Monte Carlo Simulations of Gravimetric Terrain Corrections Using LIDAR Data

J. A. Rod Blais
Dept. of Geomatics Engineering
Pacific Institute for the Mathematical Sciences
University of Calgary, Calgary, AB
www.ucalgary.ca/~blais

Outline

- Overview of Gravimetric Terrain Corrections
- Example of Current Application with Airborne Gravimetry
- Computation Approaches for Gravimetric Terrain Corrections
- Airborne LIDAR Dense Grids of Accurate Terrain Data
- Simulations for Gravimetric Terrain Corrections
- Accuracy of Simulated Terrain Corrections
- Concluding Remarks

Gravimetric Terrain Correction

Newtonian Potential U(P) at some point P = (x, y, z):

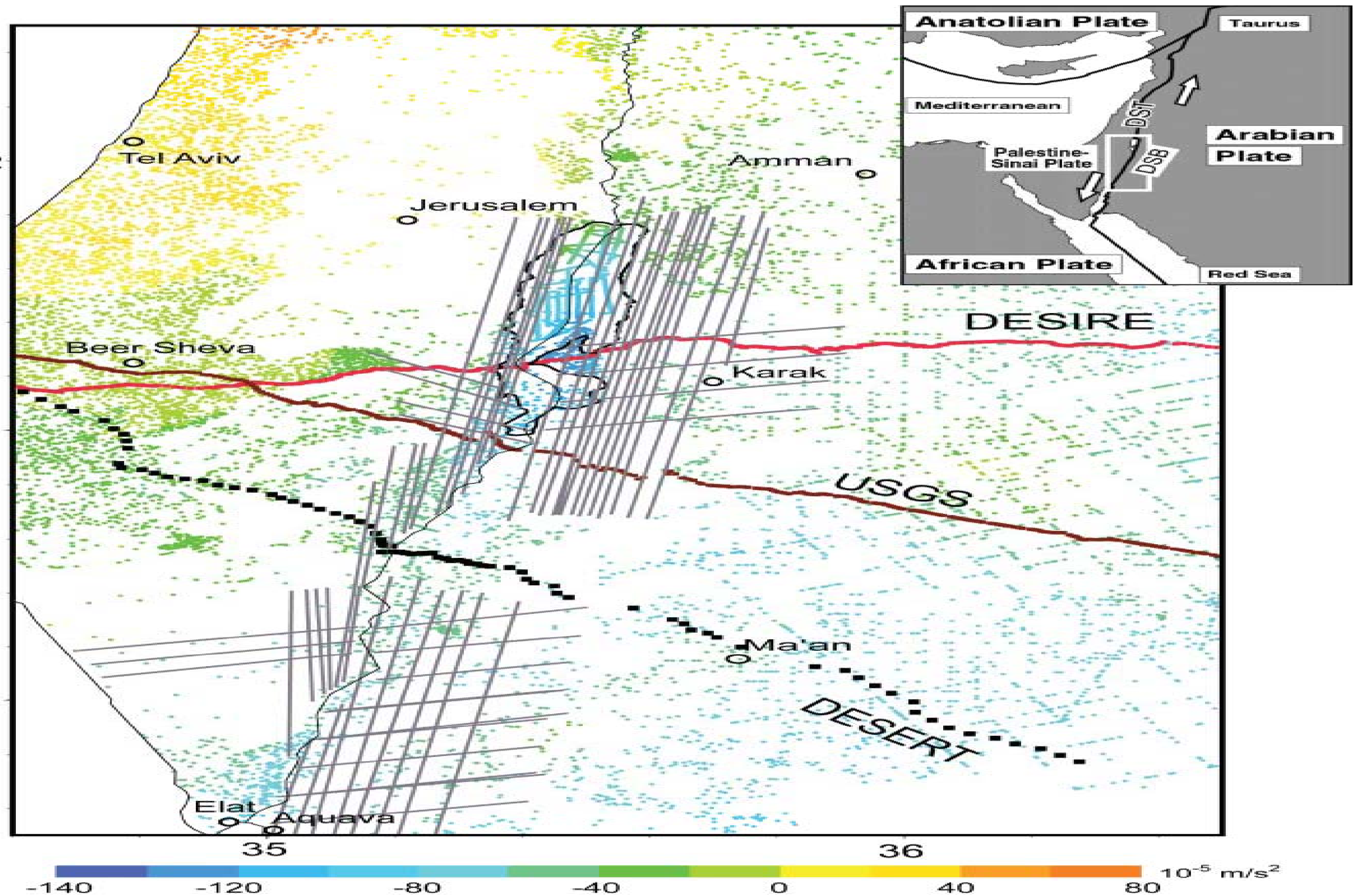
$$\mathbf{U}(\mathbf{P}) = \mathrm{G}\iiint_E \frac{\rho(\mathbf{Q})}{|\mathbf{P}-\mathbf{Q}|}\,\mathrm{d}\sigma(\mathbf{Q})$$

