Some new randomized methods for control and optimization

B.T. Polyak

Institute for Control Sciences, Russian Academy of Sciences, Moscow, Russia

June 2009

with E. Gryazina, P. Scherbakov

Workshop in honor of B.Barmish for the occasion of his 60th birthday

CONGRATULATIONS!

Bob, it's been a long time we have worked together, but I remember this time with delight. I remember your eyes shining when we discovered e^E picture for randomized robustness...
Remain as young and vital as ever!

Outline

- Historical remarks
- 2 problems — generation of points in a set and optimization
- "Ideal" Monte Carlo and convergence estimates
- Markov Chain Monte Carlo
  - Boundary oracle
  - Hit-and-Run
  - Shake-and-Bake
  - Exploiting barriers
- Applications to control
- Applications to optimization
- Conclusions

Historical remarks

- First random search methods for optimization
  Rastrigin (1960-ies)
- Randomized methods for control
  Stengel and Ray (1990)
  Barmish and Polyak (1996)
  Tempo, Calafiore, Dabbene (2004)
- Revival of randomized approaches for optimization
  Bertsimas and Vempala (2004), Dabbene, Shcherbakov and Polyak (2008)

Historical remarks

- First random search methods for optimization
  Rastrigin (1960-ies)
- Randomized methods for control
  Stengel and Ray (1990)
  Barmish and Polyak (1996)
  Tempo, Calafiore, Dabbene (2004)
- Revival of randomized approaches for optimization
  Bertsimas and Vempala (2004), Dabbene, Shcherbakov and Polyak (2008)

Historical remarks

- First random search methods for optimization
  Rastrigin (1960-ies)
- Randomized methods for control
  Stengel and Ray (1990)
  Barmish and Polyak (1996)
  Tempo, Calafiore, Dabbene (2004)
- Revival of randomized approaches for optimization
  Bertsimas and Vempala (2004), Dabbene, Shcherbakov and Polyak (2008)

2 main problems

1. Generation of points in a set $X \in \mathbb{R}^n$. It can be nonconvex and non-connected. Applications: generate stabilizing controllers and calculate performance specifications, generate perturbations and check robustness.
2. Convex optimization

$$\min \ c^T x$$
$$\text{s.t. } x \in X$$

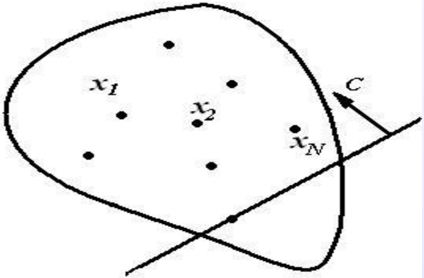
X is a convex bounded closed set in $\mathbb{R}^n$ with nonempty interior.

"Ideal" Monte Carlo

$x_1, x_2, \ldots x_N$ independent uniformly distributed points in X

Cutting plane method for optimization

1. Set $X_1 = X$
2. Generate uniform $x_1, x_2, \ldots x_N \in X_k$
3. Find $f_k = \min c^T x_i$
4. Set $X_{k+1} = X_k \bigcap \{x : c^T x \leq f_k\}$
   go to Step 2.



Convergence estimates

$$f^* = \max_{x \in X} \ c^T x, \quad f_* = \min_{x \in X} \ c^T x, h = f^* - f_*$$

Theorem

After k iterations of the algorithm

$$E\left[f_k\right] - f_* \leq q^k, \quad q = \frac{h}{n} B\left(N + 1, \frac{1}{n}\right),$$

where $B(a, b)$ is Euler beta-function.

Dabbene, Scherbakov, Polyak, 47th CDC, 2008

Radon theorem and center of gravity method

Case of a special interest: $N = 1,\ k = 1$

Theorem

Let x_1 be a random point uniformly distributed in X. Then

$$E\left[c^T x_1\right] - f_* \le h\left(1 - \frac{1}{n+1}\right).$$

$$E\left[x_1\right] = g \text{ (center of gravity of } X\text{)}, \quad \Rightarrow \quad c^T g - f_* \le h\left(1 - \frac{1}{n+1}\right)$$

[Radon theorem (1916)]

center of gravity method

$$x^k = g^k, \quad X_{k+1} = X_k \bigcap \{x : c^T x \le c^T g^k\}$$

Implementable random algorithms: MCMC

How to implement Monte Carlo method?

