Level Set Systems, Inc.

High Frequency Wave Propagation Using the Level Set Method

Stan Osher

Joint with:
L.-T. Cheng, M. Kang, H. Shim and Y. Tsai

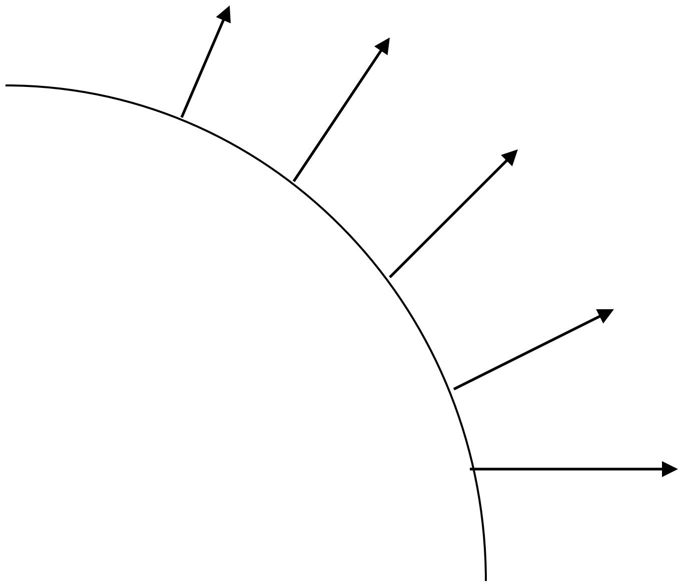
Supported by AFOSR

Paper(s) Available: www.levelset.com

Level Set Systems, Inc.

GRID BASED GEOMETRIC OPTICS

Why? Problems with ray tracing:



Diverging Rays – Miss large regions

PDE's on grids have advantages: e.g. other physics can be easily attached, Self interpolation, accurate finite difference schemes…

e.g. Scalar Wave Equation:

$$\frac{\partial^2 u}{\partial t^2} = c^2(x)\Delta u(x,t),\ x \in R^d,\ t > 0$$

$$u(x,0) = u_0(x),\ x \in R^d$$

$$\frac{\partial u}{\partial t}(x,0) = u_1(x),\ x \in R^d.$$

Use ansatz

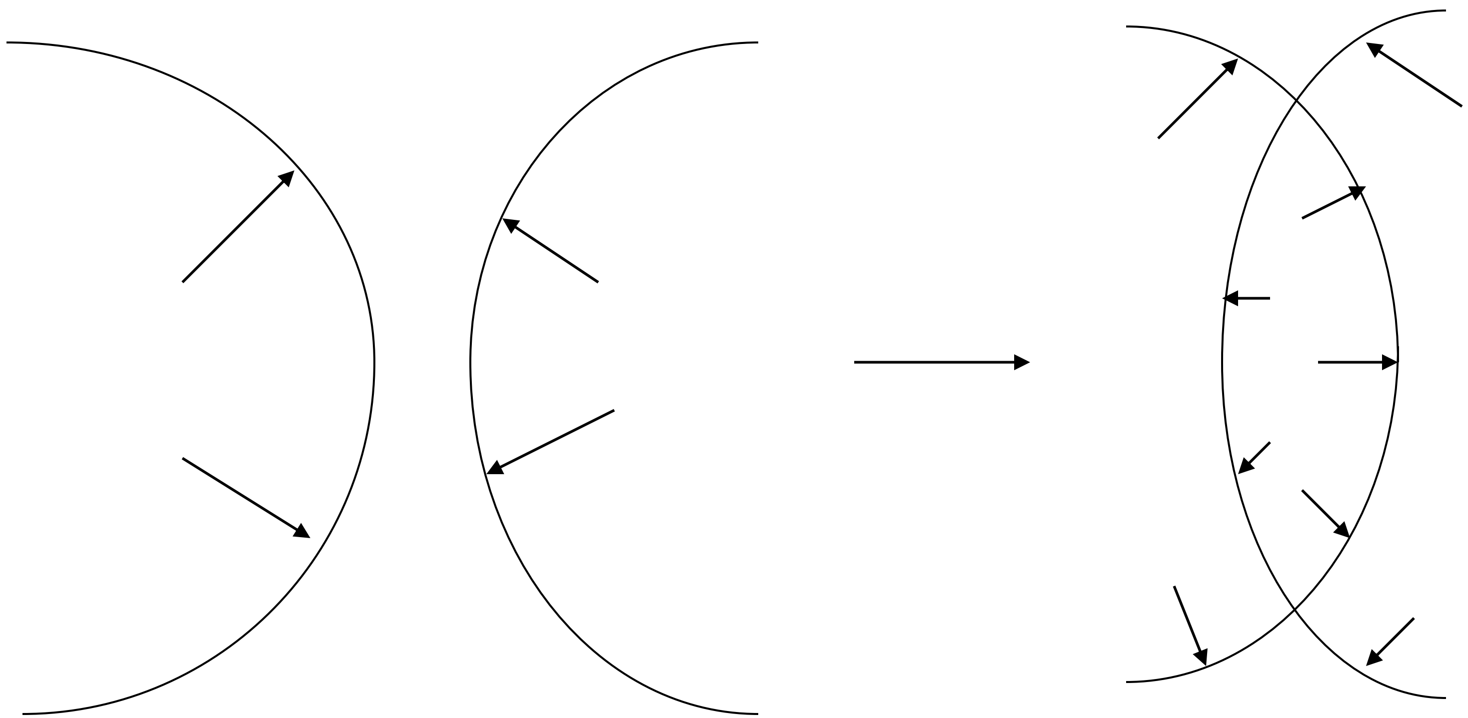
$$u \sim e^{i\omega\varphi(x,t)} \sum_{j=0}^{\infty} a_j(x,t)(i\omega)^{-j}$$

get, for $|\omega| \to \infty$

$$\frac{\partial \varphi}{\partial t} + c(x)|\nabla \varphi| = 0$$

eikonal equation.

We want the *multivalued* solution to this eikonal equation. The eikonal equation is nonlinear, but solutions to the wave equation superpose linearly.



cross through each other

This can be computed by the method of characteristics from

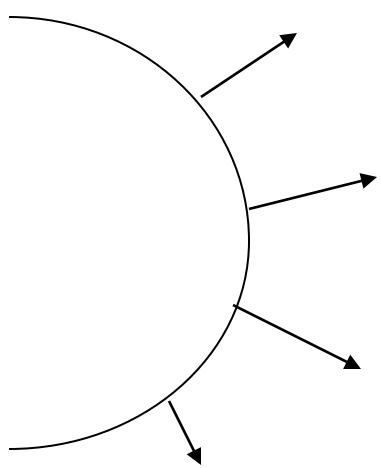
$$\varphi_t + c\,|\nabla\varphi| = 0$$

$$\nabla\varphi = (p_1, \cdots, p_d) = p$$

$$\frac{dx}{dt} = \frac{p}{|p|}c$$

$$\frac{dp}{dt} = -(\nabla c)\,|p|$$

⟺ Ray Tracing

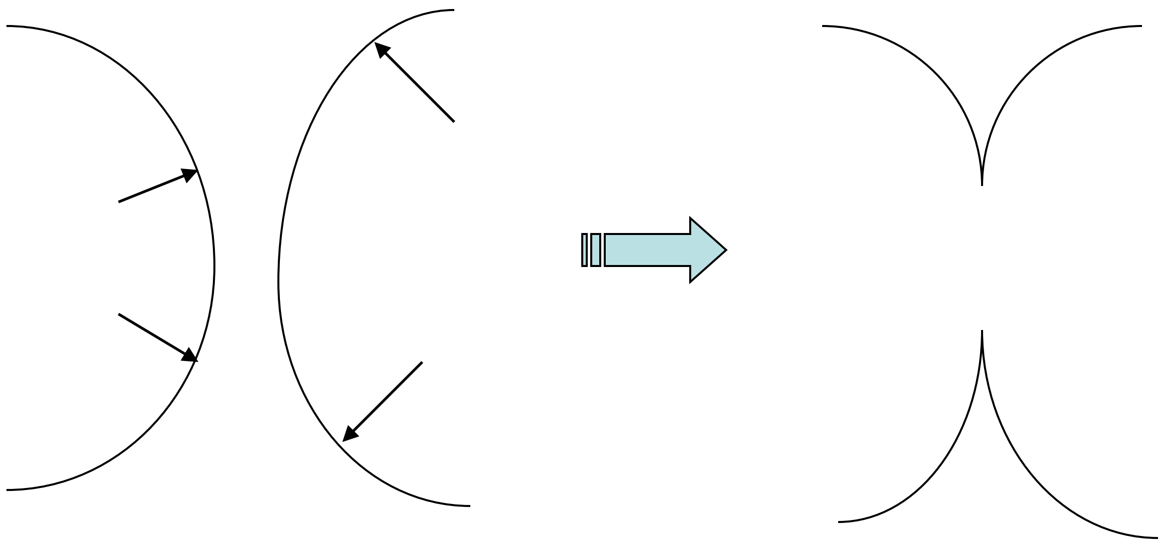


Diverging Rays cause problems

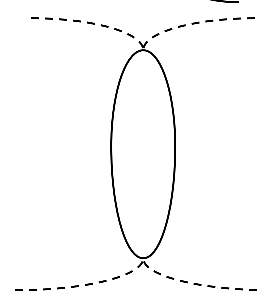
Grid Based PDE Approaches

(Vidale, Fatemi-Engquist-Osher, Benamou)

Viscosity solution of Hamilton-Jacobi eikonal equation
(Crandall-Lions,...)



single valued, loses



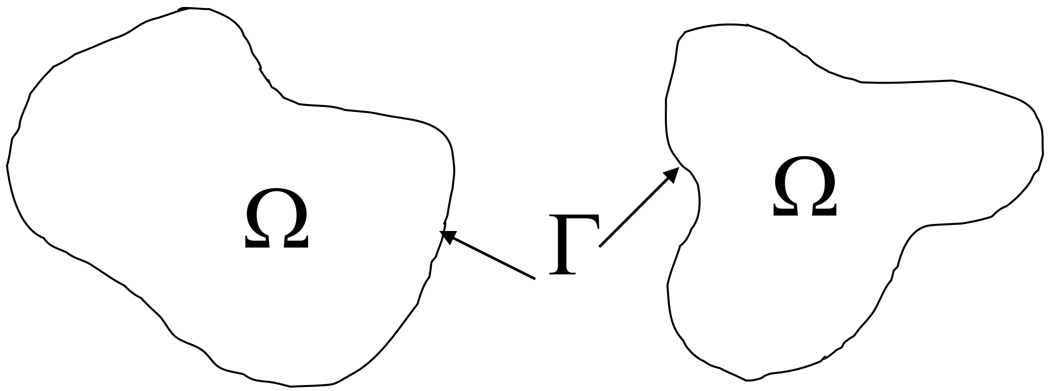
Patch together somehow

+ uses upwind H.-J. monotone, ENO, WENO schemes
(O-Shu, O-Sethian, Jiang-Peng)

Self interpolates

- Loses multivaluedness – need to patch things together.

Level Set Method



Γ Represented by $\vec{x}$ with: $\varphi(\vec{x}) = 0$

$\Omega : \vec{x}$ with $\varphi(\vec{x}) < 0$

Implicit surface, use grid

Key idea: move $\Gamma(t)$ with velocity $\vec{v}$, get

$$\varphi_t + \vec{v} \cdot \nabla \varphi = 0$$

or

$$\varphi_t + v_n |\nabla \varphi| = 0$$

Easy geometry extraction, topological changes

Vector Level Set Method

e.g.

$$\begin{pmatrix} \varphi_1(x_1, x_2, x_3, t) \\ \varphi_2(x_1, x_2, x_3, t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

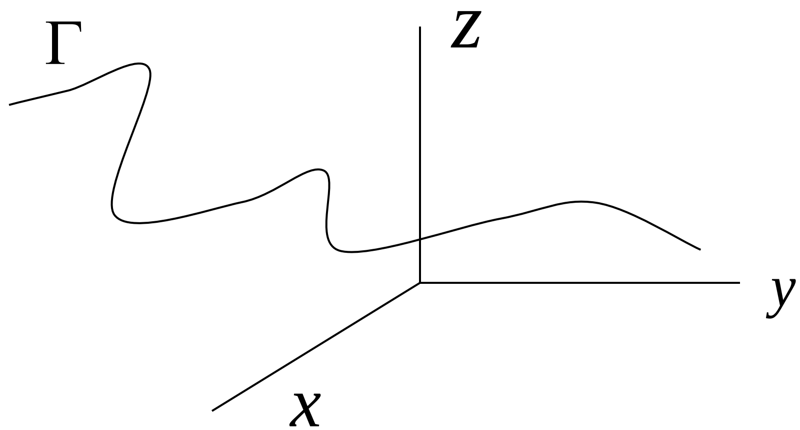
Intersection of 2 level sets can evolve by moving φ_1 , φ_2

So curve $\Gamma(t)$ in R^3 can evolve by moving φ_1 , φ_2 .

Level Set Systems, Inc.

New idea, based on using vector valued level set method to move high codimension stuff, e.g. curves in R^3
(VVLSM invented by Burchard, Cheng, Merriman & O, 1999)

(suggested as a theoretical device by Ambrosio-Soner)



Γ curve in R^3

$$\varphi = \begin{pmatrix} \varphi_1(x_1, x_2, x_3, t) \\ \varphi_2(x_1, x_2, x_3, t) \end{pmatrix}$$

$\varphi = 0$ defines $\Gamma(x,t)$

Geometry of Γ

$$\frac{\nabla\varphi_1 \times \nabla\varphi_2}{|\nabla\varphi_1 \times \nabla\varphi_2|} = \tau = \text{tangent to } \Gamma \text{ on } \varphi = 0$$

(curvature) normal = $\frac{\partial}{\partial\tau}\tau$, on $\varphi = 0$

All geometry comes simply from φ_1 and φ_2 on their zero level sets restricted to Γ

Use Liouville (Vlasov??) equation

Solve for $w(x, p, t)$, with $x = (x_1, \cdots, x_d)$, $p = (p_1, \cdots, p_d)$, t
2d + 1 independent variables

$$w_t + c(x)\frac{p}{|p|}\nabla_x w - |p|\nabla_x c \cdot \nabla_p w = 0$$

linear equation, characteristics

$$\frac{dx}{dt} = c\frac{p}{|p|}, \quad \frac{dp}{dt} = -\nabla_x c\,|p|$$

same as for the eikonal equation along these rays in 2D space
w is constant along characteristic!

To represent a curve $\Gamma(x, t)$ in R^d space we let

$$w = \begin{pmatrix} w_1(x, p, t) \\ w_2(x, p, t) \\ \vdots \\ w_3(x, p, t) \end{pmatrix}$$

Γ is projection on x space of $w \equiv 0$

Can lower the dimension by one

$$\frac{d}{dt}\frac{c^2|p|^2}{2} = |p|^2 c\nabla c\cdot\frac{dx}{dt}+c^2 p\cdot\frac{dp}{dt}$$

$$= c^2\nabla c\cdot p|p| - c^2 p\cdot\nabla c|p| = 0$$

$$\therefore\ |p(x,t)| = |p(x,0)|$$

Thus we can use angle variables for d = 2. Need only

$\theta,\ p(x,t) = |p(x,0)|(\cos\theta,\sin\theta)$

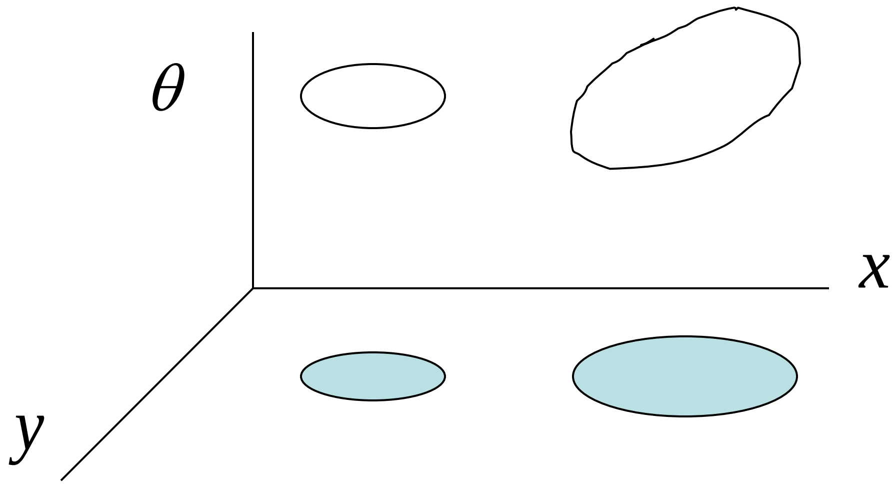
θ = angle of normal to Γ

Equations in 2D become

$$u = \begin{pmatrix} u^1 \\ u^2 \end{pmatrix} = u(x,y,\theta,t)$$

$$u_t + c(x,y)(\cos\theta u_x + \sin\theta u_y) + (c_x\sin\theta - c_y\cos\theta)u_\theta = 0$$

Two linear, decoupled equations



<u>NOTE</u> if Γ has disjoint components in x , y , θ space at $t = 0$, they never intersect at later t in this (cotangent space) representation.

Also Note: propagation speed involves

$$\frac{d\theta}{dt} = \nabla c \cdot (\cos\theta, \sin\theta)^{\perp}$$

so different time step restriction, generally a bit more restrictive.

In 3D we have spherical coordinates

$$P = |P| \cdot (\cos\theta_1 \cos\theta_2, \ \cos\theta_1 \sin\theta_2, \sin\theta_1)$$

Reduce it to a 5 dimension + time problem.

System of 3 equations:

$$u_t + c(\cos\theta_1 \cos\theta_2 u_x + \cos\theta_1 \sin\theta_2 u_y + \sin\theta_1 u_z)$$
$$+ (c_x \sin\theta_1 \cos\theta_2 + c_y \sin\theta_1 \sin\theta_2 - c_z \cos\theta_1) u_{\theta_1}$$
$$+ (c_x \sin\theta_2 - c_y \cos\theta_2) \frac{u_{\theta_2}}{\cos\theta_1} = 0, \quad -\frac{\pi}{2} \le \theta_1 \le \frac{\pi}{2}, -\pi \le \theta_2 \le \pi$$

Trouble at $\theta_1 = \pm\frac{\pi}{2}$ (north & south poles)

if $\nabla c \neq 0$ at such points. Easily fixed.

Complexity seems high. But we can use local level set.

In principle we are looking at a manifold of dimension d-1 + time.

Should be complexity $O(n^{d-1}\log n)$ to update, also low storage.

We now have that.

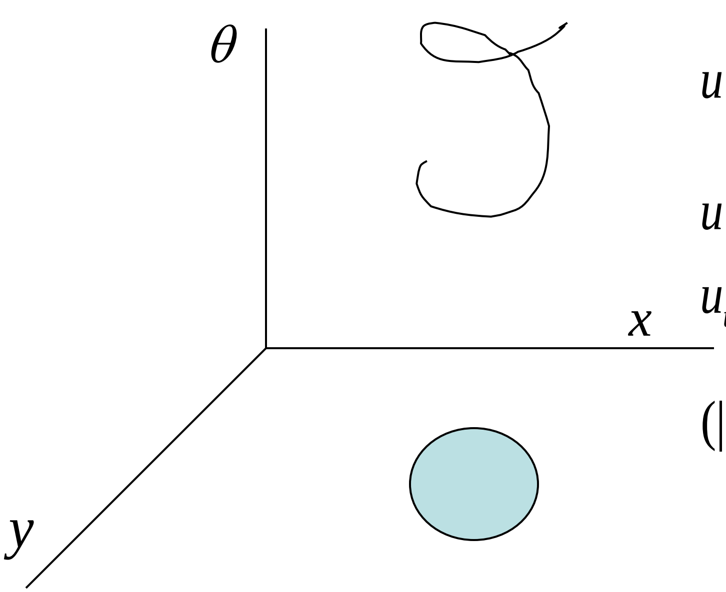
Solve only near where $u = 0$.

Earlier work: Engquist, Runborg, Tornberg

Use segment projection: needs logic – many many segments can develop.

Complexity is the same, intricate programming

Our method: Review: for d = 2



$$u = \begin{pmatrix} u_1(x,y,\theta,t) \\ u_2(x,y,\theta,t) \end{pmatrix}$$

$$u = 0 \quad \text{on } \Gamma$$

$$u_t + c(x,y)(\cos\theta u_x + \sin\theta u_y) + (c_x \sin\theta - c_y \cos\theta) u_\theta = 0$$

$$(|c| + |\nabla c|) \frac{\Delta t}{\Delta x, \Delta y, \Delta\theta} \leq \frac{1}{2}$$

Intensity

$$\frac{A(x,y,\theta,t)}{A(x,y,\theta,0)} = \frac{\sqrt{x_\theta^2 + y_\theta^2(0)}}{\sqrt{x_\theta^2 + y_\theta^2(t)}}$$

and

$$x_\theta^2 + y_\theta^2 = \frac{J^2(u;(y,\theta)) + J^2(u;(x,\theta))}{J^2(u;(x,y))}$$

J = Jacobian

$$J(u;(y,\theta)) = \det\begin{pmatrix} (u_1)_y & (u_1)_\theta \\ (u_2)_y & (u_2)_\theta \end{pmatrix}$$

Easy (passive) calculation.

Also 3D.

Initialization

Given an initial surface via a level set function

$$\varphi(x, y) = 0$$

Initialize

$$\varphi(x, y, \theta, 0) = \varphi(x, y)$$

$$\psi(x, y, \theta, 0) = \varphi_x(x, y)\cos\theta - \varphi_y(x, y)\sin\theta$$

In 3D

$$\varphi(x, y, z) = 0$$

$$\varphi(x, y, z, \theta_1, \theta_2, 0) = \varphi(x, y, z)$$

$$\psi(x, y, \theta_1, \theta_2, 0) = \varphi_x \sin\theta_1 - \varphi_z \cos\theta_1 \cos\theta_2$$

$$\eta(x, y, \theta_1, \theta_2, 0) = \varphi_y \sin\theta_1 - \varphi_z \cos\theta_1 \sin\theta_2$$

Example: 2D

Start with an ellipse

$$x_0 = b\cos\eta, \qquad b > 0$$

$$y_0 = \sin\eta$$

$$\eta = \tan^{-1}(b\tan\theta)$$

$$u_1(x,y,\theta,0) = x - \frac{b}{\sqrt{1+b^2\tan^2\theta}}$$

$$u_2(x,y,\theta,0) = y - \frac{b\tan\theta}{\sqrt{1+b^2\tan^2\theta}}$$

Say $c(x,y) \equiv 1$

Then

$$u_1 = x + t\cos\theta - \frac{b}{\sqrt{1+b^2\tan^2\theta}}$$

$$u_2 = y + t\sin\theta - \frac{b\tan\theta}{\sqrt{1+b^2\tan^2\theta}}$$

Curve $u_1 = u_2 = 0$, is :

$$x = -t\cos\theta + \frac{b}{\sqrt{1+b^2\tan^2\theta}}$$

$$y = -t\sin\theta + \frac{b\tan\theta}{\sqrt{1+b^2\tan^2\theta}}$$

Other example in 2D

Suppose initial data for eikonal equation is given at $y = 0$

$$\varphi(x,0,0) = h(x)$$

Then

$$\varphi_x(x,0,0) = h'(x) = p_0(x,0)$$

$$\varphi_y(x,0,0) = q_0(x,0) = \pm\sqrt{\frac{1}{c^2} - h'(x)^2}$$

$$\theta_0(x,0) = \tan^{-1}\left(\pm\sqrt{\frac{1}{c^2(h')^2} - 1}\right)$$

$$u_1(x,y,\theta,0) = y$$

$$u_2(x,y,\theta,0) = \cos\theta - h'(x)c(x,0)$$

simple initialization

Level Set Systems, Inc.

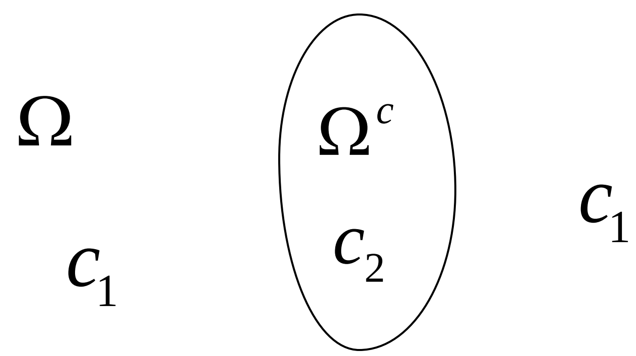
3D example : Initially ellipse

$$
\begin{aligned}
x &= a\cos s_1 \cos s_2 \\
y &= b\cos s_1 \sin s_2 \\
z &= c\sin s_1 \\
\frac{x^2}{a^2} &+ \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1
\end{aligned}
$$

Can initialize :

$$
\begin{aligned}
u_1 &= x - \frac{a^2\cos\theta_1\cos\theta_2}{\sqrt{c^2\sin^2\theta_1+\cos^2\theta_1(a^2\cos^2\theta_2+b^2\sin^2\theta_2)}} \\
u_2 &= y - \frac{b^2\cos\theta_1\sin\theta_2}{\sqrt{c^2\sin^2\theta_1+\cos^2\theta_1(a^2\cos^2\theta_2+b^2\sin^2\theta_2)}} \\
u_3 &= z - \frac{c^2\sin\theta_1}{\sqrt{c^2\sin^2\theta_1+\cos^2\theta_1(a^2\cos^2\theta_2+b^2\sin^2\theta_2)}}
\end{aligned}
$$

Reflection: let $c = c_1(x)$ in Ω, $c = c_2(x)$ in Ω^c



Can solve reflection & refraction easily in this framework

$c = c_1 + (c_2 - c_1) H(\psi(x))$, ψ is level set function for Ω :

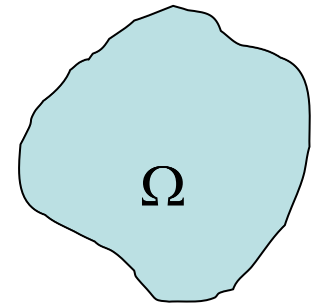
$$\psi(x) > 0 \text{ in } \Omega^c$$

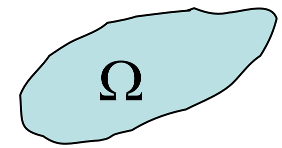
$$\psi(x) < 0 \text{ in } \Omega$$

$$H(x) \equiv 1,\ x > 0,\ \ H(x) \equiv 0,\ x < 0$$

inwards normal to $\Omega^c = \dfrac{\nabla \psi}{|\nabla \psi|} = (\cos\theta_n, \sin\theta_n)$, (in 2D).