Vertical gradient assuming z ~ height:

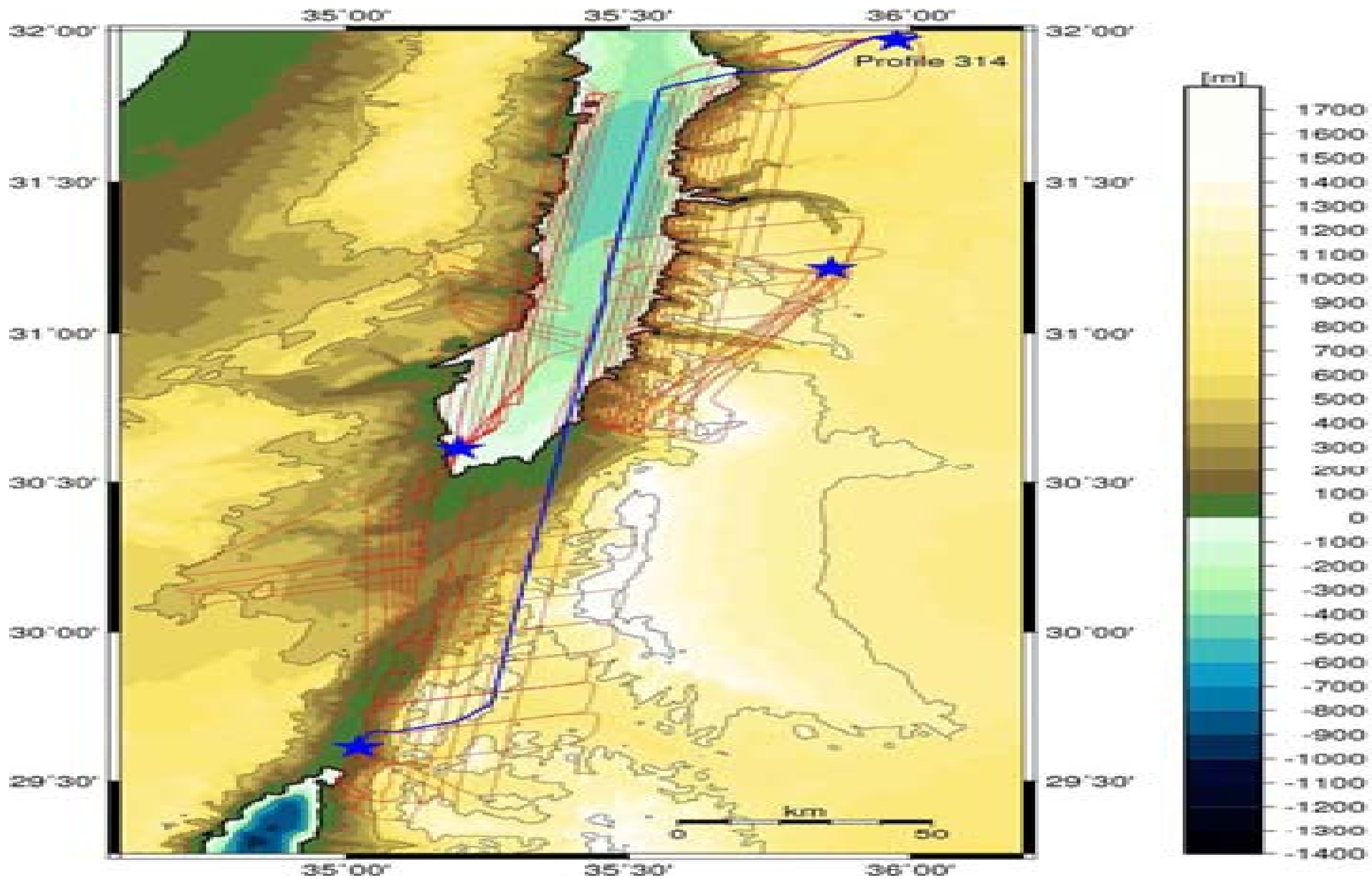
$$\frac{\partial}{\partial \mathrm{z}}\mathbf{U}(\mathbf{P}) = -\mathrm{G}\iiint_E \frac{\rho(\mathbf{Q})\mathrm{z}(\mathbf{Q})}{|\mathbf{P}-\mathbf{Q}|^3}\,\mathrm{d}\sigma(\mathbf{Q})$$

