Quasi-Monte Carlo Methods for Stochastic Optimization

Tito Homem-de-Mello

Northwestern University
Department of Industrial Engineering and Management Sciences

Rubinstein Conference July 2008

The Problem

We consider the optimization problem

$$\min_{x \in X} \{g(x) := \mathbb{E}[G(x,\xi)]\}$$

where:

- X is a subset of $\mathbb{R}^n$
- ξ is a random vector in $\mathbb{R}^s$
- $G : \mathbb{R}^n \times \mathbb{R}^s \mapsto \mathbb{R}$ is a real-valued function

Suppose that $\mathbb{E}[G(x,\xi)]$ cannot be easily calculated.

Sample Average Approximation

Basic idea:

- Let $\xi^1, \ldots, \xi^N$ be a random sample drawn from F.
  We assume momentarily that $\xi^1, \ldots, \xi^N$ are i.i.d.

Sample Average Approximation

Basic idea:

- Let $\xi^1, \ldots, \xi^N$ be a random sample drawn from F.
  We assume momentarily that $\xi^1, \ldots, \xi^N$ are i.i.d.
- Estimate $g(x) = \mathbb{E}G(x, \xi)$ by

$$\hat{g}_N(x) = \frac{1}{N} \sum_{i=1}^{N} G(x, \xi^i).$$

Sample Average Approximation

Basic idea:

- Let $\xi^1, \ldots, \xi^N$ be a random sample drawn from F.
  We assume momentarily that $\xi^1, \ldots, \xi^N$ are i.i.d.
- Estimate $g(x) = \mathbb{E}G(x, \xi)$ by

$$\hat{g}_N(x) = \frac{1}{N} \sum_{i=1}^{N} G(x, \xi^i).$$

- Solve $\min_{x \in \Theta} \ \hat{g}_N(x)$

  using a deterministic optimization algorithm, and take its optimal solution $\hat{x}_N$ and optimal value $\hat{v}_N$ as estimates of true optimal solution and true optimal value.

Sample Average Approximation (II)

Rubinstein and Shapiro (1993) call this approach stochastic counterpart method.

Sample Average Approximation (II)

Rubinstein and Shapiro (1993) call this approach stochastic counterpart method.

- They show that $\hat{x}_N$ and $\hat{v}_N$ converge to the "right values" as $N \longrightarrow \infty$.

Sample Average Approximation (II)

Rubinstein and Shapiro (1993) call this approach stochastic counterpart method.

- They show that $\hat{x}_N$ and $\hat{v}_N$ converge to the "right values" as $N \to \infty$.

  Another class of results deals with *rates* of convergence, i.e., how fast the estimation error (e.g., $|\hat{v}_N - v^*|$) goes to zero.

  - Such rates are usually $O_p(N^{-1/2})$, as a consequence of the Central Limit Theorem.
- Results of this type have been well studied in the literature.

Rates of Convergence

The convergence rate of $O_p(N^{-1/2})$ can be slow, especially if large sample sizes are needed.

Rates of Convergence

The convergence rate of $O_p(N^{-1/2})$ can be slow, especially if large sample sizes are needed.

This problem arises even in context of pointwise estimation.

- That is, let U be an s-dimensional $(0,1)$ uniform random vector and suppose we want to estimate $I := \mathbb{E}[f(U)]$.
- Let $\xi^1, \ldots, \xi^N$ be numbers distributed on the box $(0,1)^s$, and estimate I with $\hat{I} := \frac{1}{N} \sum f(\xi^i)$.
- If $\xi^1, \ldots, \xi^N$ are standard (Monte Carlo) samples, then the error $|\hat{I} - I|$ is $O_p(N^{-1/2})$.

Rates of Convergence

The convergence rate of $O_p(N^{-1/2})$ can be slow, especially if large sample sizes are needed.

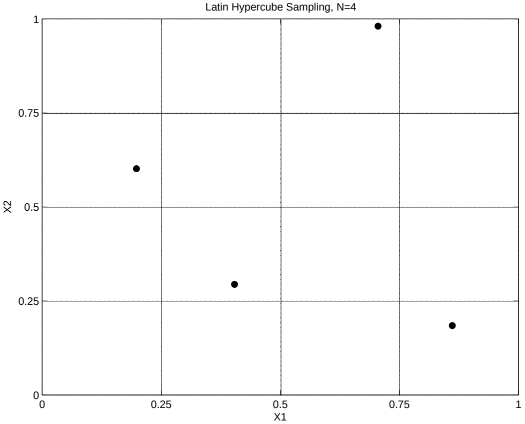
This problem arises even in context of pointwise estimation.

- That is, let U be an s-dimensional $(0,1)$ uniform random vector and suppose we want to estimate $I := \mathbb{E}[f(U)]$.
- Let $\xi^1, \ldots, \xi^N$ be numbers distributed on the box $(0,1)^s$, and estimate I with $\hat{I} := \frac{1}{N}\sum f(\xi^i)$.
- If $\xi^1, \ldots, \xi^N$ are standard (Monte Carlo) samples, then the error $|\hat{I} - I|$ is $O_p(N^{-1/2})$.

QUESTION: Are there other sampling techniques that yield better rates?

Latin Hypercube Sampling

Latin hypercube sampling (LHS) is a stratified sampling technique aimed at reducing the variance of estimators (McKay et al. 1979)



Comments about LHS

- LHS is very simple to implement, and often very effective.
- Asymptotically, variance of estimators constructed with LHS is no worse than that obtained with Monte Carlo (Stein 1987).
    - However, *rate* of convergence is still the same as Monte Carlo (Owen 1992)
- Impact of LHS is higher if the underlying function is close to being additive.

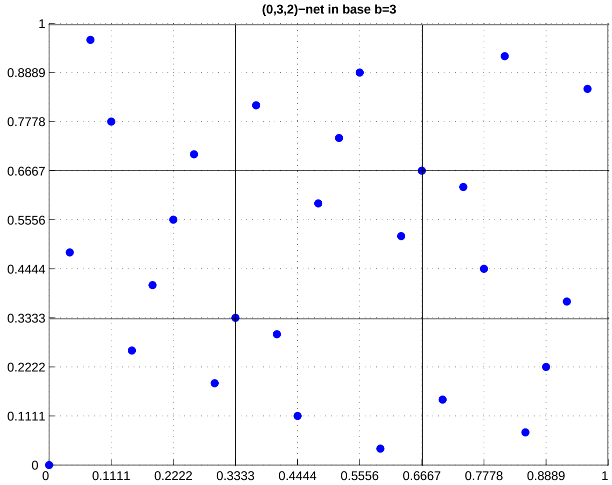
Comments about LHS

- LHS is very simple to implement, and often very effective.
- Asymptotically, variance of estimators constructed with LHS is no worse than that obtained with Monte Carlo (Stein 1987).
  - However, *rate* of convergence is still the same as Monte Carlo (Owen 1992)
- Impact of LHS is higher if the underlying function is close to being additive.
- In the context of stochastic optimization, sampling with LHS preserves convergence properties (HM 2006).
  - In particular, pathwise convergence is guaranteed; however, rate of convergence is the same as Monte Carlo (at least for a certain class of problems), even though estimators may have smaller variance.
  - An exception occurs in case the function is additive — rate is much faster then.

Quasi-Monte Carlo Sampling

- Sample points are chosen deterministically.
- Goal is for the point set to resemble a uniform distribution.
- Deviation from the uniform distribution is measured by the *discrepancy*. One such measure is the *star-discrepancy* $D_N^*(\xi^1, \ldots, \xi^N)$ on $[0,1)^s$.
- Low-discrepancy point sets are desirable; the two main classes of low-discrepancy points are (t, m, s)-nets and lattice rules.

Example: (t, m, s)-net



Error rates for QMC

- Estimation error is typically $O\left(\frac{(\log N)^s}{N}\right)$ (Niederreiter 1992).
- For a particular type of randomized QMC, estimation error is $O_p\left(\left[\frac{(\log N)^{(s-1)}}{N^3}\right]^{1/2}\right)$ (Owen 1997).

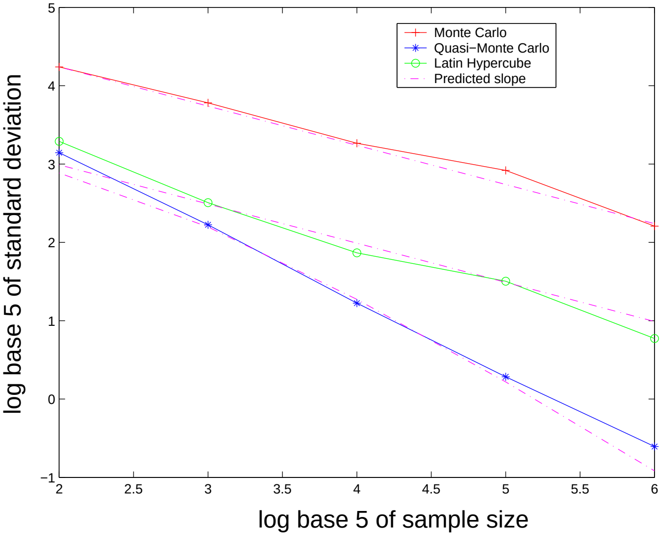
Error rates for QMC

- Estimation error is typically $O\left(\frac{(\log N)^s}{N}\right)$ (Niederreiter 1992).

