Walks around Monte-Carlo
Part 2: Applications

Boris Polyak

with Elena Gryazina

Institute for Control Science, Moscow

Seminar MFTI

March, 2012

Uniformly generated points: applications

- Center of gravity. Volume.
- Multidimensional integration.
- Convex optimization.
- Global optimization.
- Modelling of uncertainty. Robustness.
- Control applications.

Center of gravity

$Q \subset R^n$ bounded and measurable,

$$x_g = \frac{\int_Q x dx}{\int_Q dx} \text{ — center of gravity.}$$

Center of gravity (centroid) is affine invariant. How to calculate x_g?

1. If Q has a center of symmetry O, then $x_g = O$.
2. If $Q = conv\{a_1, \dots, a_{n+1}\}$ then $x_g = \frac{1}{n+1} \sum a_i$.
3. If Q is an union of simple sets (positive or negative) then x_g is a weighted sum of centers of gravity of these sets. Examples: disk with a hole, triangulization.

However in general calculation of center of gravity (even for convex sets) is hard (there are rigorous results on complexity).

Center of gravity via sampling

Points $x_1, \ldots, x_N \in Q$ are i.u.d. (independent uniformly distributed) points in Q. Then their average is the estimate of x_g:

$$\widehat{x} = \frac{1}{N} \sum x_i, \quad E\widehat{x} = x_g$$

Estimate the accuracy $E(\widehat{x} - x_g)(\widehat{x} - x_g)^T$. Hint: moment of inertia.

Theorems on center of gravity

1. Radon 1916. Q is a convex compact body in R^n, $f(x) = (c, x)$, $f^* = \max\limits_{x \in Q} f(x)$, $f_* = \min\limits_{x \in Q} f(x)$, $f_g = f(x_g)$, $h = f^* - f_*$, then

$$\frac{1}{n+1} \leq \frac{f^* - f_g}{h} \leq \frac{n}{n+1}.$$

.

Worst-case — simplex.

2. Grunbaum 1960, Mityagin 1969. Q is a convex compact body in R^n, $H = \{x : (c, x) \geq (c, x_g)\}$, $v_1 = Vol(H)$, $v = Vol(Q)$. Then

$$\frac{v_1}{v} \leq (1 - \frac{1}{n+1})^n < 1 - \frac{1}{e}.$$

Worst-case — simplex.

In general x_g is the only point with these properties.

Applications to optimization

Linear optimization

$$\min(c, x), \quad x \in Q$$

Q is a convex compact body in R^n, x^* is a solution, we assume that x_g is available.

Cutting plane method

Start: $Q_0 = Q, x_0 = x_g(Q_0)$

k-th iteration: $x_k = x_g(Q_{k-1}), Q_k = Q_{k-1} \bigcap \{x : (c, x) \leq (c, x_k)\}$

Theorem

$$(c, x_k) - (c, x^*) \leq ((c, x_0) - (c, x^*))(1 - \frac{1}{n+1})^k.$$

Convergence — geometric progression with ratio $q = \dfrac{n}{n+1}$, not depending on geometry of Q! We need n iterations to increase accuracy $e = 2.78\ldots$ times.

Applications to optimization 2

Convex optimization

$$\min f(x), \quad x \in Q$$

Q is a convex compact body in R^n, $f(x)$ is a convex function defined on Q, x^* is a solution, we assume that x_g and subgradient $\partial f(x)$ are available.

Center of gravity method: Levin 1965, Newman 1965.

Start: $Q_0 = Q, x_0 = x_g(Q_0)$

k-th iteration: $x_k = x_g(Q_{k-1}), Q_k = Q_{k-1} \bigcap \{x : (\partial f(x_{k-1}), x) \leq (\partial f(x_{k-1}), x_k)\}$

Theorem

$$\min_{0 \leq i \leq k} f(x_i) - f(x^*) \leq (f(x_0) - f(x^*))(1 - \frac{1}{e})^{k/n}.$$