- X is a simple set (box, ball, simplex etc.)
- Rejection method
- Markov-Chain Monte Carlo schemes (random walks in X)

P.Diaconis "The Markov chain Monte Carlo revolution", Bull. of the AMS, 2009, Vol. 46. No 2. pp. 179–205.

Implementable random algorithms: MCMC

How to implement Monte Carlo method?

- X is a simple set (box, ball, simplex etc.)
- Rejection method
- Markov-Chain Monte Carlo schemes (random walks in X)

P.Diaconis "The Markov chain Monte Carlo revolution", Bull. of the AMS, 2009, Vol. 46. No 2. pp. 179–205.

Implementable random algorithms: MCMC

How to implement Monte Carlo method?

- X is a simple set (box, ball, simplex etc.)
- Rejection method
- Markov-Chain Monte Carlo schemes (random walks in X)

P.Diaconis "The Markov chain Monte Carlo revolution", Bull. of the AMS, 2009, Vol. 46. No 2. pp. 179–205.

Implementable random algorithms: MCMC

How to implement Monte Carlo method?

- X is a simple set (box, ball, simplex etc.)
- Rejection method
- Markov-Chain Monte Carlo schemes (random walks in X)

P.Diaconis "The Markov chain Monte Carlo revolution", Bull. of the AMS, 2009, Vol. 46. No 2. pp. 179–205.

Boundary oracle

Given $x_0 \in X$, d — vector specifying the direction in $\mathbb{R}^n$

Boundary oracle

$L = \{t \in \mathbb{R} : x^0 + td \in X\}$

For convex sets $L = (\underline{t}, \bar{t})$,
where $\underline{t} = \inf\{t : x^0 + td \in X\}$, $\bar{t} = \sup\{t : x^0 + td \in X\}$

Complete boundary oracle

$L = \{t \in \mathbb{R} : x^0 + td \in X\}$ + inner normals to X at the boundary points

Boundary oracle

Given $x_0 \in X$, d — vector specifying the direction in $\mathbb{R}^n$

Boundary oracle

$L = \{t \in \mathbb{R} : x^0 + td \in X\}$

For convex sets $L = (\underline{t}, \bar{t})$,
where $\underline{t} = \inf\{t : x^0 + td \in X\}$, $\bar{t} = \sup\{t : x^0 + td \in X\}$

Complete boundary oracle

$L = \{t \in \mathbb{R} : x^0 + td \in X\}$ + inner normals to X at the boundary points

Boundary oracle is available for numerous sets

- LMI set

$$X = \left\{ x \in \mathbb{R}^n : A_0 + \sum_{i=1}^{n} x_i A_i \leq 0 \right\}$$

- LMI constrained set of symmetric matrices P

$$X = \left\{ P : AP + PA^T + C \leq 0,\ P \geq 0 \right\}$$

- Quadratic matrix inequalities set

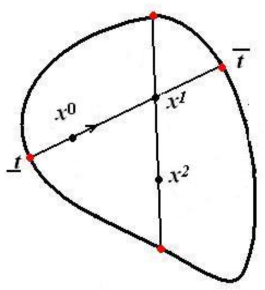
$$X = \left\{ P : AP + PA^T + PBB^T P + C \leq 0,\ P \geq 0 \right\}$$

- Linear algebraic inequalities set

$$X = \left\{ x \in \mathbb{R}^n : c_i^T x \leq a_i,\ i = 1, \ldots, m \right\}$$

Hit-and-Run

1. $i = 0$, $x^0 \in X$
2. Choose random direction d uniformly distributed on the unit sphere
3. $x^{i+1} = x^i + t_1 d$, t_1 is uniformly distributed on $L = (\underline{t}, \overline{t})$
4. L is updated with respect to x^{i+1}, go to Step 2.



Theorem (Turchin (1971), Smith (1984))

Let X be bounded open or coincides with the closure of interior points of X. Then for any measurable set $A \subset X$ probability $P_i(A) = P(x^i \in A|x^0)$ can be estimated as $|P_i(A) - P(A)| \leq q^i$, where $P(A) = \frac{\mathrm{Vol(A)}}{\mathrm{Vol(X)}}$ and $q < 1$ does not depend on x^0.

Shake-and-Bake: an alternative way for generating points

Points are *asymptotically* uniformly distributed in the boundary of X.
Complete boundary oracle is exploited.

