INVDISK - INVERSION OF DISK MODEL "IMAGES" USING MEAN, VARIANCE & VARIOGRAM DATA

David L B Jupp

(Original document was written June, 1991; Modified 1997 and edited for this release in 1999. The computer program Invdisk has since been modified but outputs given here are the same as were produced in 1991. The current release supports a review by Ni and Jupp (1999) of image variance studies.)

CONTENTS

1. INTRODUCTION

The disk model is possibly the simplest of the models to which the methods outlined in Jupp *et al*. (1988,1989) can be applied. It comprises a scene consisting of disks (of possibly varying size) with centres distributed with Poisson density on an unbounded plane. The disk and background have distinct "colours" (or graylevel values) and the colour does not change where disks overlap. The first and second order statistics of such a "scene" were given in Jupp *et al*. (1989). Despite its simplicity, this scene is not unlike a vertically viewed forest.

The image model for this simple scene is one where the image data are integrals of radiance over pixels by a "sensor" with a disk shaped IFOV. The pixels are sampled equally in two directions to produce an image with finite extent. The geometric pixel size is the distance between samples and is often less than the diameter of the IFOV to avoid missing data. The sampling also covers only a portion of the plane creating a finite sample from the assumed unbounded scene. The integration of the radiances into pixel data has been called "regularisation" and the process that occurs as pixel size ranges from smaller than the disk sizes to larger than the disk sizes is illustrated in Figure 1.

From such a finite image, a number of statistics can be obtained. For example, the histogram of graylevel values can be computed (see Figure 1) and summarised by its mean and variance which, as illustrated in Figure 1, are a function of both the scene and the pixel size. Spatial statistics can also be obtained. In particular, the (semi-) variogram can be computed at lags of integral geometric pixel steps in the vertical and horizontal direction as well as at various fractional steps in diagonal directions. Images realising this simple model can be easily simulated on a computer.

Invdisk is a computer program developed to study this simple situation. Questions being addressed with Invdisk include whether such data are sufficient to completely "invert" the data to estimate disk size, spacing and even graylevel values of disk and background? The data in this case consist only of the image data and statistics derived from it. It should be clear that in terms of the general descriptions of image models given in Strahler *et al*. (1986) that the feasibility of the answer will be sensitive to whether the problem is H-resolution (IFOV much less than disk size) or L-resolution (IFOV larger than disk size). Indeed, it is known that, as a special case of the discrete canopy models of Strahler and Li (Li & Strahler, 1985, 1986), the inversion of size and spacing is possible in the L-resolution case when the graylevel values of disk and background are known. This study therefore aims at establishing what is generally possible given different relationships between pixel size and disk size and spacing.

The disk model contains much of the essential structure of the complete inversion problem for a forest canopy which is being modelled by a Li-Strahler spheroid model. It may be shown (see Franklin & Strahler, 1988; Strahler, Wu & Franklin, 1988 and Strahler & Jupp, 1991) that provided the distribution of trees (the stem density) can be effectively described by a Poisson distribution, the variance in that case is driven by the first and second order statistics of fraction of visible sunlit background and pixel size. Hence, if

- The "disk" is used to describe the composite projection of tree crown and shadow on the background (for example, by using a disk with the same area and centre of mass),
- The colour of the "disk" is taken as the visible area weighted mean of sunlit crown, shaded crown and shadow on background for the composite projection, and
- The colour of the disk model "background" is taken to be that of sunlit background,

then the disk model becomes very close to being an effective two component model for the forest distribution of colour and (spatial) variance. In fact, inverting actual forest canopy data with Invdisk in this way has been an objective of this project.

2. INVDISK MODEL

The model currently used for Invdisk is one of disks with constant diameter D_1 and area A_c ($A_c = \pi D_1^2 / 4$) having centres distributed with Poisson density λ. For this "Boolean Model", the probability of not hitting a disk is:

$$q = e^{-\lambda A_c}$$

from which it follows that, over any area, the expected graylevel (G) is:

$$
\begin{aligned}
G &= q\, g_B + (1-q)\, g_D \\
&= g_D + q\,(g_B - g_D)
\end{aligned}
$$

where g_B is the graylevel (colour) of the background and g_D is the graylevel of a disk.

In terms of the Li/Strahler model, q is the expected fraction of background in a pixel, or K_G and (1-q) is the expected fraction covered by disks, or K_C. There are no shadow fractions here. If this were the only datum available, however, since there are three unknowns and one piece of information, inversion would be impossible so other information must be provided and modelled.

The variance, variogram function and covariance function for the disk model are derived in ARDI-II (Jupp *et al.*, 1989). For the underlying scene model, in the case where the graylevels of disk and background are zero and one, the punctual covariance for step h (in terms of the normalised variable $s = h/D_1$) is:

$$
Cov(h) = q^2 (e^{\lambda A_c T(s)} - 1)
$$

where $T(s)$ is the fraction of overlap area between two disks of unit area:

If we define the auxiliary variable θ by:

$$
\cos\frac{\theta}{2} = s \qquad\qquad when\ h < D_1
$$

Then it follows that:

$$
\begin{aligned}
T(s) &= \frac{1}{\pi}(\theta - \sin\theta) \qquad && when\ h < D_1 \\
&= 0 && when\ h \geq D_1
\end{aligned}
$$

When $h > D_1$ (or, equivalently, s>1) it follows that $Cov(h)=0$ and for $h=0$ it follows that $Cov(0)=q(1-q)$ which is the binomial variance.

From the work presented in ARDI-I (abbreviation for Jupp *et al.*, 1988) and ARDI-II (abbreviation for Jupp *et al.*, 1989) it follows that for pixels with disk shaped IFOV of diameter D_2, the variance (σ_Z^2) can be modelled as:

$$
\sigma_Z^2 = 8\int_0^1 t\, T(t)\, Cov(D_2\, t)\, dt
$$

and the covariance at step $s = h/D_2$ is:

$$Cov_Z(s) = \frac{8}{\pi}\int_0^s t\,\Phi(t,s)\,Cov(D_2\;t)\,dt$$

from which the (semi)variogram can be computed as:

$$\gamma_Z(s) = \sigma_Z^2 - Cov_Z(s)$$

where s represents the lag (h) in units of multiples of the IFOV (D_2).

These expressions for image variance and variograms must be now be multiplied by $(g_D\text{-}g_B)^2$ as the equations derived in ARDI assume gray levels of zero and one. With this in place, a model for the image mean, variance and variogram data at lags between one pixel and the sill is available as a function of disk size, spacing and colour as well as pixel IFOV and geometric size. These are, of course, population values and when it comes to a simulated image the sampling distribution will need to be taken into account.

It is also important to take into account that the pixel IFOV need not be the same as the geometric pixel. The IFOV is the angle or area over which data are integrated by the sensor and the geometric pixel depends on the sampling within and between lines that provides the image data. In traditional scanners, such as Landsat MSS, Daedalus Scanners etc, the width and length of the geometric pixel are always functions of digitizer sampling and mirror scanning rate rather than optics. For theoretical studies, an equivalent “square” pixel could be taken with size 0.886 of D_2 and lags in vertical and horizontal directions taken relative to this size. This represents a square pixel with the same area as the disk. Lags in the diagonal directions would correspond to s=1.253 for the first step variogram - or local variance.

As a consequence of the sampling, in the large pixel (or L-resolution) case, data other than mean and variance are usually limited to the local variance, or variogram at lag one pixel. This is because the sill of the variogram is at a lag (D_1+D_2) which is less than two pixels when $D_1<D_2$ and the geometric pixel size is close to D_2. As can be seen from the discussion above, there are some advantages to having oversampled scenes when the IFOV is large and to using variogram data in the vertical and horizontal directions separately from data for the diagonal directions.

3. Application to the Disk Model

3.1 Inverse Problem Formulation

The inversion theory used in Invdisk presented here is based on the outline in Jupp and Vozoff (1975) with some modification of notation. That is, suppose there exist M data values y_i for $i=1,M$ which will be represented by the M by 1 vector:

$$\mathbf{y} = [y_1 \quad y_2 \quad y_3 \quad \ldots \quad y_M]^T$$

and suppose there is a model for the data **f**(**x**, **θ**) which is a function of specified parameters **x** and a vector of *N* model parameters $\boldsymbol{\theta}_i$ for i=1,N. For any choice of $\boldsymbol{\theta}$ the residuals:

$$e_i = y_i - f(\mathbf{x}_i, \mathbf{q})$$

or, in vector notation:

$$\mathbf{e} = \mathbf{y} - \mathbf{f}(\theta)$$

represent the misfit between model and data.

An iterative non-linear Least Squares inversion method takes a starting estimate θ_0 and iteratively finds the closest final estimate θ^* which minimises the weighted sum of squared residuals:

$$\begin{aligned} F(\theta) &= \sum_{i=1}^{M} w_i \; e_i^2 \\ &= \|\mathbf{e}\|_W \\ &= \mathbf{e}^T \mathbf{W} \mathbf{e} \end{aligned}$$

where W is a non-negative symmetric weight matrix (usually diagonal) which in certain cases can be identified with an estimate for the inverse of the covariance matrix of the random residuals.

The numerical method used here to apply this formulation is described in Appendix 1 together with the way its outputs may be used to analyse the parameter sensitivity and importance.

3.2 Invdisk Formulation

The information outlined in Section 2 may be used to define the model data (*f*) and Jacobian matrix (*J*) for the Disk Model. There will, in general, be (N_v*+2*) pieces of data available to be modelled, consisting of the graylevel mean (*G*), N_v values for the variogram (V_i for i=1,N_v) at specified lags (h_i for i=1,N_v) and the global variance (*S*).