in which ρ(Q) denotes the density of the Earth (*E*) at location Q and G is Newtonian's gravitational constant, i.e. G = 6.672×10^{-11} m^3 s^{-2} kg^{-1}

Note: ρ(crust) ≈ 2.67 g cm^{-3} and 1 mgal = 10^{-5} m s^{-2} = 10^{-8} km s^{-2}

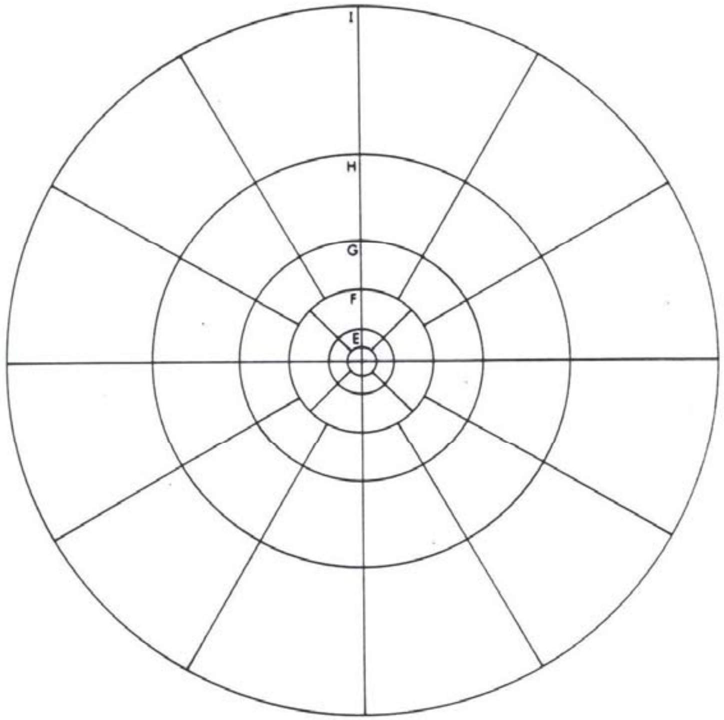


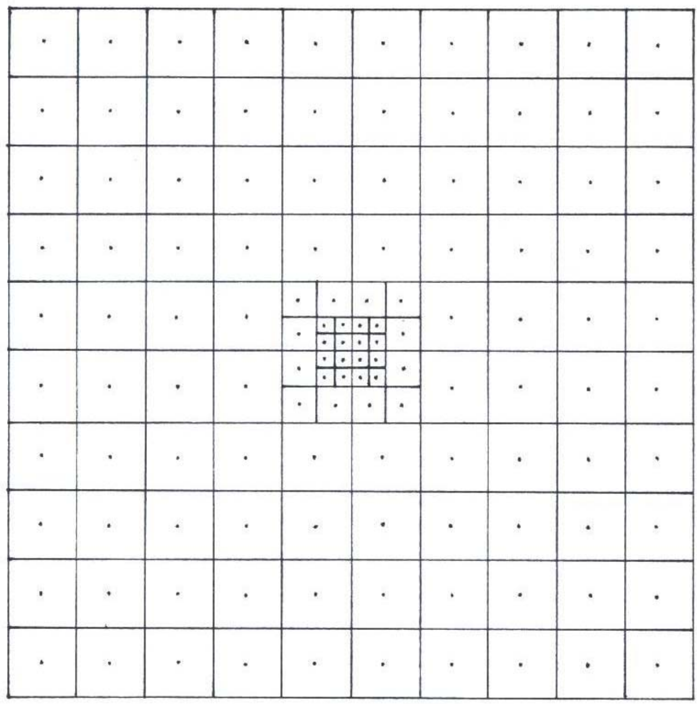
Source: EOS, Vol.91, No.12, 23 March 2010