Reflection





u defined $u(x, y, \theta, t)$ for $(x, y) \notin \Omega$

Treat as an initial – boundary value problem:

The value of u corresponding to Ω_B , θ_I which is incoming = value of u at $\theta_R = 2\theta_B - \theta_I - \pi$

θ_B is the angle of the outwards normal. Discrete values of θ are given so we interpolate in θ .

Transmission: via Snell's Law; solve initial boundary value problem and Ω

$$u^T(x, y, \theta, t) = u^I\left(x, y, \theta_n + \sin^{-1}\left(\frac{c_I}{c_I}\sin(\theta - \theta_n)\right), t\right)$$

$$u^T(x, y, \theta, 0) = \text{large positive}$$

Transmitted wave where $u^T = 0$ in Ω^c .

Level Set Systems, Inc.

More about reflection

Give (x, y) lying on boundary of Ω and any θ,

$$u(x,y,\theta,t) = u(x,y,2\theta_B - \theta - \pi, t)$$

This translates into :

No boundary conditions needed for incoming ray
(upwind differencing takes care of this)
Reflection boundary conditions as above for reflected rays.
Since we use x, y differencing, we

(a) Interpolate if an incoming ray "upwinds" the wrong way at the boundary
(b) use a subgrid interpolant to go to the boundary for the reflected wave
(c) Make sure that we stop at least $\frac{\Delta x}{2}$ away from this subgrid point
(d) Limit the ENO stencils so as not to cross the boundary

Works well in 2 and 3D.

Alternative for transmission:

Just solve directly

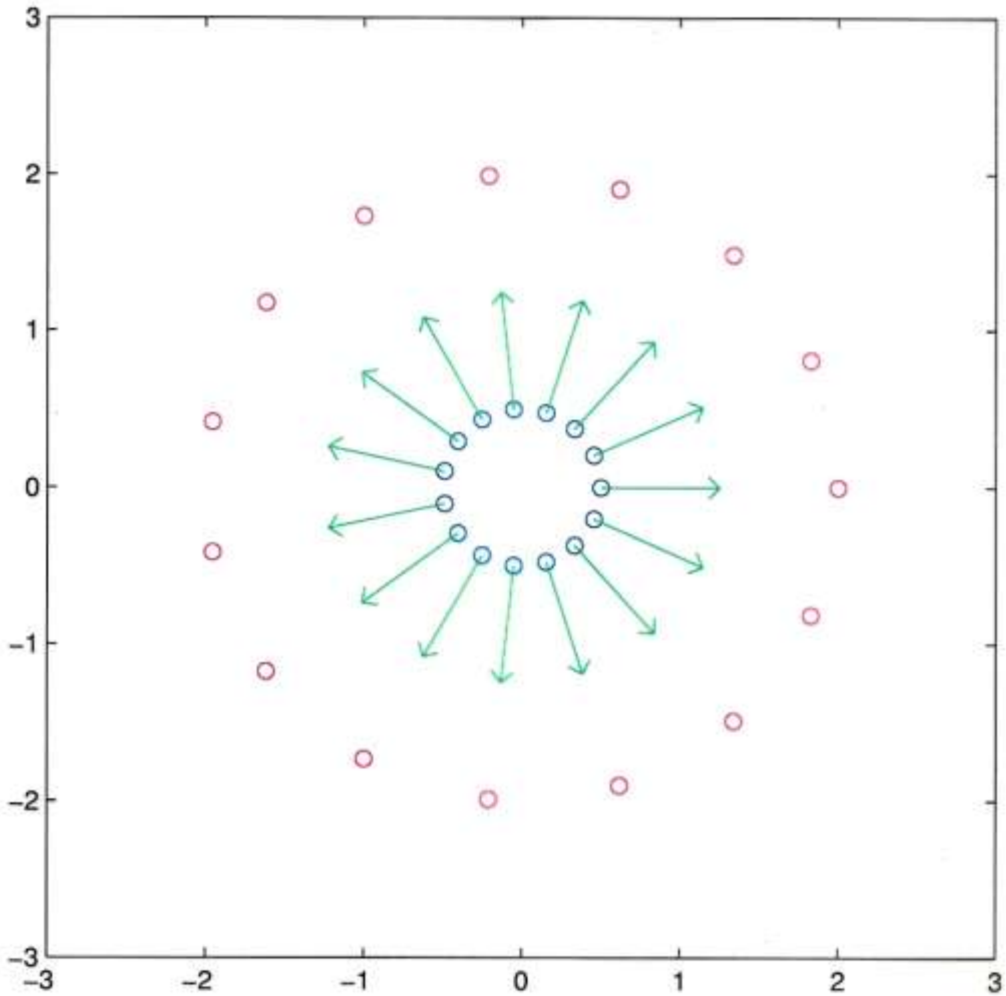
$$u_t + c(\cos\theta u_x + \sin\theta u_y) + (\sin\theta c_x - \cos\theta c_y)u_\theta = 0$$

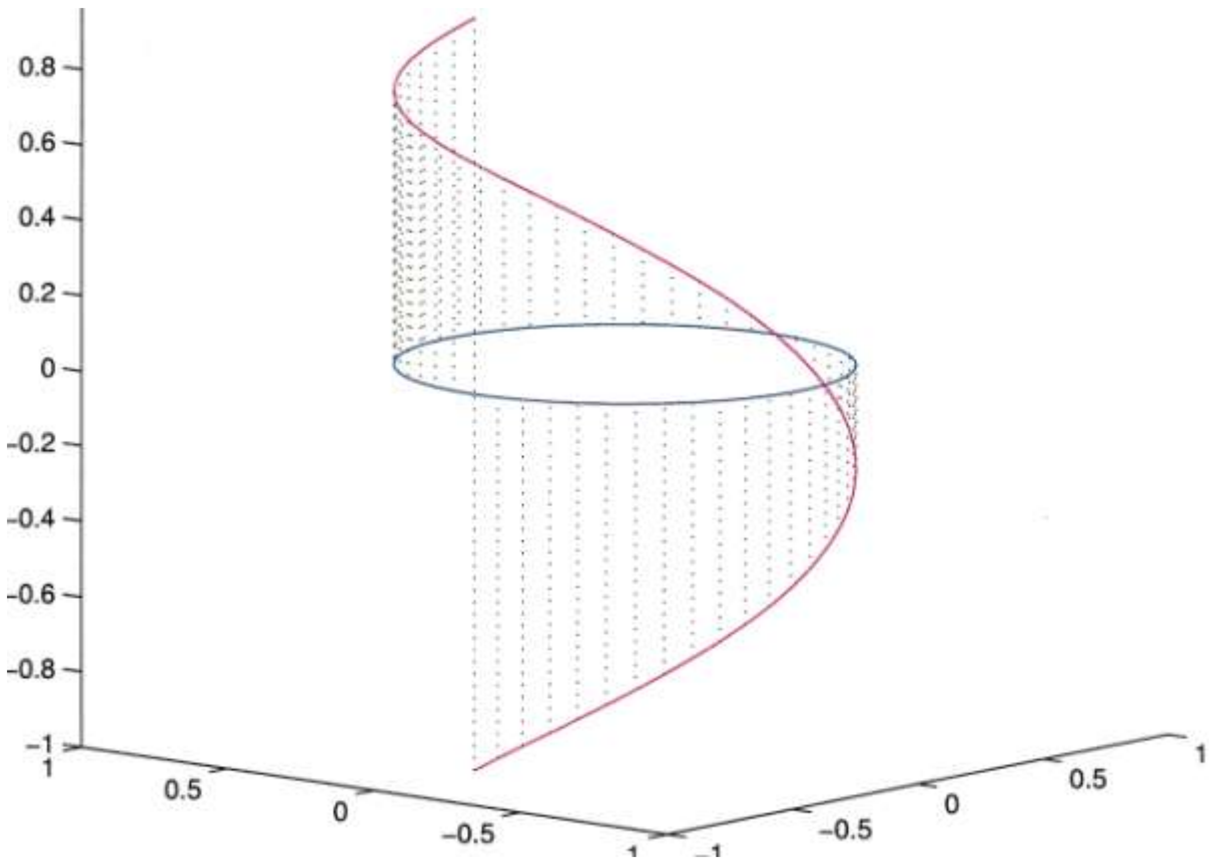
Have approximate delta function coefficients of u_θ, just solve
By smoothing c slightly and restricting

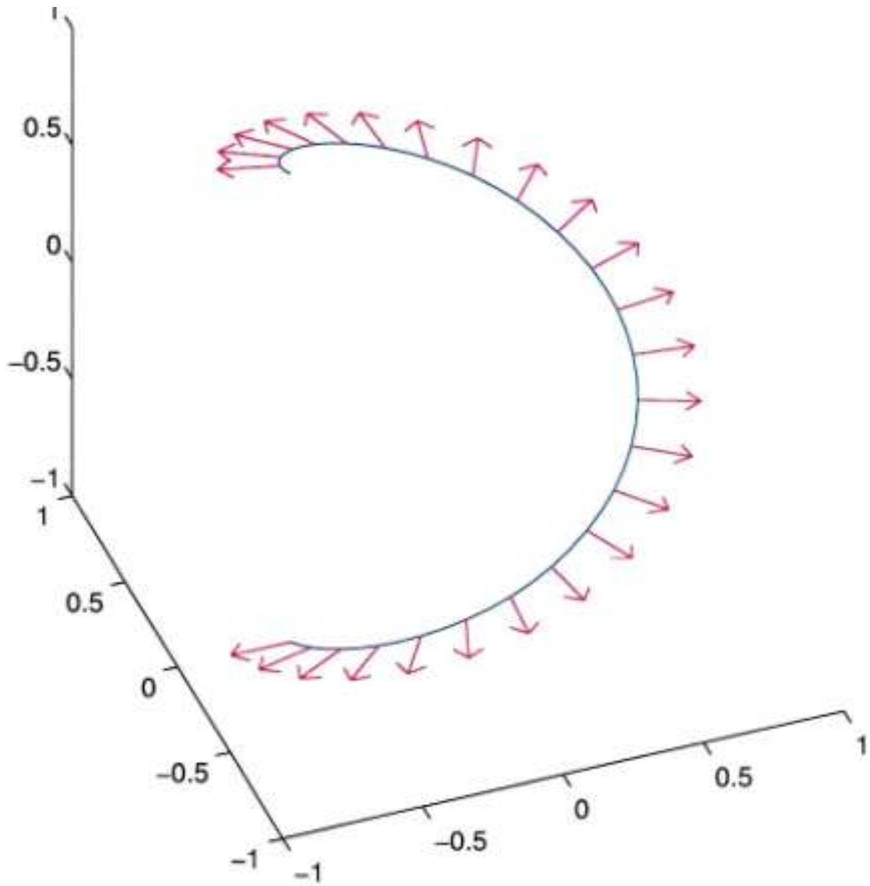
$$\frac{\Delta t}{(\Delta\theta)^2} \leq K$$

Works, but slower, actually works even with $\frac{\Delta t}{\Delta\theta} \leq K$!!
Automatically get Snell's Law.

