Gradient-based Adaptive Stochastic Search

Enlu Zhou

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

November 5, 2014

Outline

Outline

1. Introduction
2. GASS for non-differentiable optimization
3. GASS for simulation optimization
4. Conclusions

Introduction

- We consider

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x)$$

Given any $x \in \mathcal{X}$, $H(x)$ can be evaluated exactly.

Introduction

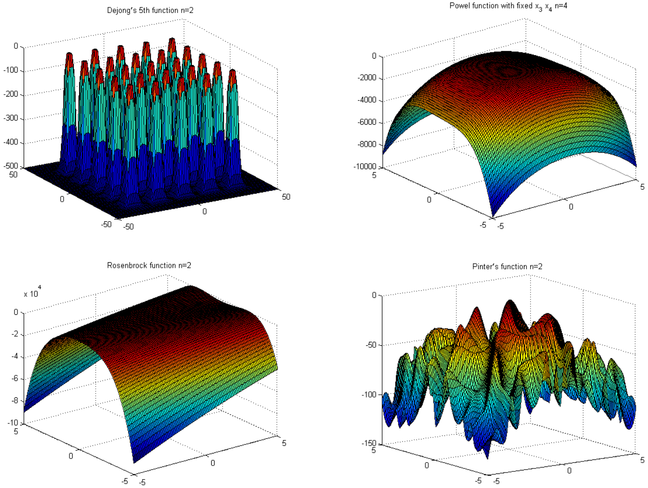
- We consider

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x)$$

  Given any $x \in \mathcal{X}$, $H(x)$ can be evaluated exactly.

- We are interested in objective functions:
  - lack structural properties (such as convexity and differentiability)
  - have multiple local optima
  - only be assessed by “black-box” evaluation

Examples of Objective Functions



Background

- Stochastic Search: use randomized mechanism to generate a sequence of iterates

Background

- Stochastic Search: use randomized mechanism to generate a sequence of iterates

  e.g., simulated annealing (Krikpatrick et al. 1983), genetic algorithms (Goldberg 1989), tabu search (Glover 1990), nested partitions method (Shi and Ólafsson 2000), pure adaptive search (Zabinsky 2003), sequential Monte Carlo simulated annealing (Zhou and Chen 2011), model-based algorithms (survey by Zlochin et al. 2004).

Background

- Stochastic Search: use randomized mechanism to generate a sequence of iterates

  e.g., simulated annealing (Krikpatrick et al. 1983), genetic algorithms (Goldberg 1989), tabu search (Glover 1990), nested partitions method (Shi and Ólafsson 2000), pure adaptive search (Zabinsky 2003), sequential Monte Carlo simulated annealing (Zhou and Chen 2011), model-based algorithms (survey by Zlochin et al. 2004).

- Model-based Algorithms: generate candidate solutions from a sampling distribution (i.e., probabilistic model)

Background

- Stochastic Search: use randomized mechanism to generate a sequence of iterates

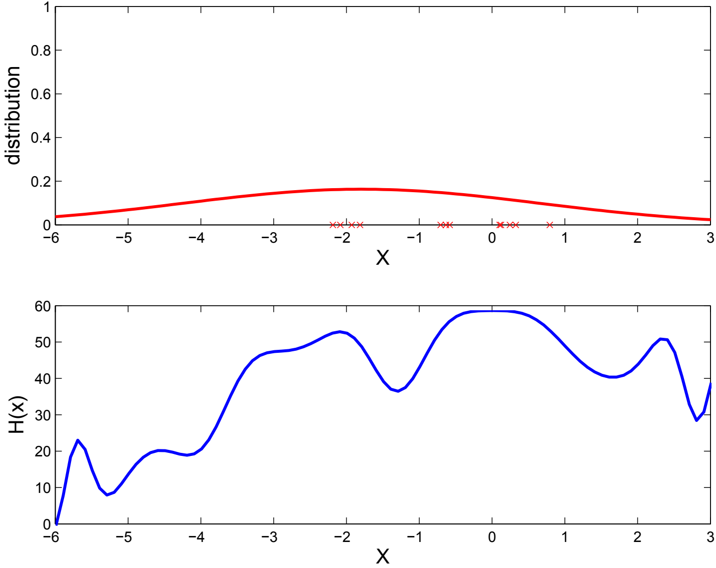
  e.g., simulated annealing (Krikpatrick et al. 1983), genetic algorithms (Goldberg 1989), tabu search (Glover 1990), nested partitions method (Shi and Ólafsson 2000), pure adaptive search (Zabinsky 2003), sequential Monte Carlo simulated annealing (Zhou and Chen 2011), model-based algorithms (survey by Zlochin et al. 2004).

- Model-based Algorithms: generate candidate solutions from a sampling distribution (i.e., probabilistic model)

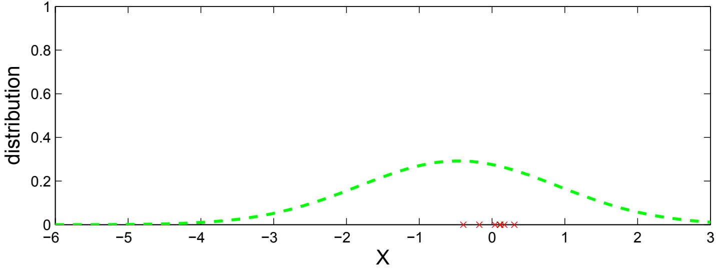
  e.g., ant colony optimization (Dorigo and Gambardella 1997), annealing adaptive search (Romeijn and Smith 1994), estimation of distribution algorithms (Muhlenbein and Paaß 1996), the cross-entropy method (Rubinstein 1997), model reference adaptive search (Hu et al. 2007).