Source: EOS, Vol.91, No.12, 23 March 2010

Templates for Multigrid Quadratures





Integral Approach

Direct Integration

$$\delta g(x_o, y_o, z_o) = G\bar{\rho}\int_{-L}^{L}\int_{-L}^{L}\int_{0}^{H(x,y)} \frac{z\,dz\,dy\,dx}{((x-x_o)^2 + (y-y_o)^2 + (z-z_o)^2)^{3/2}}$$

or

$$\delta g(r_o, \theta_o, h_o) = G\bar{\rho}\int_{0}^{R}\int_{0}^{2\pi}\int_{0}^{H(r,\theta)} \frac{r\,h\,dh\,d\theta\,dr}{((r-r_o)^2 + (h-h_o)^2)^{3/2}}$$

$$= 2\pi G\bar{\rho}\int_{0}^{R}\int_{0}^{H(r)} \frac{r\,h\,dh\,dr}{((r-r_o)^2 + (h-h_o)^2)^{3/2}}$$

Cartesian Prism Approach

Direct Integration

$$\delta g = -G\bar{\rho}\left|\left|\left| x\log(y+r) + y\log(x+r) + z\arctan\frac{zr}{xy} \right|_{x_1}^{x_2}\right|_{y_1}^{y_2}\right|_{z_1}^{z_2}$$

or simplifying to a known cross-section s

$$\delta g \approx G\bar{\rho}\, s\int_0^h \frac{z\,dz}{(d^2+z^2)^{3/2}} = G\bar{\rho}\, s\left(\frac{1}{d} - \frac{1}{\sqrt{d^2+h^2}}\right)$$

which is usually called the line mass formula.