- LMI set

$$X = \left\{ x \in \mathbb{R}^n : A_0 + \sum_{i=1}^{n} x_i A_i \leq 0 \right\}$$

$n_i = -(A_i e, e)$, where e is the eigenvector corresponding to zero eigenvalue of the matrix $A_0 + \sum_{i=1}^{n} x_i^0 A_i$.

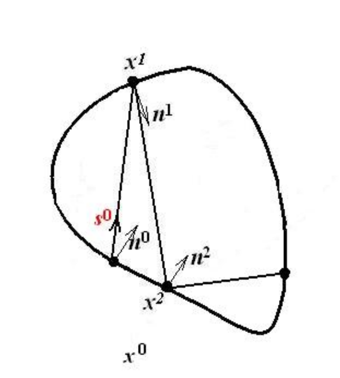
- LMI constrained set of symmetric matrices P

$$X = \left\{ P : AP + PA^T + C \leq 0 \right\}$$

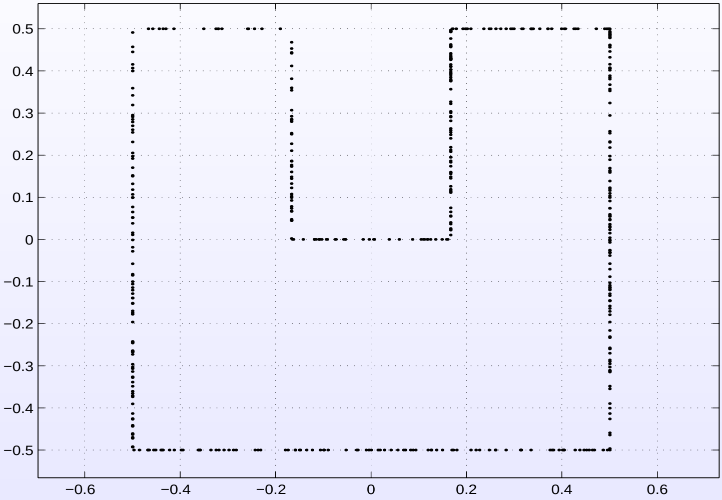
$N = -(ee^T A + A^T ee^T)$, where e is the eigenvector corresponding to zero eigenvalue of the matrix $AP_0 + P_0 A^T + C$.

Shake-and-Bake: the algorithm

1. $i = 0$, $x^0 \in \partial X$, n^0 is the normal.
2. Choose random direction s^i, $s^i = \sqrt{1 - \xi^{\frac{2}{n-1}}} n^0 + r$, ξ uniform random in $(0, 1)$, r is random unit uniform direction $(n^0, r) = 0$.
3. $x^{i+1} = x^i + \bar{t}s$, $\bar{t}$ is given by the boundary oracle for the direction s.
4. L is updated with respect to x^{i+1}, go to Step 2.



Shake-and-Bake for nonconvex sets



Numerical simulation

Standard SDP of the form

$$\min \; c^T x$$
$$\text{s.t. } A_0 + \sum_{i=1}^{n} x_i A_i \leq 0$$

A_i, $i = 0, 1, \ldots n$ — symmetric real matrices $m \times m$; $c = [0, \ldots, 0, 1]$
We applied modified HR where $\min \; x_i$ was replaced with averaged X_i
$+$ various heuristic acceleration methods (scaling, projecting, accelerating step)
Open problem: number of HR points in every step.

Disadvantages of HR

- Jams in a corner.
- Jams for long or thin bodies.

(Lovasz, Vempala. Hit-and-Run from a corner, 2007) Number of iterations to achieve accuracy ε

$$N > 10^{10}\frac{n^2R^2}{r^2}\ln\frac{M}{\varepsilon}$$

How to accelerate?

Smith (1998)

d – uniformly on a sphere $\rightarrow$ another distribution H

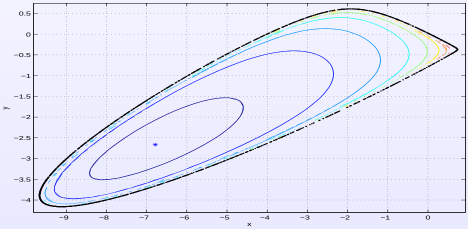
Barrier functions in convex optimization

Yu. Nesterov, A. Nemirovski, S. Boyd

Convex function $F(x)$ is a barrier for a convex set X, if $F(x)$ is defined on interior of X and $F(x) \to \infty$ for $x \to \partial X$.