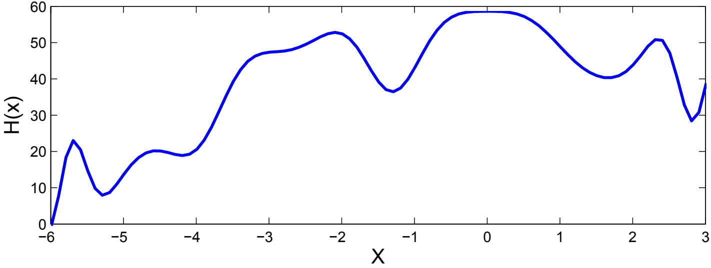
Model-based optimization

Model-based optimization

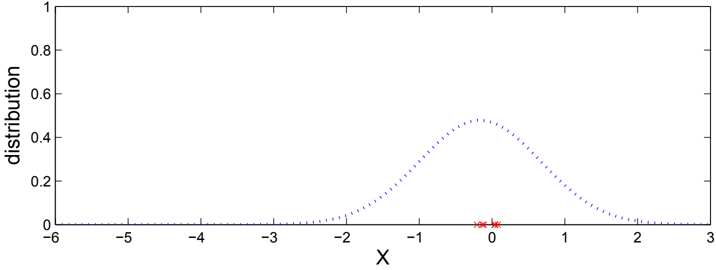


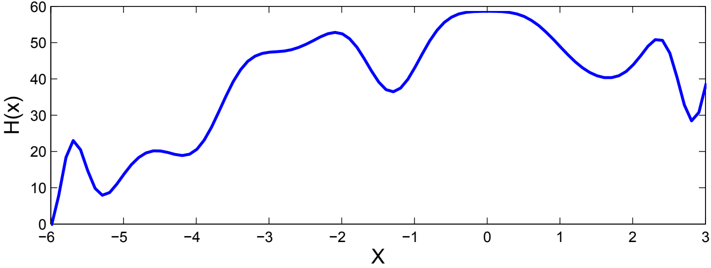
Model-based optimization



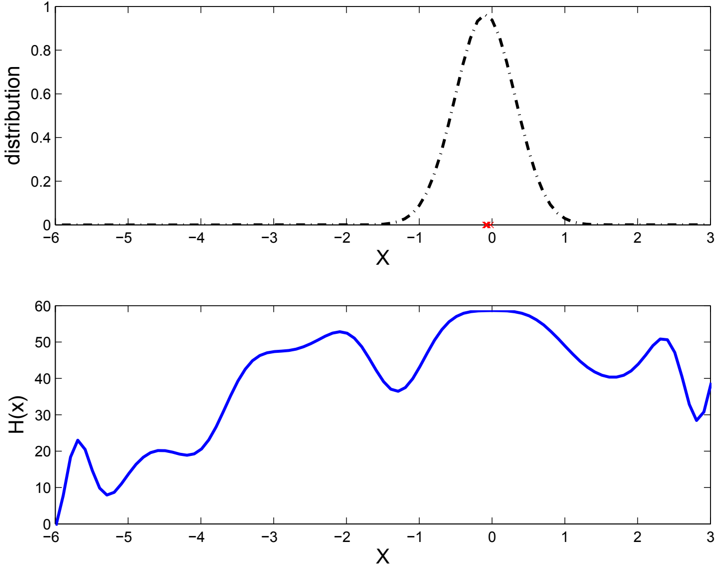


Model-based optimization





Model-based optimization



Outline

1. Introduction
2. GASS for non-differentiable optimization
3. GASS for simulation optimization
4. Conclusions

Reformulation

- Original problem:

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x), \quad \mathcal{X} \subseteq \mathbb{R}^n.$$

Reformulation

- Original problem:

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x), \quad \mathcal{X} \subseteq \mathbb{R}^n.$$

- Let $\{f(x;\theta)\}$ be a parameterized family of probability density functions on $\mathcal{X}$.

$$\int H(x) f(x;\theta) dx \leqslant H(x^*) \triangleq H^*, \quad \forall \theta \in \mathbb{R}^d.$$

Reformulation

- Original problem:

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x), \quad \mathcal{X} \subseteq \mathbb{R}^n.$$

- Let $\{f(x;\theta)\}$ be a parameterized family of probability density functions on $\mathcal{X}$.

$$\int H(x) f(x;\theta) dx \leqslant H(x^*) \triangleq H^*, \quad \forall \theta \in \mathbb{R}^d.$$

"=" is achieved if and only if $\exists\ \theta^*$ s.t. the probability mass of $f(x;\theta^*)$ is concentrated on a subset of the optimal solutions.

Reformulation

- Original problem:

$$x^* \in \arg\max_{x \in \mathcal{X}} H(x), \quad \mathcal{X} \subseteq \mathbb{R}^n.$$

- Let $\{f(x;\theta)\}$ be a parameterized family of probability density functions on $\mathcal{X}$.

$$\int H(x) f(x;\theta) dx \leqslant H(x^*) \triangleq H^*, \quad \forall \theta \in \mathbb{R}^d.$$

  "=" is achieved if and only if $\exists\ \theta^*$ s.t. the probability mass of $f(x;\theta^*)$ is concentrated on a subset of the optimal solutions.

- New problem:

$$\theta^* \in \arg\max_{\theta \in \mathbb{R}^d} \int H(x) f(x;\theta) dx.$$

Why reformulation?