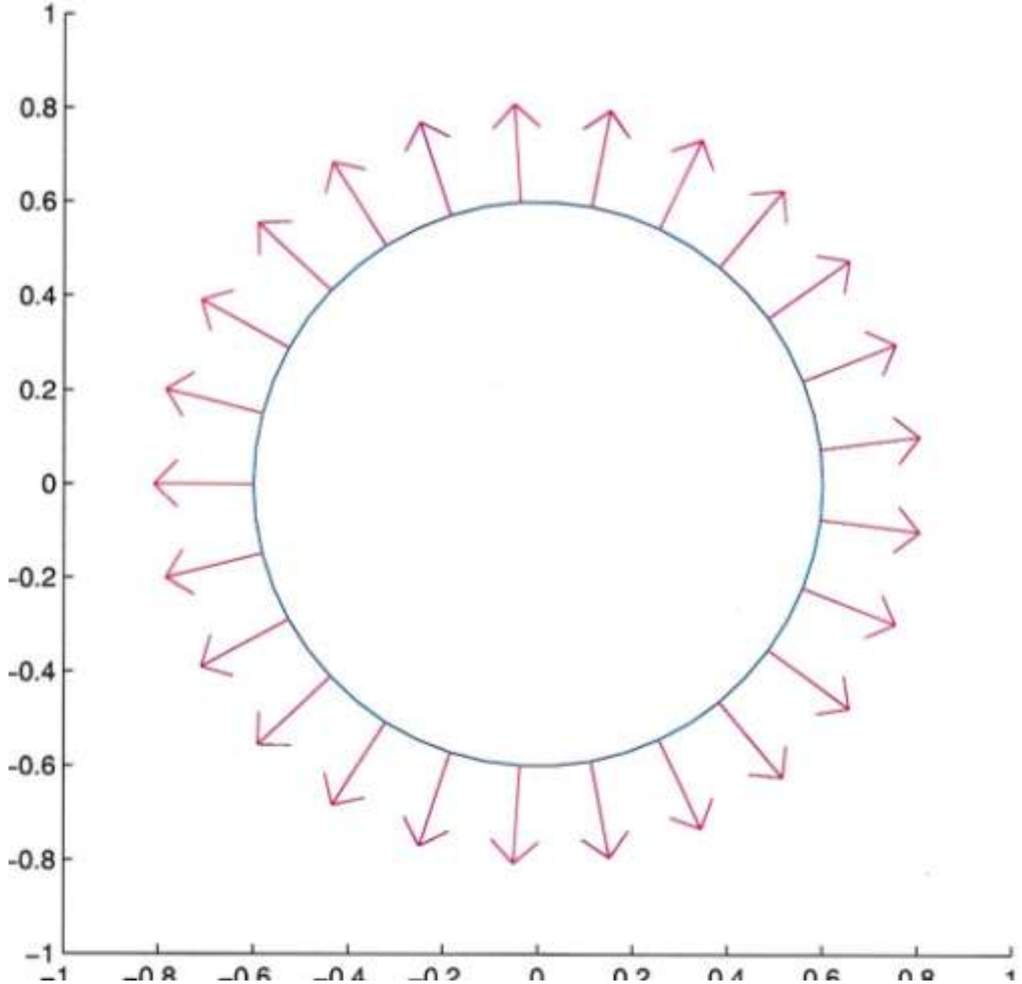
3D also simple.

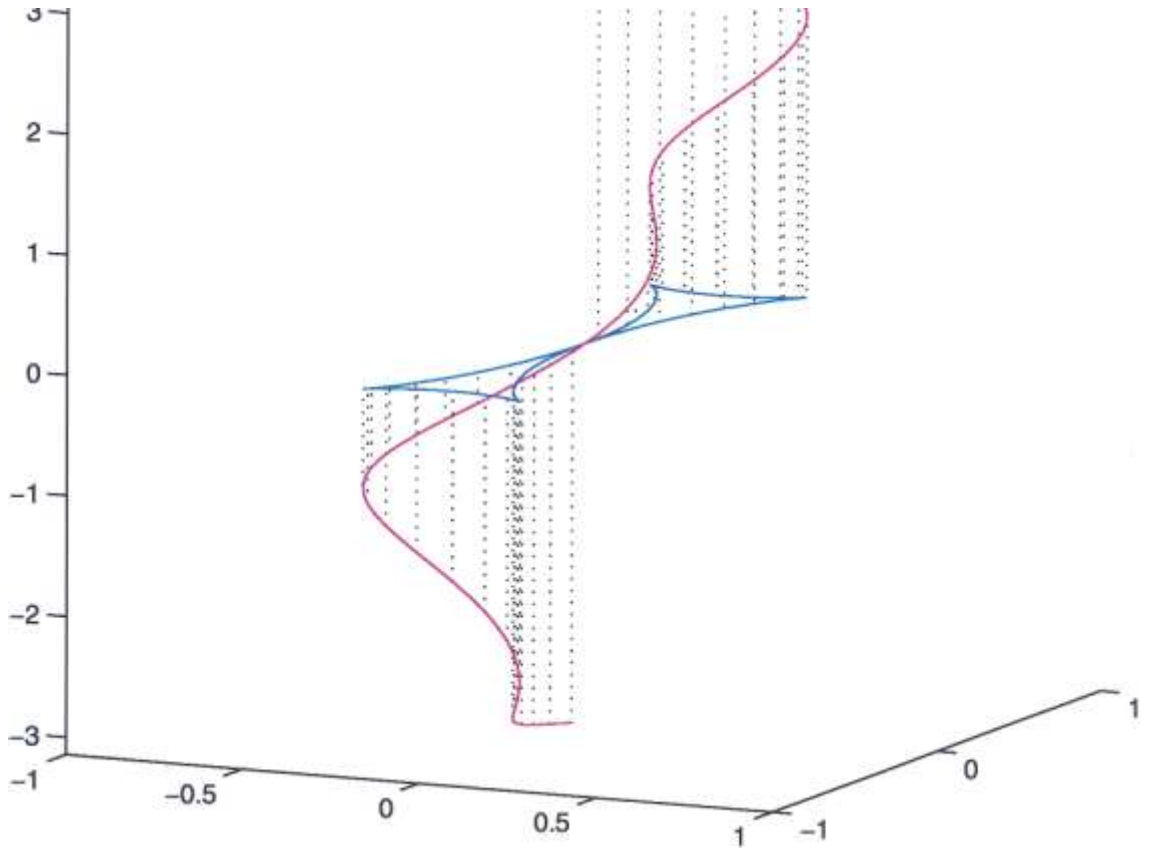

3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3


0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
1
0.5
0
-0.5
-0.5
0
0.5
1


1
0.5
0
-0.5
-1
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

Level Set Systems, Inc.




3
2
1
0
−1
−2
−3
−1
−0.5
0
0.5
1
−1
0
1

Exciting New Stuff

Computing Multiphase Semiclassical Limits of Schrödinger Equation

S. Osher, S. Jin, Y.-H. Tsai, H. Liu and L.-T. Cheng

$$i\varepsilon\psi_t^\varepsilon + \frac{\varepsilon^2}{2}\Delta\psi^\varepsilon = V(x)\psi^\varepsilon \quad \text{for} \quad |\varepsilon| \ll 1, \quad \varepsilon \text{ Real}$$

$$x = (x_1, \cdots, x_n)$$

$$\psi^\varepsilon(x,t) = A(x)e^{i\frac{\varphi(x,t)}{\varepsilon}}, \quad A = A_0 + \varepsilon A_1 + \cdots$$

H-J $$\varphi_t + \frac{|\nabla\varphi|^2}{2} + \nabla V = 0$$

$A_0^2 = \rho$, we get $\rho_t + \nabla \cdot \rho \nabla \varphi = 0$

Level Set Systems, Inc.

Define $\nabla\varphi = u$

$$u_t + u\cdot\nabla u + \nabla\cdot V = 0$$

Use vector level set method,
get linear Liouville system

$$w(x,u,t) \equiv 0$$

Solve for $w(x,y,t)$

multivalued w

(Li) $w_t + y\cdot w_x - \nabla_x V\cdot w_y = 0, \qquad w = (w^1,\cdots .w^n)$

$$w_0(x,y) = y - \nabla_x\varphi(x,0)$$

Let $J = \det\left\{w^i_{y_j}\right\}$

Compute single valued density $\rho(x, y, t)$
Then :

$\rho\,|\,J\,|$ also satisfies Liouville equation

Can desingularize problem
Compute

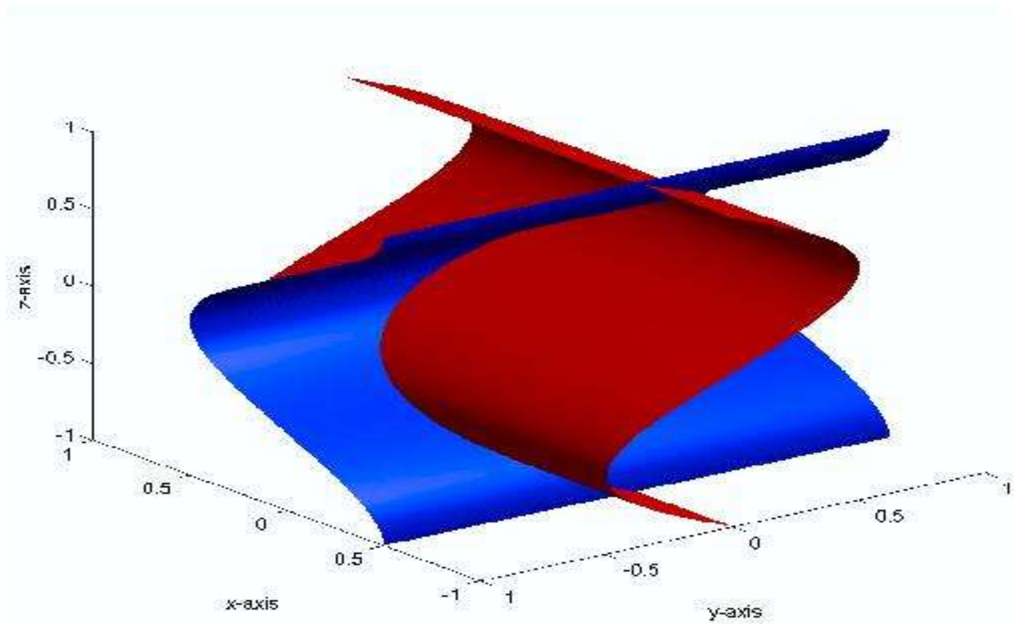
$$I = \int dy\,\rho(x,y,t)\prod_{i=1}^{n}\delta(w^{i}(x,y,t))\,|\,J\,|$$

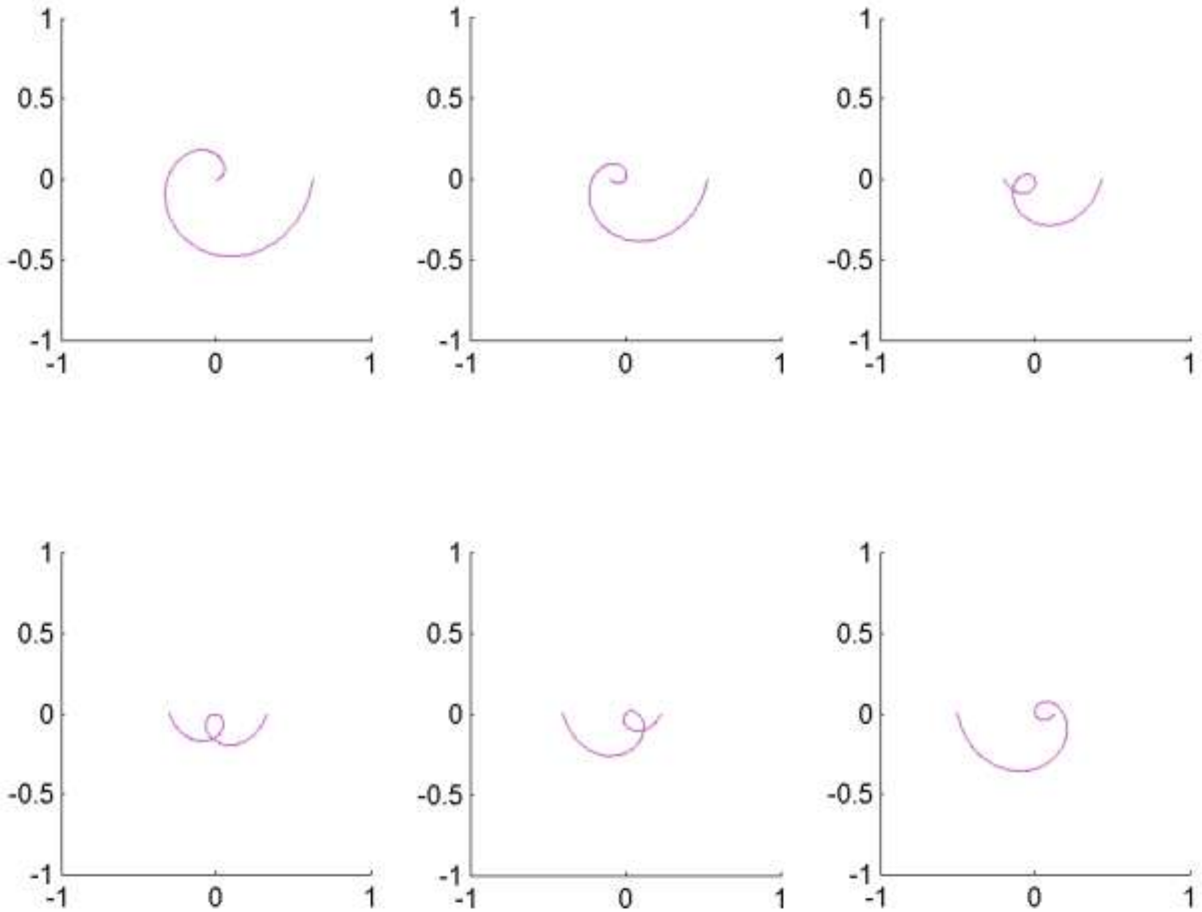
gives mass

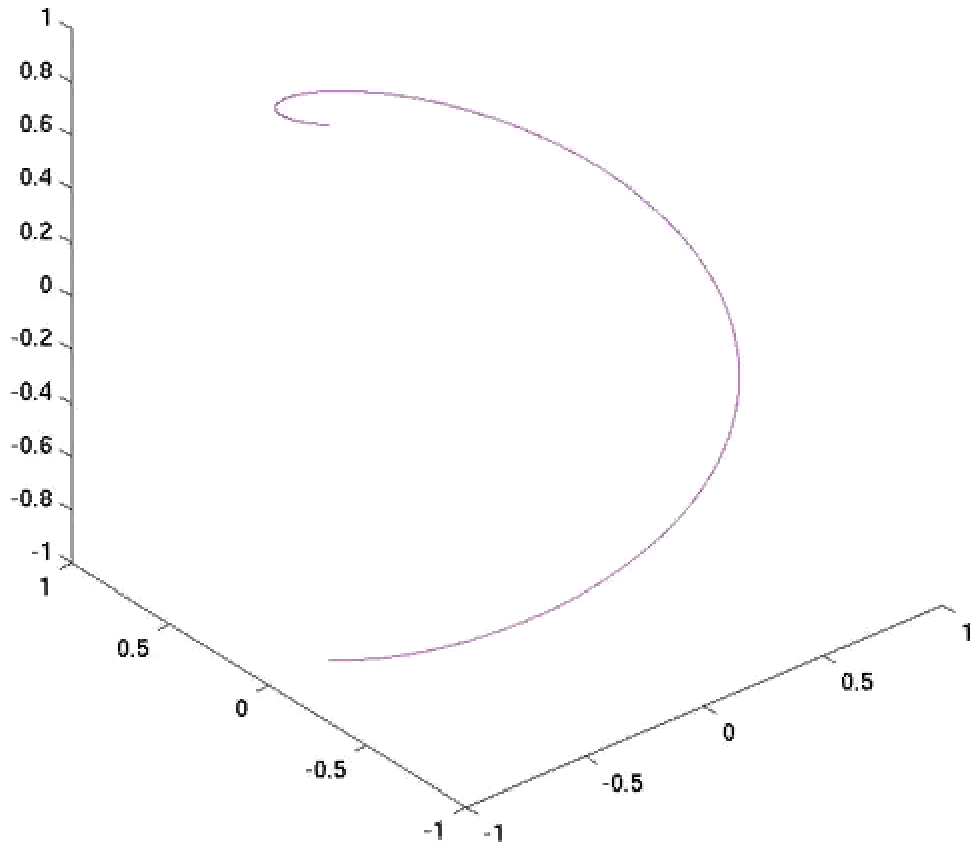
$$J_i = \int dy\; y_i\,\rho(x,y,t)\prod_{i=1}^{n}\delta(w^{i}(x,y,t))\,|\,J\,|$$

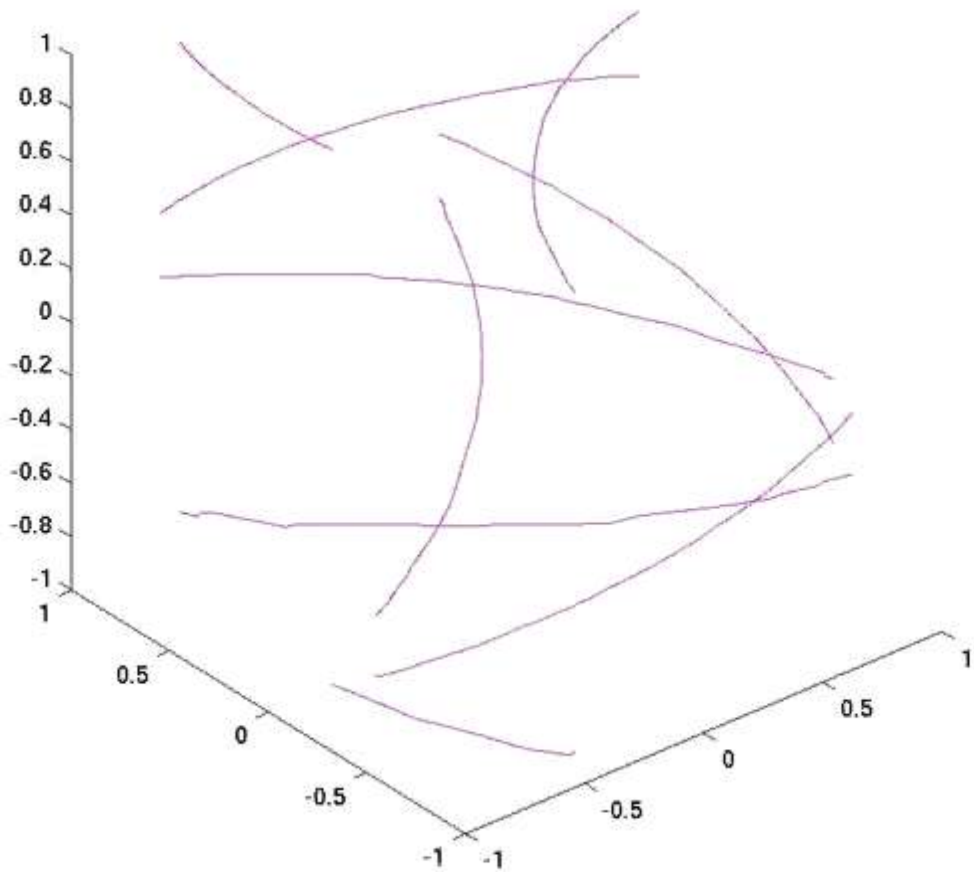
given ith component of momentum

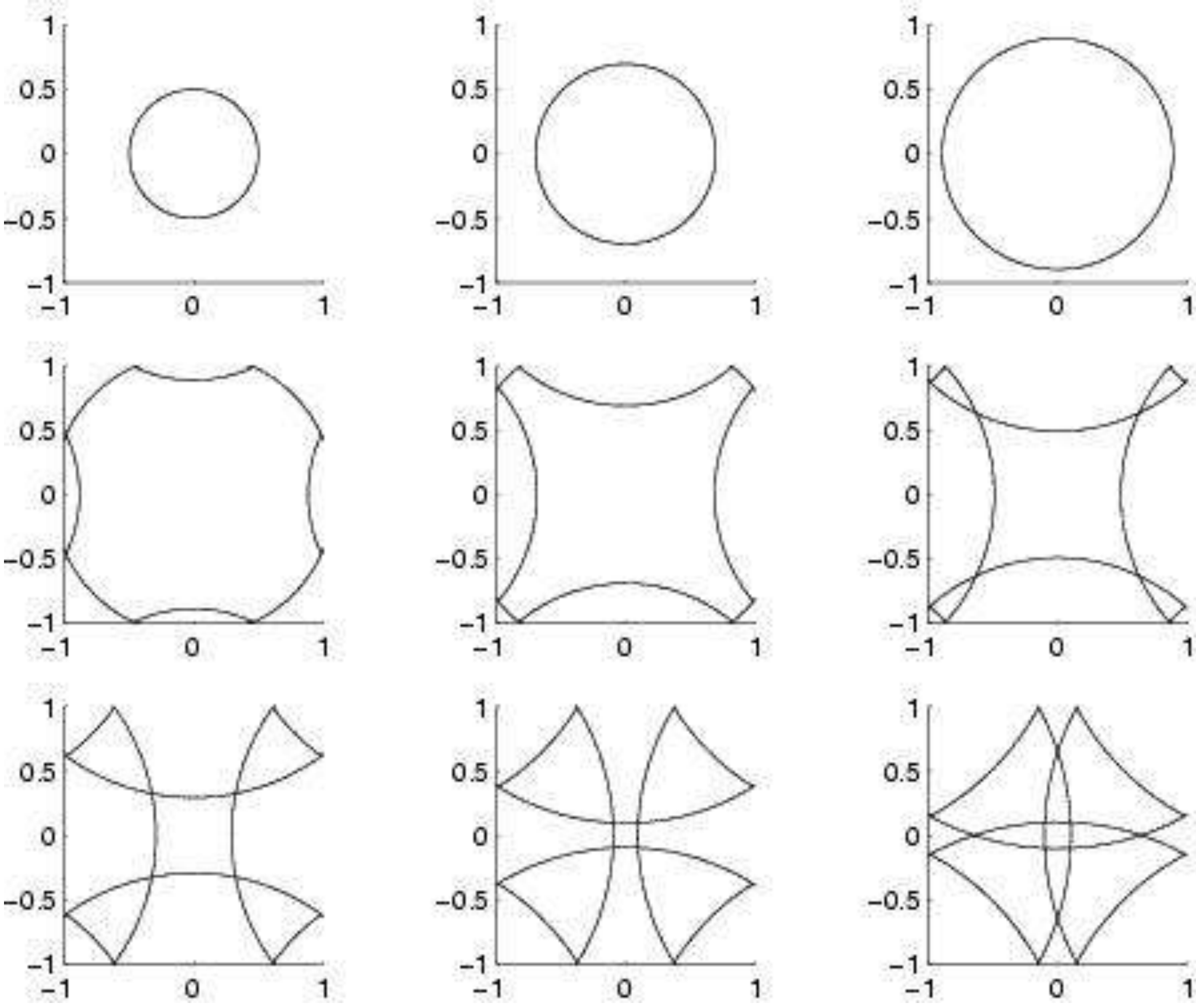
<u>New Approach</u>
No moments
No Wigner Transform
Works in multi-dimension

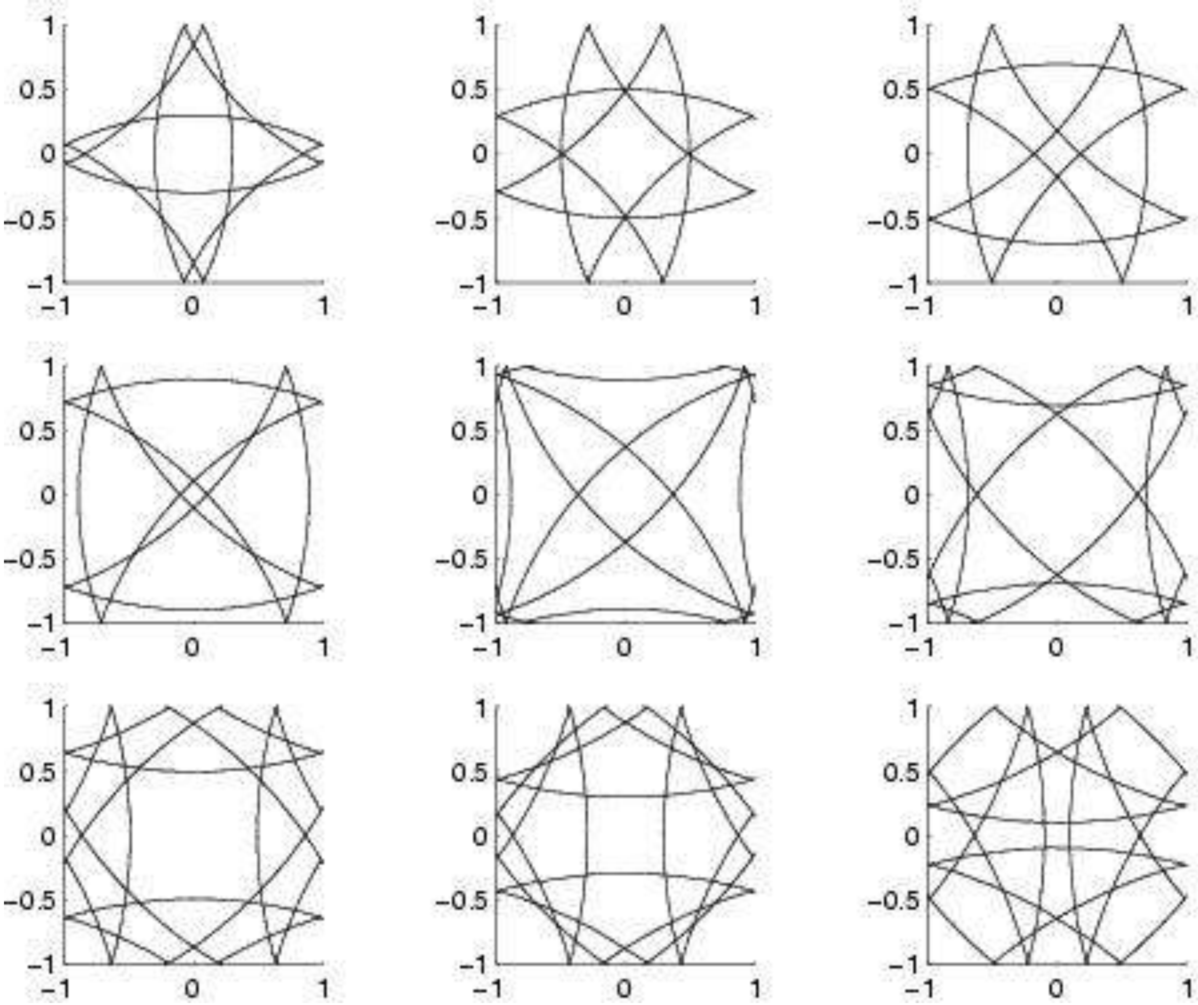

1
0.5
0
-0.5
-1
z-axis
1
0.5
0
-0.5
-1
x-axis
1
-0.5
0
0.5
1
y-axis

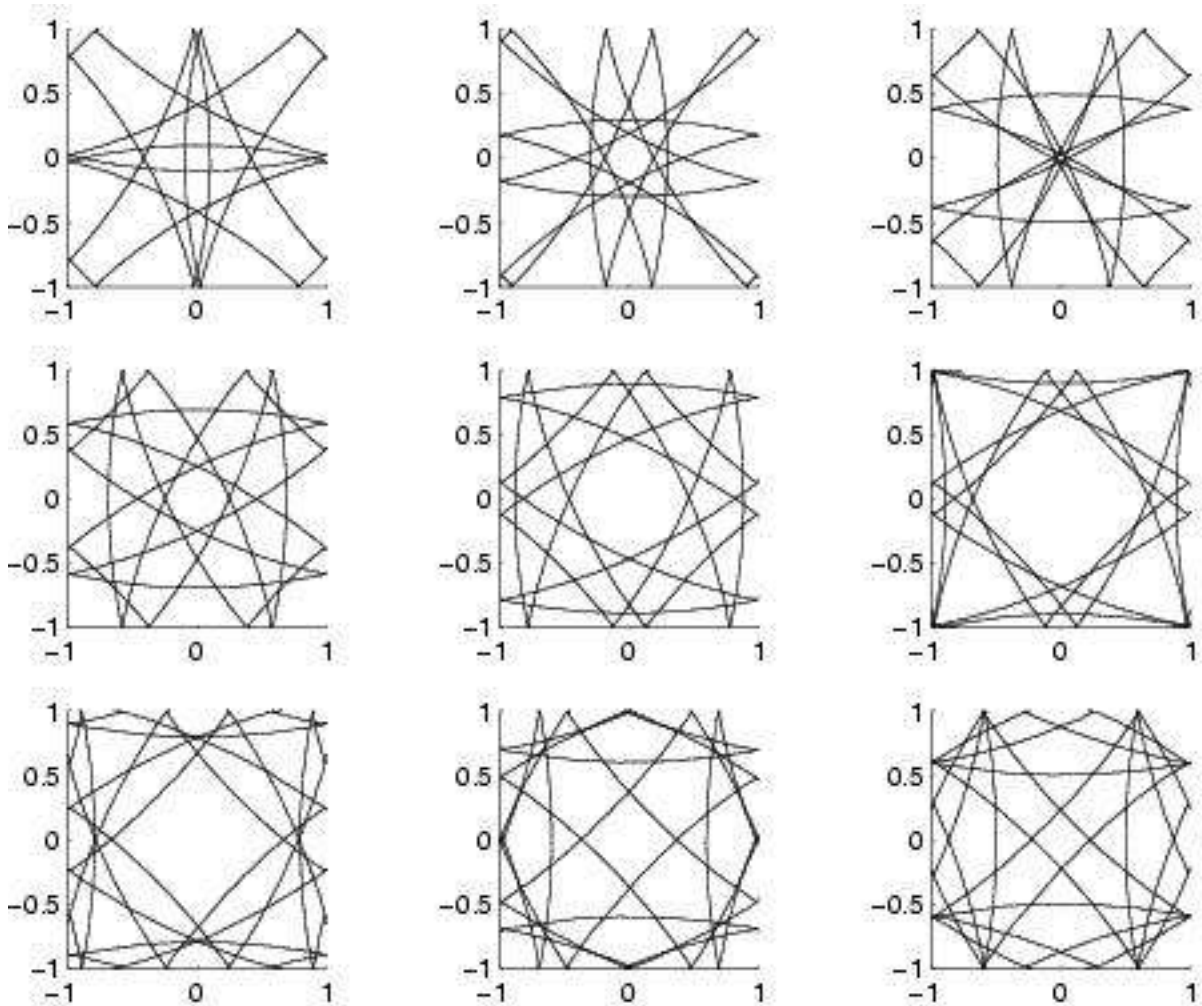

1
0.5
0
-0.5
-1
-1
0
1

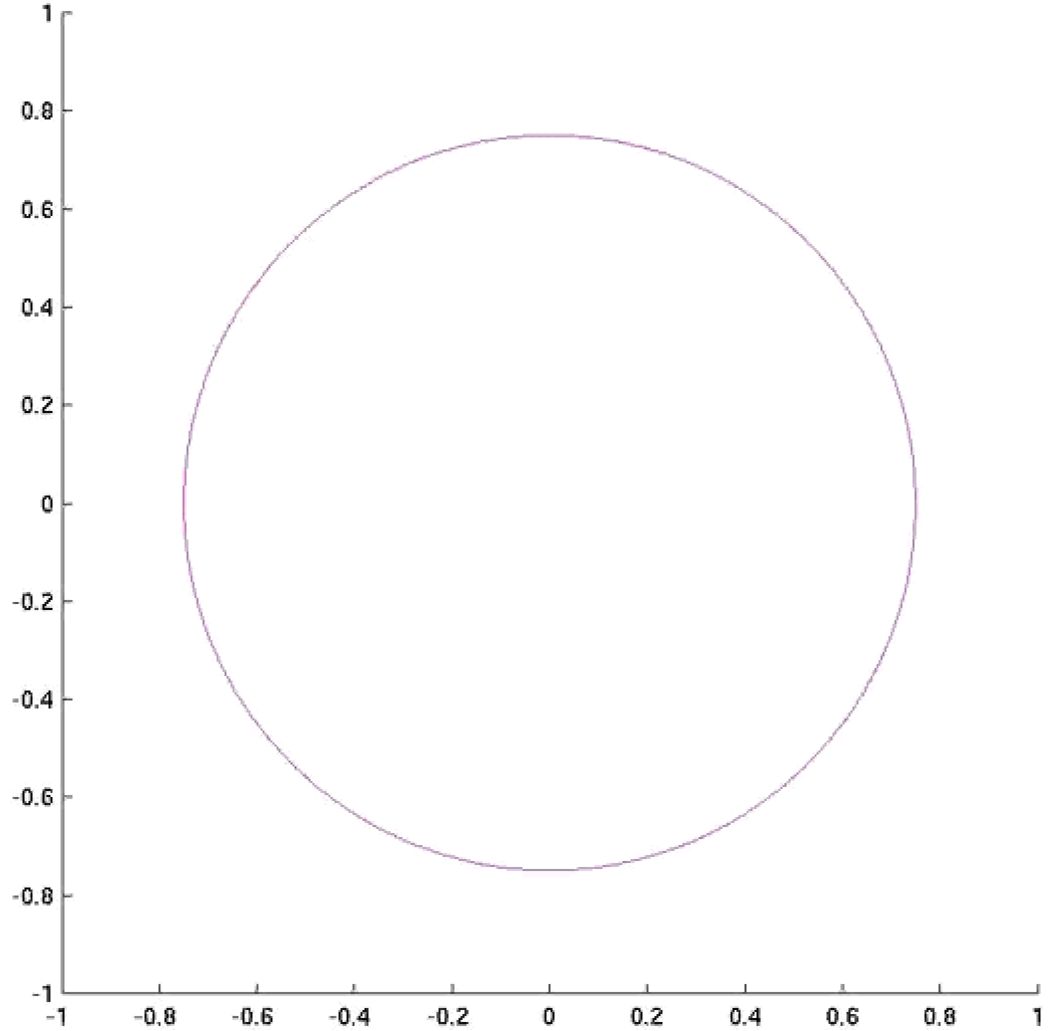

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

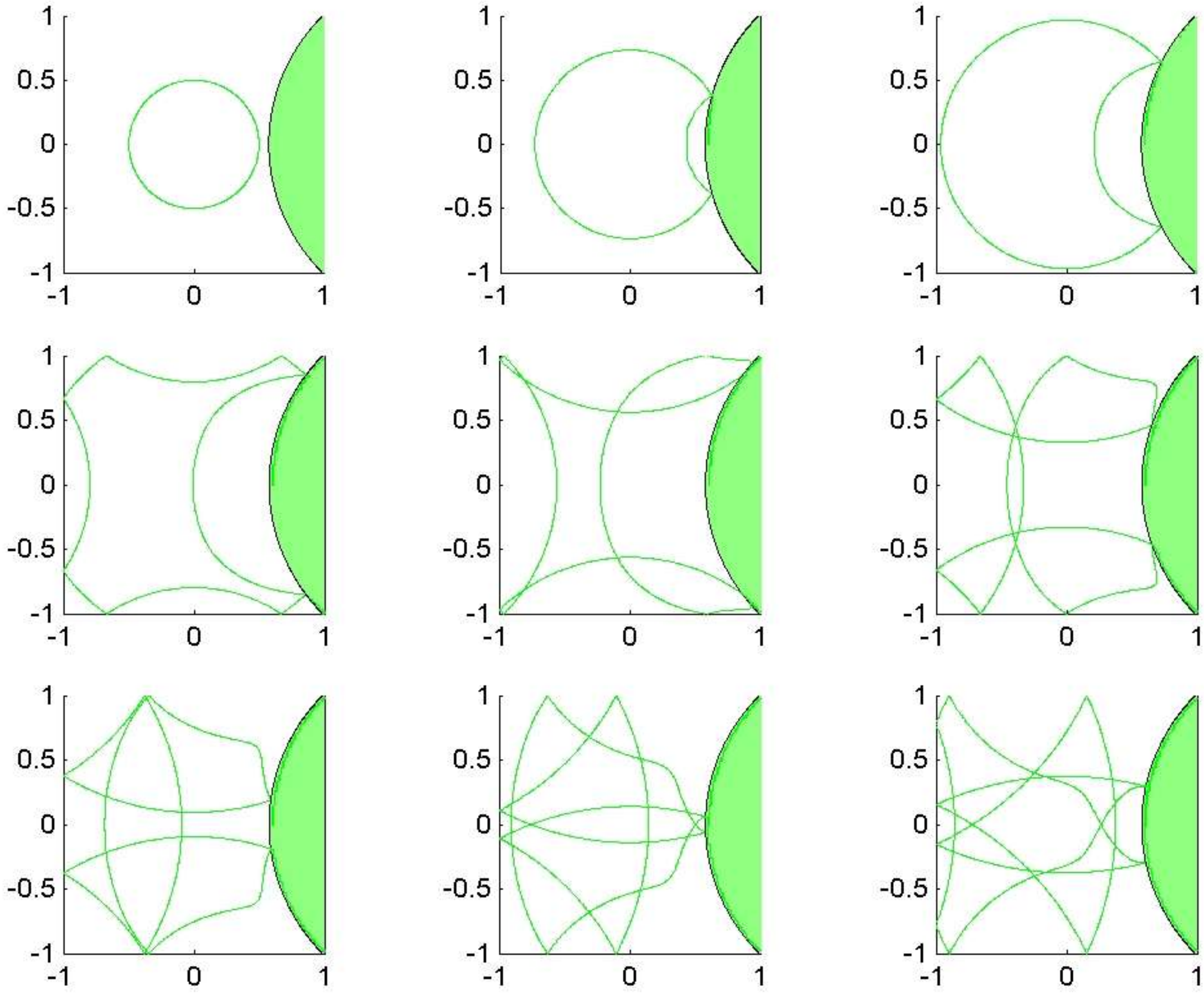

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

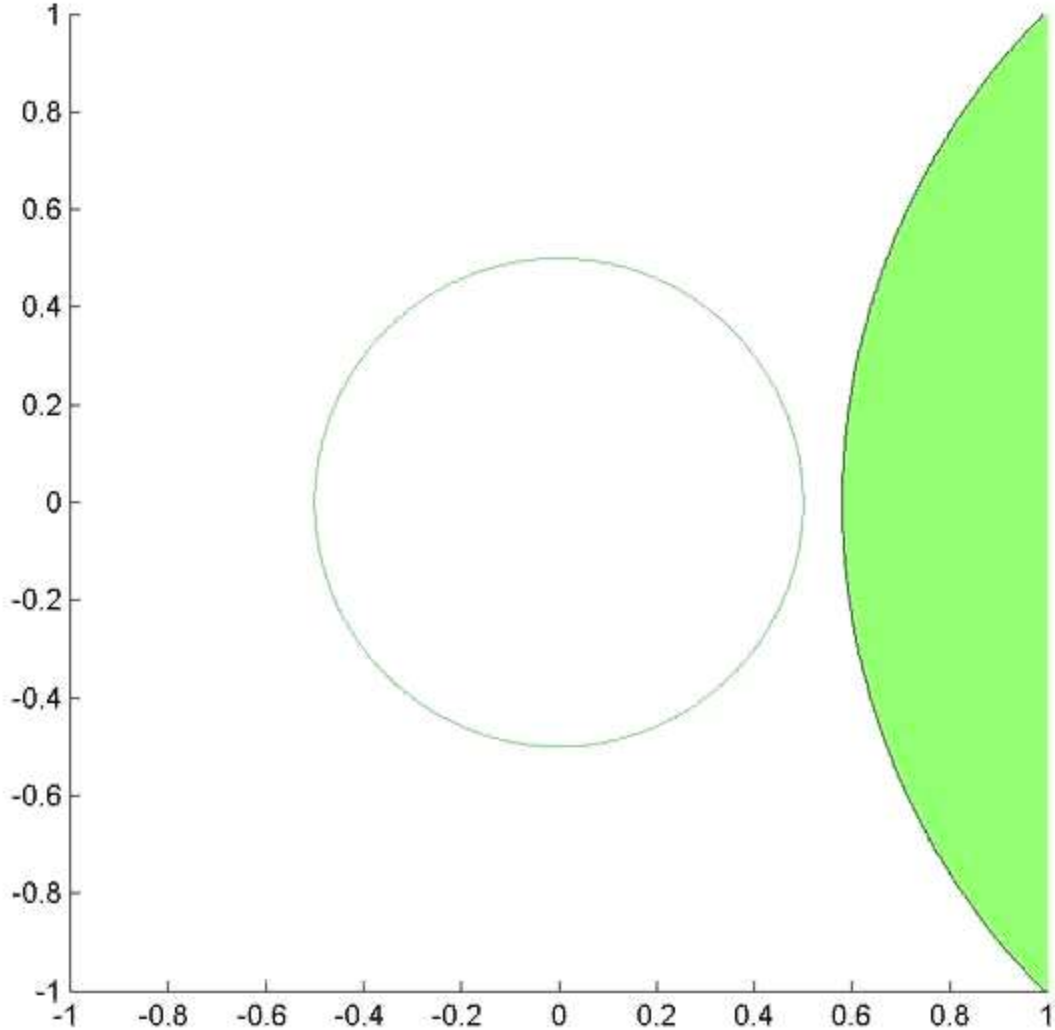

1
0.5
0
−0.5
−1
−1
0
1

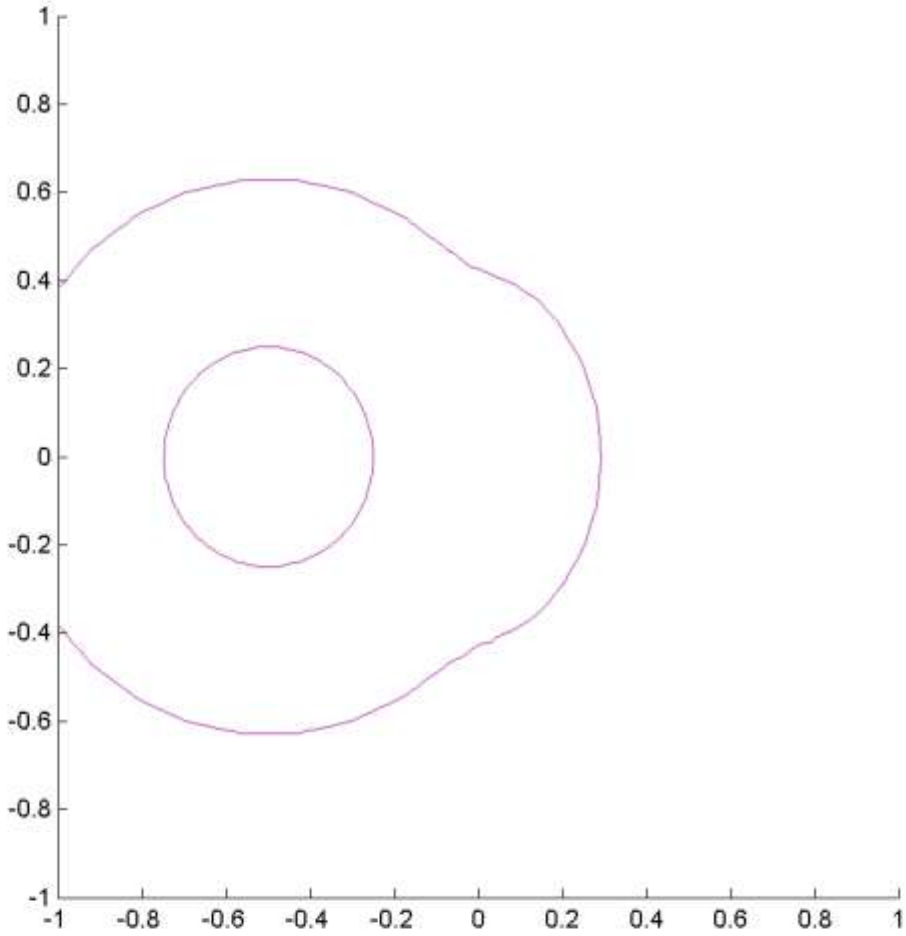

1
0.5
0
−0.5
−1
−1
0
1

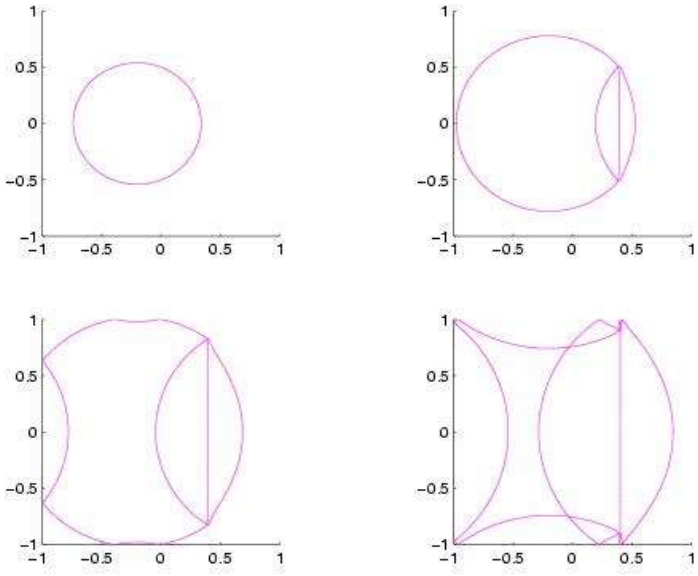

1
0.5
0
-0.5
-1
-1
0
1

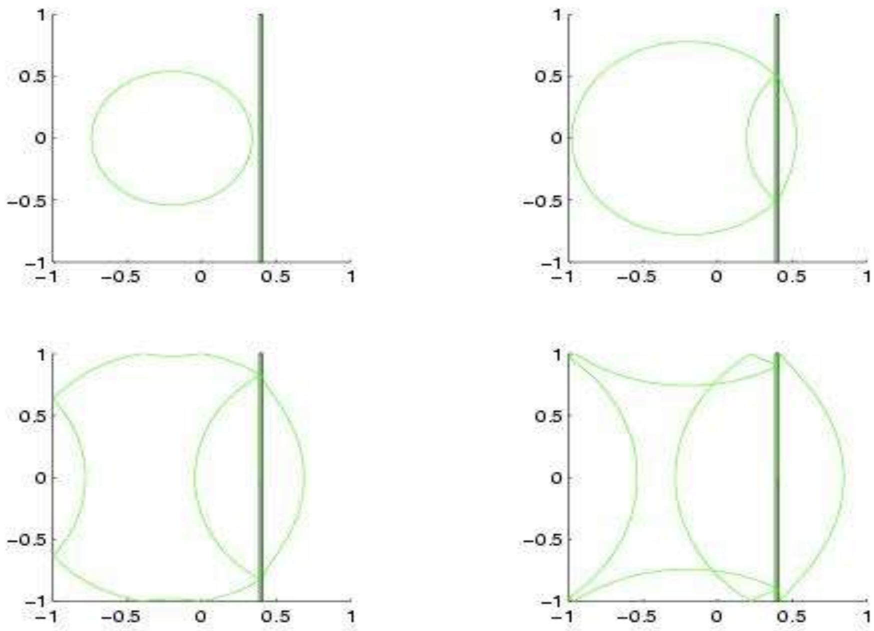

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

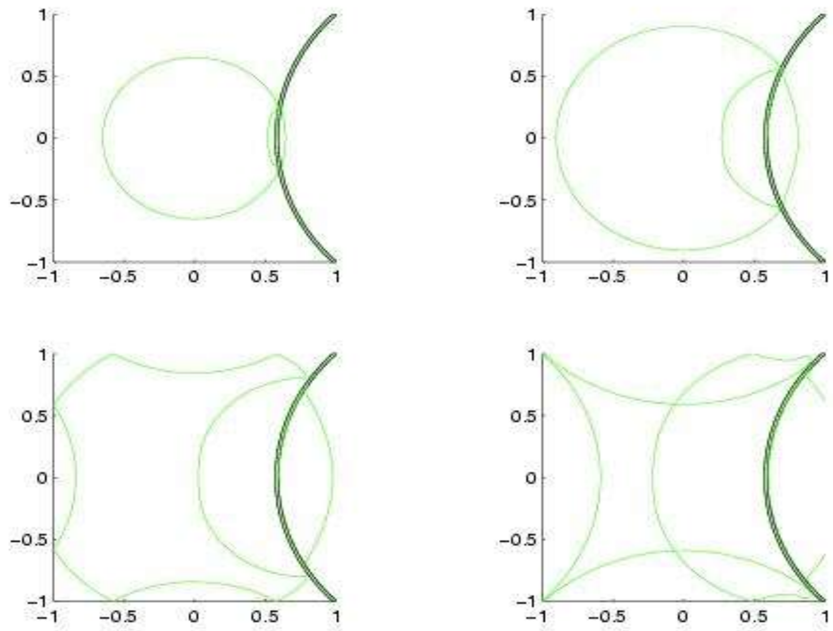

1
0.5
0
-0.5
-1
-1
0
1

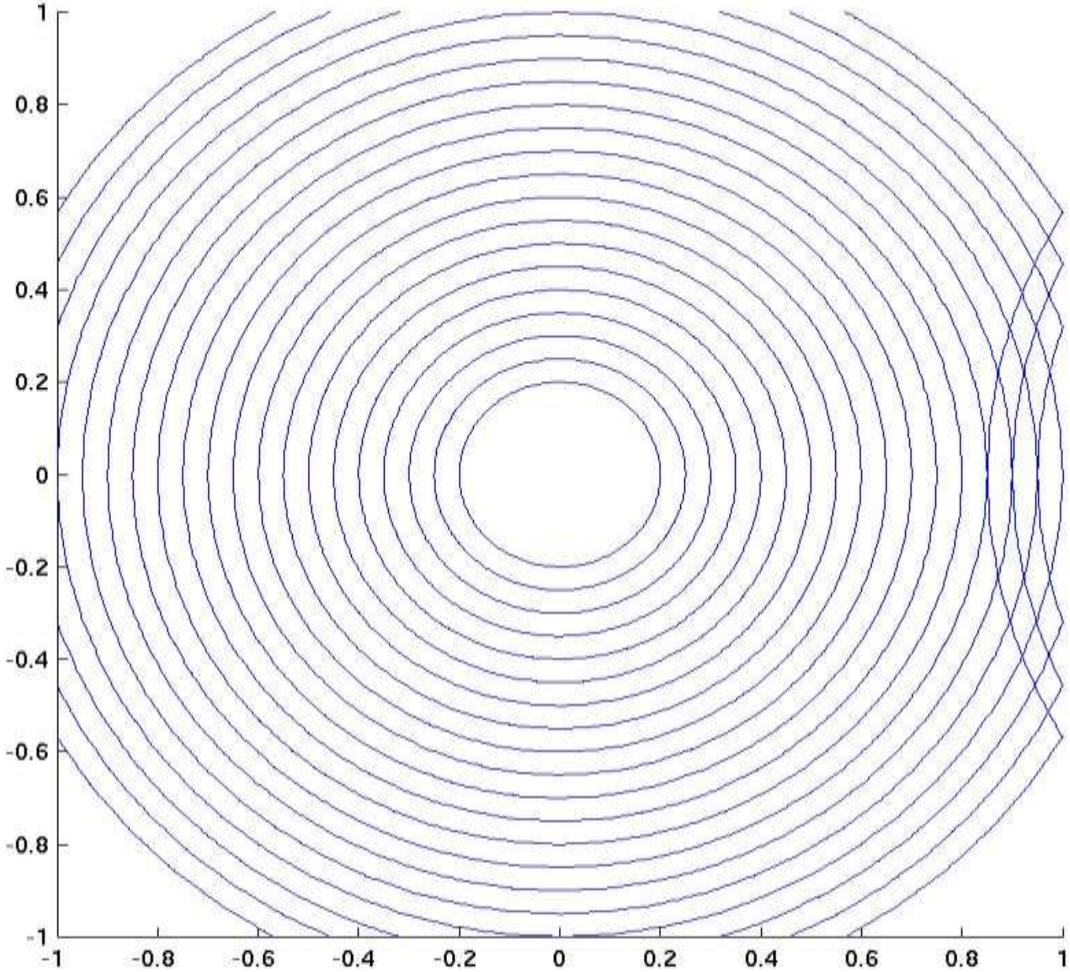

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

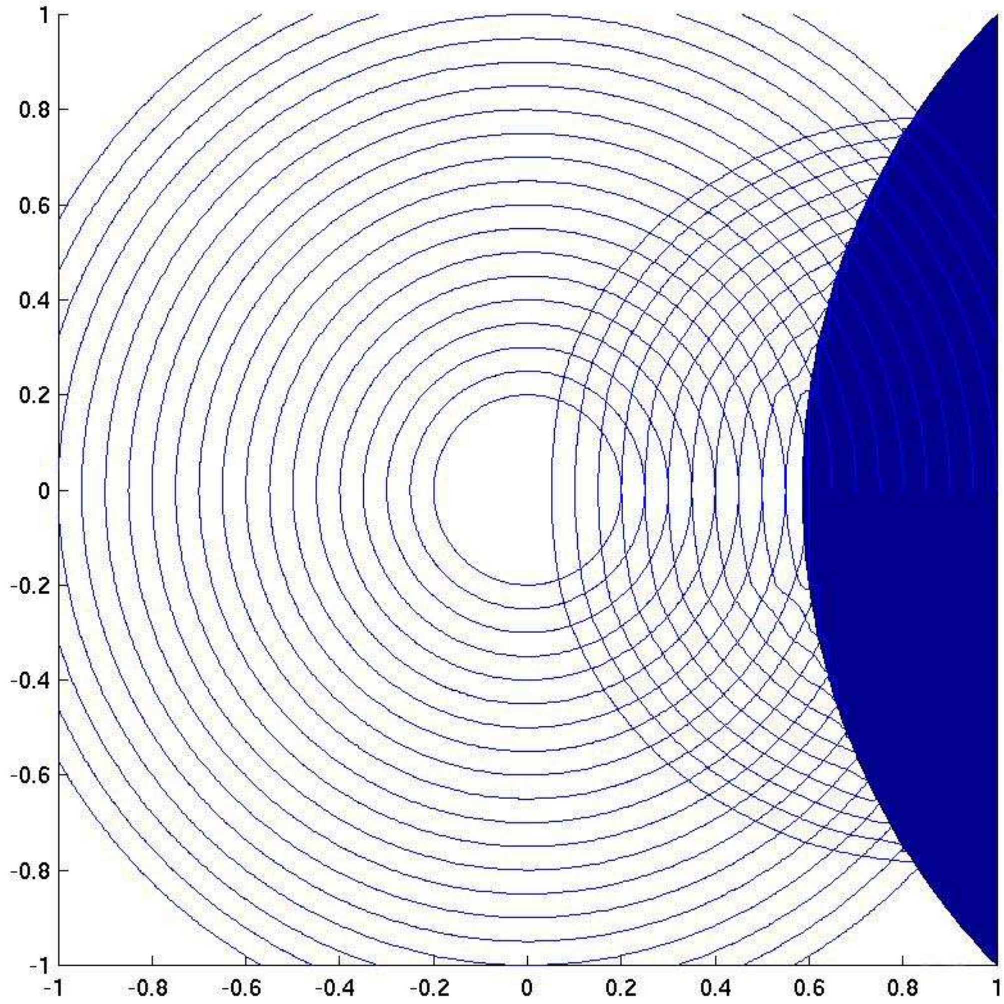

1
0.5
0
−0.5
−1
−1
−0.5
0
0.5
1

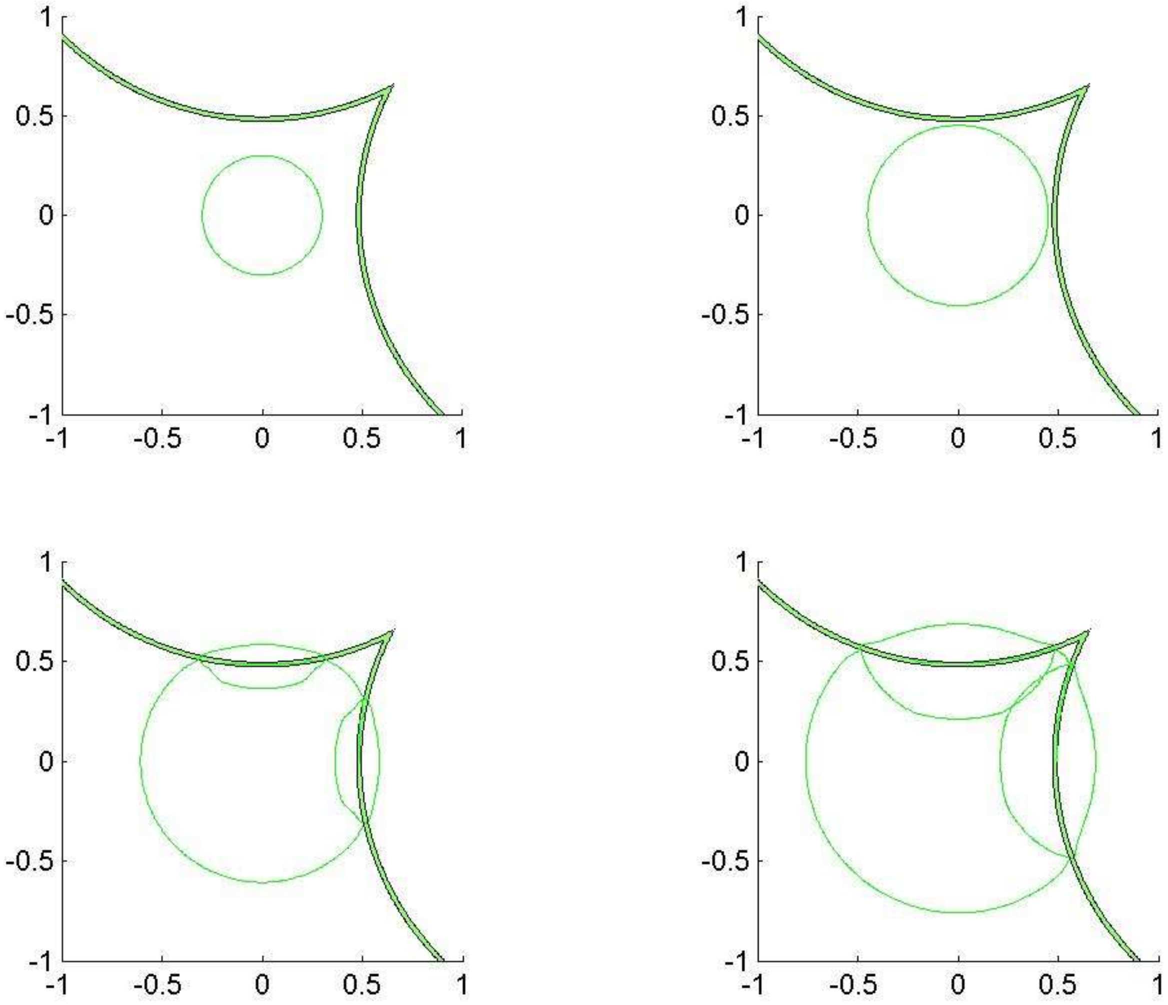

1
0.5
0
−0.5
−1
−1
−0.5
0
0.5
1

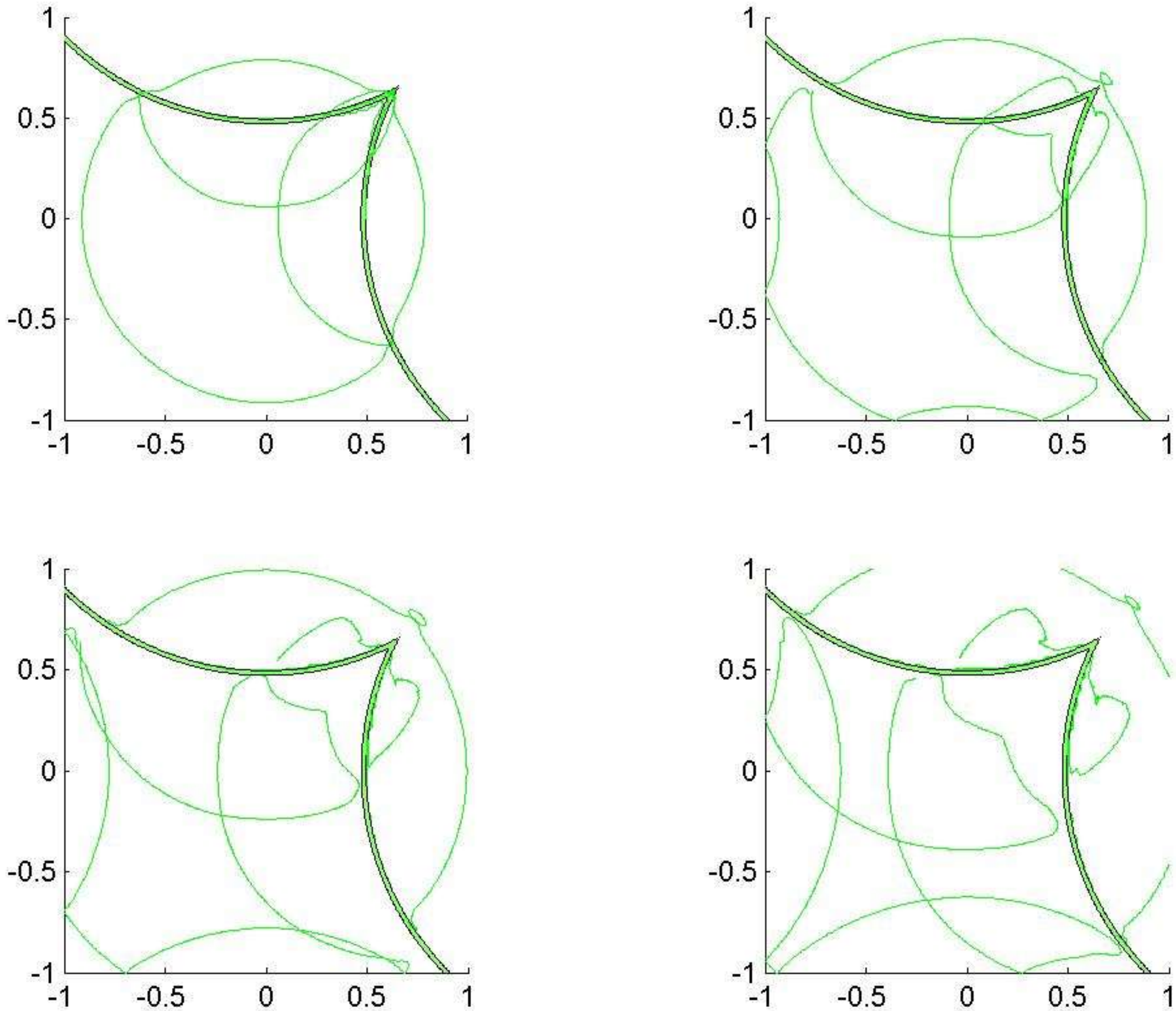

1
0.5
0
−0.5
−1
−1
−0.5
0
0.5
1


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

Level Set Systems, Inc.




1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1


1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

Level Set Systems, Inc.