Self-concordant barriers, Newton method

Interior-point methods are highly effective for convex optimization!



Modified Hit-and-Run by use of barriers

$F(x)$ – self-concordant barrier for X

$$F(x^0 + y) = F(x^0) + \underbrace{(\nabla F(x^0), y) + \frac{1}{2}(\nabla^2 F(x^0)y, y)}_{\tilde{F}(y)} + o(y^2),$$

$\tilde{F}(x)$ – quadratic approximation of $F(x)$ at x^0.

Dikin's ellipsoid

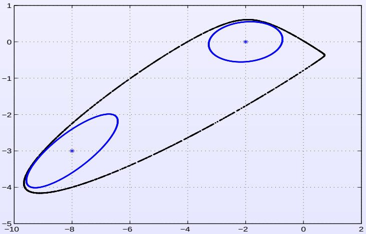
$$E = \{y : (\nabla^2 F(x^0)(y - x^0), y - x^0) \leq 1\} \subset X$$

Random direction

$$d = (\nabla^2 F)^{-1/2}\xi, \quad \xi - \text{ uniformly on a sphere}$$

Next point

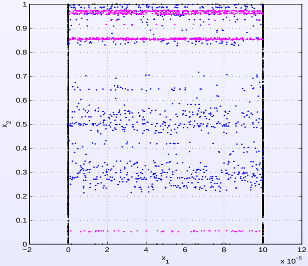
$$x = x^0 + \lambda d, \quad \lambda \in [\underline{\lambda}, \overline{\lambda}] \text{ as in Hit-and-Run}$$



Modified Hit-and-Run by use of barriers: example

$$Q = \{x \in \mathbb{R}^2 : \quad 0 \le x_i \le a_i\}, \quad a = [10^{-4}, 1] \quad \text{“strip”}$$

1000 points in $\mathbb{R}^2$



Magenta — Hit-and-Run

Blue — Modified Hit-and-Run

Barrier MC: Phase 1

$F(x)$ – self-concordant barrier for X

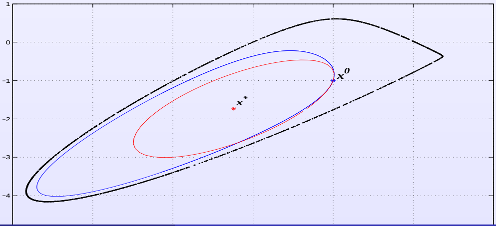
$$F(x^0 + y) = F(x^0) + \underbrace{(\nabla F(x^0), y) + \frac{1}{2}(\nabla^2 F(x^0) y, y)}_{\tilde{F}(y)} + o(y^2),$$

$E = \{y : \tilde{F}(y) \leq 0\}$, or $E = \{y : (\nabla^2 F(x^0)(y - x^*), y - x^*)) \leq \delta\}$,
$x^* = -(\nabla^2 F(x^0))^{-1} \nabla F(x^0)$, $\delta = ((\nabla^2 F(x^0))^{-1} \nabla F(x^0), \nabla F(x^0))$

Random direction

$$d = x^* + \left(\frac{\nabla^2 F}{\delta}\right)^{-1/2} \xi,$$

ξ — uniformly on a sphere



Advantages

- Departs corners.
- Well walks in long and thin sets.

Conjecture x^k asymptotically distributed with density $\rho(x) > 0$ on X.

Phase 2.

Ellipsoid is defined by matrix $H_i = \left(\frac{\nabla^2 F}{\delta}\right)^{-1/2}$.