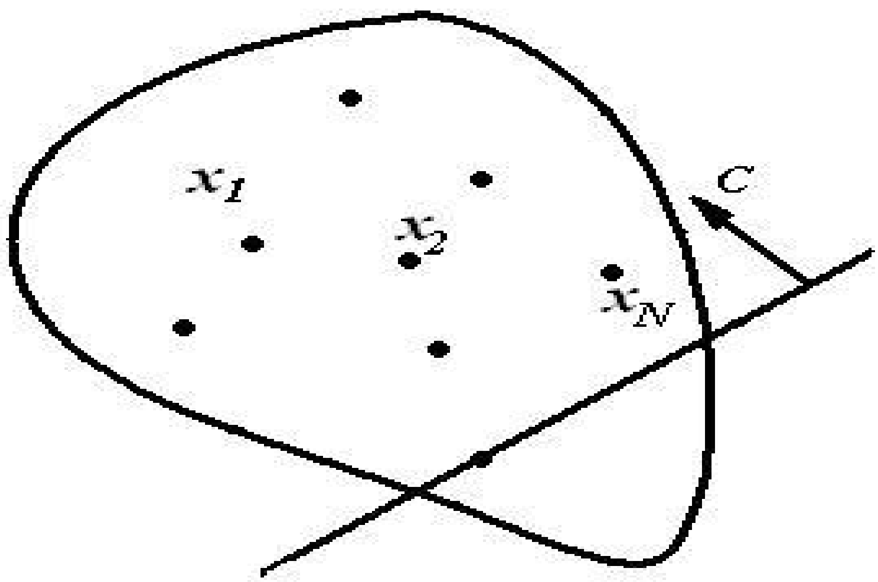
This is slower rate of convergence than for linear optimization (approx. e times more iterations), but the method is optimal in some sense. The only problem is: the method is not implementable!

Random version of cutting plane

$$\min(c, x), \quad x \in Q$$

$x_1, \ldots, x_N$ i.u.d. on Q.

Iteration: $\bar{f} = \min_{1 \le i \le N} (c, x_i), \quad Q_{new} = Q \bigcap \{x : (c, x) \le \bar{f}\}$



Theorem

$$E\bar{f} - f_* \le \frac{h}{n} B\left(N+1, \frac{1}{n}\right) \le \left(\frac{1}{N+1}\right)^{1/n},$$

where $B(a, b)$ is Euler beta-function. Simplex is the worst-case set.

F.Dabbene, B.Polyak and P.S.Scherbakov) "A Randomized Cutting Plane Method with Probabilistic Geometric Convergence", SIAM Journal on Optimization, 2010, V. 20, No 6, 3185–3207.

Case $N = 1$ is the fastest version (Radon theorem).

Volume

Other types of random walks (e.g. in cubic grid):

M.Dyer, A.Frieze, R.Kannan, A Random Polynomial-Time Algorithr for Approximating the Volume of Convex Bodies, Journal of the ACM, Volume 38 Issue 1, Jan. 1991

Lovasz, L., Simonovits, M. Random walks in a convex body and an improved volume algorithm, Random Structures and Algorithms, 1993

But how can one estimate $Vol(Q)$ exploiting i.u.d. sample in Q? Idea: there is $S \subset Q, Vol(S)$ known. Then

$$Vol(Q) \simeq Vol(G)\frac{N}{m}$$

m is the number of points $x_1, \ldots, x_N$ which are in S. Examples: G is Dikin ellipsoid, G is a simplex, etc.

Integration

Goal: calculation of

$$I = \int_Q f(x)dx$$

Let $x_1, \ldots, x_N$ be i.u.d. sample in Q, then

$$I \simeq Vol(Q)\frac{1}{N}\sum f(x_i)$$

For simple sets (cubes, simplices, balls etc.) it is very simple and attractive.

Global optimization

Multistart method: numerous initial points + local search.

However it is hopeless to rely on ANY methods of global optimization: see Nesterov, “Introduction to convex optimization”, p.32: finding optimum value of Lipschitz-continuous function on unit cube with relative accuracy 0.01 for $n = 11$ requires millions of years of modern computer calculations, and randomness does not help!

Of course it does not contradicts to ability to solve special classes of global optimization problems.

Concave programming

$$\min f(x), \quad x \in Q$$

Q is a convex compact body in R^n, $f(x)$ is a concave function. Then minimum is achieved on the boundary, however local minima are possible.

Local minimization method: Frank-Wolfe or conditional gradient method:

$$x_{k+1} = \arg\min_{x \in Q} (f'(x_k), x)$$

If Q is a polytope, the method is finite. Thus having $x_1, \dots, x_N$ i.u.d. sample in Q, we make this local descent for each of them.

Example: finding the diameter

$$\max_{x,y \in Q} ||x - y||, \quad Q = \{x : Ax \leq b\}$$

$$x_{k+1} = \arg\max_{x \in Q} (x_k - y_k, x)$$

$$y_{k+1} = \arg\max_{y \in Q} (x_k - y_k, y)$$

We have a sample in Q (obtained by Hit-and-Run or Shake-and-Bake), choose several most distant pairs and provide such descent.

Numerical experiments are welcome!

Applications to control

Sets with available boundary oracle

- Stability set for polynomials. Polynomial $p(s)$ is stable, if all its roots lie in the open left half-plane of complex plane. Given affine family of polynomials with parameters k,

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

- Stability set for matrices. Matrix is stable, if all its eigenvalues lie in the open left half-plane of complex plane. Typical problem is to find all stabilizing controllers K:

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

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

- Robust stability set for polynomials. Vectors $q \in Q$ denote uncertainty parameters. Describe

$$\mathcal{K} = \{k : p_0(s,q) + \sum_{i=1}^{n} k_i p_i(s,q) \;\text{ is stable } \forall q \in Q\}, \quad Q \subset \mathbb{R}^m$$

- Quadratic stability set. Matrix $P > 0$ (i.e. positive definite) describes quadratic Lyapunov function:

$$\dot{x} = Ax$$

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

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

So-called D-decomposition problem for real scalar parameter t is easily solvable.

