Quasi-Monte Carlo – halftoning in high dimensions?

Ken Hanson

CCS-2, Methods for Advanced Scientific Simulations
Los Alamos National Laboratory

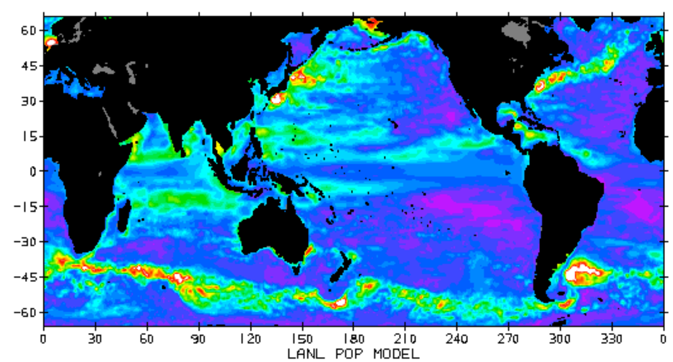
This presentation available at
http://www.lanl.gov/home/kmh/

Overview

- Digital halftoning – purpose and constraints
  - direct binary search (DBS) algorithm for halftoning
  - minimize cost function based on human visual system
- Quasi-Monte Carlo (QMC) – purpose, examples
- Minimum Visual Discrepancy (MVD) algorithm for points, analogous to DBS
  - examples
  - integration tests
- Extensions; higher dimensions, non-uniform sampling
  - possible approaches - Voronoi, electrostatic repulsion, …

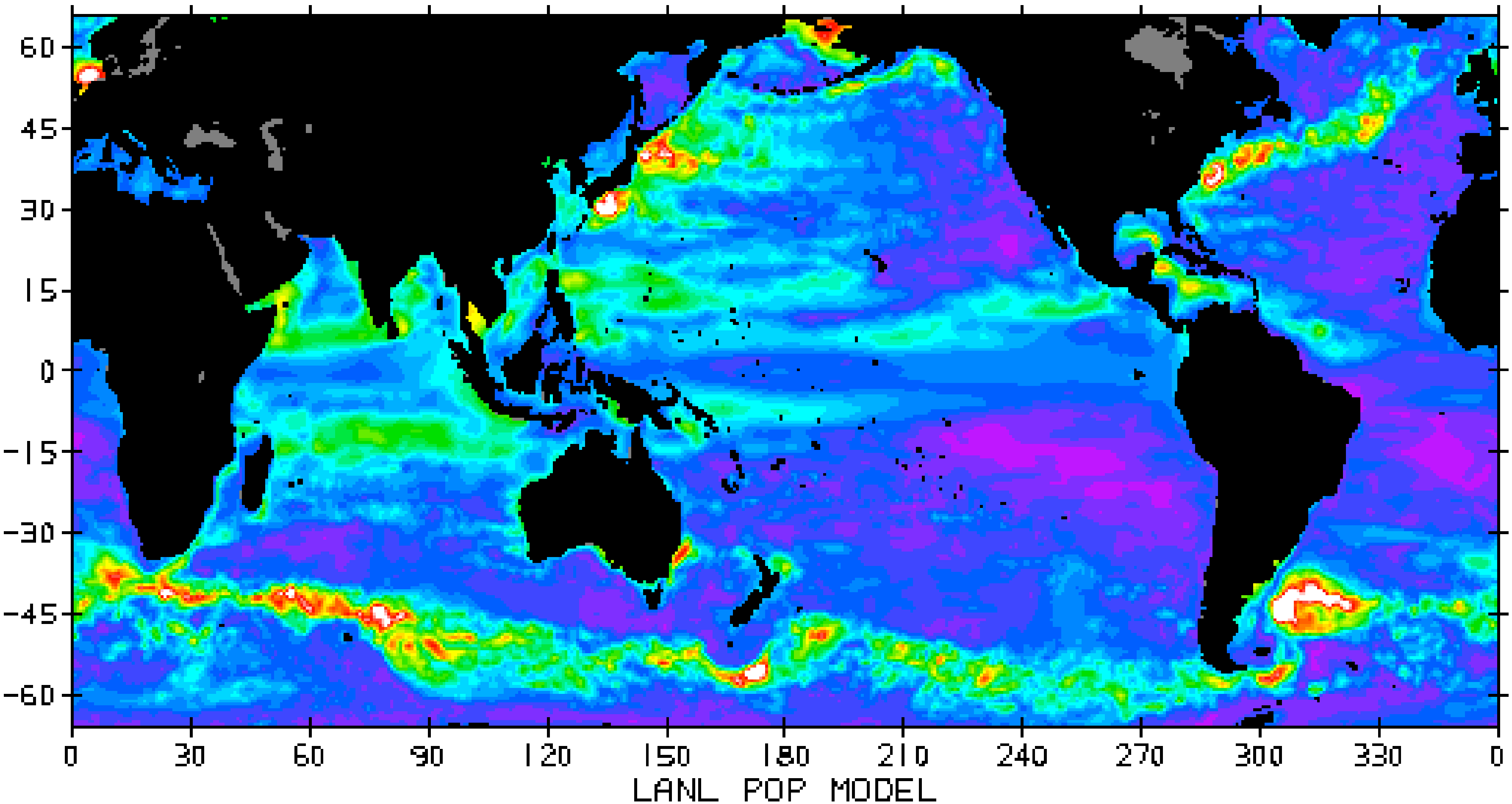
Validation of physics simulation codes

- Computer simulation codes
  - many input parameters, many output variables
  - very expensive to run; up to weeks on super computers
- It is important to validate codes - therefore need
  - to compare codes to experimental data; make inferences
  - advanced methods to estimate sensitivity of simulation outputs on inputs
    - Latin square (hypercube), stratified sampling, quasi-Monte Carlo
- Examples of complex simulations
  - ocean and atmosphere modeling
  - aircraft design, etc.
  - casting of metals