After N iterations let $H = \frac{1}{N}\sum H_i$ and generate random direction

$$d = H\xi, \quad \xi - \text{ uniformly on unit sphere}$$

Barrier MC can be applied for

- Polyhedral sets in $\mathbb{R}^n$

$$X = \{x \in \mathbb{R}^n : (a_i, x) \le b_i, \quad i = 1, \dots, m\}.$$

$$F(x) = -\sum_{i=1}^{m} \ln(b^i - (a^i, x)), \; \nabla F(x^0) = \sum \frac{a^i}{1 - (a^i, x^0)},$$

$$\nabla^2 F(x^0) = \sum \frac{a^i {a^i}^T}{(1 - (a^i, x^0))^2}$$

- LMI in standard format

$$X = \{x \in \mathbb{R}^\ell : A(x) = A_0 + \sum_{i=1}^{\ell} x_i A_i \succeq 0, \quad A_i \in \mathbb{S}^{n \times n}\}.$$

$$F(x) = -\ln \det(A(x))$$

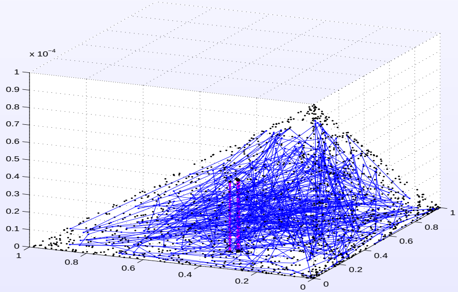
- LMI with matrix variables

$$Q = \{X \succeq 0, \quad \mathrm{tr} CX \le 1, \quad X, C \in \mathbb{S}^{n \times n}\}.$$

$$F(X) = -\ln \det X - \ln(1 - \mathrm{tr} CX)$$

Example: "diamond"

$$X = \{x \in \mathbb{R}^n : \quad x_i \geq 0, \quad (a, x) \leq 1\}, \quad a = [1, \dots, 1, 10^4],$$
$$F(x) = -\sum \ln x_i - \ln(1 - (a, x))$$



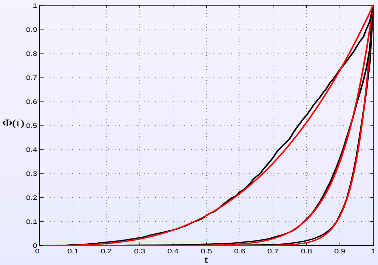
Comparison with pure MC

$$X = \{x \in \mathbb{R}^n : \quad x_i \geq 0, \quad (a,x) \leq 1\}, \quad a = [1, \dots, 1, 10^4]$$

$$f_i = (a, x^i), \quad 0 \leq f \leq 1, \quad x \in Q$$

If x^i are uniform on X, then cdf

$$\Phi(t) = \int_0^t p(f)df \sim t^n$$

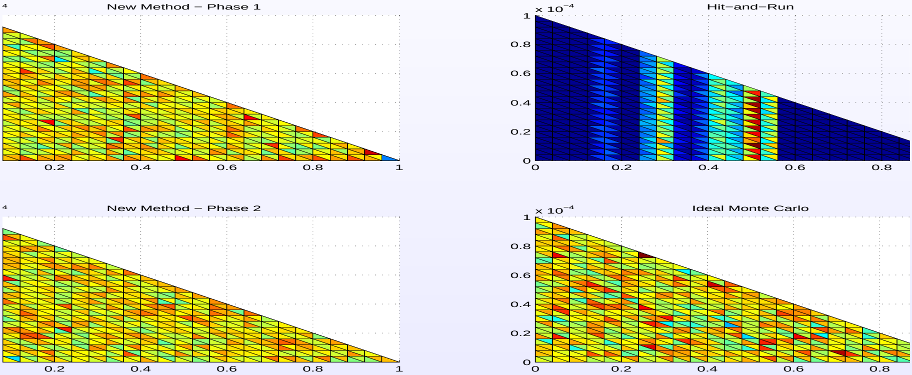


$n = 3 \qquad N = 2000$

$n = 10 \qquad N = 10\,000$

$n = 20 \qquad N = 10\,000$

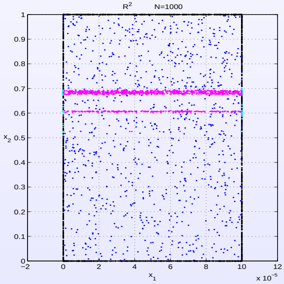
Empirical density ($N = 50\ 000, \quad \mathbb{R}^2$)

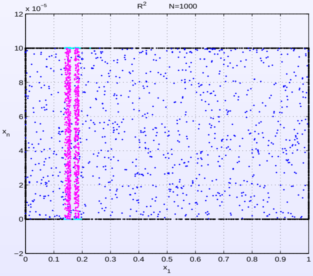


Example: "stick" and "sheet" $\mathbb{R}^2$

$$Q = \{x \in \mathbb{R}^n : \quad 0 \le x_i \le a_i\}$$

$$a = [10^{-4}, \dots, 10^{-4}, 1] \qquad\qquad a = [1, \dots, 1, 10^{-4}]$$

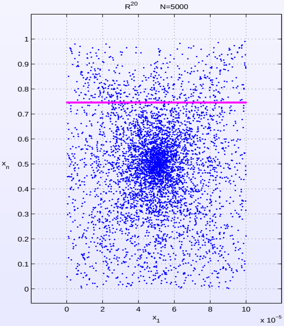


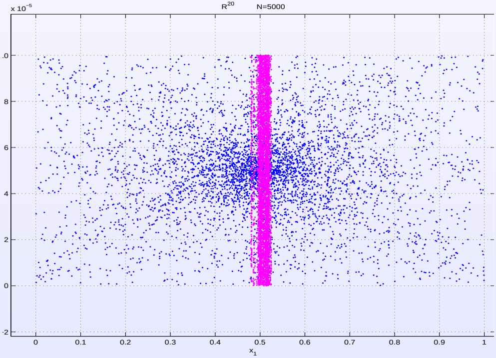


Example: "stick" and "sheet" $\mathbb{R}^{20}$

$$Q = \{x \in \mathbb{R}^n : \quad 0 \le x_i \le a_i\}$$

$$a = [10^{-4}, \ldots, 10^{-4}, 1] \qquad\qquad a = [1, \ldots, 1, 10^{-4}]$$

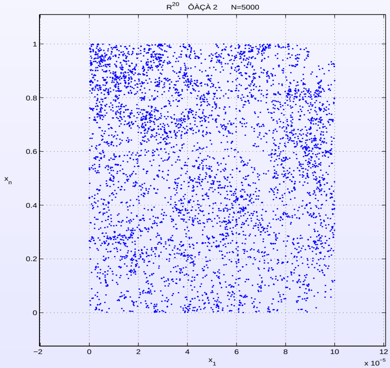


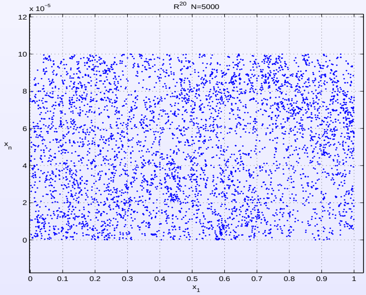


Example: "stick" and "sheet" $\mathbb{R}^{20}$ Phase 2

$$Q = \{x \in \mathbb{R}^n : \quad 0 \le x_i \le a_i\}$$

$$a = [10^{-4}, \ldots, 10^{-4}, 1] \qquad\qquad a = [1, \ldots, 1, 10^{-4}]$$

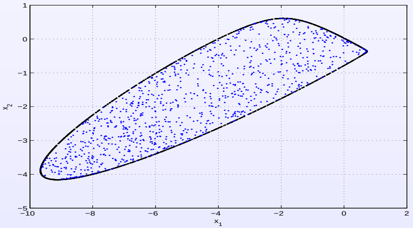




Standard LMI

$$X = \{x \in \mathbb{R}^\ell : A(x) = A_0 + \sum_{i=1}^{\ell} x_i A_i \succeq 0, \quad A_i \in \mathbb{S}^{n\times n}\}$$

$$F(x) = -\ln \det(A(x))$$
$$\nabla F(x^0)_i = -\mathrm{tr} A_i A(x^0)^{-1},$$
$$\nabla^2 F(x^0)_{ij} = \mathrm{tr} A(x^0)^{-1} A_i A(x^0)^{-1} A_j$$



A_0, A_1, A_2 — random matrices
4×4, $A_0 \succ 0$
1000 points

LMI with matrix variables

$$Q = \{X \succeq 0, \quad \text{tr}CX \le 1, \quad X, C \in \mathbb{S}^{n\times n}\}.$$

$F(X) = -\ln \det X - \ln(1 - \text{tr}CX), \qquad \varepsilon = 1 - \text{tr}CX_0$

Ellipsoid $E = \{Y : \tilde{F}(y) \le 0\}$ is

$$\langle X_0 + \frac{X_0 C X_0}{\varepsilon}, Y\rangle + \frac{1}{2}\langle Y + \frac{X_0 C Y C X_0}{\varepsilon^2}, Y\rangle \le 0$$

Center of ellipsoid Y^*– solution of Lyapunov equation

$$AXB^T - X + G = 0$$

where $A = -\dfrac{X_0 C}{\varepsilon^2}$,$B = CX_0$, $G = -\left(\dfrac{X_0 C X_0}{\varepsilon} - X_0\right)$

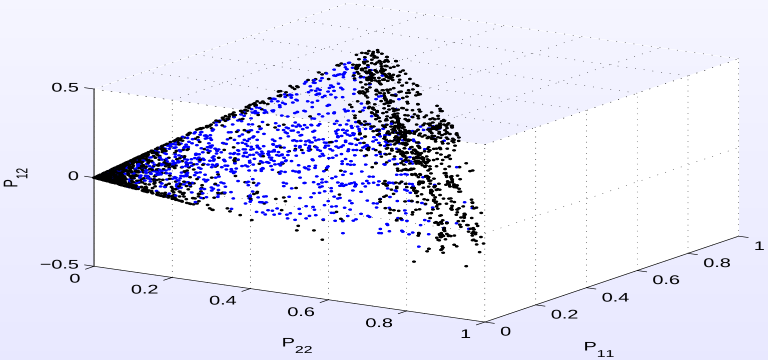
Choice of direction for ellipsoid $\langle AXA + BXB, X\rangle \le \delta$

Z - uniform on $||Z||_F = 1$, λ uniform on $[-\alpha, \alpha]$,

$\alpha : \alpha^2\langle AZA + BZB, Z\rangle = \delta$, then $\lambda Z \in E$.

$X : 2 \times 2,\ C = I$

$$Q = \left\{ \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \succeq 0, \quad p_{11} + p_{22} \leq 1 \right\} \Leftrightarrow \left\{ \begin{array}{c} p_{11} > 0, \quad p_{22} > 0 \\ p_{11} \cdot p_{22} \geq p_{12}^2 \\ p_{11} + p_{22} \leq 1 \end{array} \right\}$$



Comparison with pure MC

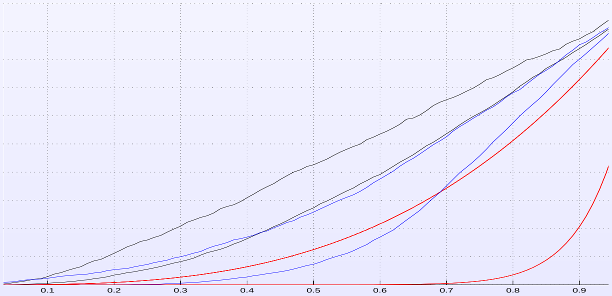
Theorem If x_i are uniform i.i.d. on $X \subset \mathbb{R}^n$, $f = (c, x)$, $f_* = \min_X (c, x)$, $h = \max_X (c, x) - f_*$ then

$$\frac{E[\min f_i] - f_*}{h} \leq \frac{B\left(N+1, \frac{1}{n}\right)}{n}, \quad i = 1, \dots, N$$

n	N	C	$E[\min f_i]$	$1 - E[\max f_i]$	2
2	1000	I	0.013	0.0004	
2	2000	diag(1,100)	0.018	0.0002	
5	5000	I	0.042	0.0001	
5	5000	diag(1,. . . ,1,100)	0.0124	0.0001	

Comparison with pure MC

$$\Phi(t) = \int_0^t p(f)df \sim t^m, \quad m = \frac{n(n+1)}{2}$$



Sets with available boundary oracle

- Stability set for polynomials

$$\mathcal{K} = \{k \in \mathbb{R}^n : p(s,k) = p_0(s) + \sum_{i=1}^{n} k_i p_i(s) \text{ is stable}\}$$

- Stability set for matrices

$$A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{l\times n}$$

$$\mathcal{K} = \{K \in \mathbb{R}^{m\times l} : A + BKC \text{ is stable}\}$$

- Robust stability set for polynomials

$$\mathcal{K} = \{k : P_0(s,q) + \sum_{i=1}^{n} k_i P_i(s,q) \ \text{ is stable } \forall q \in Q\}, \quad Q \subset \mathbb{R}^m$$

- Quadratic stability set

$$\dot{x} = Ax$$

$$\mathcal{K} = \{P > 0 : AP + PA^T \leq 0\}$$

Stability set for polynomials

$$\mathcal{K} = \{k \in \mathbb{R}^n : p(s,k) = p_0(s) + \sum_{i=1}^{n} k_i p_i(s) \text{ is stable}\}$$

$k^0 \in \mathcal{K}$ i.e. $p(s, k^0)$ is stable,

$d = s/||s||, s =$ `randn(n,1)` — random direction

Boundary oracle: $L = \{t \in \mathbb{R} : k^0 + td \in \mathcal{K}\}$,

i.e. $\{t \in \mathbb{R} : p(s,k^0) + t \sum d_i p_i(s) \text{ is stable}\}$.