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

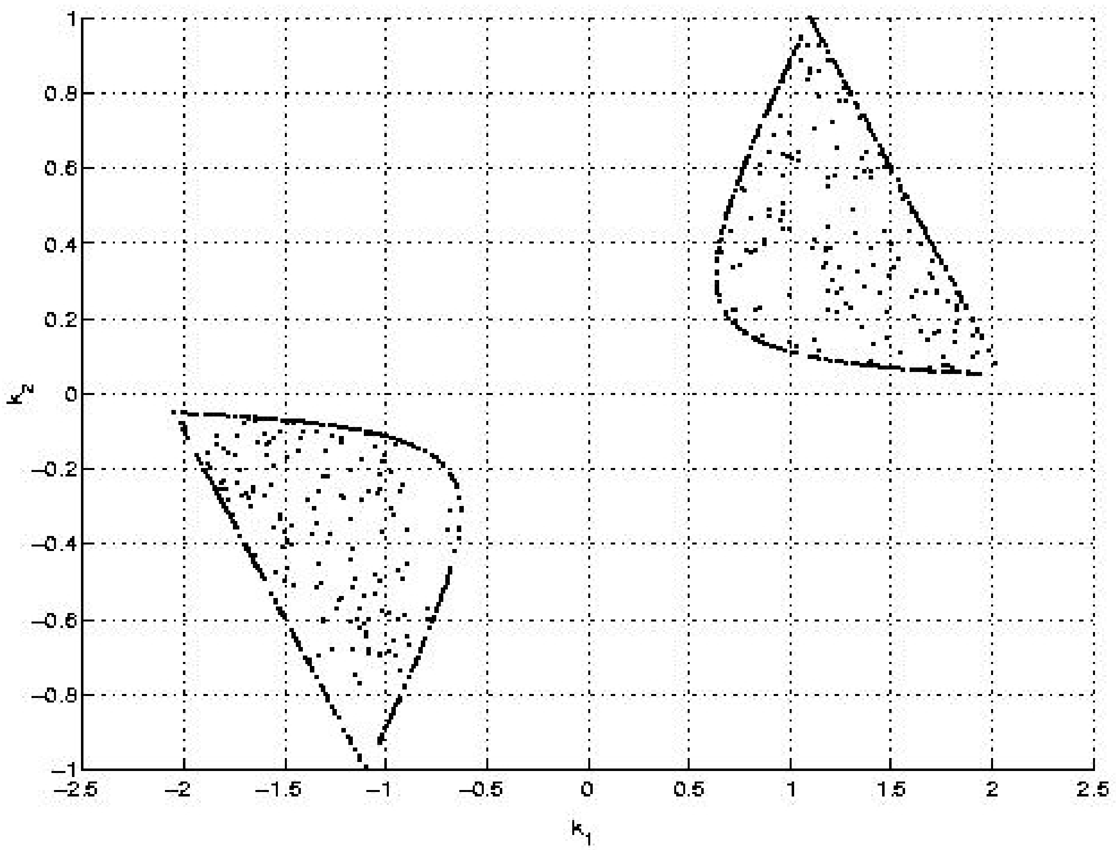
Example: Stability set for polynomials

$$\mathcal{K} = \{k \in \mathbb{R}^2 : p(s,k) = p_0(s) + \sum_{i=1}^{2} k_i p_i(s) \text{ is stable}\},$$

$$p_0(s) = 2.2s^3 + 1.9s^2 + 1.9s + 2.2,$$

$$p_1(s) = s^3 + s^2 - s - 1,$$

$$p_2(s) = s^3 - 3s^2 + 3s - 1$$



Set $\mathcal{K} \subset \mathbb{R}^2$ is nonconvex and disconnected.

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
$D = Y/||Y||, Y = \mathrm{randn}(m, l)$ — 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:
$f(t) = \max \Re \, \mathrm{eig}(F + tG)$
numerical solution of the equation $f(t) = 0$, $t > 0$ (MatLab command fsolve)

Quadratic stability

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

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

$\mathcal{K}$ 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: $L = (-\underline{t}, \overline{t})$,
where $\overline{t} = \min \lambda_i$, $\underline{t} = \min \mu_i$;
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.

Open problems

- Global: are there random methods of convex optimization superior over deterministic ones?
- Estimate rigorously complexity (as function of N, n, ε) of multistart global optimization e.g. for quadratic concave minimization subject to linear constraints.
- Discrete optimization via Hit-and-Run and similar methods.