- For a particular type of randomized QMC, estimation error is $O_p\left(\left[\frac{(\log N)^{(s-1)}}{N^3}\right]^{1/2}\right)$ (Owen 1997).

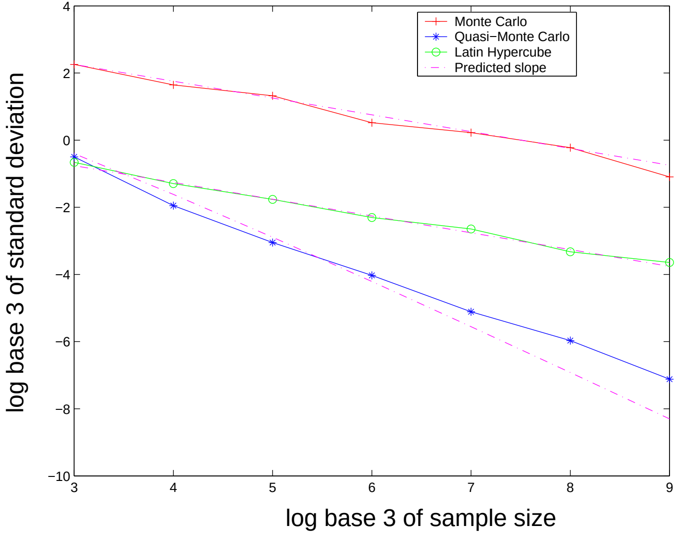
- By applying these results in the optimization context, it is possible to show that, under some (restrictive) assumptions, error rate $|\hat{v}_N - v^*|$ is also $O_p\left(\left[\frac{(\log N)^{(s-1)}}{N^3}\right]^{1/2}\right)$ (HM 2006).
  - Even when assumptions are not satisfied, error rate is often still better than $O(N^{-1/2})$.

Example 1



Monte Carlo
Quasi−Monte Carlo
Latin Hypercube
Predicted slope
log base 5 of standard deviation
log base 5 of sample size
5
4
3
2
1
0
−1
2
2.5
3
3.5
4
4.5
5
5.5
6

Example 2



Monte Carlo
Quasi−Monte Carlo
Latin Hypercube
Predicted slope
log base 3 of standard deviation
log base 3 of sample size
4
2
0
−2
−4
−6
−8
−10
3
4
5
6
7
8
9

Comments about QMC

- It is clear that, asymptotically, the above error rates are better than the error with Monte Carlo ($O(N^{-1/2})$).

Comments about QMC

- It is clear that, asymptotically, the above error rates are better than the error with Monte Carlo ($O(N^{-1/2})$).
- However, this happens because these problems are low-dimensional.

Comments about QMC

- It is clear that, asymptotically, the above error rates are better than the error with Monte Carlo ($O(N^{-1/2})$).

- However, this happens because these problems are low-dimensional.

- To see why, notice that for $\left[\frac{(\log N)^{(s-1)}}{N^3}\right]^{1/2}$ to beat $N^{-1/2}$ when s is large, N must be *very large*.

Comments about QMC

- It is clear that, asymptotically, the above error rates are better than the error with Monte Carlo ($O(N^{-1/2})$).

- However, this happens because these problems are low-dimensional.

- To see why, notice that for $\left[\frac{(\log N)^{(s-1)}}{N^3}\right]^{1/2}$ to beat $N^{-1/2}$ when s is large, N must be *very large*.

- What matters is the effective dimension of the problem — roughly speaking, this is the number of variables that make up for most of the variance.

Employing QMC in Lower Dimensions

One way to approach the dimensionality problem in QMC is the following:

- Use QMC on the most "important" random variables
- Use some other method (Midpoint, Monte Carlo, LHS) on the remaining variables to "pad" the sample (e.g., Owen 1998).

