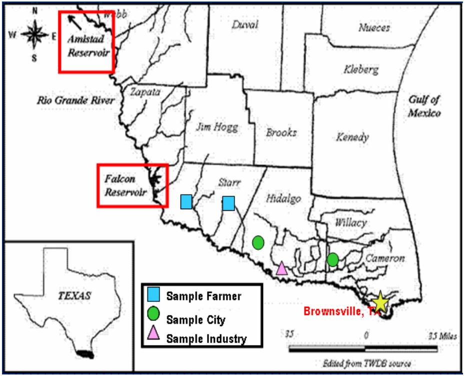


Figure 1. North Carolina Drought Management Council (http://www.ncdrought.org/).

Lower Rio Grande Valley



- City's water supply comes from
  - Permanent rights (hard to sell/buy, fixed on Jan 1)
  - Spot market leases (monthly decisions)
  - Options (purchase Jan 1, exercise May 31)
- Decisions: buy leases, buy/exercise options
  based on expected supply S_E and demand D_E
- Supply/Demand simulated by random sampling of history

Six Design Variables

- Number of rights and options: R, O
- Jan 1 – May 31: $S_E/D_E < \alpha_1$ lease (or exercise options in May) until $S_E/D_E = \beta_1$
- July 1 – Dec 31: $S_E/D_E < \alpha_2$ until $S_E/D_E = \beta_2$

Minimize Cost

$$\text{Cost} = Rp_R + Op_O + E(X)p_X + E\left(\sum_{months} L_t p_{L_t}\right)$$

R, O are amount of rights and options (design variables) purchased Jan 1.
Prices p_R, p_O known.
X = exercised options, price p_X (depends on data + design)
L_t = leases in month t, price p_L (depends on data + design)
Data randomly generated from historical record using several realizations.
Simple bound constraints and hidden constraints.

Variance Reduction via Control Variates: I

Assume that $f = E(\hat{f})$.
Estimate noise as standard error $\sigma/\sqrt{n}$ in $\hat{f}$
Objective: reduce variance σ; tune number of realizations n
Let Z be a random variable that is well correlated to $\hat{f}$ for which $E(Z)$ is known.
Define

$$\theta = \hat{f}(x) + c(Z - E(Z)),$$

and c is tuned to minimize variance.
We can use $f = E(\theta)$ since $E(Z)$ is known.

Variance Reduction via Control Variates: II

Since

$$Var(\theta) = Var(\hat{f}) - 2cCov(\hat{f}, Z) + c^2 Var(Z)$$

the optimal value of c is

$$c^* = Cov(\hat{f}, Z)/Var(Z).$$

We get an estimate of $Var(\theta)$ too:

$$Var(\theta) = (1 - \rho^2) Var(\hat{f})$$

where ρ is the correlation between $\hat{f}$ and Z.

Variance Reduction via Control Variates: III

One can use more than one control variate:

$$\theta = \hat{f} + \sum c_i (Z_i - E(Z_i)).$$

So, how to you invent the Z's?

- Lease price at beginning of the year has known expectation (data).
- Net supply at end of April (prior to option exercise month). Compute from inflows and demand. Expectations known (data).

Then estimate $Var(\hat{f})$ and $Cov(\hat{f}, Z)$ with a small pilot study.

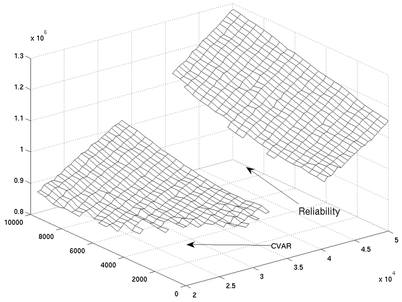
Benefits of Variance Reduction

- Smoother landscape for give number of realizations
- Reduced realizations per function call by 50%
- Promise of coupling to optimization algorithm

Hidden Constraints

- Reliability
  - Probability of serious shortage $< .005$ (every 16.7 years).
  - Tested after simulation runs.
- Conditional value-at-risk (CVAR)
  - Mean of costs above 95th percentile.
  - CVAR $\leq 1.25\times$ total portfolio costs
  - Tested after simulation runs.