- Possible Scenarios:

Original Problem $\arg\max_{x \in \mathcal{X}} H(x)$	New Problem $\arg\max_{\theta} \int H(x) f(x; \theta) dx$
Discrete in x	Continuous in θ
Non-differentiable in x	Differentiable in θ

Why reformulation?

- Possible Scenarios:

Original Problem $\arg\max_{x \in \mathcal{X}} H(x)$	New Problem $\arg\max_\theta \int H(x) f(x;\theta) dx$
Discrete in x	Continuous in θ
Non-differentiable in x	Differentiable in θ

- Incorporate model-based optimization into gradient-based optimization:
  1). Generate candidate solutions from $f(\cdot;\theta)$ on the solution space $\mathcal{X}$.
  2). Use a gradient-based method to update the parameter θ.

Why reformulation?

- Possible Scenarios:

Original Problem $\arg\max_{x\in\mathcal{X}} H(x)$	New Problem $\arg\max_{\theta} \int H(x)f(x;\theta)dx$
Discrete in x	Continuous in θ
Non-differentiable in x	Differentiable in θ

- Incorporate model-based optimization into gradient-based optimization:
  1). Generate candidate solutions from $f(\cdot;\theta)$ on the solution space $\mathcal{X}$.
  2). Use a gradient-based method to update the parameter θ.

- Combine the robustness of model-based optimization with the relative fast convergence of gradient-based optimization.

More reformulation

- For an arbitrary but fixed $\theta' \in \mathbb{R}^d$, define the function

$$l(\theta; \theta') \triangleq \ln\left(\int S_{\theta'}(H(x))f(x;\theta)dx\right).$$

More reformulation

- For an arbitrary but fixed $\theta' \in \mathbb{R}^d$, define the function

$$l(\theta;\theta') \triangleq \ln\left(\int S_{\theta'}(H(x))f(x;\theta)dx\right).$$

- The shape function $S_\theta(\cdot) : \mathbb{R} \to \mathbb{R}^+$ is chosen to ensure

$$0 < l(\theta;\theta') \leq \ln\left(S_{\theta'}(H^*)\right) \quad \forall\,\theta,$$

  and "=" is achieved if $\exists$ a θ^* s.t. the probability mass of $f(x;\theta^*)$ is concentrated on a subset of global optima.

More reformulation

- For an arbitrary but fixed $\theta' \in \mathbb{R}^d$, define the function

$$l(\theta;\theta') \triangleq \ln\left(\int S_{\theta'}(H(x))f(x;\theta)dx\right).$$

- The shape function $S_\theta(\cdot) : \mathbb{R} \to \mathbb{R}^+$ is chosen to ensure

$$0 < l(\theta;\theta') \leq \ln\left(S_{\theta'}(H^*)\right) \quad \forall\,\theta,$$

  and "=" is achieved if $\exists$ a θ^* s.t. the probability mass of $f(x;\theta^*)$ is concentrated on a subset of global optima.

- So consider

$$\max_\theta l(\theta;\theta').$$

Parameter updating

- Suppose $\{f(\cdot;\theta)\}$ is an exponential family of densities, i.e.,

$$f(x;\theta) = \exp\{\theta^T T(x) - \phi(\theta)\}, \quad \phi(\theta) = \ln\{\int \exp(\theta^T T(x))dx\}.$$

Parameter updating

- Suppose $\{f(\cdot;\theta)\}$ is an exponential family of densities, i.e.,

$$f(x;\theta) = \exp\{\theta^T T(x) - \phi(\theta)\}, \quad \phi(\theta) = \ln\{\int \exp(\theta^T T(x))dx\}.$$

Then

$$\begin{aligned}
\nabla_\theta l(\theta;\theta')|_{\theta=\theta'} &= E_{p(\cdot;\theta')}[T(X)] - E_{\theta'}[T(X)], \\
\nabla_\theta^2 l(\theta;\theta')|_{\theta=\theta'} &= \mathrm{Var}_{p(\cdot;\theta')}[T(X)] - \mathrm{Var}_{\theta'}[T(X)],
\end{aligned}$$

where $p(x;\theta') \triangleq \frac{S_{\theta'}(H(x))f(x;\theta')}{\int S_{\theta'}(H(x))f(x;\theta')dx}$.

Parameter updating

- Suppose $\{f(\cdot;\theta)\}$ is an exponential family of densities, i.e.,

$$f(x;\theta) = \exp\{\theta^T T(x) - \phi(\theta)\}, \;\; \phi(\theta) = \ln\{\int \exp(\theta^T T(x))dx\}.$$

Then

$$\begin{aligned} \nabla_\theta l(\theta;\theta')|_{\theta=\theta'} &= E_{p(\cdot;\theta')}[T(X)] - E_{\theta'}[T(X)], \\ \nabla_\theta^2 l(\theta;\theta')|_{\theta=\theta'} &= \mathrm{Var}_{p(\cdot;\theta')}[T(X)] - \mathrm{Var}_{\theta'}[T(X)], \end{aligned}$$

where $p(x;\theta') \triangleq \frac{S_{\theta'}(H(x))f(x;\theta')}{\int S_{\theta'}(H(x))f(x;\theta')dx}$.