The corresponding model data are:

$$\begin{aligned} f(x_1, \theta) &= g_B \; q + g_D \; (1-q) \\ f(x_i, \theta) &= (g_D - g_B)^2 \; \gamma_Z(h_{i-1}) \quad for \; i = 2, N_v + 1 \\ f(x_{N_v+2}, \theta) &= (g_D - g_B)^2 \; \sigma_Z^2 \end{aligned}$$

with θ as the vector of four parameters:

$$\theta = \left(g_D, g_B, \lambda, A_c \right)$$

and the x_i simply represent the fixed parameters (such as lag) indexing or defining the different modelled data components.

The Least Squares objective function for this formulation is:

$$\begin{aligned} F(\theta) &= w_1 (G - f(x_1, \theta))^2 + \sum_{i=2}^{N_v+1} w_i (V_{i-1} - f(x_i, \theta))^2 + w_{N_v+2} (S - f(x_{N_v+2}, \theta))^2 \\ &= \sum_{i=1}^{N_v+2} w_i \left| e_i \right|^2 \\ &= \left\| e \right\|_W^2 \end{aligned}$$

where, if N_v is zero, only data mean and variance are being used.

The inversion program is based on the Gauss method and the outputs can be analysed by the sensitivity analysis outlined in the Appendix. It is therefore possible to proceed once the Jacobian matrix (J) has been computed. The partial derivatives required are derived as follows.

3.3 *Partial derivatives of* f(xi, θ) *wrt* gD

The partial derivatives of $f(x_i, \theta)$ with respect to g_D are:

$$\begin{aligned} \frac{\partial f(x_1, \theta)}{\partial g_D} &= (1 - q) \\ \frac{\partial f(x_i, \theta)}{\partial g_D} &= 2\,(g_D - g_B)\,\gamma_Z(h_{i-1}) \qquad i = 2, N_v + 1 \\ \frac{\partial f(x_{N_v+2}, \theta)}{\partial g_D} &= 2\,(g_D - g_B)\,\sigma_Z^2 \end{aligned}$$

3.4 *Partial derivatives of* f(xi, θ) *wrt* gB

The partial derivatives of $f(x_i, \theta)$ with respect to g_B are:

$$\frac{\partial f(x_1, \theta)}{\partial g_B} = q$$

$$\frac{\partial f(x_i, \theta)}{\partial g_B} = -2\,(g_D - g_B)\,\gamma_Z(h_{i-1}) \qquad i = 2, N_v + 1$$

$$\frac{\partial f(x_{N_v+2}, \theta)}{\partial g_B} = -2\,(g_D - g_B)\,\sigma_Z^2$$

3.5 Partial derivatives of f(x_i, θ) *wrt* λ

The partial derivatives of $f(x_i, \theta)$ with respect to λ are:

$$\frac{\partial f(x_1, \theta)}{\partial \lambda} = (g_D - g_B)\,A_c\,q$$

$$\frac{\partial f(x_i, \theta)}{\partial \lambda} = (g_D - g_B)^2\,\frac{\partial \gamma_Z(h_{i-1})}{\partial \lambda} \qquad i = 2, N_v + 1$$

$$\frac{\partial f(x_{N_v+2}, \theta)}{\partial \lambda} = (g_D - g_B)^2\,\frac{\partial \sigma_Z^2}{\partial \lambda}$$

3.6 Partial derivatives of f(x_i, θ) *with respect to* A_c

The partial derivatives of $f(x_i, \theta)$ with respect to A_c are:

$$\frac{\partial f(x_1, \theta)}{\partial A_c} = (g_D - g_B)\,A_c\,q$$

$$\frac{\partial f(x_i, \theta)}{\partial A_c} = (g_D - g_B)^2\,\frac{\partial \gamma_Z(h_{i-1})}{\partial A_c} \qquad i = 2, N_v + 1$$

$$\frac{\partial f(x_{N_v+2}, \theta)}{\partial A_c} = (g_D - g_B)^2\,\frac{\partial \sigma_Z^2}{\partial A_c}$$

3.7 Putting all the partial derivatives together

From Section 2, it follows that:

$$\frac{\partial \sigma_z^2}{\partial \theta_j} = 8\int_0^1 tT(t)\,\frac{\partial Cov(D_2 t)}{\partial \theta_j}\,dt$$

and

$$\frac{\partial \gamma_Z(h)}{\partial \theta_j} = \frac{\partial \sigma_Z^2}{\partial \theta_j} - \frac{\partial Cov_Z(h)}{\partial \theta_j}$$

where

$$\frac{\partial Cov_Z(h)}{\partial \theta_j} = \frac{8}{\pi}\int_0^s t\Phi(t,s)\frac{\partial Cov(D_2 t)}{\partial \theta_j}dt$$

Hence, all that is needed to complete the partial derivatives is:

$$\frac{\partial Cov(h)}{\partial \theta_j}$$

where θ_j is λ or A_c. Since the same numerical integration can be used for the partials as for the base model data, only the integrand will change.

That is, from Section 2, if $h \le D_1$ (i.e. $s \le 1$):

$$\frac{\partial Cov(h)}{\partial \lambda} = q^2 A_c \left(2-(2-T(s))\, e^{\lambda A_c T(s)}\right)$$

$$\frac{\partial Cov(h)}{\partial A_c} = q^2 \lambda \left(2-(2-T(s)-A_c\frac{\partial T(s)}{\partial A_c})e^{\lambda A_c T(s)}\right)$$

where, $s = h / D_1$ for $h \le D_1$ and when $h > D_1$ (or, alternatively, $s > 1$), each partial is zero.

Finally, the remaining expression is:

$$A_c \frac{\partial T(s)}{\partial A_c} = \frac{2}{\pi} s(1-s^2)^{1/2}$$

and the partial derivatives and J are, in principal, computed.

4. Running Invdisk

This theory has been implemented in a fortran program Invdisk. Invdisk allows you to create exact model data by entering a model and computing the selected mean, variance and variograms as well as entering or reading existing data for inversion. The parameters are entered as disk and background gray level and disk Diameter (D_1) and the percent cover (*Cover*) - even though the program itself uses λ and A_c in the inversion. Disk size and percent cover seem to be more intuitive ways to define the model. That is:

$$\lambda A_c = -Log(1-Cover/100)$$

and

$$A_c = \frac{\pi}{4} D_1^2$$

define λ and A_c from D_1 and *Cover*.

As well as this, Invdisk has some special features which need more detailed explanation.

4.1 Parameter and Data Transformation

To help scale the problem and build important constraints into the solution, both the parameters and data have been scaled. The parameters have been scaled to new parameters such that:

$$\begin{aligned} \theta_1 &= Log\left|G - g_D\right| \\ \theta_2 &= Log\left|g_B - G\right| \\ \theta_3 &= Log(\lambda) \\ \theta_4 &= Log(A_c) \end{aligned}$$

which enforces EITHER $g_B > G > g_D$ OR $g_D > G > g_B$ and therefore assumes knowledge of at least which of disk or background is brighter and assumes no error in G for the purpose of the transformation.

These transformations are easily handled without the user needing to know and partial derivatives are easily derived as:

$$\frac{\partial F}{\partial \theta_1} = (g_D - G)\frac{\partial F}{\partial g_D}$$

$$\frac{\partial F}{\partial \theta_2} = (g_B - G)\frac{\partial F}{\partial g_B}$$

$$\frac{\partial F}{\partial \theta_3} = \lambda \frac{\partial F}{\partial \lambda}$$

$$\frac{\partial F}{\partial \theta_4} = A_c \frac{\partial F}{\partial A_c}$$

The transformations need to be taken into account when the eigenvectors of the Jacobian matrix are being interpreted. Linear combinations of Logarithms become products of original parameters – which in this case is very convenient.

4.2 Li/Strahler Estimates for Starting Values

As pointed out before, the starting values given to the program are very significant and should be well chosen for a useful result. The program, for example, will find the "closest" local minimum to the starting value but cannot guarantee a global minimum. Moreover, the stabilisation will leave the very poorly resolved ("unimportant") components at their starting values. Since the "unimportant" parameters are different for different models and depend on the starting values it is best to get the starting values as close as possible.

If the values for the disk and background graylevels were known, then estimates for the disk size and spacing based only on the data mean and variance are available from the Li/Strahler formula. Even if the colours of the disk and background are only approximately known, or the colour of one and some idea of total cover is known, this provides a good method for starting the inversion.

If g_D and g_B are known, then the data mean (G with $g_D \le G \le g_B$) can be used to estimate the cover fraction, and therefore q:

$$q = \frac{G - g_D}{g_B - g_D}$$

In the Li/Strahler terminology, q is K_G or the expected fraction of background in a pixel. If the actual background fraction in a given pixel is denoted "lower case" k_G then the variance of k_G is simply:

$$\begin{aligned} V(k_G) &= S/(g_D - g_B)^2 \\ &= \sigma_Z^2 \end{aligned}$$

The Li/Strahler estimate, however, is not in terms of k_G but in terms of a parameter m which is related to the total area of tree, or disk, associated with a pixel. It is convenient here to use πm (which will be denoted *b*) rather than *m* - but the results are equivalent. That is, define *b* as:

$$b = \frac{1}{A_p} \sum_{j=1}^{n_c} A_{c,j}$$

where A_p is the area of a pixel and A_c is the area of a disk. The sum is taken over all disks (or in general, all objects) whose centres fall in the pixel. In a specific case, there will be n_c such disks and the j'th one will have area $A_{c,j}$. The change of notation to "b" is to point out the link this has with basal area in the case where the disks are cross-sections of tree trunks at Breast Height.

If the disk centres have a Poisson distribution, it follows that the expectation (mean) of b (denoted B) is:

$$\begin{aligned} B &= N_c A_c / A_p \\ &= \lambda A_c \end{aligned}$$

where λ and A_c are as before and N_c is the expected number of centres which fall into a pixel of size A_p. For disks of constant size, the variance of b is:

$$\begin{aligned} V(b) &= N_c \frac{A_c^2}{A_p^2} \\ &= \frac{B^2}{N_c} \\ &= B \frac{A_c}{A_p} \end{aligned}$$

which leads to the expressions for A_c and λ of:

$$\begin{aligned} A_c &= \frac{V(b)}{B} A_p \\ \lambda &= \frac{B}{A_c} \end{aligned}$$

This expression for A_c is available if B and $V(b)$ are known, and this has been used to good effect in low density situations by Strahler, Li and Franklin in various papers by relating b and k_G.

The issue, however, is that the data we have are not of the random variable "b" but of the background proportion, k_G. Much of the effort put towards the Li-Strahler model has been put towards estimating b (or "m") from the image data. For the simple disk model there are two cases that are considered. These are the low density case where disks are assumed not to overlap and the high density case where disks are often overlapping.

The low density case was used in early papers describing the Li/Strahler inversion (eg see Strahler and Li, 1981). In this case, it is assumed that λ is small enough so that overlap between disks is negligible. In addition, it is assumed that areas of disks with centres outside the pixel, but intersecting it, are matched (as was done in Li & Strahler, 1985) by including into the pixel summation the complete areas of those disks with centres in the pixel but which intersect the boundary. (Note that this becomes invalid when the pixel is small with respect to the disk size).

Then, it follows (simply) that:

$$k_G = 1 - b$$

from which B and $V(b)$ can be immediately inferred if K_G and $V(k_G)$ are known.

Invdisk computes the low density estimate for A_c and (by implication) λ when estimates for g_D and g_B are provided. However, there are significant limitations to this estimate in both the higher density case and when the pixel size is small. It is a big pixel and low density model.

For the high density case, a second estimate is available. That is, in every case (high or low density, small or large pixel) we know that (exactly):

$$
\begin{aligned}
B &= \lambda A_c \\
&= -Log\ q \\
&= -Log\ K_G
\end{aligned}
$$

but the estimate for V(b) needs care.

The problem is that b and k_G cannot actually be identified except statistically. That is, in general there can be many k_G values corresponding to a single b value and vice versa. One way to overcome the problem is to associate an effective background fraction (k'_G) with every realisation of b such that:

$$
k'_G = e^{-b}
$$

and then use the first order approximation to obtain a variance estimate:

$$
V(k'_G) \approx K_G^2 V(b)
$$

so that, identifying the variances of k_G and k'_G we get:

$$
V(b) \approx \frac{V(k_G)}{K_G^2}
$$

This “second estimate” for the variance is used with the exact value for B in Invdisk to estimate an initial A_c and λ from the supplied g_D and g_B. The results are very good in the large pixel case over most densities - but are poor in the H-resolution case. The problem in the H-resolution case is understandable when you consider (for example) that in this case there will be many situations when there will be no disk centres in a pixel but it will be totally covered by disk. The relationship between k_G and b will be much more complex in that situation. Analysing these limits is another opportunity Invdisk offers.

[recent simulations of the disk model, including the case where disk diameters are lognormally distributed, have shown that this approach works very well.]

4.3 Choice of parameters for inversion

The complete inversion problem cannot be solved in the L-resolution case using only these data. When the pixel size is large then ONLY the mean, variance and possibly the local variance are available as independent data. In this case, four parameters cannot be estimated. For this case, and to stabilize other cases, Invdisk allows three choices of parameter combination for inversion.

- Parameter combination Type 0 corresponds to all four parameters allowed to vary.
- Parameter combination Type 1 allows only λ and A_c to vary and
- Parameter combination Type 2 allows λ, A_c and g_B to vary.

The Type 1 selection allows efficient inversion when the graylevels for the disk and background are known. If only mean and variance are used, ($N_v=0$), the resulting iteration is very fast and leads to the fully optimised solution of the original problem. However, it works for every case - L- and H-resolution, low and high density of disks. In the presence of noise, Invdisk adds the possibility of using mean, variance AND local variance for this solution.

The Type 2 selection allows inversion in cases where the colour of the disks is known, but not that of the background. The possibility of this case when the local variance is known in addition to the mean and global variance is very interesting and may be quite feasible with SPOT or TM data in real data situations.

Finally, in the H-resolution case, where variograms at various steps are available up to the sill, the Type 0 selection allows complete inversion. However, the stability of the solution needs to be investigated if the procedure is to be used with real (noisy) data such as SPOT panchromatic or scanner data.

5. Example outputs & discussion

An outline of how Invdisk may be set up and run is provided in Appendix 2. Outputs from Invdisk need to be saved in a file and plotted in a separate package – such as a spreadsheet. Some initial runs have been made as simple illustrations of the Invdisk outputs for a model where the disk diameter is 10 metres and the disk and background graylevels are 17 and 28 respectively. Runs were made for IFOV diameters of 2.5, 5, 10, 20, 40 and 80 metres and for cover fractions (of disk) of 10, 35, 50, 65 and 90 percent. These covers correspond to the mid-ranges of the Open Woodland, Woodland, Open Forest and Forest categories as used in the example in ARDI-II.

5.1 General Output Statistics

Figures 2.1 to 2.5 plot the analytical variograms for the models as computed by Invdisk with each cover (10%, 35%, 50%, 65% and 90%) as a separate plot. Figures 3.1 to 3.6 plot the variograms with each pixel size (2.5m, 5m, 10m, 20m, 40m and 80m) as a separate plot. Clearly, the range of image covariance increases with pixel size but overall variance decreases. The underlying image variance is maximum for a cover of 50%.

Figure 4 plots the image variance as a function of pixel size (scaled by disk size) for each of the selected covers. There is a general trend of reducing variance as a function of pixel size and maximum overall variance at 50%. In these runs, disk size has not been varied. For a given cover and pixel size it is found that image variance increases as the disk size increases.

Figure 5 plots variance as a function of cover with separate plots for each of the pixel sizes. For a given pixel size a similar result is obtained with the "boomerangs" indexed by disk size. It has been proposed that plots of local variance against colour (related to cover) can characterise areas of different crown size – but this has not been well established at this time.

5.2 Li-Strahler Variance Model Results

The results for the Li/Strahler estimate were very interesting. They are presented as two sets of two graphs shown in Figures 6.1 and 6.2 and 7.1 and 7.2. The first two graphs plot the estimated disk size (for exactly known values of disk and background gray levels of D_1=10 metres, gD=17 and gB=28) based on the low density approximation, or LS_1 (Figure 6.1) and overlapping approximation, or LS_2 (Figure 6.2) as a function of pixel size and with plots indexed by cover.

For an IFOV of 80 metres and low density (10% cover) the LS_1 estimate is close - but that is the only time it is. The worst case is high density and small IFOV (i.e. the H-resolution case). Figure 6.2 plots the second estimate in the same way. Here, the results are better but varied. The best results are when the pixel size is more than 20 metres and covers are around 50%. For high covers, for example, the 80 metre IFOV case significantly OVER-estimates D_1. It would seem in this case that SPOT MSS and Landsat TM resolutions are as fine as you should use. At least, this may be proposed for covers where tree plus shadow forms a patch of the same area as a disk with diameter about 10 metres.

Figures 7.1 and 7.2 display the same data (LS_1 and LS_2) plotted against cover and with plots indexed by pixel size. The poor performance of the LS_1 estimate is clear as is the need for pixel sizes to be greater than 20m and covers to be about 50% for good performance of the LS_2 estimate for this disk model. Invdisk can easily be used to widen this discussion to include a range of disk sizes. That is left to the user.

It is clear that in this simplest case there are messages for the use of the Li-Strahler inversion model. It is a relatively large pixel and mid-range density model. However, simulations also show that it is sensitive to knowledge of the object and background colours and to variance that does not arise from the geometric model (such as varying spectral signatures and a complex background).

5.3 Parameter Sensitivity for Inversion

For IFOV diameters other than 2.5 and 5 only three data values were computed - mean, local variance and global variance - so that only inversions of Type 1 and 2 made sense. For the IFOV of 5, four values were computed with two steps of the variogram available before the sill was reached. There were no actual inversions made for this test but rather the singular values were computed, the V matrix inspected and the Li/Strahler estimates checked as illustrated in the program listing of Appendix 2. Numerical simulations based on these data with various forms of added disturbance have been made for reporting elsewhere.

The singular values for the initial runs generally showed that when the colours of the disk and background are known (Type 1) then the problem is very well posed and stable - even in the L-resolution case, except possibly for very sparse (10% cover) or dense (90% cover) cases. Moreover, in the H-resolution case, the complete inversion is possible using the additional local variance and variogram data (Type 0). When a good estimate for disk (i.e. tree and shadow) colour is available, and the pixel is not too large (e.g. Landsat TM data) it is possible to invert size, spacing and background colour using local variance data (Type 2) as added information. However, in these last two cases the data must be very accurate since the fourth (Type 0) and third (Type 2) singular values were always very small.

The cover should also be in the range of 20 to 80 percent for good results. In all cases the problem was better posed for the covers of 35% and 65% than for 10% and 90%. This might be expected intuitively but is worth supporting by a numerical experiment.

5.4 Practical Implementation

The main problem in practice is data noise, sampling variance and other perturbing effects. One effect worth considering is that of larger scales of pattern on the global variance. It may be possible to use the variogram at lag 2 pixels rather than global variance to avoid these effects. Again, variogram data may need to be estimated robustly, and data for the vertical and horizontal as well as diagonals separated as data. Also, it may be useful to compute the covariogram and use a form of PCA that maximises spatial rather than spectral variance to choose the axes for computing k_G. Quite a lot of attention to field data, data processing and image statistics is needed at this stage to get the best out of the inversions.

Invdisk may also be made more "realistic" and better posed in a number of ways. These developments are ongoing and described briefly in Appendix 3.

6. References

Bard, Y. (1974). Non-Linear parameter estimation. Academic Press, New York.

Franklin, J. and Strahler, A.H. (1988). Invertible canopy reflectance modeling of vegetation structure in semi-arid woodland. *IEEE Trans. Geosci. Remote Sensing*, 26(6):809-825.

Jupp, D.L.B. and Vozoff, K. (1975). stable iterative methods for the inversion of geophysical data. *Geophysical Journal of the Royal astronomical Society*, 42:957-976.

Jupp, D.L.B., Strahler, A.H., and Woodcock, C.E. (1988). Autocorrelation and regularization in digital images. I. Basic theory. *IEEE Trans. on Geoscience & Remote Sensing*, 26, 463-7

Jupp, D.L.B., Strahler, A.H., and Woodcock, C.E. (1989). Autocorrelation and regularization in digital images. II. Simple image models. *IEEE Trans. on Geoscience & Remote Sensing*, 27, 247-258.

Kowalik, J. and Osborne, M.R. (1968). *Methods for unconstrained optimization problems*. American Elsivier, New York.

Li, X. and Strahler, A.H. (1985). Geometric-optical modeling of a conifer forest canopy. *IEEE Trans. on Geosci. Remote Sensing*, GE23(5):705-721.

Li, X. and Strahler, A.H. (1986). Geometric-optical bidirectional reflectance modeling of a coniferous forest canopy. *IEEE Trans. on Geosci. Remote Sensing*, GE24(6):906-919.

Strahler, A.H. and Li, X. (1981). An invertible coniferous forest canopy reflectance model. In *Proceedings of the 15th International Symposium on Remote Sensing of Environment,* Ann Arbor, MI, 1237-1244.

Strahler, A.H., Woodcock, C.E. and Smith, J.A. (1986). On the nature of models in remote sensing. *Remote Sensing of Environment*, **20**, 122-139.

Strahler, A.H., Wu, Y. and Franklin, J. (1988). Remote estimation of tree size and density from satellite imagery by inversion of a geometric-optical canopy model. Proceedings 22nd Int. Symp. on Remote Sensing of Environment, Abidjan, October 20-26, 1988.

Strahler, A.H. and Jupp, D.L.B. (1991). Modeling bidirectional reflectance of forests and woodlands using boolean models and geometric optics. *Remote Sensing of Environment*, **34**, 153-166.

7. Appendix 1 - Inversion and Sensitivity Analysis

7.1 Inversion Method

The particular non-linear inversion method used here is a modification of the Gauss Method (Kowalick and Osborne, 1968). It has the efficiency of the Newton Method when the current parameter estimate is near the solution but only uses first derivatives of the objective function. In addition, stability during the iterations is increased by modifications similar to those of the Marquardt method. The derivation runs briefly as follows.

Consider a current estimate for the parameters θ_k and a perturbation of it of size $\delta\theta$. Then the solution we seek is the minimum of:

$$
\begin{aligned}
F(\delta\theta) &= \|\mathbf{e}(\theta_k + \delta\theta)\|_W \\
&= \|\mathbf{y} - \mathbf{f}(\theta_k + \delta\theta)\|_W
\end{aligned}
$$

as a function of $\delta\theta$.

Now consider the Taylor expansion of **f** about θ_k:

$$\mathbf{f}(\theta_k + \delta\theta) = \mathbf{f}(\theta_k) + \mathbf{J}\ \delta\theta + \mathbf{R}(f, \delta\theta)$$

where $\mathbf{R}(\mathbf{f}, \theta_k)$ is the remainder, which under reasonable conditions can be considered $O(\|\delta\theta\|^2)$, and **J** is the *M* by *N* Jacobian matrix:

$$J = \left[\frac{\partial f_i}{\partial \theta_j}\right]$$

If $\mathbf{R}(\mathbf{f}, \theta_k)$ can be neglected, the problem could be replaced by the Linear Least Squares problem to minimize:

$$F(\delta\theta) = \|\mathbf{e}_k - \mathbf{J}\delta\theta\|_W$$

as a function of $\delta\theta$ which has a well known solution (as described in more detail in Jupp and Vozoff, 1975):

$$\delta\theta = \mathbf{J}^{+}\mathbf{e}_k$$

where $\mathbf{J}^+$ is the "generalized inverse" of **J**.

When **J** has rank *N* this solution has the familiar Linear Least Squares form:

$$\delta\theta = \left(\mathbf{J}^T \mathbf{W}\mathbf{J}\right)^{-1} \mathbf{J}^T W \mathbf{e}_k$$

and will solve the problem exactly when **f** is a linear function.

For a nonlinear problem, not only will $\|\mathbf{R}\|$ be significant for many values of θ_k but also **J** will be different at ($\theta_k + \delta\theta$). The Gauss method therefore uses $\delta\theta$ as a correction step and iterates until the actual solution is achieved.

There are two things, in particuler, which make the method especially useful and convenient. They both follow by noting that the gradient of *F* at θ is:

$$\nabla F(\theta) = -2\ J^T W \mathbf{e}$$

and that the solution is characterised by the condition that $\nabla F = 0$.

The first thing is that when the solution to the linear problem is reached, the step is zero and $\nabla \mathrm{F}(\theta)=0$ to the same degree, since:

$$\delta\theta = -\frac{1}{2}\,(\mathrm{J}^{\mathrm{T}}\mathrm{WJ})^{-1}\ \nabla \mathrm{F}(\theta)$$

That is, the nonlinear problem solution is characterised by that of the linearised problem.

The second thing is that the direction of the vector $\delta\theta$ is "downhill" at θ_k. That is:

$$\begin{aligned}\nabla F^T \delta\theta &= -2\ e^T J^+ e \\ &= -\frac{1}{2} \nabla F^T \left(J^T W J\right)^{-1} \nabla F\end{aligned}$$

which is ALWAYS ≤0 as JJ^+ is a non-negative definite symmetric matrix.

The implication of this is that even if:

$$F(\theta_k + \delta\theta) > F(\theta_k)$$

there will be a value in the same direction but with size between zero and that of $\delta\theta$ which will reduce *F*. That is, *F* can always be reduced by a "line search" in the direction $\delta\theta$. If this step to a point between zero and the complete Gauss step is denoted $\delta\theta_k$ then the sequence of solutions generated from θ_0 can form a "minimizing sequence" for F and converge to a local minimum, θ^*.

How this "line-search" may be done most efficiently using a combination of Marquardt type methods and the Goldstein condition is described in Jupp and Vozoff (1975) and Kowalick and Osborne (1968). The result is that if a starting estimate, θ_0, is provided as well as a function subroutine providing F, $\boldsymbol{e}$ and J when given a current value for θ_k, a stable minimizing sequence of estimates can be generated which converge to the "closest" local minimum of F to θ_0. These methods have been implemented in a fortran routine called NLLSQ which is used in Invdisk to invert the disk model.

7.2 Parameter Sensitivity - Analysis of the Jacobian

In Jupp and Vozoff (1975), an analysis of the Jacobian matrix is used to study the stability of both the iterative method for finding a solution to the Least Squares problem as well as the final model as a function of data perturbations. The perturbations were purposely not modelled using a statistical model but rather in terms of quadratic "norms" such as:

$$\begin{aligned}\|\delta e\|_W &= \sum_{i=1}^{M}\sum_{j=1}^{M} w_{ij}\ \delta e_i \delta e_j \\ &= \delta e^T W \delta e\end{aligned}$$

for perturbations in data space and

$$\begin{aligned}\|\delta\theta\|_H &= \sum_{i=1}^{N}\sum_{j=1}^{N} h_{ij}\ \delta\theta_i \delta\theta_j \\ &= \delta\theta^T H \delta\theta\end{aligned}$$

for perturbations to parameters.

The perturbations could easily be modelled as statistical. However, there are advantages to considering a more general model in non-linear inversion. First, near the starting model and often even at the "solution" when a generalised model is being fit to data, the component in the residual due to model inadequacy may be much larger than that ascribable to statistical "error" or "noise". Analysing the effects of general, even systematic, perturbations is therefore much more realistic. At the solution, when the model is a "good" model for the data, the assumption of statistical error dominating the residuals may apply and can be easily accommodated along the lines described by Bard (1974).

Given that viewpoint, consider we are at some set of values of parameters and data (which may not be a local solution to the Least Squares problem) and consider the data perturbation (δe_1) defined by:

$$\delta e_1 = J\ \delta\theta_1$$

and the model perturbation ($\delta\theta_2$) defined by:

$$\delta\theta_2 = J^+ \ \delta e_2$$

for respective model and data perturbations $\delta\theta_1$ and δe_2.

The first case represents a linearised approximation to the way in which the model data (and hence the residuals) can change due to small changes ($\delta\theta_1$) in the model parameters. The second represents an approximation to the way in which the complete solution step would alter because of small changes (δe_2) in the data - for whatever reason.

In that the two problems are inverse to one another, it is not surprising that parameters or parameter combinations which produce very little effect in the model data when perturbed are also the ones which change most to small perturbations in the data under inversion. These are the unstable and "unimportant" parameters. In effect, the variance attributable to these parameters is easily dominated by other effects - such as random error and noise - and their inversion is very poorly posed. If a parameter or parameter combination produces NO change in the model data whatever its perturbation then it cannot be inverted and it is called "irrelevant".

A precise analysis of the parameters, and a characterisation of parameter relevance and importance, can be accomplished using an eigenvalue analysis of the Jacobian called the "Singular Value Decomposition" (or SVD). This is described in Jupp and Vozoff (1975) and is the description of J as the product:

$$\begin{aligned} J &= U \ S \ V^T \\ &= \sum_{j=1}^{N} s_j \ \mathbf{u}_j \mathbf{v}_j \end{aligned}$$

where U and V are orthonormal matrices such that U is M by N and:

$$U^T U = I_M$$

and V is N by N with:

$$V^T V = I_N$$

The matrix S is diagonal such that each element is the non-negative square root of the eigenvalues of $J^T J$ and ordered so that:

$$s_1 \geq s_2 \geq \ldots \geq s_N \geq 0$$

Assuming that M≥N, the matrix V is the matrix of eigenvectors for J^TJ and U is the matrix formed by the first N eigenvectors of JJ^T.

If the weight matrix $W=I$ (which can be achieved by suitable scaling) then the generalised inverse has a very useful expression in terms of the SVD:

$$J^+ = V\ S^+\ U^T$$

where, S^+ is the diagonal matrix with:

$$s_j^+ = \begin{cases} \dfrac{1}{s_j} & if \quad s_j > 0 \\ 0 & if \quad s_j = 0 \end{cases}$$

Therefore, returning to the previous perturbation equations and defining:

$$\delta p = V^T \delta\theta$$

as an orthogonal transformation of the parameters, and

$$\delta g = U^T \delta e$$

as an orthogonal transformation of the data, then the equations can be written:

$$\delta g_1 = S\ \delta p_1$$

or, in component form:

$$\delta g_{j,1} = s_j\ \delta p_{j,1}$$

and:

$$\delta p_2 = S^+\ \delta g_2$$

or, in component form:

$$\delta p_{j,2} = \begin{cases} \dfrac{\delta g_{2,j}}{s_j} & if \;\; s_j > 0 \\ 0 & if \;\; s_j = 0 \end{cases}$$

In terms of this decomposition into independent parameter and data components, the "irrelevant" parameters are those in the span of eigenvectors corresponding to zero singular values. That is, in the null space of *J*. The "unimportant" parameters are those corresponding to small singular values - or in the span of the eigenvectors corresponding to the small (but non-zero) singular values and the "important" parameters correspond to those in the linear span of the eigenvectors coorresponding to the large singular values.

The reason for this is that the model data are not very sensitive to parameters which are combinations of eigenvectors corresponding to small singular values and dually, small perturbations in the data - for instance, due to noise or error - will lead to large, often very large, perturbations of the parameters during inversion - due to the effect of the factors 1/sj. During non-linear iterations, where an inversion is made at each step, this can lead to quite erroneous solutions being reached with both stable and unstable parameters being well removed from their "true" values. A stable iterative method, therefore, damps the effect of irrelevant and unimportant parameters and even at a solution, it is these parameters which are least well supported by the data.

The problem is to decide at which point parameters become "unimportant". It is here that statistics can help a little. The area is rather heuristic and not fully satisfactory, but a rough rule used in geophysics depends on using a quantity called the "Noise-to-Signal-Ratio" (NSR) for the model fit to the data defined as:

$$NSR = \frac{100}{\sqrt{F}}$$

where *F* is the F-ratio for the best least squares fit by the model to the data and r is the number of non-zero singular values. Similar quantities are defined for the model:

$$NSR_r = \frac{100}{\sqrt{F_r}}$$

where

$$F_r = \frac{\sum_{j=1}^{r} s_j^2 \Big/ r}{\sum_{j=r+1}^{N} s_j^2 \Big/ N - r}$$

The "threshold" between the important and unimportant parameters is taken to be the point (r) where NSR_r falls below NSR. Intuitively, this corresponds to a point where the variance ascribable to the parameter combinations spanned by the eigenvectors corresponding to the singular values less than the threshold are within the noise level of the fit to the data.

This is heuristic and not rigorous in a statistical sense. However, it has been found to provide a useful analysis. For any division of the components defined by the SVD, it provides a measure of what level of accuracy may be needed to resolve the parameter combinations made up of the eigenvectors down to that level. All that is needed for this analysis is the V matrix and the singular values.

8. Appendix 2 - Running Invdisk

Invdisk is a Fortran program that runs in a DOS window. A typical run would take the following form:

Program Invdisk - Compute Disk Model Variograms

Invdisk Option (?=Menu;H=Help;Q=Quit) > h

[H provides some information about Invdisk]

Invdisk allows you to compute or enter data for a simple
disk model and invert the data
The program is useful if the disk is identified with tree and
shadow combined and background with sunlit background.

Data may be computed exactly, read in or entered manually.
Data are graylevel mean, variance and some variogram values
at specified lags (steps). Note that in the L-resolution case
the Sill is reached at (D1 + D2). If it is not, there may be
effects other than subpixel (tree) based variance

Pixel IFOV is taken as a disk with diameter D2. Geometric
pixel size is usually smaller and defines the variogram lags
If IFOV unknown use D2=1.128*Pixel geometric width.

Invdisk Option (?=Menu;H=Help;Q=Quit) > ?

[? Lists the menu of options]

Invdisk Options are :
C .... Compute disk model data
R .... Read in data from a file
E .... Enter data from keyboard
L .... List the data
I .... Run the inversion
O .... Change output device
H .... Help about Invdisk
Q .... Quit Invdisk
? .... List Menu

[If you want to try some cases use C to compute data then I to invert the computed data. For data generated in other ways use E to enter from the keyboard or R to read from a file. Here we will compute some data and invert it.]

Invdisk Option (?=Menu;H=Help;Q=Quit) > c

Enter Disk Diameter (D1) and COVER (%) > 10 50

Enter graylevel for Disk (Grd) and Background (Grb) > 17 28

"Entered Disk Model:"
"Diameter (D1) =" 10.0000
"Cover =" 50.0000 %
"Density of disks (dens) =" 0.0088
"Area of disk (Ac) =" 78.5398

"Disk graylevel (Grd) =" 17.0000
"Background (Grb) =" 28.0000
"Mean graylevel =" 22.5000

Model OK (Y/N) ? > y

Enter Diameter of pixel IFOV (D2) > 20
Supply Max lag (Hmax) and Number of lags (NH) > 80 4

[The variogram data are computed by supplying a maximum lag and the number to sample at assumed equally spaced steps. In this case, the pixel size and variogram step are kept the same, 20 metres]

"Covariance Data with pixel IFOV modelled as a disk"
"Effective IFOV Disk diameter =" 20.0000

Regularized Variogram and Derivatives

#	h	Gz(h)	d/dGrd	d/dGrb	d/dDens	d/dAc
0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	20.0000	4.3144	-0.7844	0.7844	-92.6702	0.0324
2	40.0000	4.5077	-0.8196	0.8196	-98.4419	0.0358
3	60.0000	4.5077	-0.8196	0.8196	-98.4419	0.0358
4	80.0000	4.5077	-0.8196	0.8196	-98.4419	0.0358

Mean = 22.5000
Derivatives = 0.5000 0.5000 -431.9690 -0.0485
Variance = 4.5077
Derivatives = -0.8196 0.8196 -98.4419 0.0358

[These statistics are from the program. If the data are saved to file using the O option the derivatives are not listed.]

"Strahler/Li estimates for Disk size and density"

	"Low Density"	"Second Estimate"
"Mean m (M) ="	0.1592	0.2206
"Variance V(m) ="	0.0038	0.0151
"Parameter Estimates"		
"Diameter (D1) ="	5.4592	9.2733
"Cover ="	50.0000	50.0000 %
"Density (Dens) ="	0.0296	0.0103
"Area (Ac) ="	23.4073	67.5391

[These are the LS_1 and LS_2 estimates that are described in the text.]

Invdisk Option (?=Menu;H=Help;Q=Quit) > i

Enter starting model
Enter graylevel for Disk (Grd) and Background (Grb) > 17 28

Strahler/Li estimates for Disk size and density
Low Density Second Estimate

Mean m (M) = 0.1592 0.2206
Variance V(m) = 0.0038 0.0151
Parameter Estimates
Diameter (D1) = 5.4592 9.2733
Cover = 50.0000 50.0000 %
Density (Dens) = 0.0296 0.0103
Area (Ac) = 23.4073 67.5391

Select disk diameter (D1) and % Cover
Options are Low Density Li/Strahler (L)
Second Estimate Li/Strahler (S)
or Enter your own estimates (E) > s

[You can enter starting values for the disk diameter or use the LS_1 or LS_2 estimates. Here, the LS_2 estimate is used]

"Initial Disk Model for inversion:"
"Diameter (D1) =" 9.2733
"Cover =" 50.0000 %
"Density of disks (dens) =" 0.0103
"Area of disk (Ac) =" 67.5391

"Disk graylevel (Grd) =" 17.0000
"Background (Grb) =" 28.0000
"Mean graylevel =" 22.5000

Model OK (Y/N) ? > y

Enter Parameter Selection Type (0/1/2/3) (?=Help) > 0

Itype = 0
All 4 parameters (Grd, Grb, Dens & Ac) free to vary

[You can try and invert all four parameters or selections as described in the text. Here we try all four.]

****
Inversion Output.
****

Nonlinear Least Squares
Second Order Marquardt with Singular Value Decomposition
Iteration Limit = 7

Initial Standard Error = 0.1597
Relative Singular Value Threshold (RSVT) = 0.010

Convergence on Predicted Decrease - 4 Iterations
Standard Error = 0.1349E-03

Initial Grd = 17.0000 Final = 17.2208 Actual = 17.0000
Initial Grb = 28.0000 Final = 28.2955 Actual = 28.0000
Initial D1 = 9.2733 Final = 10.0113 Actual = 10.0000
Initial Cover = 50.0000 Final = 52.3331 Actual = 50.0000 %

[The result is good but since the starting values are very close to the truth it is not a very surprising result. The parameter sensitivity is more significant in this run.]

Jacobian Matrix Singular Values
3.4414 0.3005 0.0239 0.0000

[The zero fourth SV indicates that there is a redundant parameter combination. The small third SV indicates there is a very poorly resolved parameter combination.]

Eigenvectors of the Jacobian matrix

0.6197	-0.3601	0.5127	-0.4727
0.6798	0.4607	0.0274	0.5700
-0.1769	-0.5910	0.4151	0.6687
0.3501	-0.5557	-0.7510	0.0677

[The loadings of the fourth column indicate the redundant parameter is a combination of the difference between disk and background colour and the disk size. That is, increasing disk size will reduce the variance but reducing contrast between the disk and background can compensate. The third Eigenvector probably expresses the cover/colour tradeoff.]

List Data and Errors ? > n

[There is no need to list the errors in this case]

Invdisk Option (?=Menu;H=Help;Q=Quit) > q
Invdisk finished

As mentioned in the run, the "O" option will allow listing to go to a file. The file output has a separate form from the data above but summarises the output ready to import into a spreadsheet. Invdisk runs can be set up and run using batch files and input scripts.

9. Appendix 3 – Current Modifications to Invdisk

In recent changes to Invdisk, new parameters have been made available. These are to define the inversion in terms of λA_c and *D*. The new partials are obtained from:

$$\frac{\partial F}{\partial \lambda A_c} = \frac{1}{A_c}\frac{\partial F}{\partial \lambda} + \frac{1}{\lambda}\frac{\partial F}{\partial A_c}$$
$$\frac{\partial F}{\partial D} = \frac{\pi D}{2}\frac{\partial F}{\partial A_c}$$

An alternative parameterisation provided is λA_c and $\pi\lambda D$. This seeks to emphasise the clearing size and cover as two parameters.

In the implementation, this leads to:

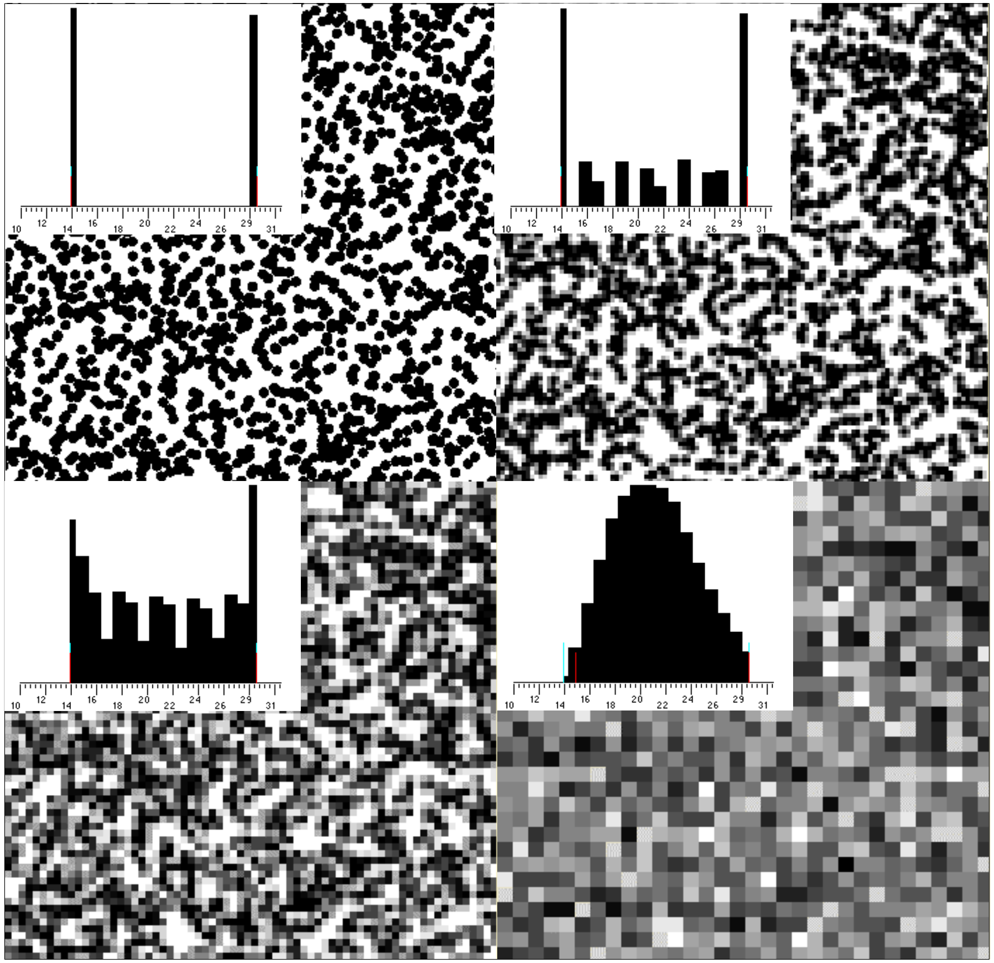
$$\theta_3 = Log(\lambda A_c)$$
$$\theta_4 = Log(D) \quad or \quad Log(\lambda D)$$

There is reason to believe that these will lead to better interpretations of the parameters. A study of this idea has not, however, been done at this time.

The recent version of Invdisk also allows for λ, A_c and g_D to vary so that cases where the objects are bright on a dark background can be modelled.

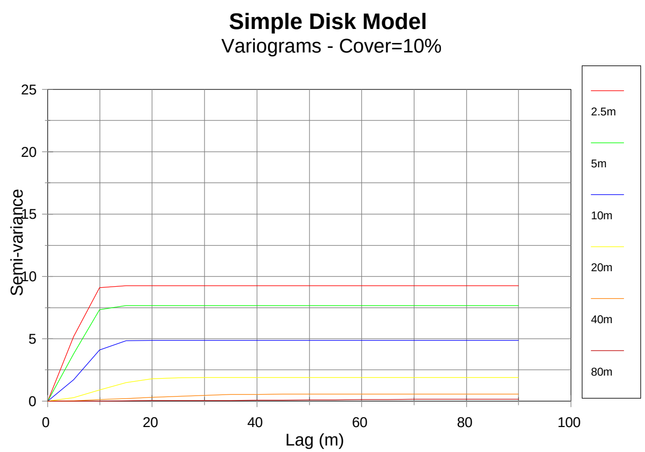
Finally, it is possible to include in Invdisk the cases of lognormal disk sizes and Neyman Type-A density. This has been done to increase the generality and scope of the investigations. Another possibility that has been discussed is to add a superimposed set of disks that are given the same colour as the background. These are intended to model a superimposed gap structure.

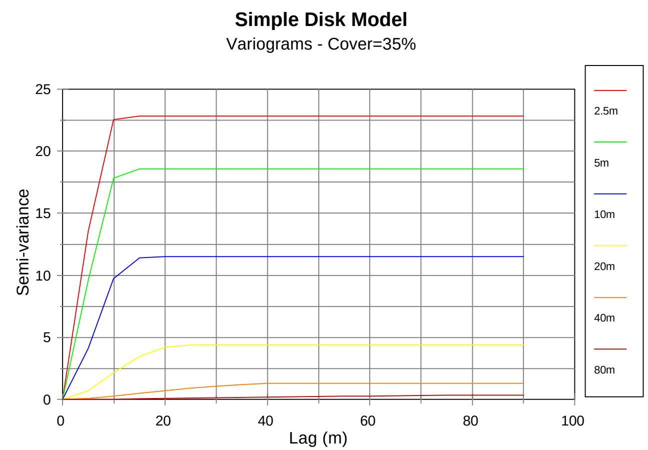
Figure 1: Effects of Size & Spacing on Cover Variance

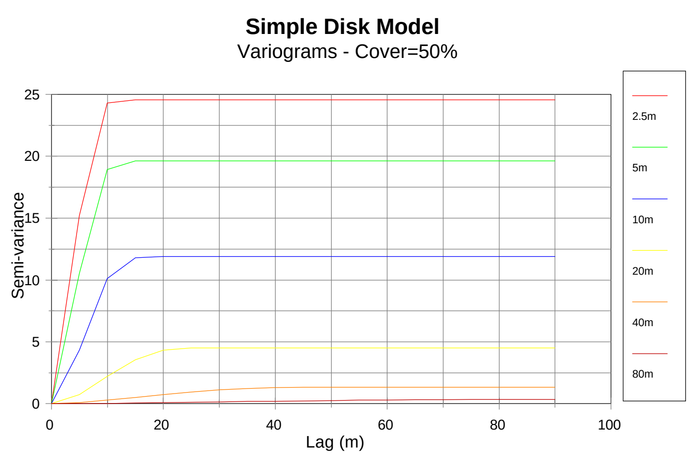


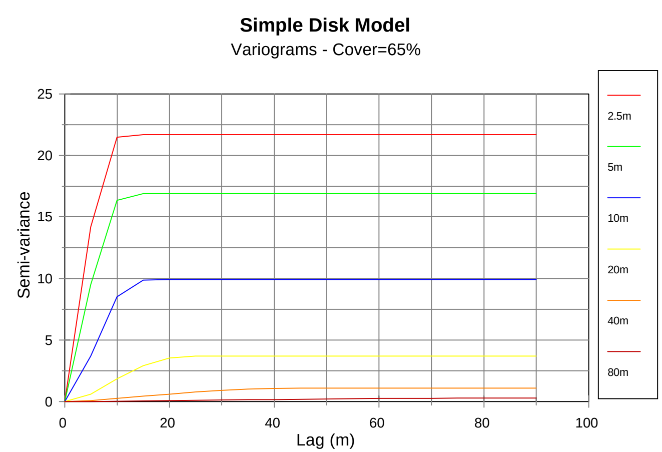
4_disks.cdr

Invdisk Figures









Figures 2.1 to 2.4: Variograms for 10 metres disks for varying cover (10% to 65%) between plots.