Optimization Landscape



Results of Optimization

- Water Resources
  - Leases/Options vs only permanent rights
    lower annual costs with no reliability penalty
  - Leases alone reduce costs even more
    but with higher variability
- Optimization
  - Implicit filtering was robust: could cross gaps
    restarts not necessary
    final costs within 3% for varying initial iterates
  - Feasible initial iterate important

Algorithm Design

- Algorithms based on coordinate search
- Termination
  - budget, estimate of necessary conditions, . . .
- Hidden constraints
- Trivial(?) Parallelism

Noisy Optimization Problems

- Implicit filtering is designed for noisy problems.
  - Perturbations of smooth problems
  - Internal iterations
  - Stochastic models
- Handles Hidden Constraints or failed points

Stencil-Based Sampling Methods

- Given x_c, scale h_c, and directions $V = (v_1, \ldots, v_k)$
  - Evaluate f at $x_c + h_c v_j$ for $1 \leq j \leq k$.
  - Assign NaN, Inf, or artificial value to failed point.
  - Decide what to do next.
  - Example: Coordinate Search
    - Take best point $f(x_c + h_c v_j) = min$ unless ...
    - stencil failure: $f(x_c)$ is best. In that case, reduce h_c.

Coordinate search implicitly filters out noise.

Convergence Theory: smooth f

If

- f has bounded level sets
- $f \in C^1$
- V is a positive basis

Stencil failure implies $\nabla f = O(h)$ and so ...

- $h_n \to 0$ (bounded level sets) therefore
- $\nabla f(x_n) \to 0$

Due to many authors in various forms.
True, but not very exciting.

Convergence Theory: noisy f

If

- f has bounded level sets
- $f = f_{smooth} + \phi$, $f_{smooth} \in C^1$
- V is a positive basis

then $h_n \to 0$, and if

$$\phi(x_n)/h_n \to 0$$

then

$$\nabla f_{smooth}(x_n) \to 0$$

Theory (Audet-Dennis) for Lipschitz f.

Example: coordinate search

Sample f at x on a stencil centered at x, scale$=h$

$$S(x,h) = \{x \pm he_i\}$$

- Move to the best point.
- If x is the best point, reduce h.

Necessary Conditions: No legal direction points downhill (which is why you reduce h).

What if x is the best point?

Smooth Objective

If $f(x) \leq \min_{z \in S(x,h)} f(z)$ **(stencil failure)**
then
$\|\nabla f(x)\| = O(h)$

So, if (x_n, h_n) are the points/scales generated by coordinate search and f has bounded level sets, then

- $h_n \to 0$ (finitely many grid points/level) and therefore
- any limit point of $\{x_n\}$ is a critical point of f.

Not a method for smooth problems.

Model Problem

motivated by the pictures

$$\min_{R^N} f$$

$$f = f_s + \phi$$

- f_s smooth, easy to minimize; ϕ noise
- N is small, f is typically costly to evaluate.
- f has multiple local minima
  which trap most gradient-based algorithms.

Convergence?

Stencil failure implies that

$$\|\nabla f_s(x_n)\| = O\left(h_n + \frac{\|\phi\|_{S(x_n,h_n)}}{h_n}\right)$$

where

$$\|\phi\|_{S(x,h)} = \max_{z \in S} |\phi(z)|.$$

Bottom line

So, if (x_n, h_n) are the points/scales generated by coordinate search, f has bounded level sets, **and**

$$\lim_{n \to \infty} (h_n + h_n^{-1} \|\phi\|_{S(x,h_n)}) = 0$$

then

- $h_n \to 0$ (finitely many grid points) and therefore
- any limit point of $\{x_n\}$ is a critical point of f.

Simplex Gradient

- Coordinate search is building an approximation of the gradient.
  - Let $\delta f(x : V)_j = f(x + hv_j) - f(x)$
  - Define the simplex gradient

$$Df(x : h, V) = h^{-1}(V^T)^{\dagger}\delta f(x : V)$$

- $Df(x : h, V)$ is the minimum norm least squares solution of

$$\min \|hV^T Df - \delta f\|$$

- If V is a one-sided or centered difference stencil, you get the usual difference gradient.

Exploiting the Simplex Gradient

Since we can compute the simplex gradient with no extra effort, we can

- add $-Df$ to the stencil for the next search,
- do a line search in that direction and mimic steepest descent,
- build a quasi-Newton model Hessian and use that direction, and/or
- reduce h when $\|Df\|$ is small.

Implicit filtering reduces h when $\|Df(x : h, V)\| \leq \tau h$ and uses a quasi-Newton model Hessian.

Quasi-Newton Acceleration

We begin with $H = I$. For the unconstrained case we use two updates.

- SR1

$$H_+ = H_c + \frac{(y - H_c s)(y - H_c s)^T}{(y - H_c s)^T s}$$

- BFGS

$$H_+ = H_c + \frac{yy^T}{y^T s} - \frac{(H_c s)(H_c s)^T}{s^T H_c s}$$

where $s = x_+ - x_c$ and

$$y = Df(x_+ : h_c, V) - Df(x_c : h_c, V)$$

QN Hessians make a huge difference in performance.
Obvious extension for bound constraints.

Implicit Filtering

imfilter$(x, f, pmax, \tau, \{h_n\}, amax)$

for $k = 0, \dots$ **do**

`fdquasi`$(x, f, pmax, \tau, h_n, amax)$

end for

$pmax$, τ, $amax$ are termination parameters

fdquasi = finite difference quasi-Newton method using a simplex gradient $Df(x : h, v)$

fdquasi$(x, f, pmax, \tau, h, amax)$

$p = 1$
while $p \leq pmax$ and $\|Df(x : h, v)\| \geq \tau h$ **do**
  compute f and $Df(x : h, v)$
  terminate with success on stencil failure
  update the model Hessian H if appropriate; solve $Hd = -Df(x : h, v)f(x)$
  use a backtracking line search, with at most *amax* backtracks, to find a step length λ
  terminate with failure on $>$ *amax* backtracks
  $x \leftarrow x + \lambda d$; $p \leftarrow p + 1$
end while
if $p > pmax$ report iteration count failure
if $p <= pmax$ report success

Application of Theory

Let (x_n, h_n) be the sequence from implicit filtering.
If

- ∇f_s is Lipschitz continuous.
- $\lim_{n \to \infty}(h_n + h_n^{-1} \|\phi\|_{S(x,h_n)}) = 0$
- fdquasi terminates with success for infinitely many n.

then any limit point of $\{x_n\}$ is a critical point of f_s.

Least squares gradient/Hessian approximation

- Split $H = A + C$. Approximate A with quasi-Newton. Compute C (N_C degrees of freedom).
- Compute part of the Hessian using

$$\delta f(x_c : h, V) = h_c V^T \nabla f(x_c) + \frac{h_c^2}{2} V^T H V + O(h^2)$$

by solving the least squares problem

$$\min \|\delta f(x_c : h_c, V) - h_c V^T Df(x : h, V) - \frac{h_c^2}{2} V^T C V\|$$

for N_C unknowns.

Powell (2006) has a similar idea for DFO.

Diagonal C; V central difference

- $N_C = N$, so if there are no failed points

$$\min \| f(x_+) - V^T \overline{Df} - \frac{1}{2} V^T C V \|$$

  is a square system.

Not a finite difference approximation of Hessian diagonal.

Feedback to objective

You can pass h to f. This helps if ...

- f can control its own accuracy via
  - tolerance in ODE/DAE/PDE models, or
  - number of realizations n in Monte Carlo models.
- f knows its own limiting resolution, so
  - f can tell you when to terminate the iteration.

Using the estimate of the noise

$$Df(x : h, V) = \nabla f_s + O(h + \|\phi\|/h)$$

so the noise renders the gradient estimate useless when

$$\sigma/\sqrt{n} \approx \|\phi\| \geq \|\nabla f\| h \approx \|Df(x : h, V)\| h.$$

So, if f can estimate σ, then one can tune n so that

$$\sigma/\sqrt{n} \leq M_{tune} \|Df(x : h, V)\| h.$$

Termination

Even if the gradient estimate is poor, the search may still produce good results.
However, the search fails if decreases in f do not reflect decreases in f_s:

$$\sigma/\sqrt{n} \approx \|\phi\| \geq |\delta f(x : V)_j|$$

for all j.
So, if f can tell the code what $\sigma/\sqrt{n}$ is, the code can terminate if the estimated noise is larger than the variation in f.

New Mode for Parallelism

- User managed parallelism
- Call multiple instances of objective
- Sample mpi/c/linux cluster code available coming soon

Research Issues

- Algorithms to locate neighborhoods of minimizers
- Analysis
- Asymptotic theoretical results vs tight computational budget
- Parallel computing: I/O, load balancing
- Designing feedback between function and optimization method
  - Noise estimation and control
  - Termination of iteration
- Other Applications
  Electronics, Automotive, Algorithm Tuning

How to get imfil.m

- Email me at `tim_kelley@ncsu.edu`
- Get it directly from
  `http://www4.ncsu.edu/~ctk/imfil.html`

Under construction.