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k(\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1}\left(E_{p(\cdot;\theta_k)}[T(X)] - E_{\theta_k}[T(X)]\right),$$

$$\alpha_k > 0, \epsilon > 0.$$

Parameter updating

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k (\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1} \left(E_{p(\cdot;\theta_k)}[T(X)] - E_{\theta_k}[T(X)] \right),$$

$$\alpha_k > 0, \epsilon > 0.$$

Parameter updating

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k(\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1}\left(E_{p_{(\cdot;\theta_k)}}[T(X)] - E_{\theta_k}[T(X)]\right),$$

$$\alpha_k > 0, \epsilon > 0.$$

$\mathrm{Var}_{\theta_k}[T(X)] = E[(\nabla_\theta \ln f(X;\theta_k))^2]$ is the Fisher information matrix, leading to the following facts:

Parameter updating

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k (\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1} \left(E_{p(\cdot;\theta_k)}[T(X)] - E_{\theta_k}[T(X)] \right),$$

$$\alpha_k > 0, \epsilon > 0.$$

$\mathrm{Var}_{\theta_k}[T(X)] = E[(\nabla_\theta \ln f(X;\theta_k))^2]$ is the Fisher information matrix, leading to the following facts:

- $\mathrm{Var}_{\theta_k}[T(X)]^{-1}$ is the minimum-variance step size in stochastic approximation.

Parameter updating

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k(\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1}\left(E_{p(\cdot;\theta_k)}[T(X)] - E_{\theta_k}[T(X)]\right),$$

$$\alpha_k > 0, \epsilon > 0.$$

$\mathrm{Var}_{\theta_k}[T(X)] = E[(\nabla_\theta \ln f(X;\theta_k))^2]$ is the Fisher information matrix, leading to the following facts:

- $\mathrm{Var}_{\theta_k}[T(X)]^{-1}$ is the minimum-variance step size in stochastic approximation.
- $\mathrm{Var}_{\theta_k}[T(X)]^{-1}$ adapts the gradient step to our belief about promising regions. (Think about $T(X) = X$...)

Parameter updating

A Newton-like scheme for updating θ

$$\theta_{k+1} = \theta_k + \alpha_k(\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1} \left(E_{p_{(\cdot;\theta_k)}}[T(X)] - E_{\theta_k}[T(X)] \right),$$

$$\alpha_k > 0, \epsilon > 0.$$

$\mathrm{Var}_{\theta_k}[T(X)] = E[(\nabla_\theta \ln f(X; \theta_k))^2]$ is the Fisher information matrix, leading to the following facts:

- $\mathrm{Var}_{\theta_k}[T(X)]^{-1}$ is the minimum-variance step size in stochastic approximation.
- $\mathrm{Var}_{\theta_k}[T(X)]^{-1}$ adapts the gradient step to our belief about promising regions. (Think about $T(X) = X$...)
- $\mathrm{Var}_{\theta_k}[T(X)]^{-1} \nabla_\theta l(\theta; \theta_k)|_{\theta=\theta_k}$ is the gradient of $l(\theta; \theta_k)$ on the statistical manifold equipped with Fisher metric.

Main algorithm: GASS

Gradient-based Adaptive Stochastic Search (GASS)

- *Initialization:* set $k = 0$.
- *Sampling:* draw samples $x_k^i \overset{\text{iid}}{\sim} f(x; \theta_k), i = 1, 2, \ldots, N_k$.
- *Updating:* update the parameter θ according to

$$\theta_{k+1} = \theta_k + \alpha_k (\widehat{\mathrm{Var}}_{\theta_k}[T(X)] + \epsilon I)^{-1} (\widehat{E}_{p_k}[T(X)] - E_{\theta_k}[T(X)]),$$

  where $\widehat{\mathrm{Var}}_{\theta_k}[T(X)]$ and $\widehat{E}_{p_k}[T(X)]$ are estimates using the samples $\{x_k^i, i = 1, \ldots, N_k\}$.
- *Stopping:* If some stopping criterion is satisfied, stop and return the current best sampled solution; else, set $k := k + 1$ and go back to step 2).

Accelerated algorithm: GASS_avg

- GASS can be viewed as a stochastic approximation algorithm in finding θ^*.

Accelerated algorithm: GASS_avg

- GASS can be viewed as a stochastic approximation algorithm in finding θ^*.
- Accelerated GASS: use Polyak averaging with online feedback

$$
\begin{aligned}
\theta_{k+1} &= \theta_k + \alpha_k \left(\widehat{\mathrm{Var}}_{\theta_k}[T(X)] + \epsilon I \right)^{-1} \left(\widehat{E}_{p_k}[T(X)] - E_{\theta_k}[T(X)] \right) \\
&\quad + \alpha_k c(\bar{\theta}_k - \theta_k), \\
\bar{\theta}_k &= \frac{1}{k} \sum_{i=1}^{k} \theta_i.
\end{aligned}
$$

Convergence analysis

- The updating of θ can be rewritten in the form of a generalized Robbins-Monro iterates:

$$\theta_{k+1} = \theta_k + \alpha_k [D(\theta_k) + b_k + \xi_k],$$

where $D(\theta_k)$ is the gradient field, b_k is the bias term, and ξ_k is the noise term.

$$D(\theta_k) \quad = \quad \left(\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I\right)^{-1} \nabla_\theta l(\theta_k; \theta_k).$$

Convergence analysis

- The updating of θ can be rewritten in the form of a generalized Robbins-Monro iterates:

$$\theta_{k+1} = \theta_k + \alpha_k [D(\theta_k) + b_k + \xi_k],$$

  where $D(\theta_k)$ is the gradient field, b_k is the bias term, and ξ_k is the noise term.

$$D(\theta_k) \quad = \quad (\mathrm{Var}_{\theta_k}[T(X)] + \epsilon I)^{-1} \nabla_\theta l(\theta_k; \theta_k).$$

- It can be viewed as a noisy discretization of the ordinary differential equation (ODE)

$$\dot{\theta}_t = D(\theta_t), \quad t \geq 0.$$

Convergence analysis

$$\begin{aligned}\theta_{k+1} &= \theta_k + \alpha_k[D(\theta_k) + b_k + \xi_k],\\ \dot{\theta}_t &= D(\theta_t), \quad t \geq 0.\end{aligned}$$

Convergence analysis

$$\begin{aligned} \theta_{k+1} &= \theta_k + \alpha_k [D(\theta_k) + b_k + \xi_k], \\ \dot{\theta}_t &= D(\theta_t), \quad t \geq 0. \end{aligned}$$

Assumption

$\alpha_k \searrow 0$ as $k \to \infty$, $\sum_{k=0}^{\infty} \alpha_k = \infty$.