Airborne LIDAR

Light Detection and Ranging

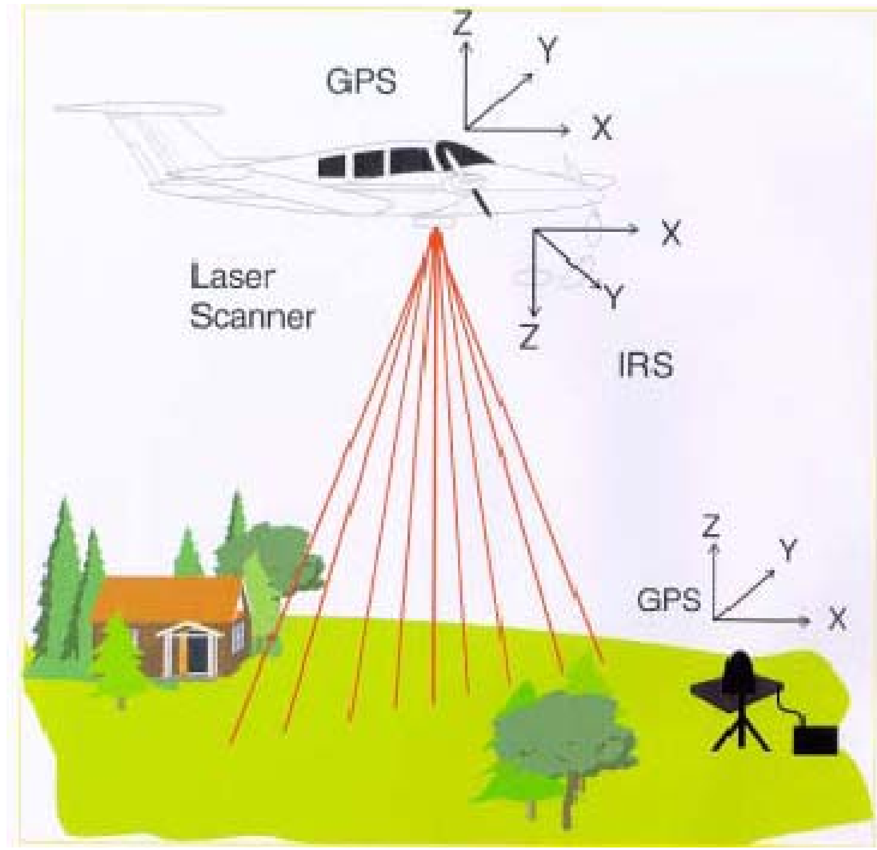
Airborne laser, GPS & INS

DEM rapid data collection

Grid with sub-metre resolution

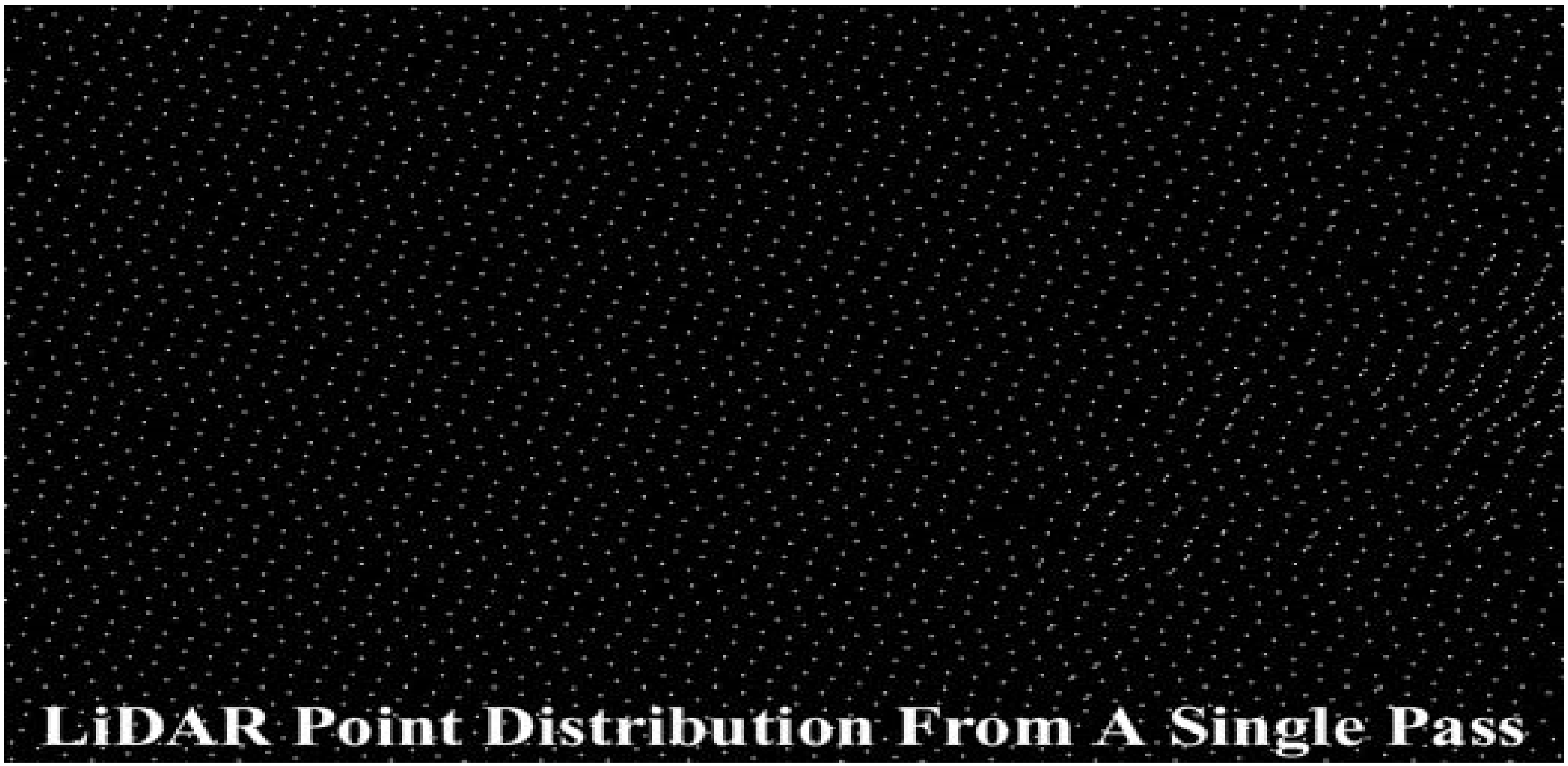
Height accuracy: 15-25 cm

Ideal for special projects (e.g., www.ambercore.com)