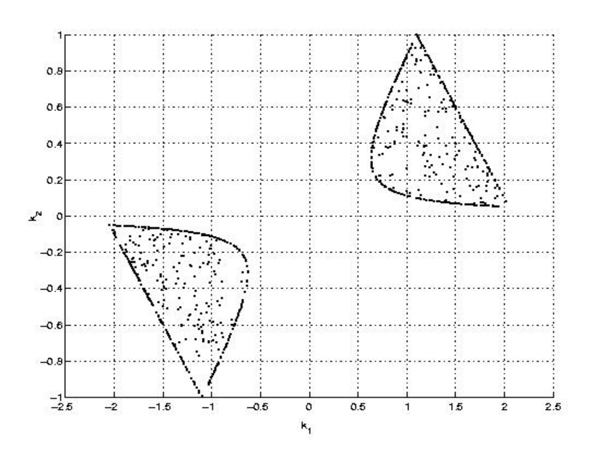
D-**decomposition problem for real scalar parameter** t**!**

Gryazina E. N., Polyak B. T. Stability regions in the parameter space: D-decomposition revisited //Automatica. 2006. Vol. 42, No. 1, P. 13–26.

Example: Generating points in the disconnected set

$$\mathcal{K} = \{k \in \mathbb{R}^n : p(s,k) = p_0(s) + \sum_{i=1}^{n} k_i p_i(s) \text{ is stable}\},$$

$$p(s,k) = 2.2s^3 + 1.9s^2 + 1.9s + 2.2 + k_1(s^3 + s^2 - s - 1) + k_2(s^3 - 3s^2 + 3s - 1)$$



Stability set for matrices

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad u = Ky$$

$$A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{l\times n}; \quad \mathcal{K} = \{K \in \mathbb{R}^{m\times l} : A + BKC \text{ is stable}\}$$

$K^0 \in \mathcal{K}$, i.e. $A + BK^0C$ is stable

$D = Y/||Y||, Y =$ `randn`(m, l) — random direction in the matrix space K

$A + B(K^0 + tD)C = F + tG$, where $F = A + BK^0C$, $G = BDC$

Boundary oracle: $L = \{t \in \mathbb{R} : F + tG \text{ is stable}\}$

Total description of L is hard:

$f(t) = \max Re \text{ eig}(F + tG)$

numerical solution of the equation $f(t) = 0$, $t > 0$ (MatLab command `fsolve`)

Quadratic stability

$$\dot{x} = Ax + Bu, \quad u = Kx$$

$$\mathcal{K} = \{K : \exists P > 0, A_c^T P + PA_c \leq 0\}, \quad A_c = A + BK$$

$\mathcal{K}$ is convex and bounded.

$$Q = P^{-1} > 0, \quad QA^T + AQ + BY + Y^T B^T < 0, \quad Y = KQ.$$

$k^0 \in \mathcal{K}$, $Q_0 = P_0^{-1}$, $Y_0 = K_0 Q_0$ — starting points
$Q = Q_0 + tJ$, $Y = Y_0 + tG$, where J and G are random directions in the matrix space.
initial inequality $\Longleftrightarrow F + tR < 0$
Boundary oracle: $L = (-\underline{t}, \bar{t})$,
where $\bar{t} = \min \lambda_i$, $\underline{t} = \min \mu_i$;
λ_i — real positive eigenvalues for the pair of matrices
$F = Q_0 A^T + AQ_0 + BY_0 + Y_0^T B^T$ and $-R = JA^T + AJ + BG + G^T B^T$;
μ_i correspondingly for matrices F, R.

Conclusions

- New versions of MCMC are effective
- Randomized approaches for optimization are promising.
- Proposed methods are simple in implementation and give an opportunity to solve large-dimensional problems.