Employing QMC in Lower Dimensions

One way to approach the dimensionality problem in QMC is the following:

- Use QMC on the most "important" random variables
- Use some other method (Midpoint, Monte Carlo, LHS) on the remaining variables to "pad" the sample (e.g., Owen 1998).

ISSUES:

- How to select the important random variables?
  - General methods exist, but we want to exploit the structure of underlying problem.
  - A number of papers exist in the context of finance problems.

Employing QMC in Lower Dimensions

One way to approach the dimensionality problem in QMC is the following:

- Use QMC on the most "important" random variables
- Use some other method (Midpoint, Monte Carlo, LHS) on the remaining variables to "pad" the sample (e.g., Owen 1998).

ISSUES:

- How to select the important random variables?
  - General methods exist, but we want to exploit the structure of underlying problem.
  - A number of papers exist in the context of finance problems.
- What kind of properties do the padded estimators have?
  - In particular, we are interested in checking if these estimators satisfy a Central Limit Theorem.

A CLT for Padded Estimators

Ökten, Tuffin and Burago (2006) show that estimators constructed with QMC padded with Monte Carlo satisfy a Central Limit Theorem.

- Proof relies heavily on the independence of the padding components.

A CLT for Padded Estimators

Ökten, Tuffin and Burago (2006) show that estimators constructed with QMC padded with Monte Carlo satisfy a Central Limit Theorem.

- Proof relies heavily on the independence of the padding components.

Drew and HM (2007) show the following result:

Theorem

Estimators constructed with QMC padded with LHS satisfy a Central Limit Theorem.

Moreover, the asymptotic variance is no larger than the variance from pure Monte Carlo sampling or from QMC sampling padded with Monte Carlo.

General Methods to Determine Important Variables

From the literature on sensitivity analysis, there are a number of different methods to determine important variables, such as:

- Screening methods
- Regression/Correlation coefficients
- Variance based methods
- Principal component analysis
- Etc.

Two-stage stochastic program with fixed recourse

Recall the two-stage problems of the form

$$\min_{x \in X} \{g(x) := c^T x + \mathbb{E}[Q(x, \xi)]\}$$

where

$$Q(x, \xi) := \inf\{q^T y : Wy \geq h - Tx, \ y \geq 0\}.$$

Random variables: $\xi = (h, T)$

Two-stage stochastic program with fixed recourse

Recall the two-stage problems of the form

$$\min_{x \in X} \{g(x) := c^T x + \mathbb{E}[Q(x,\xi)]\}$$

where

$$Q(x,\xi) := \inf\{q^T y : Wy \geq h - Tx,\ y \geq 0\}.$$

Random variables: $\xi = (h, T)$

Once samples $\xi^1, \ldots, \xi^N$ are obtained, the sampled problem can be solved using standard techniques.

- Thus, our focus is on estimating $\mathbb{E}[Q(x,\xi)]$.

The second-stage dual problem

Dual Problem:

$$\sup\{\pi^T(h - Tx) : \pi^T W \leq q^T, \pi \geq 0\}$$

Thus,

$$Q(x, \xi) = \sum_k \pi_k^*(h_k - \sum_j T_{kj}x_j)$$

where π^* are the optimal dual multipliers.

The second-stage dual problem

Dual Problem:

$$\sup\{\pi^T(h - Tx) : \pi^T W \leq q^T, \pi \geq 0\}$$

Thus,

$$Q(x, \xi) = \sum_k \pi_k^*(h_k - \sum_j T_{kj} x_j)$$

where π^* are the optimal dual multipliers.

- The multiplier π_k measures, in a sense, the importance of the term $h_k - \sum_j T_{kj}$.
- Goal is to combine use of π with variance information of individual random variables.

Determining the set of important variables

We use some heuristics to estimate V_k, the overall contribution to the variance from random variable k.

Determining the set of important variables

We use some heuristics to estimate V_k, the overall contribution to the variance from random variable k.

- Let us decompose $Q(x, \xi)$ as $\sum_{i=1}^{m} Z_i$ where each Z_i contains either 0 or 1 random components of ξ.

Determining the set of important variables

We use some heuristics to estimate V_k, the overall contribution to the variance from random variable k.

- Let us decompose $Q(x, \xi)$ as $\sum_{i=1}^{m} Z_i$ where each Z_i contains either 0 or 1 random components of ξ.

  Then: $\mathrm{Var}(Q(x, \xi)) = \sum_{i,j} \mathrm{Cov}(Z_i, Z_j) = \sum_{k=1}^{s} V_k$.