Invdisk Figures

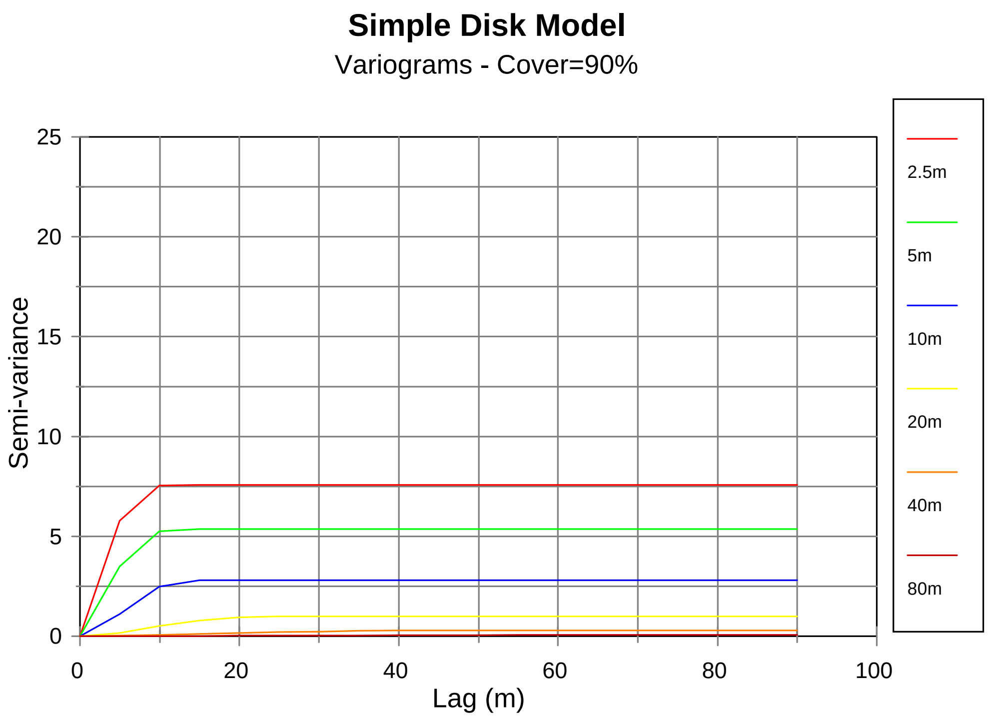
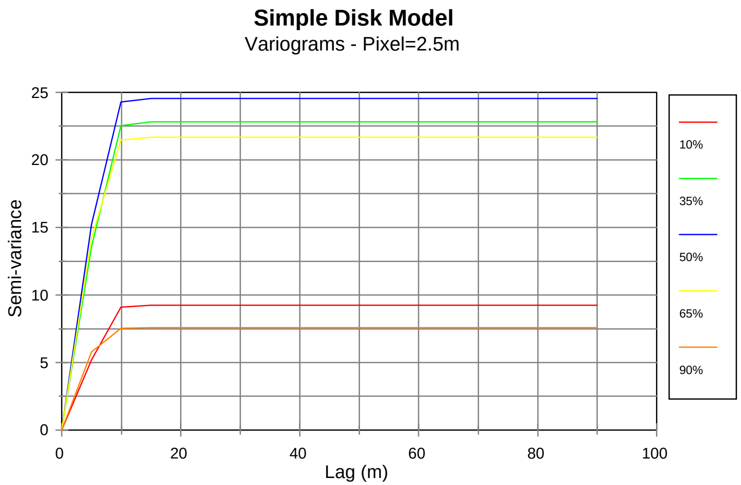
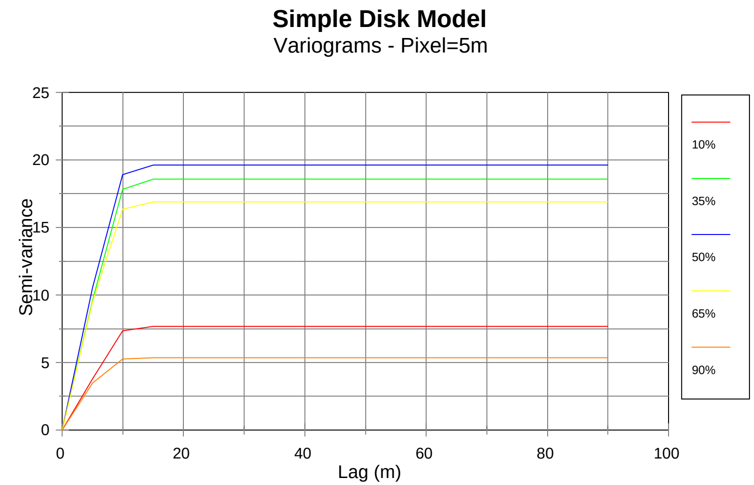
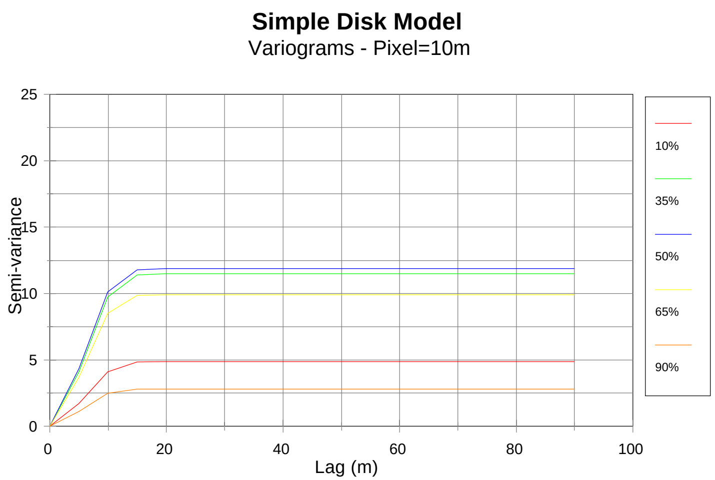
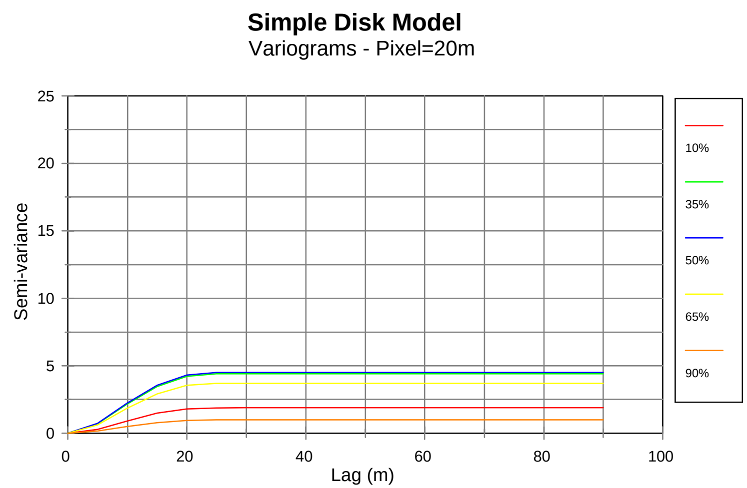


Figure 2.5- Variograms for 10 metres disks for 90% cover. (Cf. Figures 2.1 to 2.4).



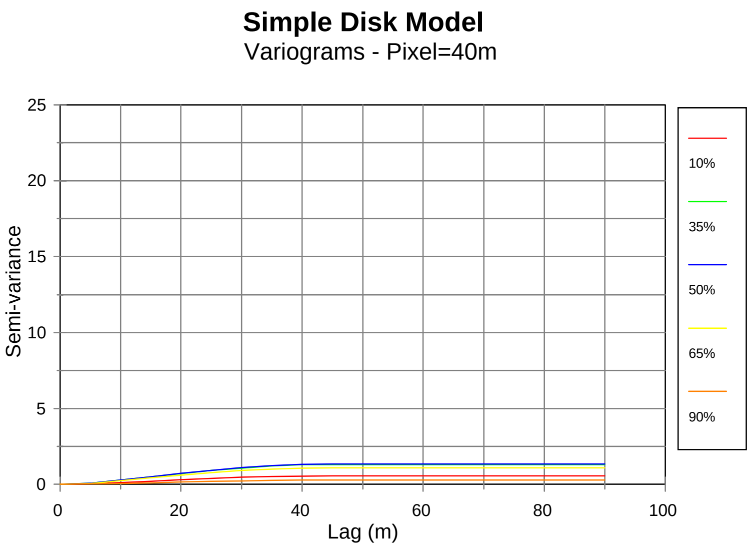


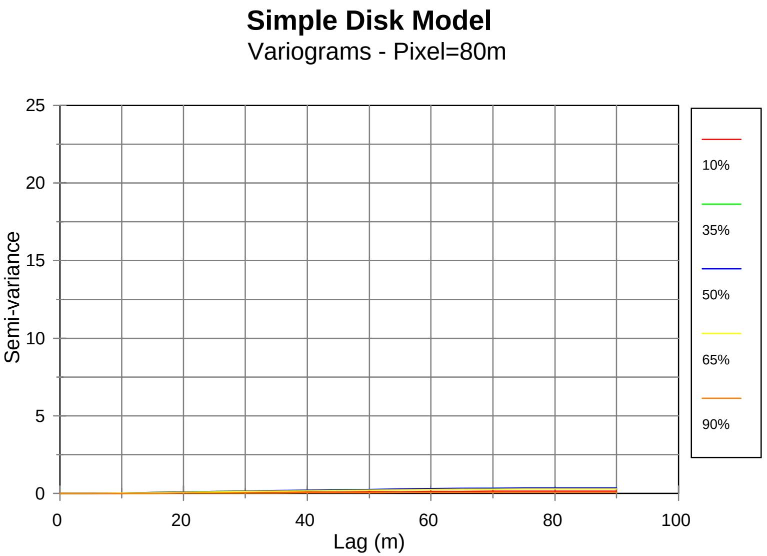




Figures 3.1 to 3.4: Variograms for 10 metre disks for varying pixel size (2.5-20 metres)

Invdisk Figures





Figures 3.5 to 3.6: Variograms for 10 metre disks for varying pixel size (40 and 80 metres)

Invdisk Figures

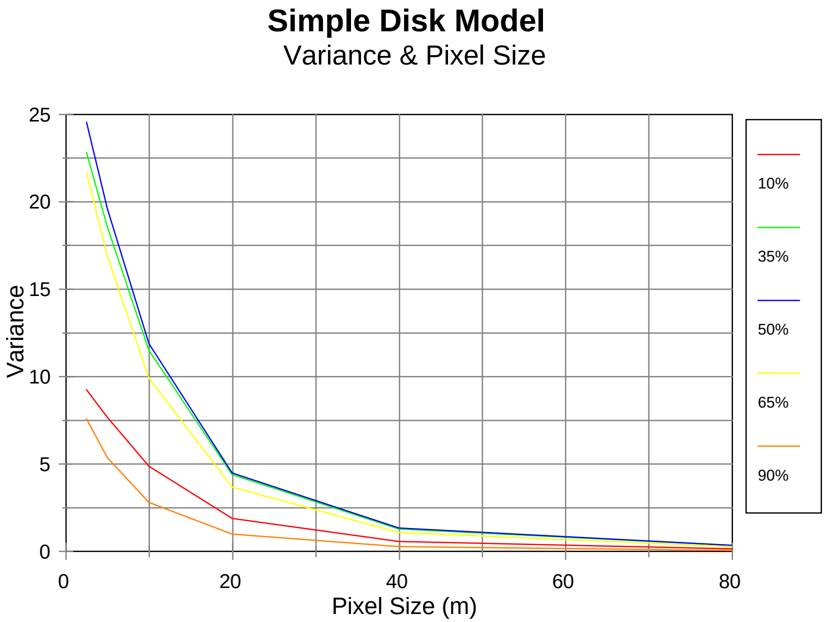


Figure 4: Variance as a function of pixel size

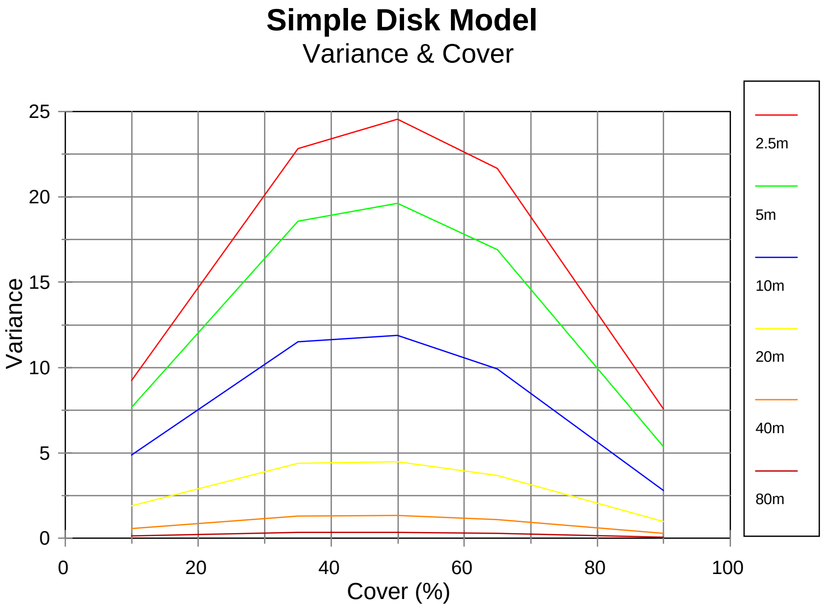
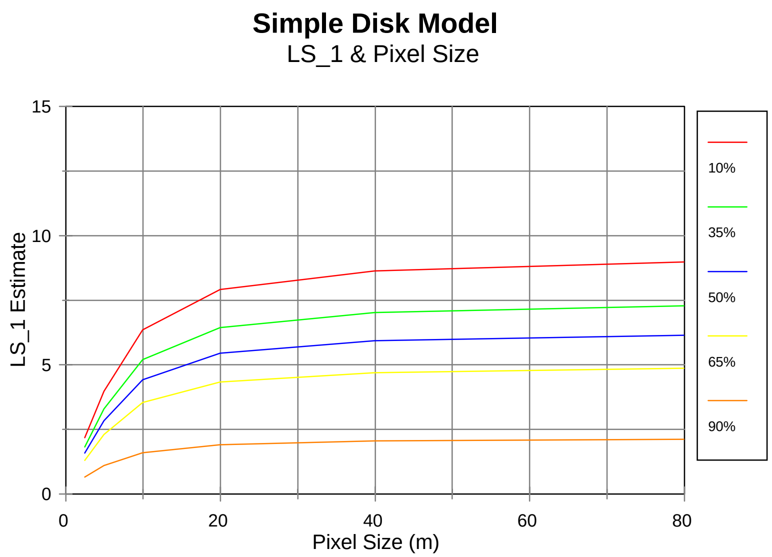
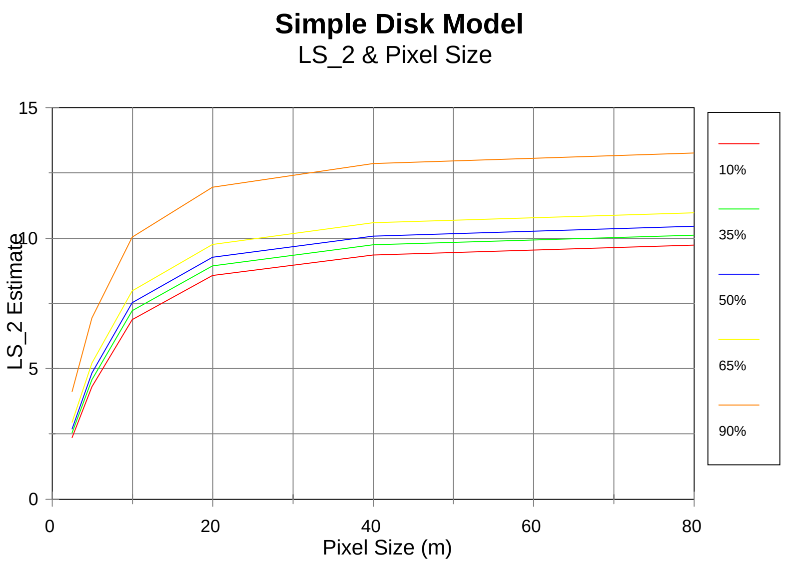


Figure 5: Variance as a function of cover

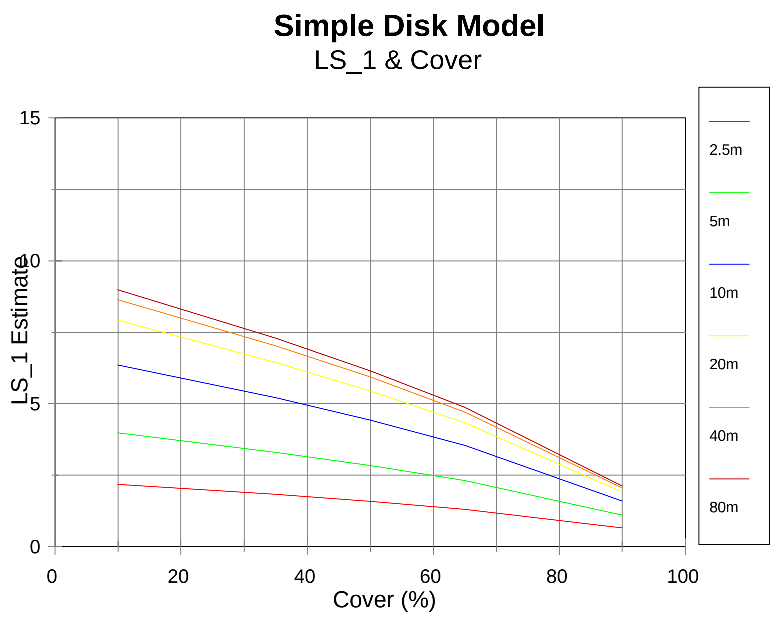
Invdisk Figures

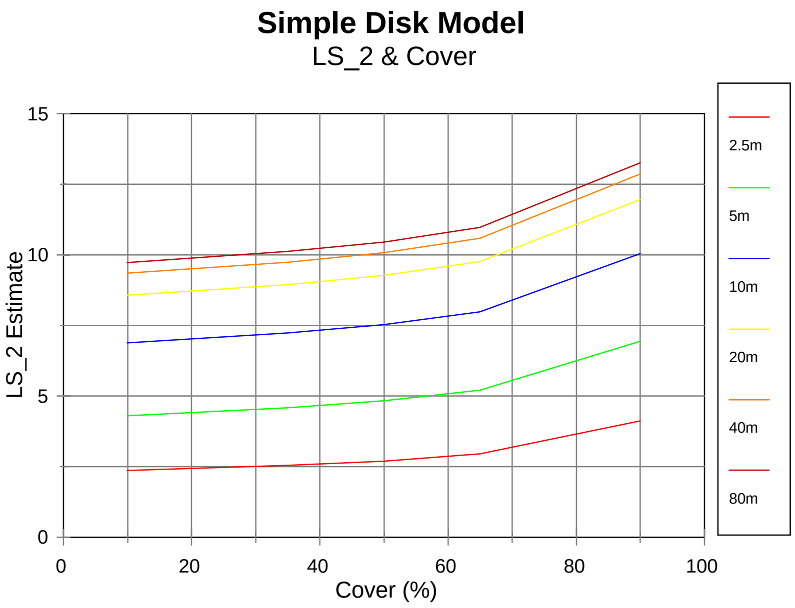




Figures 6.1 & 6.2: Li-Strahler estimates of disk size for Varying pixel size. 6.1 is Low Density estimate (LS_1) And 6.2 is the High Density estimate (LS_2). (Actual=10m)

Invdisk Figures





Figures 7.1 & 7.2: Li-Strahler estimates of disk size for Varying Cover. 7.1 is Low Density estimate (LS_1) And 7.2 is the High Density estimate (LS_2). (Actual=10m)

Invdisk Figures