Example of ocean model simulation

1/6 degree resolution – rms dev. in ocean height

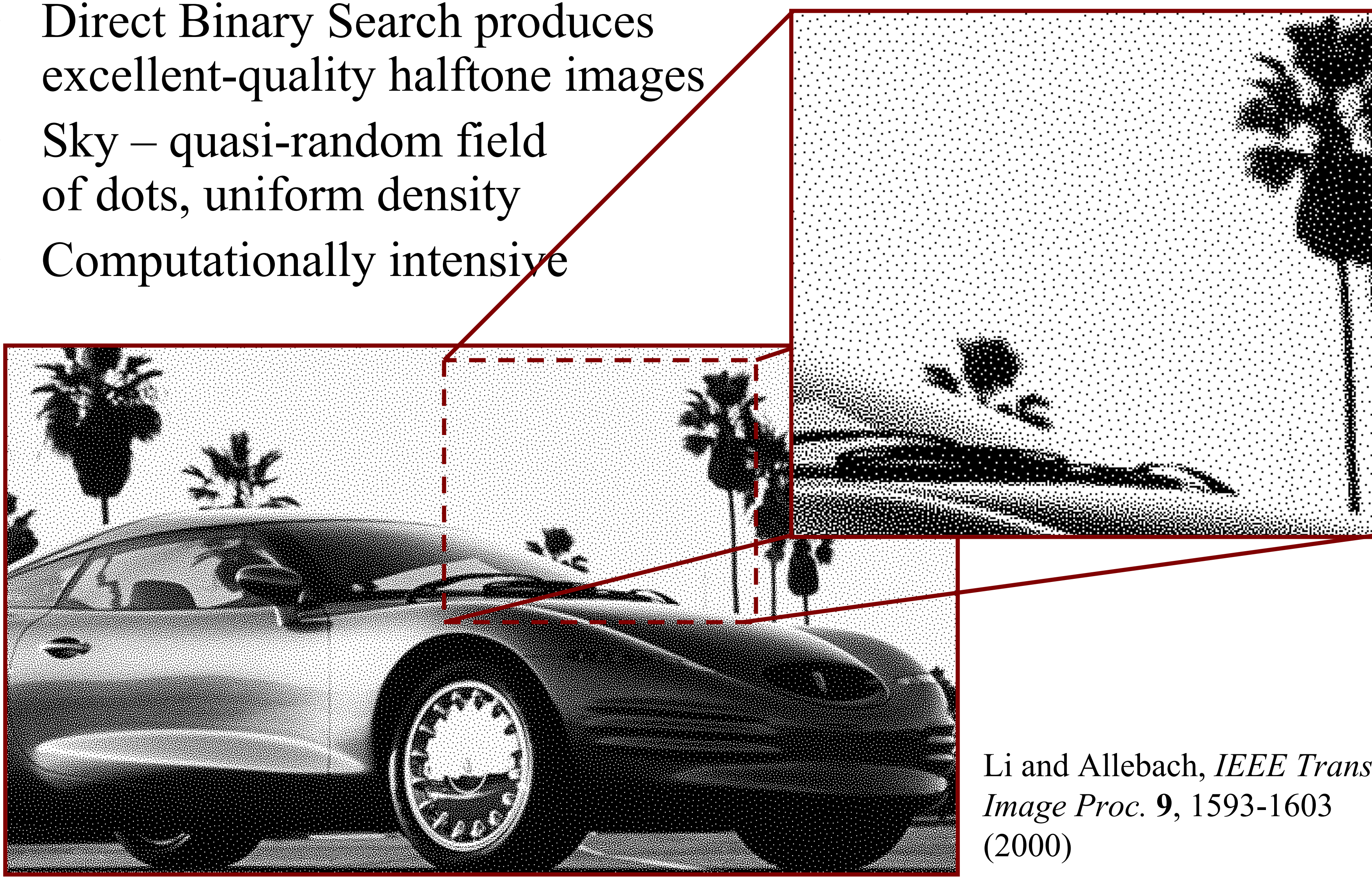


Digital halftoning techniques

- Purpose
  - render a gray-scale image by placing black dots on white background
  - make halftone rendering **look** like original gray-scale image
- Constraints
  - resolution – size and closeness of dots, number of dots
  - speed of rendering
- Various algorithmic approaches
  - error diffusion, look-up tables, blue-noise, …
  - concentrate here on Direct Binary Search

DBS example

- Direct Binary Search produces excellent-quality halftone images
- Sky – quasi-random field of dots, uniform density
- Computationally intensive



Li and Allebach, *IEEE Trans. Image Proc.* **9**, 1593-1603 (2000)

Direct Binary Search (DBS) algorithm

- Consider digital halftone image to be composed of black or white pixels
- Cost function is based on perception of two images

$$\varphi = \left| \mathbf{h} * (\mathbf{d} - \mathbf{g}) \right|^2$$

  - where **d** is the dot image, **g** is the gray-scale image to be rendered, **h** is the image of the blur function of the human eye, and * represents convolution
- To minimize φ
  - start with a collection of dots with average local density ~ **g**
  - iterate sequentially through all image pixels;
  - for each pixel, swap value with neighborhood pixels, or toggle its value to reduce φ

Monte Carlo integration techniques

- Purpose
  - estimate integral of a function over a specified region R in m dimensions, based on evaluations at n sample points

$$\int_R f(x)\,dx = \frac{V_R}{n}\sum_{i=1}^{n} f(x_i)$$

- Constraints
  - integrand not available in analytic form, but calculable
  - function evaluations may be expensive, so minimize them
- Algorithmic approaches
  - focus on accuracy in terms of # function evaluations n
  - quadrature (Simpson) – good for few dimensions; rms err ~ n^{-1}
  - Monte Carlo – useful for many dimensions; rms err ~ $n^{-1/2}$
  - quasi-Monte Carlo – reduce # evaluations; rms err ~ n^{-1}