- Now we just need heuristics to decide which covariance terms should be assigned to which V_k

PCA heuristics

Let $S := \text{Cov}(Z)$. Then, we can write $S = U\lambda U^T$, where Λ is a diagonal matrix with the eigenvalues of S and U is an orthonormal matrix with the eigenvectors of S.

PCA heuristics

Let $S := \text{Cov}(Z)$. Then, we can write $S = U\lambda U^T$, where Λ is a diagonal matrix with the eigenvalues of S and U is an orthonormal matrix with the eigenvectors of S.

Then,

- By sorting the eigenvalues such that $\lambda_1 \geq \ldots \geq \lambda_m$, we can find the number of important variables k such that

$$\frac{\sum_{i=1}^{k} \lambda_i}{\text{trace}(S)} \;\geq\; \rho,$$

  where ρ is some pre-specified threshold (say, 0.9).

PCA heuristics

Let $S := \text{Cov}(Z)$. Then, we can write $S = U\lambda U^T$, where Λ is a diagonal matrix with the eigenvalues of S and U is an orthonormal matrix with the eigenvectors of S.

Then,

- By sorting the eigenvalues such that $\lambda_1 \geq \ldots \geq \lambda_m$, we can find the number of important variables k such that

$$\frac{\sum_{i=1}^{k} \lambda_i}{\text{trace}(S)} \;\geq\; \rho,$$

  where ρ is some pre-specified threshold (say, 0.9).

- To determine which ones are the important variables, we can look at the largest elements of the eigenvectors corresponding to the largest eigenvalues.

A stochastic programming algorithm with padded sampling

An External Sampling Algorithm using QMC with Padding (ES-QMCP)

- An iterative algorithm
- Number of samples increases at each iteration
- At each iteration:
  - Use current first stage solution to estimate the covariance matrix for the second stage problem and determine the important subset at that point;
  - Then use QMC with Padding to obtain new estimates of optimal value and optimal first-stage solution.
- Test of stopping criteria is based on statistical properties of gap estimators.

Stopping Criteria

We implement the stopping criteria developed by Bayraksan and Morton (2006), adapted to our padded sampling context.

- Let $\tilde{x}$ be a candidate solution, and let x_N^* be the optimal solution of the sample-average stochastic program obtained with padded QMC+LHS samples.

- Calculate

$$\mathrm{Gap}_N(\tilde{x}, x_N^*) := \frac{1}{N}\sum_{i=1}^{N}(G(\tilde{x},\xi^i) - G(x_N^*,\xi^i)) = \bar{g}(\tilde{x}) - \bar{g}(x_N^*).$$

  and

$$s_N^2(\tilde{x}, x_N^*) := \frac{1}{N-1}\sum_{i=1}^{N}((G(\tilde{x},\xi^i) - G(x_N^*,\xi^i)) - (\bar{g}(\tilde{x}) - \bar{g}(x_N^*)))^2$$

Stopping Criteria

Theorem

Suppose that $\tilde{x} \in X$, and that $\xi^1, \ldots, \xi^N$ are from a padded QMC+LHS sample of ξ. Then, under mild assumptions on G, given $0 < \alpha < 1$ we have

$$\liminf_{N\to\infty} P\left(g(\tilde{x}) - \nu^* \leq \mathrm{Gap}_N(\tilde{x}, x_N^*) + \frac{z_\alpha s_N(\tilde{x}, x_N^*)}{\sqrt{N}}\right) \geq 1 - \alpha,$$

where ν^* is the optimal value of the problem.

Stopping Criteria

Theorem

Suppose that $\tilde{x} \in X$*, and that* $\xi^1, \ldots, \xi^N$ *are from a padded QMC+LHS sample of* ξ*. Then, under mild assumptions on* G*, given* $0 < \alpha < 1$ *we have*

$$\liminf_{N \to \infty} P\left(g(\tilde{x}) - \nu^* \leq \mathrm{Gap}_N(\tilde{x}, x_N^*) + \frac{z_\alpha s_N(\tilde{x}, x_N^*)}{\sqrt{N}} \right) \geq 1 - \alpha,$$

where ν^* *is the optimal value of the problem.*

- This result yields a stopping criterion: stop the algorithm when the estimated gap is "small enough."
- Bayraksan and Morton (2006) prove this result for the i.i.d. case.
- The above extension to padded QMC+LHS uses the Central Limit Theorem for that type of sampling we showed earlier.
  - Since padded QMC+LHS has a smaller asymptotic variance, we expect the coverage given by the above theorem to be better than with Monte Carlo.

