Running in Circles

26 May 2009

Christoph Maier

Initialization

```
Off[General::spell, General::spell1]

LSolve = Last[Solve[##]] &

Last[Solve[##1]] &

PeS = PowerExpand[Simplify[#]] &
FpPeS = FixedPoint[PeS, #] &

PowerExpand[Simplify[#1]] &

FixedPoint[PeS, #1] &

unitsquare = Function[p, Line[p + #/2 & /@ {{1, 1}, {-1, 1}, {-1, -1}, {1, -1}, {1, 1}}]]

Function[p, Line[(p + #1/2 &) /@ {{1, 1}, {-1, 1}, {-1, -1}, {1, -1}, {1, 1}}]]
```

Basic algorithm

Defining equation for a circle

x is a vector, *r* is the radius (well, a sphere in an Length[*x*]-dimensional vector space, actually)

```
circleeqn = Function[{x, r}, x.x - r^2 == 0]

Function[{x, r}, x.x - r^2 == 0]

circleeqn[{x, y}, R]
```

$-R^2 + x^2 + y^2 == 0$

Iterations

```
circleeqn[{x + 1, y}, R]
```

$-R^2 + (1 + x)^2 + y^2 == 0$

```
circleeqn[#, R]⟦1⟧ & /@ {{x + 1, y}, {x, y}}
Simplify[Subtract @@ %]
```

$\{-R^2 + (1 + x)^2 + y^2, -R^2 + x^2 + y^2\}$

$1 + 2x$

```
Simplify[Subtract @@ (circleeqn[#, R]⟦1⟧ & /@ #)] & /@
 {{{x + 1, y}, {x, y}}, {{x, y - 1}, {x, y}}, {{x + 1, y - 1}, {x, y}}}
```

$\{1 + 2x, 1 - 2y, 2(1 + x - y)\}$

Slope

```
circleeqn[{x, y[x]}, R]
D[%, x]
LSolve[%, y'[x]]
```

$-R^2 + x^2 + y[x]^2 == 0$

$2x + 2y[x]\,y'[x] == 0$

$\left\{y'[x] \to -\frac{x}{y[x]}\right\}$

Different iteration step strategies

We can define a drawing algorithm such that the only possible directions are either horizontal or vertical. In this case, we need to split a full circle into 4 quadrants.
The discrimination criteria between quadrants are the signs of the coordinates: {+, +}, {−, +}, {−, −}, {+, −}.

Alternatively, we can define a drawing algorithm such that the possible directions are horizontal, vertical, or diagonal. In this case, we need to split a full circle into 8 octants.
The discrimination criteria between octants are the signs of the coordinates and the relative magnitude of the partial derivatives of the discriminating function with respect to the coordnates:
$\left\{+, +, \left|\frac{\partial\Delta}{\partial x}\right| < \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{+, +, \left|\frac{\partial\Delta}{\partial x}\right| > \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{-, +, \left|\frac{\partial\Delta}{\partial x}\right| > \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{-, +, \left|\frac{\partial\Delta}{\partial x}\right| < \left|\frac{\partial\Delta}{\partial y}\right|\right\},$
$\left\{-, -, \left|\frac{\partial\Delta}{\partial x}\right| < \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{-, -, \left|\frac{\partial\Delta}{\partial x}\right| > \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{+, -, \left|\frac{\partial\Delta}{\partial x}\right| > \left|\frac{\partial\Delta}{\partial y}\right|\right\}, \left\{+, -, \left|\frac{\partial\Delta}{\partial x}\right| < \left|\frac{\partial\Delta}{\partial y}\right|\right\}$

Orthogonal~~only~~ iteration

Assumption: We iterate in mathematically positive sense of orientation.

Quadrant	δx	δy
{+, +}	−	+
{−, +}	−	−
{−, −}	+	−
{+, −}	+	+

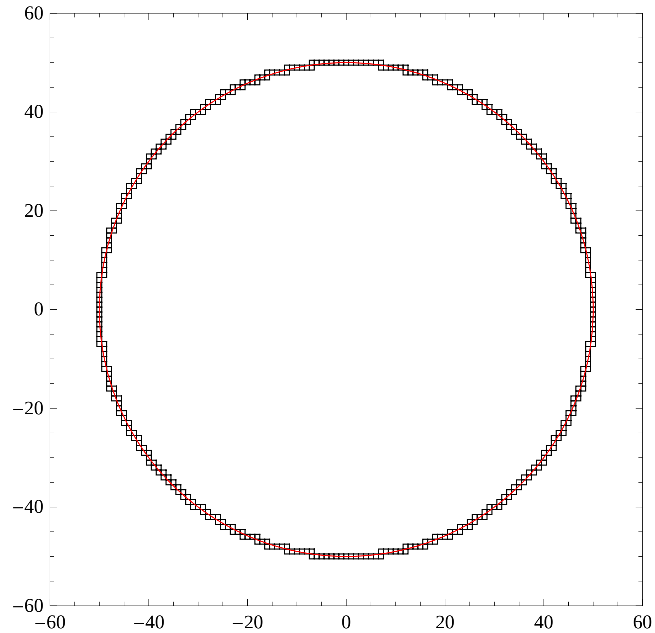
Underlying general rules:
$\text{Sign}[\delta x] = -\text{Sign}[y]$; $\text{Sign}[\delta y] = \text{Sign}[x]$.

We want to choose the alternative that minimizes the error, i.e., gives us min $\{|\Delta + 1 + 2\,x\,\delta x|, |\Delta + 1 + 2\,y\,\delta y|\}$

```
OrthogonalIterate =
 Function[{x, Δ}, Block[{δx1 = If[x⟦2⟧ < 0, 1, -1], δx2 = If[x⟦1⟧ < 0, -1, 1], Δx, Δy},
   Δx1 = 1 + 2 x⟦1⟧ δx1; Δx2 = 1 + 2 x⟦2⟧ δx2;
   If[Abs[Δ + Δx1] < Abs[Δ + Δx2], {x + {δx1, 0}, Δ + Δx1}, {x + {0, δx2}, Δ + Δx2}]]
 ]
```

```
Function[{x, Δ}, Block[{δx1 = If[x⟦2⟧ < 0, 1, -1], δx2 = If[x⟦1⟧ < 0, -1, 1], Δx, Δy},
  Δx1 = 1 + 2 x⟦1⟧ δx1; Δx2 = 1 + 2 x⟦2⟧ δx2;
  If[Abs[Δ + Δx1] < Abs[Δ + Δx2], {x + {δx1, 0}, Δ + Δx1}, {x + {0, δx2}, Δ + Δx2}]]]
```

```
NestList[OrthogonalIterate @@ # &, {{50, 0}, 0}, 399];
Show[
 Graphics[unitsquare[First[#]] & /@ %, Frame → True, AspectRatio → 1,
  PlotRange → 60 {{-1, 1}, {-1, 1}}], Graphics[{Hue[0], Circle[{0, 0}, 50]},
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



■ Diagonal iteration

Assumption: We iterate in mathematically positive sense of orientation.

Octant	Straight	Diagonal
$\left\{+, +, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert > \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{0, ++y\}$	$\{--x, ++y\}$
$\left\{+, +, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert < \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{--x, 0\}$	$\{--x, ++y\}$
$\left\{-, +, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert < \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{--x, 0\}$	$\{--x, --y\}$
$\left\{-, +, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert > \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{0, --y\}$	$\{--x, --y\}$
$\left\{-, -, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert > \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{0, --y\}$	$\{++x, --y\}$
$\left\{-, -, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert < \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{++x, 0\}$	$\{++x, --y\}$
$\left\{+, -, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert < \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{++x, 0\}$	$\{++x, ++y\}$
$\left\{+, -, \left\lvert\frac{\partial\Delta}{\partial x}\right\rvert > \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert\right\}$	$\{0, ++y\}$	$\{++x, ++y\}$

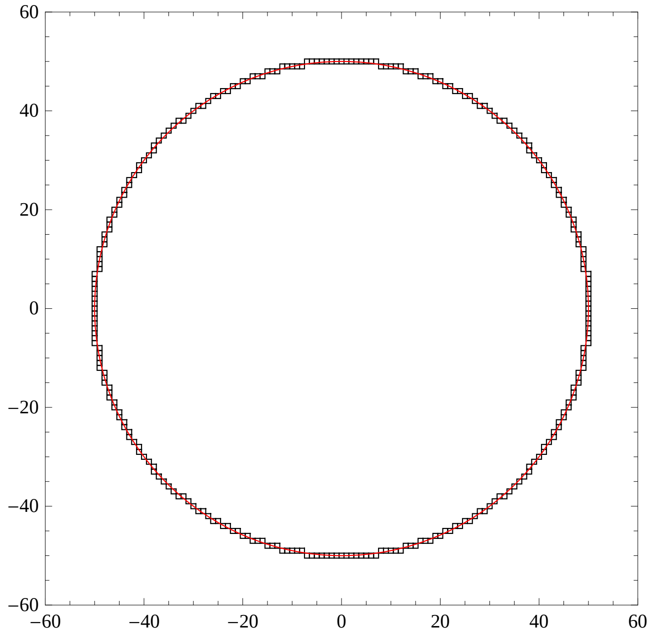
Underlying general rules:
$\mathrm{Sign}[\delta x] = -\mathrm{Sign}[y]$; $\mathrm{Sign}[\delta y] = \mathrm{Sign}[x]$,
straight direction is the direction of the smaller partial derivative.

We want to choose the alternative that minimizes the error.

```
DiagonalIterate =
 Function[{x, Δ}, Block[{δx1 = If[x⟦2⟧ < 0, 1, -1], δx2 = If[x⟦1⟧ < 0, -1, 1], Δx1, Δx2, δs, Δs},
   Δx1 = 1 + 2 x⟦1⟧ δx1; Δx2 = 1 + 2 x⟦2⟧ δx2;
   {δs, Δs} = If[Abs[x⟦1⟧] < Abs[x⟦2⟧], {{δx1, 0}, Δx1}, {{0, δx2}, Δx2}];
   If[Abs[Δ + Δs] < Abs[Δ + Δx1 + Δx2],
    {x + δs, Δ + Δs}, {x + {δx1, δx2}, Δ + Δx1 + Δx2}]
  ]
 ]
```

```
Function[{x, Δ},
 Block[{δx1 = If[x⟦2⟧ < 0, 1, -1], δx2 = If[x⟦1⟧ < 0, -1, 1], Δx1, Δx2, δs, Δs}, Δx1 = 1 + 2 x⟦1⟧ δx1;
  Δx2 = 1 + 2 x⟦2⟧ δx2; {δs, Δs} = If[Abs[x⟦1⟧] < Abs[x⟦2⟧], {{δx1, 0}, Δx1}, {{0, δx2}, Δx2}];
  If[Abs[Δ + Δs] < Abs[Δ + Δx1 + Δx2], {x + δs, Δ + Δs}, {x + {δx1, δx2}, Δ + Δx1 + Δx2}]]]
```

```
NestList[DiagonalIterate @@ # &, {{50, 0}, 0}, 283];
Show[
 Graphics[unitsquare[First[#]] & /@ %, Frame → True, AspectRatio → 1,
  PlotRange → 60 {{-1, 1}, {-1, 1}}], Graphics[{Hue[0], Circle[{0, 0}, 50]},
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



Generalization to ellipses

■ Basic equation

An ellipse with major and minor axes parallel to the coordinate axes has the equation $\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 == 0$.

An ellipse with major axis in the direction $\begin{pmatrix} \xi \\ \upsilon \end{pmatrix}$ has the equation $\frac{\langle x \;\; y | \xi \;\; \upsilon \rangle \langle \xi \;\; \upsilon | x \;\; y \rangle}{a^2 \langle \xi \;\; \upsilon | \xi \;\; \upsilon \rangle} + \frac{\langle x \;\; y | -\upsilon \;\; \xi \rangle \langle -\upsilon \;\; \xi | x \;\; y \rangle}{b^2 \langle \xi \;\; \upsilon | \xi \;\; \upsilon \rangle} - 1 == 0$.

$$\frac{(\{\xi, \upsilon\}.\{x, y\})^2}{a^2 (\{\xi, \upsilon\}.\{\xi, \upsilon\})} + \frac{(\{-\upsilon, \xi\}.\{x, y\})^2}{b^2 (\{\xi, \upsilon\}.\{\xi, \upsilon\})} - 1 == 0$$

```
Simplify[%]
```

$$-1 + \frac{(y\,\xi - x\,\upsilon)^2}{b^2 \left(\xi^2 + \upsilon^2\right)} + \frac{(x\,\xi + y\,\upsilon)^2}{a^2 \left(\xi^2 + \upsilon^2\right)} == 0$$

$$\frac{a^2 \,(y\,\xi - x\,\upsilon)^2 + b^2 \,(x\,\xi + y\,\upsilon)^2}{a^2\, b^2 \left(\xi^2 + \upsilon^2\right)} == 1$$

Equation in matrix form

```
MatrixForm /@ {{x, y}, {{ξ, -υ}, {υ, ξ}}, {{1/a^2, 0}, {0, 1/b^2}}/({ξ, υ}.{ξ, υ}), {{ξ, υ}, {-υ, ξ}}, {x, y}}
```

$$\left\{\begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} \xi & -\upsilon \\ \upsilon & \xi \end{pmatrix}, \begin{pmatrix} \frac{1}{a^2 (\xi^2+\upsilon^2)} & 0 \\ 0 & \frac{1}{b^2 (\xi^2+\upsilon^2)} \end{pmatrix}, \begin{pmatrix} \xi & \upsilon \\ -\upsilon & \xi \end{pmatrix}, \begin{pmatrix} x \\ y \end{pmatrix}\right\}$$

```
Simplify[{x, y}.{{ξ, -υ}, {υ, ξ}}.{{1/a^2, 0}, {0, 1/b^2}}/({ξ, υ}.{ξ, υ}).{{ξ, υ}, {-υ, ξ}}.{x, y} == 1]
```

$$\frac{a^2 (y\,\xi - x\,\upsilon)^2 + b^2 (x\,\xi + y\,\upsilon)^2}{a^2\, b^2 (\xi^2 + \upsilon^2)} == 1$$

The matrices that do not depend on the point coordinates $\begin{pmatrix} x \\ y \end{pmatrix}$ can be collected into one matrix:

```
MatrixForm[Simplify[{{ξ, -υ}, {υ, ξ}}.{{1/a^2, 0}, {0, 1/b^2}}/({ξ, υ}.{ξ, υ}).{{ξ, υ}, {-υ, ξ}}]]
```

$$\begin{pmatrix} \frac{b^2 \xi^2+a^2 \upsilon^2}{a^2 b^2 (\xi^2+\upsilon^2)} & \frac{(-a^2+b^2)\,\xi\,\upsilon}{a^2 b^2 (\xi^2+\upsilon^2)} \\ \frac{(-a^2+b^2)\,\xi\,\upsilon}{a^2 b^2 (\xi^2+\upsilon^2)} & \frac{a^2 \xi^2+b^2 \upsilon^2}{a^2 b^2 (\xi^2+\upsilon^2)} \end{pmatrix}$$

```
Collect[{x, y}.{{A, C}, {C, B}}.{x, y}, {x, y}]
```

$A\,x^2 + 2\,C\,x\,y + B\,y^2$

If we want to avoid all divisions, the equation is

```
(ellipsediscriminant =
   Collect[{x, y}.{{ξ, -υ}, {υ, ξ}}.{{b^2, 0}, {0, a^2}}.{{ξ, υ}, {-υ, ξ}}.{x, y} -
     a^2 b^2 {ξ, υ}.{ξ, υ}, {x, y}, Simplify]) == 0
```

$$2\,(-a^2 + b^2)\,x\,y\,\xi\,\upsilon - a^2\,b^2\,(\xi^2 + \upsilon^2) + x^2\,(b^2\,\xi^2 + a^2\,\upsilon^2) + y^2\,(a^2\,\xi^2 + b^2\,\upsilon^2) == 0$$

```
ellipsediscriminant
```

$$2\,(-a^2 + b^2)\,x\,y\,\xi\,\upsilon - a^2\,b^2\,(\xi^2 + \upsilon^2) + x^2\,(b^2\,\xi^2 + a^2\,\upsilon^2) + y^2\,(a^2\,\xi^2 + b^2\,\upsilon^2)$$

```
LSolve[D[0 == ellipsediscriminant /. y → y[x], x], y'[x]] /. y[x] → y
```

$$\left\{y'[x] \to \frac{-b^2\,x\,\xi^2 + a^2\,y\,\xi\,\upsilon - b^2\,y\,\xi\,\upsilon - a^2\,x\,\upsilon^2}{a^2\,y\,\xi^2 - a^2\,x\,\xi\,\upsilon + b^2\,x\,\xi\,\upsilon + b^2\,y\,\upsilon^2}\right\}$$

```
Collect[D[ellipsediscriminant, #], {x, y}, Simplify] & /@ {x, y}
```

$$\left\{2\,(-a^2 + b^2)\,y\,\xi\,\upsilon + 2\,x\,(b^2\,\xi^2 + a^2\,\upsilon^2),\ 2\,(-a^2 + b^2)\,x\,\xi\,\upsilon + 2\,y\,(a^2\,\xi^2 + b^2\,\upsilon^2)\right\}$$

Constant coefficients that can be calculated in advance

```
Append[Limit[ellipsediscriminant/z /.
   {{x^2 → z, y^2 → 0, x y → 0}, {x^2 → 0, y^2 → z, x y → 0}, {x^2 → 0, y^2 → 0, x y → z}}, z → ∞],
  -ellipsediscriminant /. {x^2 → 0, y^2 → 0, x y → 0}]
MapThread[Rule, {%, {A, B, 2 C, D}}]
ellipsediscriminant /. %
D[%, #] & /@ {x, y}
```

$\{b^2 \xi^2 + a^2 \upsilon^2,\ a^2 \xi^2 + b^2 \upsilon^2,\ 2(-a^2 + b^2)\xi \upsilon,\ a^2 b^2 (\xi^2 + \upsilon^2)\}$

$\{b^2 \xi^2 + a^2 \upsilon^2 \to A,\ a^2 \xi^2 + b^2 \upsilon^2 \to B,\ 2(-a^2 + b^2)\xi \upsilon \to 2C,\ a^2 b^2 (\xi^2 + \upsilon^2) \to D\}$

$-D + A x^2 + 2 C x y + B y^2$

$\{2 A x + 2 C y,\ 2 C x + 2 B y\}$

Iterated variables

The discriminant and its gradient need not be computed from scratch for each new point, but can be iterated, exploiting the fact that for continuous lines, each of the coordinate can only change by 0 or ± 1.

Discriminant

```
A x^2 + 2 C x y + B y^2 - D
TableForm[% /. {{x → x + 1}, {x → x + 1, y → y + 1}, {y → y + 1}, {x → x - 1, y → y + 1},
   {x → x - 1}, {x → x - 1, y → y - 1}, {y → y - 1}, {x → x + 1, y → y - 1}}]
TableForm[Collect[#, {x, y}, Simplify] & /@ (% - %%)]
TableForm[Simplify[{-1, 1, -1}.%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}}]
```

$-D + A x^2 + 2 C x y + B y^2$

$-D + A (1 + x)^2 + 2 C (1 + x) y + B y^2$
$-D + A (1 + x)^2 + 2 C (1 + x) (1 + y) + B (1 + y)^2$
$-D + A x^2 + 2 C x (1 + y) + B (1 + y)^2$
$-D + A (-1 + x)^2 + 2 C (-1 + x) (1 + y) + B (1 + y)^2$
$-D + A (-1 + x)^2 + 2 C (-1 + x) y + B y^2$
$-D + A (-1 + x)^2 + 2 C (-1 + x) (-1 + y) + B (-1 + y)^2$
$-D + A x^2 + 2 C x (-1 + y) + B (-1 + y)^2$
$-D + A (1 + x)^2 + 2 C (1 + x) (-1 + y) + B (-1 + y)^2$

$A + 2 A x + 2 C y$
$A + B + 2 C + 2 (A + C) x + 2 (B + C) y$
$B + 2 C x + 2 B y$
$A + B - 2 C + (-2 A + 2 C) x + 2 (B - C) y$
$A - 2 A x - 2 C y$
$A + B + 2 C - 2 (A + C) x - 2 (B + C) y$
$B - 2 C x - 2 B y$
$A + B - 2 C + 2 (A - C) x + (-2 B + 2 C) y$

$2 C$
$-2 C$
$2 C$
$-2 C$

- **Partial derivative of discriminant w.r.t. *x***

```
2 (A x + C y)
TableForm[% /. {{x → x + 1}, {x → x + 1, y → y + 1}, {y → y + 1}, {x → x - 1, y → y + 1},
   {x → x - 1}, {x → x - 1, y → y - 1}, {y → y - 1}, {x → x + 1, y → y - 1}}]
TableForm[Simplify[% - %%]]
```

```
2 (A x + C y)

2 (A (1 + x) + C y)
2 (A (1 + x) + C (1 + y))
2 (A x + C (1 + y))
2 (A (-1 + x) + C (1 + y))
2 (A (-1 + x) + C y)
2 (A (-1 + x) + C (-1 + y))
2 (A x + C (-1 + y))
2 (A (1 + x) + C (-1 + y))

2 A
2 (A + C)
2 C
-2 A + 2 C
-2 A
-2 (A + C)
-2 C
2 (A - C)
```

- **Partial derivative of discriminant w.r.t. *y***

```
2 (C x + B y)
TableForm[% /. {{x → x + 1}, {x → x + 1, y → y + 1}, {y → y + 1}, {x → x - 1, y → y + 1},
   {x → x - 1}, {x → x - 1, y → y - 1}, {y → y - 1}, {x → x + 1, y → y - 1}}]
TableForm[Simplify[% - %%]]
```

```
2 (C x + B y)

2 (C (1 + x) + B y)
2 (C (1 + x) + B (1 + y))
2 (C x + B (1 + y))
2 (C (-1 + x) + B (1 + y))
2 (C (-1 + x) + B y)
2 (C (-1 + x) + B (-1 + y))
2 (C x + B (-1 + y))
2 (C (1 + x) + B (-1 + y))

2 C
2 (B + C)
2 B
2 (B - C)
-2 C
-2 (B + C)
-2 B
2 (-B + C)
```

Gradient of discriminant

```
2 {A x + C y, C x + B y}
Outer[ReplaceAll, %, {{x → x + 1}, {x → x + 1, y → y + 1}, {y → y + 1}, {x → x - 1, y → y + 1},
   {x → x - 1}, {x → x - 1, y → y - 1}, {y → y - 1}, {x → x + 1, y → y - 1}}, 1];
TableForm[Transpose[%], TableDepth → 1]
TableForm[Transpose[Simplify[%% - %%%]], TableDepth → 1]
TableForm[
 Simplify[{-1, 1, -1}.%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}}, TableDepth → 1]
```

```
{2 (A x + C y), 2 (C x + B y)}

{2 (A (1 + x) + C y), 2 (C (1 + x) + B y)}
{2 (A (1 + x) + C (1 + y)), 2 (C (1 + x) + B (1 + y))}
{2 (A x + C (1 + y)), 2 (C x + B (1 + y))}
{2 (A (-1 + x) + C (1 + y)), 2 (C (-1 + x) + B (1 + y))}
{2 (A (-1 + x) + C y), 2 (C (-1 + x) + B y)}
{2 (A (-1 + x) + C (-1 + y)), 2 (C (-1 + x) + B (-1 + y))}
{2 (A x + C (-1 + y)), 2 (C x + B (-1 + y))}
{2 (A (1 + x) + C (-1 + y)), 2 (C (1 + x) + B (-1 + y))}

{2 A, 2 C}
{2 (A + C), 2 (B + C)}
{2 C, 2 B}
{-2 A + 2 C, 2 (B - C)}
{-2 A, -2 C}
{-2 (A + C), -2 (B + C)}
{-2 C, -2 B}
{2 (A - C), 2 (-B + C)}

{0, 0}
{0, 0}
{0, 0}
{0, 0}
```

Setting up initial conditions for the iterations

Identifying quadrants and octants

Orthogonal~~o~~nly iteration

Diagonal iteration

Assumption: We iterate in mathematically positive sense of orientation (counterclockwise)

Octants:

$\frac{\partial\Delta}{\partial x}$	$\frac{\partial\Delta}{\partial y}$	$\left\lvert\frac{\partial\Delta}{\partial x}\right\rvert - \left\lvert\frac{\partial\Delta}{\partial y}\right\rvert$	Straight	Diagonal
> 0	> 0	> 0	$\{0, +1\}$	$\{-1, +1\}$
> 0	> 0	< 0	$\{-1, 0\}$	$\{-1, +1\}$
< 0	> 0	< 0	$\{-1, 0\}$	$\{-1, -1\}$
< 0	> 0	> 0	$\{0, -1\}$	$\{-1, -1\}$
< 0	< 0	> 0	$\{0, -1\}$	$\{+1, -1\}$
< 0	< 0	< 0	$\{+1, 0\}$	$\{+1, -1\}$
> 0	< 0	< 0	$\{+1, 0\}$	$\{+1, +1\}$
> 0	< 0	> 0	$\{0, +1\}$	$\{+1, +1\}$

Underlying general rules:

$\delta x = -\mathrm{Sign}\left[\frac{\partial\Delta}{\partial y}\right]$, $\delta y = \mathrm{Sign}\left[\frac{\partial\Delta}{\partial x}\right]$, the straight direction is the direction for which the change in Δ is smaller.

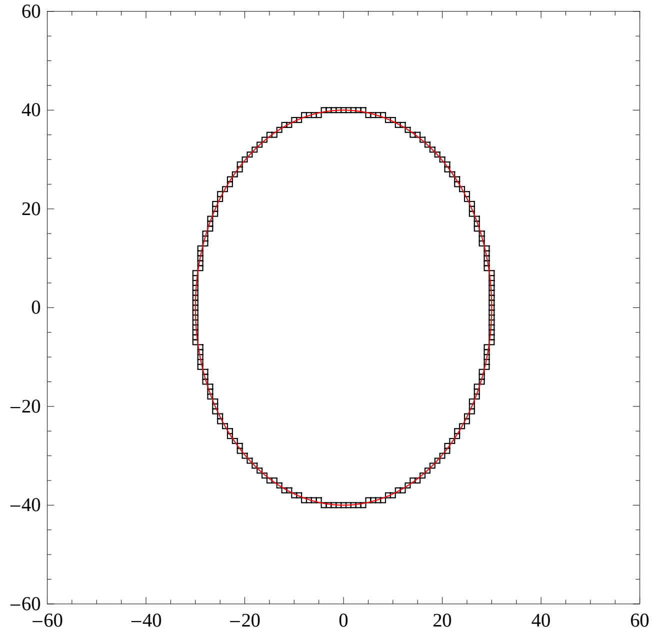
```
EllipseDiagonalIteration = Function[{x, iterationvariables, iterationconstants}, Block[{
     Δ = iterationvariables⟦1⟧,
     dΔ = iterationvariables⟦2⟧,
     A = iterationconstants⟦1⟧,
     B = iterationconstants⟦2⟧,
     C = iterationconstants⟦3⟧,
     δx1, δx2, Δx1, Δx2, δd1, δd2, δs, Δs, δds
    },
    δx1 = If[dΔ⟦2⟧ < 0, 1, -1]; δx2 = If[dΔ⟦1⟧ < 0, -1, 1];
    Δx1 = δx1 dΔ⟦1⟧; Δx2 = δx2 dΔ⟦2⟧;
    δd1 = 2 δx1 {A, C}; δd2 = 2 δx2 {C, B};
    {δs, Δs, δds} = If[Subtract @@ (Abs[dΔ]) < 0,
      {{δx1, 0}, A + Δx1, δd1}, {{0, δx2}, B + Δx2, δd2}];
    If[Abs[Δ + Δs] < Abs[Δ + A + B + 2 Sign[δx1] Sign[δx2] C + Δx1 + Δx2],
     {x + δs, {Δ + Δs, dΔ + δds}, iterationconstants},
     {x + {δx1, δx2},
      {Δ + A + B + 2 Sign[δx1] Sign[δx2] C + Δx1 + Δx2, dΔ + δd1 + δd2}, iterationconstants}]
  ]
 ]
```

```
Function[{x, iterationvariables, iterationconstants},
 Block[{Δ = iterationvariables⟦1⟧, dΔ = iterationvariables⟦2⟧, A = iterationconstants⟦1⟧,
    B = iterationconstants⟦2⟧, C = iterationconstants⟦3⟧, δx1, δx2, Δx1, Δx2,
    δd1, δd2, δs, Δs, δds}, δx1 = If[dΔ⟦2⟧ < 0, 1, -1]; δx2 = If[dΔ⟦1⟧ < 0, -1, 1];
   Δx1 = δx1 dΔ⟦1⟧; Δx2 = δx2 dΔ⟦2⟧; δd1 = 2 δx1 {A, C}; δd2 = 2 δx2 {C, B};
   {δs, Δs, δds} = If[Subtract @@ Abs[dΔ] < 0, {{δx1, 0}, A + Δx1, δd1}, {{0, δx2}, B + Δx2, δd2}];
   If[Abs[Δ + Δs] < Abs[Δ + A + B + 2 Sign[δx1] Sign[δx2] C + