Appendix: Computer Program

```
function DignomonicTiling4NCB(tiles,varargin)
% this program calculates mono- and dignomonic tilings and spirals
% USAGE
% DignomonicTiling4NCB(tiles): monognomonic with phis=Phi_crit
% DignomonicTiling4NCB(tiles,phis): regular monognomonic
% DignomonicTiling4NCB(tiles,phis,phir): regular dignomonic
% DignomonicTiling4NCB(tiles,phis,[]); dignomonic with phi2=Phi_crit^2/phis
% tiles = number of tiles in the tessellation
% Copyright 2016, CYe H. Waldman, algorithmicart@att.net
Phi_crit=1.538862046790905; % Phi for enclosed comp spirals; +/-1e16

if nargin==1
   phis=Phi_crit; phir=phis;
elseif nargin==2;
   phis=varargin{1};
   phir=phis;
elseif nargin==3
   phis=varargin{1};
   phir=varargin{2};
   if isempty(phir)
      phir=Phi_crit^2/phis;
   end
end

% test for suitability
if phis*phir<=1
   warning('Must have phis*phir > 1; they have both been inverted')
   phis=1/phis;    phir=1/phir;
end
% tiles
Z=DignomonicTiles(tiles,phis,phir);
% spirals
npts=100001;
[Zs,Ws]=DignomonicSpirals(tiles,phis,phir);
z0=Zs(1);
w0=Ws(1);
% showtime
```

```
figure;plot(Z,'k');axis equal
hold on;plot(Zs-z0,'r','LineWidth',1);
hold on;plot(Ws-w0+i,'b','LineWidth',1);
Phi=sqrt(phis*phir);
title(['[{\it\phi_s},{\it\phi_r},\Phi] = [' num2str(phis) ',' num2str(phir) ',' num2str(Phi) ']'])

return


function Z=DignomonicTiles(tiles,phis,phir)
% generate the tiling in cartesian coords
iseven=inline('rem(number,2) == 0;','number');
wrapN = @(x,N) (1+mod(x-1,N));

% three basic rectangle: seed, Gr, Gs
r=phir-1/phis; s=phis-1/phir;
seed=[0;1/phis;1/phis+i;i;0];
Gr=[0;r;r+i;i;0];
Gs=[0;phir;phir+i*s*phir;i*s*phir;0];

z1=seed;
z2=Gr-r;
z3=Gs-r-i*s*phir;
Z=[z1 z2 z3];
prevG=z3;
for k=4:tiles
   if iseven(k)
      G=Gr*(phir*phis)^((k-2)/2);
   else
      G=Gs*(phir*phis)^((k-3)/2);
   end
   n1=wrapN(k+1,4);
   n2=wrapN(k+2,4);
   z=G-G(n1)+prevG(n2);
   Z=[Z z];
   prevG=z;
end

return
```

```
function [Zs,Ws]=DignomonicSpirals(tiles,phis,phir)
% generate the spirals in the native coordinates
% create the spiral as per Gazale, p. 165 ff
% common elements
r=phir-1/phis;
s=phis-1/phir;
m=sqrt(r*s);
Phi=sqrt(phir*phis);
b=2*log(Phi)/pi;
npts=100001;

theta=linspace(0,(tiles+1)*pi/2,npts)';
cost=cos(theta);
sint=sin(theta);
y0=-1/(1+phir*phis);
x0=y0/phis;
% x,y,R in spiral coords
x=exp(b*theta).*(x0*cost-y0*r/m*sint);
y=exp(b*theta).*(x0*s/m*sint+y0*cost);
Zs=x+i*y;

% the complemntary spiral
theta=linspace(0,(tiles-1)*pi/2,npts)';
cost=cos(theta);
sint=sin(theta);
v0=-1/(1+phir*phis);
u0=v0/phis;
v0=1+v0;
% x,y,R in spiral coords
u=exp(b*theta).*(u0*cost-v0*r/m*sint);
v=exp(b*theta).*(u0*s/m*sint+v0*cost);
Ws=u+i*v;

return
```