Testing the Algorithm

- We tested our algorithm using four different sampling methods: pure Monte Carlo (MC), pure Latin Hypercube Sampling (LHS), pure Quasi-Monte Carlo (QMC) and padded QMC with LHS (QMC+LHS).
- The first three methods do not care about important variables, so they use fewer samples per iteration.
- For methods involving QMC, we use scrambled (t, m, s)-nets in base $b = 2$.
- Performance Measures:
  - Optimal value at end of algorithm
  - Number of iterations until convergence

Test Problems

- We tested
  - Three small (≤ 5 random variables) problems: **gbd**, **LandS**, **apl1p**
  - One medium-sized (40 random variables) problem: **20term**
- In all problems, we used the SRP stopping criteria from Bayraksan and Morton (2006), adapted to our padded sampling context when necessary.
- For the padded sampling method, we used the PCA heuristics to select the important random variables.
- Sampled problems were solved using ATR code of Linderoth and Wright (2005), which in turn uses a modification of Linderoth's SUTL library to handle QMC sampling.

Results

apl1p	Optimal Value		Iterations	
	Mean	95% CI	Mean	95% CI
MC	24,720	232	4.9	1.4
LHS	24,593	182	2.8	1.2
QMC	24,659	206	3.2	0.8
QMC+LHS	24,664	136	2.6	1.1
	True Optimal Value = 24,642			

LandS	Optimal Value		Iterations	
	Mean	95% CI	Mean	95% CI
MC	128.27	1.69	4.6	1.2
LHS	128.09	0.44	1.6	0.5
QMC	128.19	0.15	2.3	0.7
QMC+LHS	128.26	0.13	1.5	0.5
	True Optimal Value = 128.20			

gbd	Optimal Value		Iterations	
	Mean	95% CI	Mean	95% CI
MC	1,666	39	5.1	0.7
LHS	1,663	24	2.0	0.0
QMC	1,653	35	2.1	0.7
QMC+LHS	1,666	29	1.5	0.5
	True Optimal Value = 1,656			

20term	Optimal Value		Iterations	
	Mean	95% CI	Mean	95% CI
MC	533,251	5,012	2.4	1.2
LHS	531,052	1,601	2.0	1.1
QMC	529,489	1,537	5.1	1.0
QMC+LHS	531,610	1,265	1.9	0.9
	True Optimal Value = 531,000 ± 1,000			

Comments

- All methods yield confidence intervals for the optimal value that cover the true optimum.
- Run time is proportional to number of iterations; padded QMC+LHS is the fastest, even though it uses some extra samples just to estimate covariance terms.
- Confidence intervals with padded QMC+LHS are the tightest except for **gbd**.
  - One possible explanation is that this problem is completely *separable*, so pure LHS is the best strategy.

Summary

- Quasi-Monte Carlo techniques can be effective when solving stochastic optimization problems via sampling-based methods.

Summary

- Quasi-Monte Carlo techniques can be effective when solving stochastic optimization problems via sampling-based methods.

- However, care must be taken when applied such methods — in particular, it is suggested to apply QMC on the "most important" variables and "pad" the remaining ones with either Monte Carlo or, even better, Latin Hypercube Sampling.

Summary

- Quasi-Monte Carlo techniques can be effective when solving stochastic optimization problems via sampling-based methods.

- However, care must be taken when applied such methods — in particular, it is suggested to apply QMC on the "most important" variables and "pad" the remaining ones with either Monte Carlo or, even better, Latin Hypercube Sampling.

- It is important to incorporate the selection of the variables on which QMC is applied into an optimization algorithm.
  - Such selection procedures should use structure of the problem (e.g., dual variables) as much as possible.

Summary

- Quasi-Monte Carlo techniques can be effective when solving stochastic optimization problems via sampling-based methods.

- However, care must be taken when applied such methods — in particular, it is suggested to apply QMC on the "most important" variables and "pad" the remaining ones with either Monte Carlo or, even better, Latin Hypercube Sampling.

- It is important to incorporate the selection of the variables on which QMC is applied into an optimization algorithm.
  - Such selection procedures should use structure of the problem (e.g., dual variables) as much as possible.

- Other sampling approaches are, of course, possible; however, one needs to show that the sampling technique possesses some statistical properties (e.g., Central Limit Theorem) in order to have performance guarantees.