Airborne LIDAR System (author unknown)

LIDAR Data Coverage Example



Source: Ohio Dept. of Transportation

Typical LIDAR Sensor Characteristics [USACE, 2002]

Parameter	Typical Value(s)
Vertical Accuracy	15 cm
Horizontal Accuracy	0.2 – 1 m
Flying Height	200 – 6000 m
Scan Angle	1 – 75 deg
Scan Rate	0 – 40 Hz
Beam Divergence	0.3 – 2 mrads
Pulse Rate	05 – 33 KHz
Footprint Diameter from 1000 m	0.25 – 2 m
Spot Density	0.25 – 12 m

Monte Carlo Simulations

Numerical Recipes [Press et al, 1986] state:

$$\int_V f\, dV \approx V\langle f\rangle \pm \sqrt{\left(\langle f^2\rangle - \langle f\rangle^2\right)/N}$$

where

$$\langle f\rangle = N^{-1}\sum_{n=1}^{N} f(n)$$

and

$$Var(f) = \left\langle \left(f - \langle f\rangle\right)^2 \right\rangle = \langle f^2\rangle - \langle f\rangle^2$$

implying a standard error of $O(1/\sqrt{N})$ or a variance of $O(1/N)$

Random Numbers

- **Pseudorandom** (PRN) sequences are commonly generated using some linear congruential model applied recursively, such as

  $x_n \equiv c \odot x_{n-1}$ modulo π (for large prime π and constant c)

  or lagged Fibonacci congruential sequence, such as

  $x_n \equiv x_{n-p} \odot x_{n-q}$ modulo π (for large primes π and p, q)

  in which $\odot$ usually stands for ordinary multiplication

- **Chaotic-random** (CRN) sequences generated by e.g.

  $x_n = 4\, x_{n-1}\,(1-x_{n-1})$, $n = 1, 2, \ldots,$ (Logistic equation)

  for some seed x_0, over (0, 1), exhibits randomness with a density

  $\rho(x) = 1 / \pi\, [x\,(1 - x)]^{1/2}$ (correction needed)

- **Quasi-random** (QRN) sequences are 'equidistributed' sequences

Numerical Experimentation

PMC / QMC / CMC	N = 10	N = 10^2	N = 10^3	N = 10^4
$\int_0^1 e^x dx$	1.56693421	1.63679860	1.70388586	1.71894429
	1.56693421	1.71939163	1.71994453	1.71812988
$\cong$ **1.718281828459045**	1.67154678	1.73855363	1.76401394	1.72791977
$\int_0^1 \int_0^1 e^{xy} dxdy$	1.23409990	1.31809139	1.31787793	1.31790578
	1.23409990	1.31785979	1.31789668	1.31790120
$\cong$ **1.317902151454404**	1.21656321	1.27903348	1.34063983	1.31179521
$\int_0^1 \int_0^1 \int_0^1 e^{xyz} dxdydz$	1.14046759	1.14625944	1.14650287	
	1.14046759	1.14649963	1.14649879	
$\cong$ **1.146499072528643**	0.99503764	1.14428655		

LIDAR Terrain Simulations

Topography:

Cosine Model:

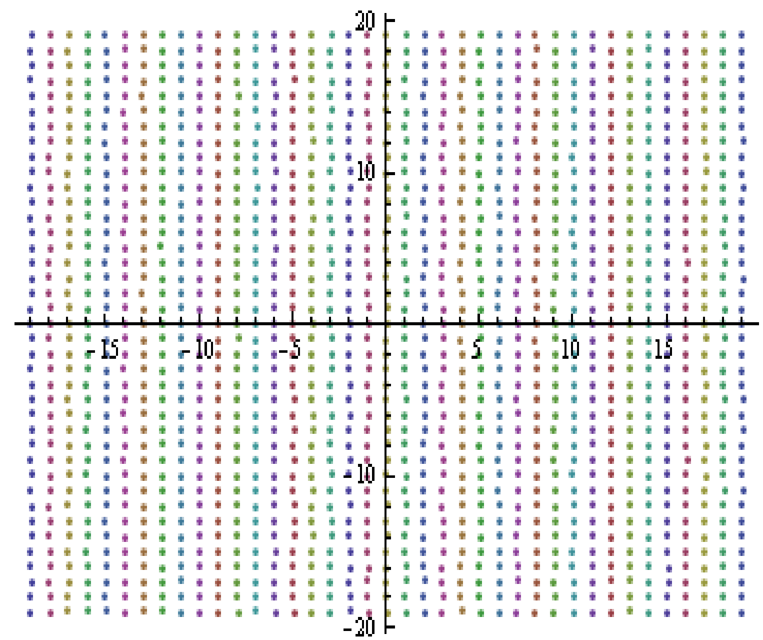
$$H_1(x,y) = k_1[1 - \cos\alpha x \cos\beta y]$$

Exponential Model:

$$H_2(x,y) = k_2[e^{-\alpha x^2 - \beta y^2} - 1]$$

Logarithmic Model:

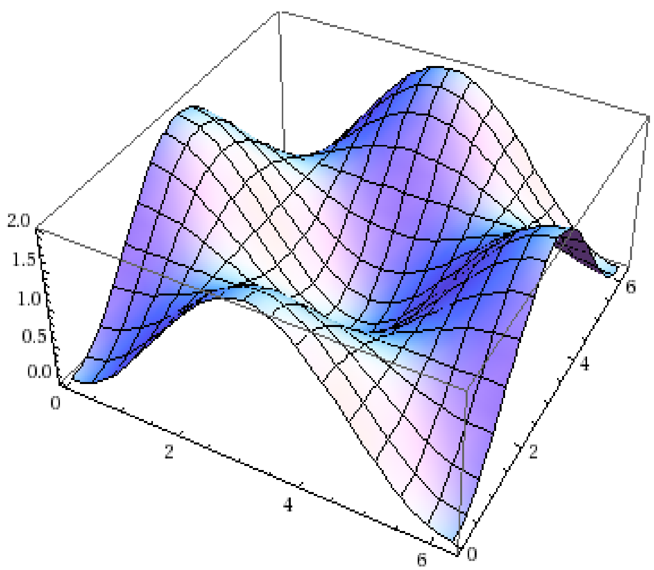
$$H_3(x,y) = k_3 \log[1 + \alpha x^2 + \beta y^2]$$



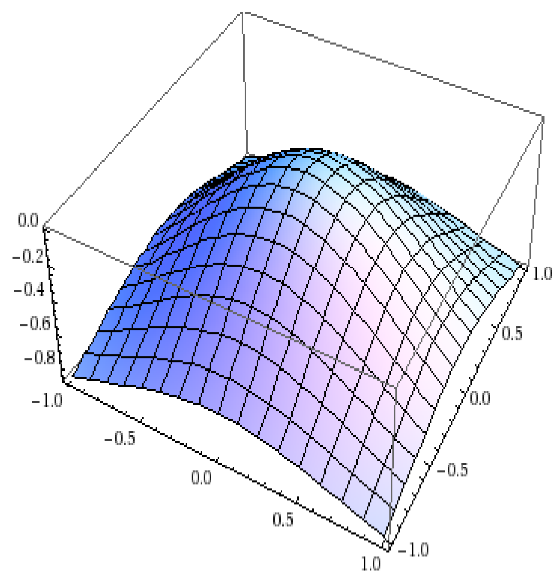
LIDAR Grid:

$$(x, y) = (i, j) + k \cdot \mathrm{UniformRandom}(0,1)$$

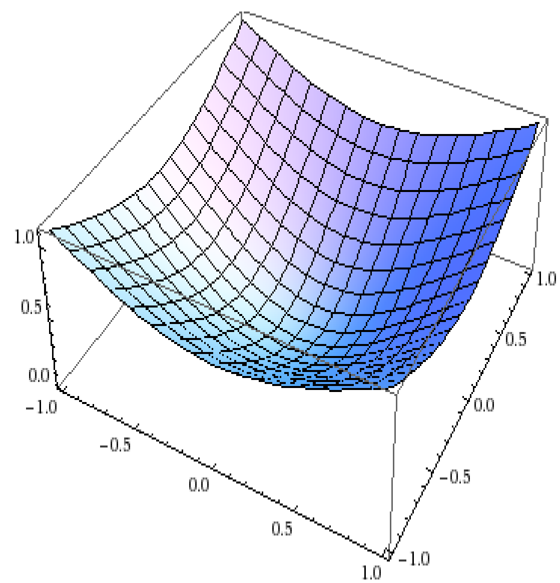
Simulated Terrain Shapes



Cosine Model



Exponential Model



Logarithmic Model

Quasi-Monte Carlo Formulation

For a gravity station at the origin,

$$\delta g(0,0,0) = G\bar{\rho}\int_{-L}^{L}\int_{-L}^{L}\int_{0}^{H(x,y)} \frac{z\,dz\,dy\,dx}{(x^2+y^2+z^2)^{3/2}}$$

then for N small prisms over an area A,

$$\delta g(0,0,0) \approx G\bar{\rho}A\left\langle \int_0^h \frac{z\,dz}{(d^2+z^2)^{3/2}} \right\rangle$$

$$\approx G\bar{\rho}A\left\langle \frac{1}{d} - \frac{1}{\sqrt{d^2+h^2}} \right\rangle$$

$$\approx \frac{G\bar{\rho}A}{N}\sum_{i=1}^{N}\left(\frac{1}{d_i} - \frac{1}{\sqrt{d_i^2+h_i^2}} \right)$$

Results of Simulations

GTC in mGal	$k = 10^3$	$k = 2 \cdot 10^3$	$k = 3 \cdot 10^3$	$k = 4 \cdot 10^3$
TERRAIN: Cosine Model* **LIDAR: » (i,j) only**	12.7892	47.9520	98.0583	155.4226
» (i,j) + URand(0, 0.2)	12.7892	47.9520	98.0583	155.4226
» scale·Urand(-0.5, 0.5)	12.7893	47.9503	98.0578	155.4180
TERRAIN: Exponential Model* **LIDAR: » (i,j) only**	45.7062	136.4131	229.2257	313.9924
» (i,j) + URand(0, 0.2)	45.7062	136.4131	229.2257	313.9924
» scale·Urand(-0.5, 0.5)	45.6991	136.4226	229.2030	314.0088
TERRAIN: Logarithmic Model* **LIDAR: » (i,j) only**	178.0971	437.7609	623.1184	746.8817
» (i,j) + URand(0, 0.2)	178.0971	437.7609	623.1184	746.8817
» scale·Urand(-0.5, 0.5)	178.0925	437.7790	623.1178	746.8773

***- All over 10 000 m x 10 000 m with gravity station in centre.**

Error Analysis Considerations

- In general, Monte Carlo simulations are known to have a standard error $O(1/N^{1/2})$, or an error variance $O(1/N)$.
- Considering the accuracy of the terrain measurements in

$$\delta g \approx G\bar{\rho}A\left\langle \frac{1}{d} - \frac{1}{\sqrt{d^2 + h^2}} \right\rangle = \frac{G\bar{\rho}A}{N}\sum_{i=1}^{N}\left(\frac{1}{d_i} - \frac{1}{\sqrt{d_i^2 + h_i^2}} \right)$$

conventional error propagation has to be carried out.

- For comparisons with conventional prism computations, some real field data are needed with gravimetric terrain corrections.

Concluding Remarks

- Practically all gravity measurements require terrain corrections
- Airborne LIDAR gives data with dm accuracy and sub-m resolution
- LIDAR terrain data can be considered as quasi-random 2D sequence
- Monte Carlo formulation uses the line mass formulation for GTC
- Numerical simulations with different terrain models are stable
- Simulation results are very convincing and useful for applications
- Real field data are needed for more complete analysis