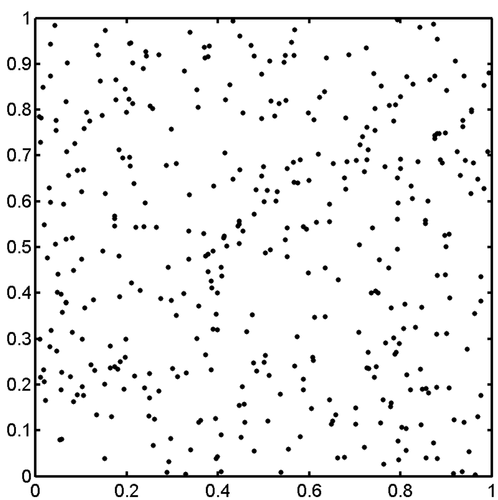
Quasi-Monte Carlo

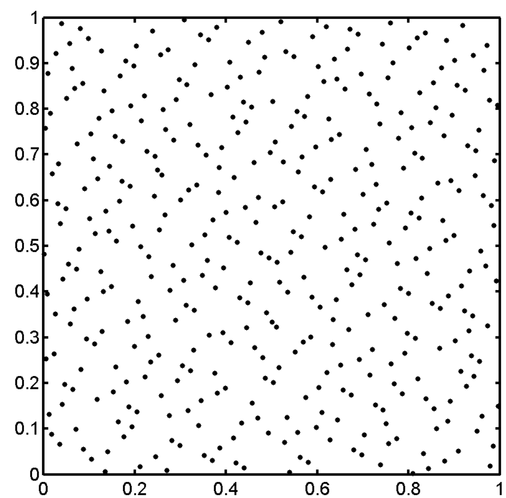
- Purpose
  - estimate integral of a function over a specified domain in *m* dimensions
  - obtain better rate of convergence of integral estimation than seen in classic Monte Carlo
- Constraints
  - integrand function not available analytically, but calculable
  - function known (or assumed) to be well behaved
- Standard QMC approaches use low-discrepancy sequences; product space
  (Halton, Sobel, Faure, Hammersley, …)
- Propose new way of generating sets of sample points

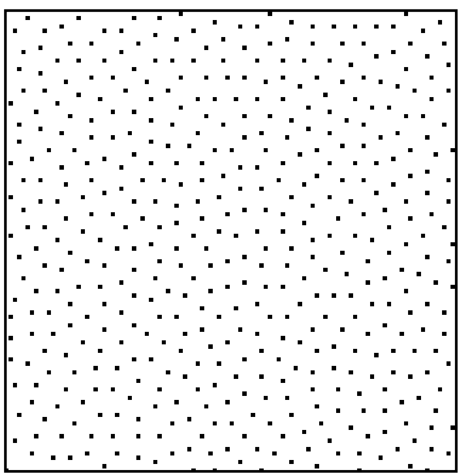
Point set examples

- Examples of different kinds of point sets
  - 400 points in each
- If quasi-MC sequences have better integration properties than random, is halftone pattern even better?

Random (independent)	Quasi-Random (Halton sequence)	Halftone (DBS sky)





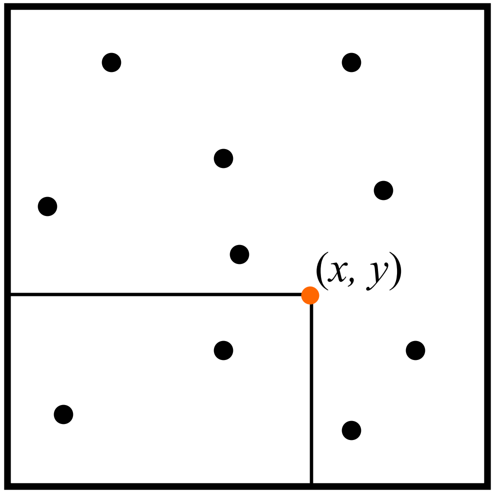


Discrepancy

- Much of QMC work is based on the discrepancy, defined for samples covering the unit square in 2D as

$$\boldsymbol{D}_2 = \int_{\mathrm{U}} \left[\boldsymbol{n}(\boldsymbol{x},\boldsymbol{y}) - \boldsymbol{A}(\boldsymbol{x},\boldsymbol{y})\right]^2 \; \boldsymbol{dxdy}$$

  - where integration is over unit square,
  - $n(x, y)$ is the number of points in the rectangle with opposite corners (0, 0) to (x, y), and
  - $A(x, y)$ is the area of the rectangle

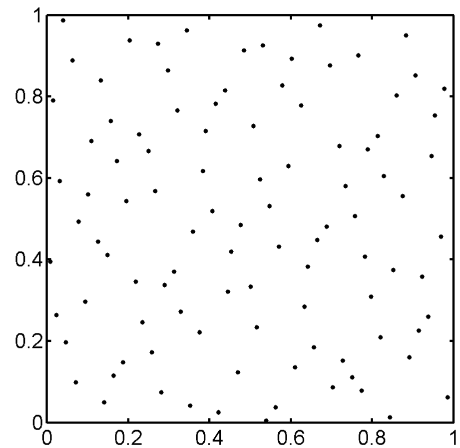


- Related to upper bounds in integr. error for class of funcs.
- Clearly a measure of uniformity of dot distribution

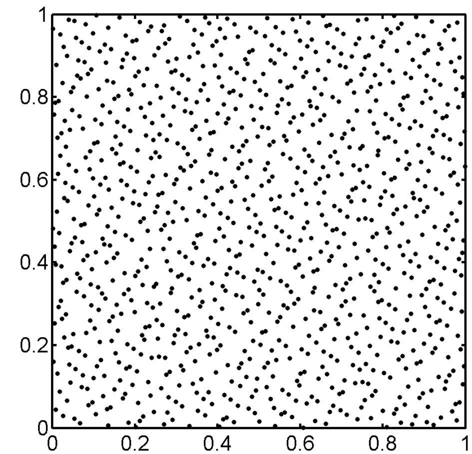
Standard Quasi-MC sequences with low D_2