Convergence analysis

$$\begin{aligned} \theta_{k+1} &= \theta_k + \alpha_k[D(\theta_k) + b_k + \xi_k], \\ \dot{\theta}_t &= D(\theta_t), \quad t \geq 0. \end{aligned}$$

Assumption

$\alpha_k \searrow 0$ as $k \to \infty$, $\sum_{k=0}^{\infty} \alpha_k = \infty$.

Lemma 1

Under certain assumptions, $b_k \to 0$ $w.p.1$ as $k \to \infty$.

Convergence analysis

$$\begin{aligned} \theta_{k+1} &= \theta_k + \alpha_k[D(\theta_k) + b_k + \xi_k], \\ \dot{\theta}_t &= D(\theta_t), \quad t \geq 0. \end{aligned}$$

Assumption

$\alpha_k \searrow 0$ as $k \to \infty$, $\sum_{k=0}^{\infty} \alpha_k = \infty$.

Lemma 1

Under certain assumptions, $b_k \to 0$ $w.p.1$ as $k \to \infty$.

Lemma 2

Under certain assumptions, for any $T > 0$,

$$\lim_{k\to\infty} \left\{ \sup_{\{n: 0 \leq \sum_{i=k}^{n-1} \alpha_i \leq T\}} \left\| \sum_{i=k}^{n} \alpha_i \xi_i \right\| \right\} = 0, \quad w.p.1.$$

Convergence results

Theorem (Asymptotic Convergence)

Assume that $D(\theta_t)$ is continuous with a unique integral curve and some regularity conditions hold. Then the sequence $\{\theta_k\}$ converges to a limit set of the ODE w.p.1. Furthermore, if the limit sets of the ODE are isolated equilibrium points, then w.p.1 $\{\theta_k\}$ converges to a unique equilibrium point.

- Implication: GASS converges to a stationary point of $l(\theta; \theta')$.

Convergence results

Theorem (Asymptotic Convergence Rate)

Let $\alpha_k = \alpha_0/k^\alpha$ for $0 < \alpha < 1$. For a given constant $\tau > 2\alpha$, let $N_k = \Theta(k^{\tau-\alpha})$. Assume the convergence of the sequence $\{\theta_k\}$ occurs to a unique equilibrium point θ^* w.p.1. If Assumptions 1, 2, and 3 hold, then

$$k^{\frac{\tau}{2}}(\theta_k - \theta^*) \xrightarrow{dist} N(0, Q\mathcal{M}Q^T),$$

where Q is an orthogonal matrix such that $Q^T(-J_{\mathcal{L}}(\theta^*))Q = \Lambda$ with Λ being a diagonal matrix, and the $(i,j)^{th}$ entry of the matrix $\mathcal{M}$ is given by $\mathcal{M}_{(i,j)} = (Q^T\Phi\Sigma\Phi^TQ)_{(i,j)}(\Lambda_{(i,i)} + \Lambda_{(j,j)})^{-1}$.

- Implication: The asymptotic convergence rate of GASS is $O(1/\sqrt{k^\tau})$.

Numerical results

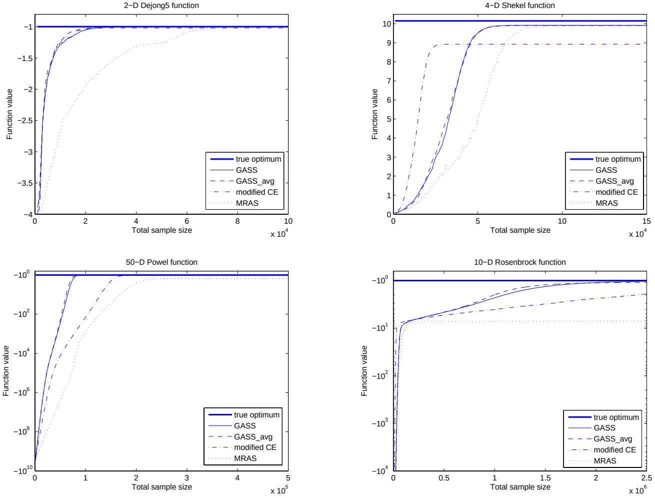


Figure : Comparison of average performance of GASS, GASS_avg, MRAS, and the modified CE.

Numerical results

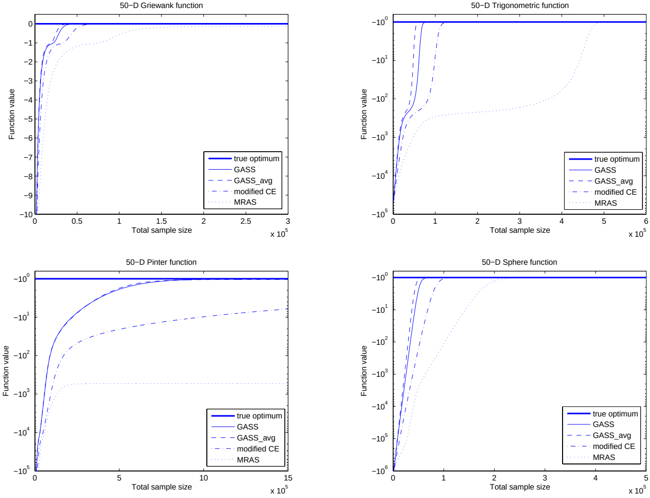


Figure : Comparison of average performance of GASS, GASS_avg, MRAS, and the modified CE.