- Halton
  - based on expansion in terms of fractions of powers of primes, for the prime p=2:
    1/2, 1/4, 3/4, 1/8, 5/8, 3/8, 7/8,…
- Sobel
  - based on primitive polynomials
- Observe similarity to halftone patterns for 100 points
  - points could be more uniformly distributed
- But objectionable patterns develop for many point

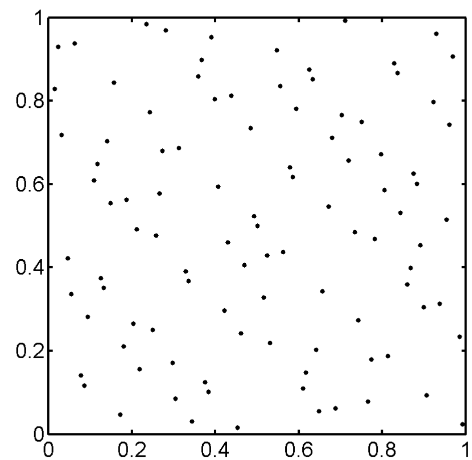
100 - Halton[2,3]



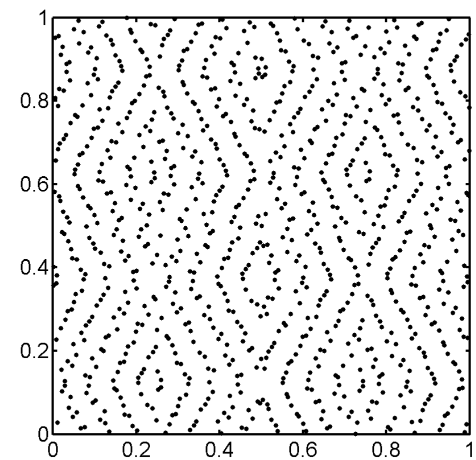
1000 - Halton[2,3]



100 - Sobel[1,2]



1000 - Sobel[1,2]



Minimum Visual Discrepancy (MVD) algorithm

Inspired by Direct Binary Search halftoning algorithm

- Start with some set of points
- Goal is to create uniformly distributed set of points
- Cost function is variance in blurred point image

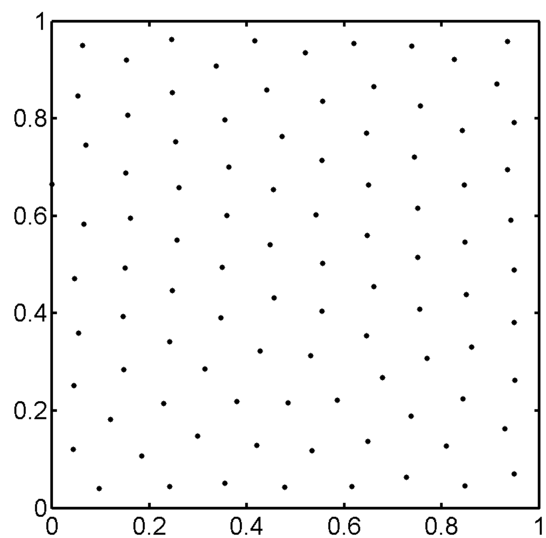
$$\psi = \mathrm{var}(\mathbf{h} * \mathbf{d})$$

  - where **d** is the point (dot) image, **h** is the blur function of the human eye, and * represents convolution
- Minimize ψ by
  - starting with some point set (random, stratified, Halton,…)
  - iterating through points in random order;
  - moving each point in 8 directions, and accept move that lowers ψ the most

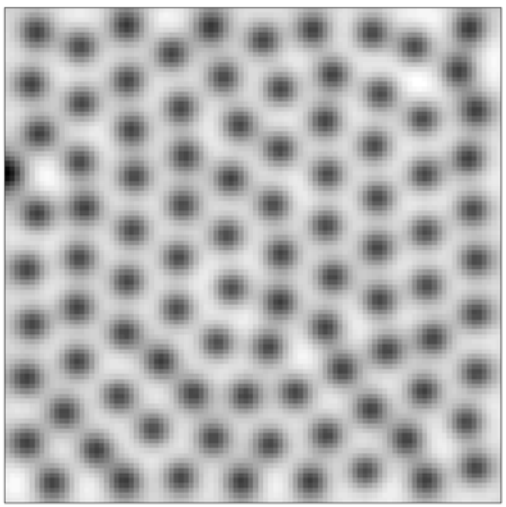
Minimum Visual Discrepancy (MVD) algorithm

- MVD result; start with100 points from Halton sequence
- MVD objective is to minimize variance in blurred image
- Effect is to force points to be evenly distributed, or as far apart from each other as possible
- Might expect global minimum is a regular pattern

100, MVD



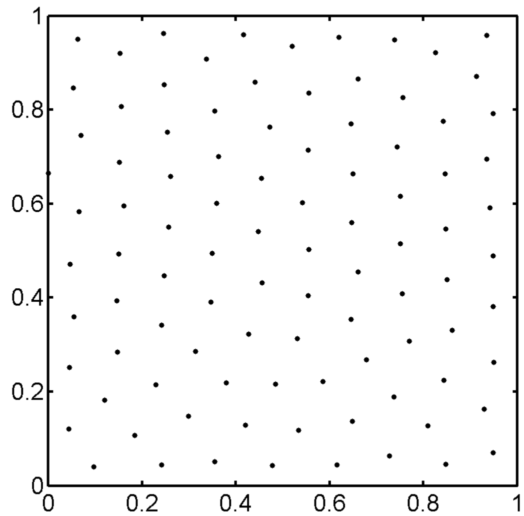
Blurred image



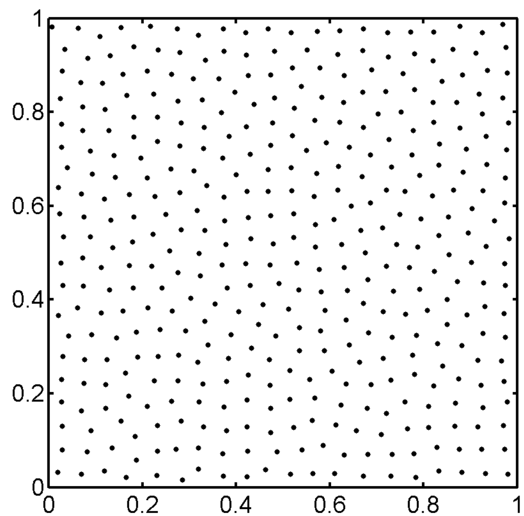
MVD results

- Final MVD distribution depends on initial point set
  - algorithm seeks local minimum, not global (as does DBS)
- Patterns somewhat resemble regular hexagonal arrary
  - similar to lattice structure in crystals
  - however, lack long-range (coarse scale) order
  - best to start with point set with good long-range uniformity

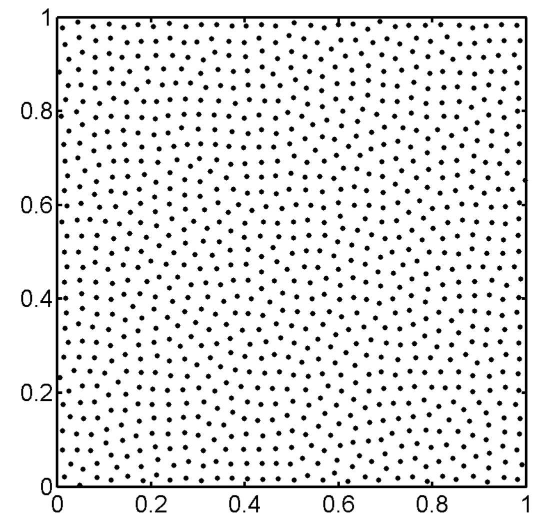
100, MVD



400, MVD



1000, MVD

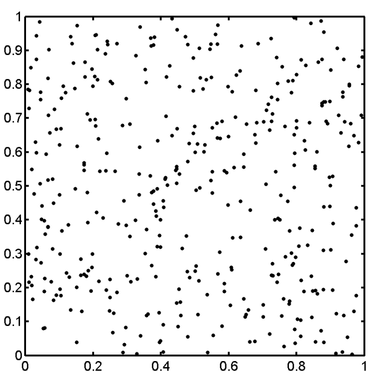


Point set examples

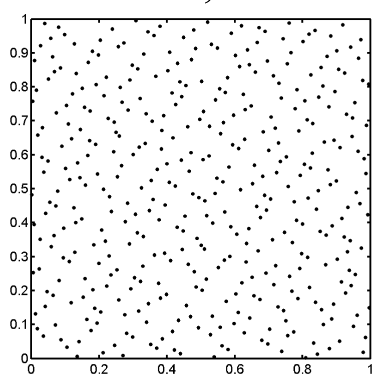
- Various kinds of point sets (400 points)
- Varying degrees of randomness and uniformity
- As the points become more uniformly distributed, the more accurate are the values of estimated integrals

RMS relative accuracies of integral of $\text{func2} = \prod_i \exp\left(-2\left|\boldsymbol{x}_i - \boldsymbol{x}_i^0\right|\right); \quad 0 < \boldsymbol{x}_i^0 < 1$

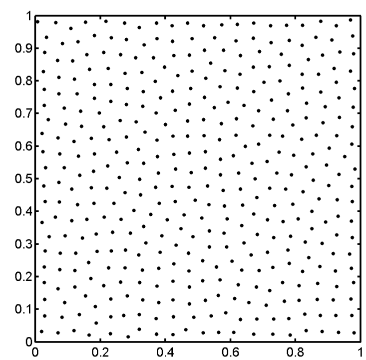
Random, 2.5%



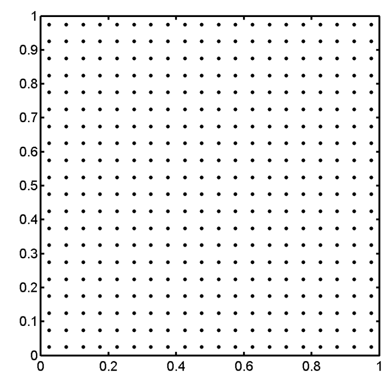
Halton, 0.5%



MVD, 0.14%

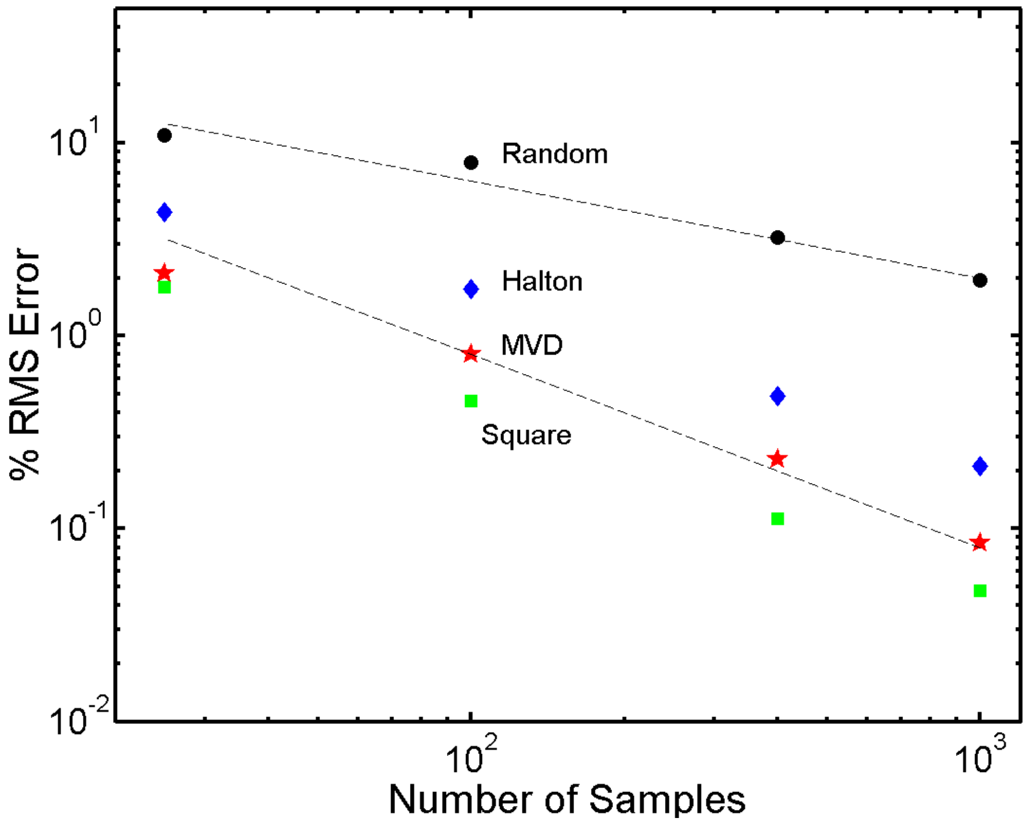


Grid, 0.09%



More Uniform, Higher Accuracy

Integration test problems

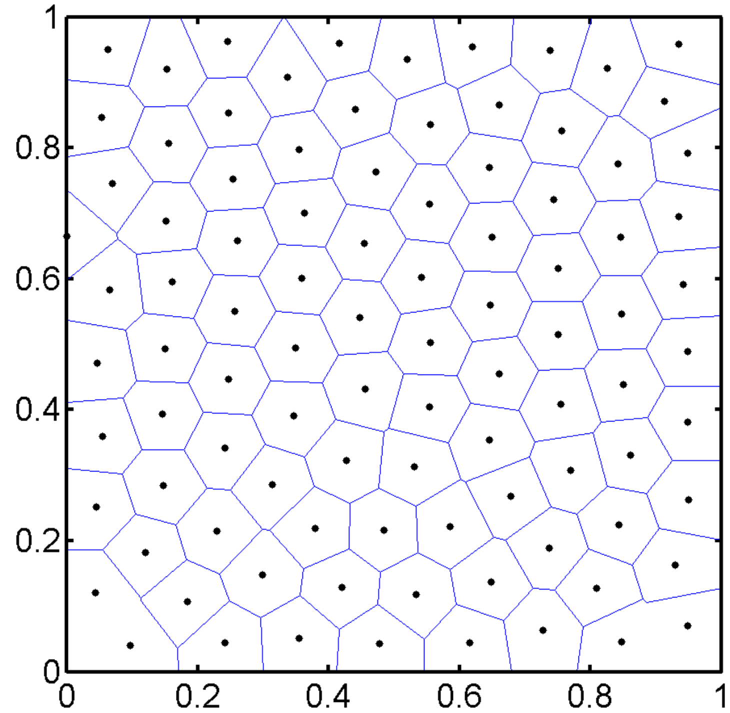


- RMS error for integral of $\mathrm{func2} = \prod_i \exp\left(-2\left|x_i - x_i^0\right|\right); \quad 0 < x_i^0 < 1$
  - from worst to best –random, Halton, MVD, square grid
  - lines show $N^{-1/2}$ (expected for MC) and N^{-1} (expected for QMC)

Voronoi analysis

- Voronoi diagram
  - partitions region of interest into polygons
  - points within each polygon are closest to one generating point, Z_i
- MC technique provides easy way to do Voronoi analysis
  - randomly throw large number of points X_i into region
  - compute distance of each X_i to all generating points $\{Z_i\}$
  - sort into those closest to each Z_i to identify
  - can compute A_i, radial moments,…
- Extensible to high dimensions

100, MVD



Metric needed to rank value of point sets

- Need to be able to identify “good” point sets
- Especially important in high dimensions where visualization is difficult or impossible
- From integration tests of several functions and many different kinds of point sets, observe:
  - discrepancy D_2 does not seem to track rms error
  - Voronoi analysis does not seem to track rms error
  - but, low rms errors are obtained when both D_2 and rms deviation of V areas are small

Conclusions

- Minimum Visual Discrepancy algorithm
  - ► produces point sets resembling uniform halftone images
  - ► yields better integral estimates than standard QMC seqs.
- Extensions
  - ► Prospects for creating good point sets in high dimensions
    - MVD will not work; need discrete representation of image (?)
    - electrostatic potential field approach is promising
      - – analogous to collection of electrons confined to box
      - – perhaps similar to 'springs' model of Atkins et al.
    - Voronoi analysis – centroidal Voronoi tesselation
  - ► Sequential development of point set
    - add one point at a time, placing it at an optimal location, that is, in holes

Bibliography

- K. M. Hanson, "Quasi-Monte Carlo: halftoning in high dimensions?," to be published in *Proc. SPIE* **5016** (2003)
- P. Li and J. P. Allebach, "Look-up-table based halftoning algorithm," *IEEE Trans. Image Proc.* **9**, pp. 1593-1603 (2000)
- H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM (1992)
- Q. Du, V. Faber, and M. Grunburger, "Centroidal Voronoi tesselations: applications and algorithms," *SIAM Review* **41**, 637-676 (1999)

This presentation and paper available at
http://www.lanl.gov/home/kmh/