Numerical results

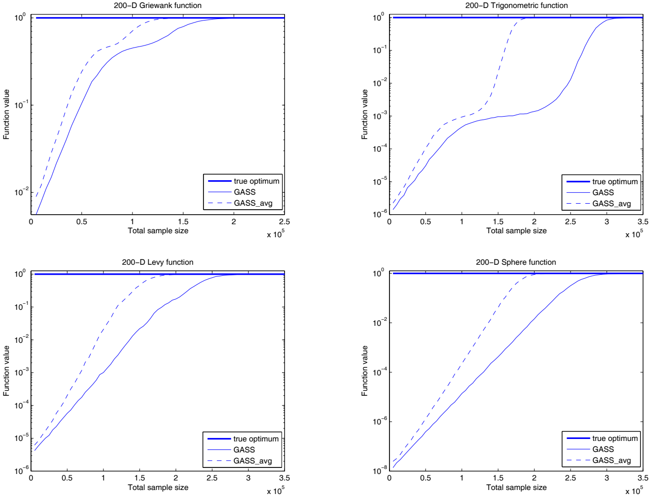


Figure : Average performance of GASS and GASS_avg on 200-dimensional benchmark problems.

Numerical results

- GASS_avg and GASS find the ϵ-optimal solutions in all the runs for 7 out of the 8 benchmark problems (except the Shekel function).

Numerical results

- GASS_avg and GASS find the ϵ-optimal solutions in all the runs for 7 out of the 8 benchmark problems (except the Shekel function).
- Accuracy: GASS_avg and GASS find better solutions than the modified CE method on badly-scaled functions and are comparable to the modified Cross Entropy method (Rubinstein 1998) on multi-modal functions; outperform Model Reference Adaptive Search (Hu et al. 2007) on all the problems.

Numerical results

- GASS_avg and GASS find the ϵ-optimal solutions in all the runs for 7 out of the 8 benchmark problems (except the Shekel function).

- Accuracy: GASS_avg and GASS find better solutions than the modified CE method on badly-scaled functions and are comparable to the modified Cross Entropy method (Rubinstein 1998) on multi-modal functions; outperform Model Reference Adaptive Search (Hu et al. 2007) on all the problems.

- Convergence speed: GASS_avg always converges faster than GASS; both are faster than MRAS on all the problems and faster than the modified CE on most problems.

Resource allocation in communication networks

- Q users may transmit or receive signals using N carriers, under a power budget B_q for the qth user. The objective is to maximize the total transmission rate (sum-rate) by optimally allocating each user's power resource to the carriers.

Resource allocation in communication networks

$$\max_{p_q(k),\forall q,\forall k} \sum_{q=1}^{Q}\sum_{k=1}^{N} \log\left(1+\frac{|H_{qq}|^2 p_q(k)}{N_0+\sum_{r=1,r\neq q}^{Q}|H_{rq}(k)|^2 p_r(k)}\right)$$

subject to:

$$\sum_{k=1}^{N} p_q(k) \leq B_q, \qquad q=1,\cdots,Q,$$

$$p_q(k)\geq 0, \quad q=1,\cdots,Q, k=1,\cdots,N.$$

Resource allocation in communication networks

$$\max_{p_q(k),\forall q,\forall k} \sum_{q=1}^{Q}\sum_{k=1}^{N} \log\left(1 + \frac{|H_{qq}|^2 p_q(k)}{N_0 + \sum_{r=1,r\neq q}^{Q} |H_{rq}(k)|^2 p_r(k)}\right)$$

subject to:

$$\sum_{k=1}^{N} p_q(k) \leq B_q, \qquad q = 1, \cdots, Q,$$

$$p_q(k) \geq 0, \quad q = 1, \cdots, Q, k = 1, \cdots, N.$$

- The sampling distribution $f(\cdot;\theta)$ is chosen to be the Dirichlet distribution, whose support is a multi-dimensional simplex.

Resource allocation in communication networks

	maximal sum-rate ($N = 10$ $Q = 5$)	maximal sum-rate ($N = 10$ $Q = 10$)
GASS	34.654	46.765
IWFA	29.671	29.219
DDPA	34.001	45.704
MADP	34.001	44.942
GPA	18.892	22.702
MINOS	33.524	43.861
Filter	33.603	44.062
Ipopt	33.479	44.239
LANCELOT	33.603	44.055

Figure : Numerical results on resource allocation in communication networks. IWFA, DDPA, MADP, GPA are distributed algorithms. Other algorithms are running multi-start versions of NEOS Solvers: http://neos-server.org/neos/.

Discrete optimization

- $\mathcal{X}$ is a discrete set.

Discrete optimization

- $\mathcal{X}$ is a discrete set.
- Discrete-GASS: use discrete distribution
  - Sampling is easy, but the parameter is of high dimension.

Discrete optimization

- $\mathcal{X}$ is a discrete set.
- Discrete-GASS: use discrete distribution
  - Sampling is easy, but the parameter is of high dimension.
- Annealing-GASS: use Boltzmann distribution
  - Parameter is always of dimension 1, but sampling (by MCMC) is more expensive and inexact.

Discrete optimization

- $\mathcal{X}$ is a discrete set.
- Discrete-GASS: use discrete distribution
  - Sampling is easy, but the parameter is of high dimension.
- Annealing-GASS: use Boltzmann distribution
  - Parameter is always of dimension 1, but sampling (by MCMC) is more expensive and inexact.
  - Annealing-GASS converges to the set of optimal solutions in probability.

Numerical results

$|\mathcal{X}| \approx 10^6$ (Shekel), 10^{16} (Rosenbrock), 10^{80} (others).

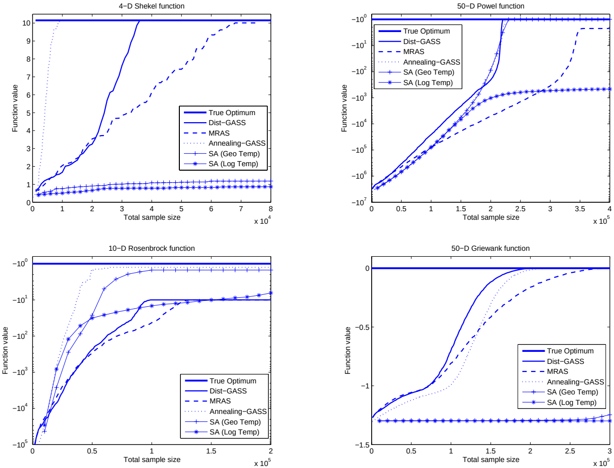


Figure : Average performance of discrete-GASS, Annealing-GASS, MRAS, SA (geometric temperature), and SA (logarithmic temperature).

Numerical results

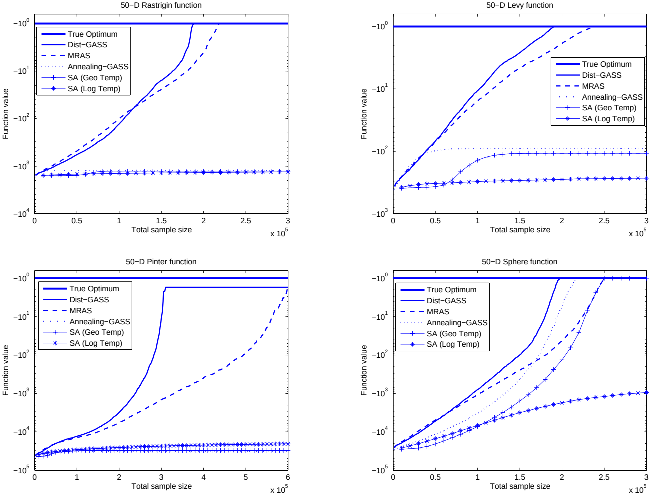


Figure : Average performance of discrete-GASS, Annealing-GASS, MRAS, SA (geometric temperature), and SA (logarithmic temperature)

Numerical results

- Discrete-GASS outperforms MRAS in both accuracy and convergence rate.

Numerical results

- Discrete-GASS outperforms MRAS in both accuracy and convergence rate.
- Annealing-GASS algorithm is an improvement of multi-start simulated annealing algorithms with geometric and logarithmic temperature schedules.

Numerical results

- Discrete-GASS outperforms MRAS in both accuracy and convergence rate.
- Annealing-GASS algorithm is an improvement of multi-start simulated annealing algorithms with geometric and logarithmic temperature schedules.
- Discrete-GASS provides accurate solutions in most of the problems; Annealing-GASS yields accurate solutions only in the low-dimensional problem and badly-scaled problems.

Numerical results

- Discrete-GASS outperforms MRAS in both accuracy and convergence rate.
- Annealing-GASS algorithm is an improvement of multi-start simulated annealing algorithms with geometric and logarithmic temperature schedules.
- Discrete-GASS provides accurate solutions in most of the problems; Annealing-GASS yields accurate solutions only in the low-dimensional problem and badly-scaled problems.
- Discrete-GASS usually needs more computation time for each iteration than Annealing-GASS, but needs less iterations to converge.

Implementation & Software

- Most tuning parameters can be set to default; need carefully choose stepsize $\{\alpha_k\}$.
- Choice of sampling distribution
  - $\mathcal{X}$ is a continuous set: (truncated) Gaussian
  - $\mathcal{X}$ is a simplex (with or without interior): Dirichlet
  - $\mathcal{X}$ is a discrete set: discrete, Boltzmann

Implementation & Software

- Most tuning parameters can be set to default; need carefully choose stepsize $\{\alpha_k\}$.
- Choice of sampling distribution
  - $\mathcal{X}$ is a continuous set: (truncated) Gaussian
  - $\mathcal{X}$ is a simplex (with or without interior): Dirichlet
  - $\mathcal{X}$ is a discrete set: discrete, Boltzmann
- Software available at http://enluzhou.gatech.edu/software.html

Outline

Simulation optimization: introduction

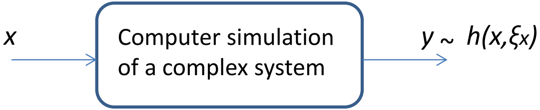
- Simulation optimization:

$$\max_{x \in \mathcal{X}} \; H(x) \triangleq E_{\xi_x}[h(x, \xi_x)].$$

Simulation optimization: introduction

- Simulation optimization:

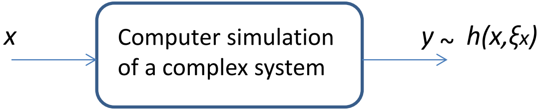
$$\max_{x \in \mathcal{X}} \; H(x) \triangleq E_{\xi_x}[h(x, \xi_x)].$$



Simulation optimization: introduction

- Simulation optimization:

$$\max_{x \in \mathcal{X}} \; H(x) \triangleq E_{\xi_x}[h(x, \xi_x)].$$

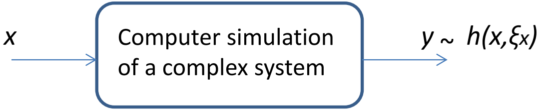


Example: a queueing system (x: service rate; H: waiting time + staffing cost; ξ_x: arrival/service times)

Simulation optimization: introduction

- Simulation optimization:

$$\max_{x \in \mathcal{X}} \; H(x) \triangleq E_{\xi_x}[h(x, \xi_x)].$$



Example: a queueing system (x: service rate; H: waiting time + staffing cost; ξ_x: arrival/service times)

- $\mathcal{X}$ is a continuous set.

Simulation optimization: introduction

- Main solution methods
  - Ranking & Selection (for problems with finite solution space)
  - Stochastic approximation
  - Response surface methods
  - Sample average approximation
  - Stochastic search methods

GASS for simulation optimization

Gradient-based Adaptive Stochastic Search (GASS)

- *Initialization*
- *Sampling:* draw samples $x_k^i \overset{\text{iid}}{\sim} f(x;\theta_k), i = 1, 2, \ldots, N_k$.
- *Estimation:* simulate each x_k^i for M_k times; estimate $\widehat{H}(x_k^i) = \frac{1}{M_k}\sum_{j=1}^{M_k} h(x_k^i, \xi_k^{i,j})$.
- *Updating:* update the parameter θ according to

$$\theta_{k+1} = \theta_k + \alpha_k(\widehat{\mathrm{Var}}_{\theta_k}[T(X)] + \epsilon I)^{-1}(\widehat{E}_{p_k}[T(X)] - E_{\theta_k}[T(X)]),$$

  where $\widehat{\mathrm{Var}}_{\theta_k}[T(X)]$ and $\widehat{E}_{p_k}[T(X)]$ are estimates using $\{x_k^i\}$ and $\{\widehat{H}(x_k^i)\}$.
- *Stopping*

Two-timescale GASS

Motivated by two-timescale stochastic approximation (Borkar 1997):

Two-timescale GASS (GASS_2T)

Assume $\alpha_k \to 0$, $\beta_k \to 0$, $\beta_k / \alpha_k \to 0$.

- Draw samples $x_k^i \overset{\text{iid}}{\sim} f(x; \theta_k),\ i = 1, \ldots, N$, and carry out computer simulation for each x_i once.
- Update the gradient and Hessian estimates in GASS on the fast timescale with step size α_k.
- Update θ_k on the slow timescale with step size β_k.

Two-timescale GASS

Motivated by two-timescale stochastic approximation (Borkar 1997):

Two-timescale GASS (GASS_2T)

Assume $\alpha_k \to 0$, $\beta_k \to 0$, $\beta_k/\alpha_k \to 0$.

- Draw samples $x_k^i \overset{\text{iid}}{\sim} f(x;\theta_k)$, $i = 1, \ldots, N$, and carry out computer simulation for each x_i once.
- Update the gradient and Hessian estimates in GASS on the fast timescale with step size α_k.
- Update θ_k on the slow timescale with step size β_k.

- Intuition: sampling distribution can be viewed as fixed while the gradient and Hessian estimates are updated over many iterations. So only a small sample size N is needed.

Numerical results

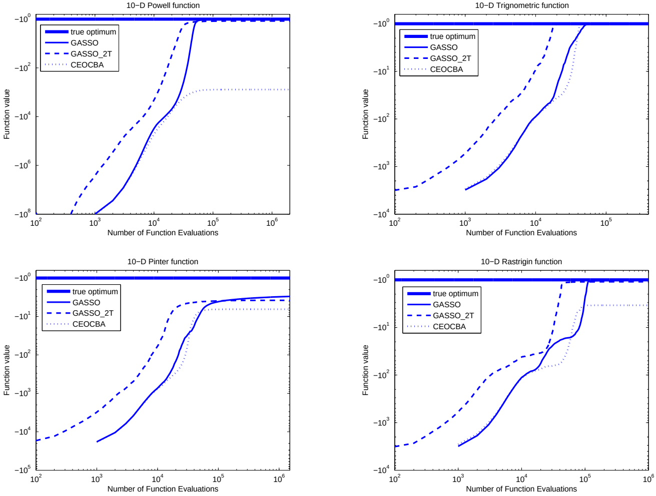


Figure : Average performance of GASS, GASS_2T, CEOCBA (He et al. 2010) on problems with independent noise

Outline

Conclusions

- By reformulating a hard optimization problem into a differentiable one, we can incorporate direct gradient search with stochastic search.

Conclusions

- By reformulating a hard optimization problem into a differentiable one, we can incorporate direct gradient search with stochastic search.

- A class of gradient-based adaptive stochastic search (GASS) algorithms for non-differentiable optimization, black-box optimization, and simulation optimization problems.

Conclusions

- By reformulating a hard optimization problem into a differentiable one, we can incorporate direct gradient search with stochastic search.
- A class of gradient-based adaptive stochastic search (GASS) algorithms for non-differentiable optimization, black-box optimization, and simulation optimization problems.
- Convergence results and numerical results show that GASS is a promising and competitive method.

References

- E. Zhou and Jiaqiao Hu, “Gradient-based Adaptive Stochastic Search for Non-differentiable Optimization”, *IEEE Transactions on Automatic Control*, 59(7), pp.1818-1832, 2014.
- E. Zhou and Shalabh Bhatnagar, “Gradient-based Adaptive Stochastic Search for Simulation Optimization over Continuous Space”, submitted.

Thank you !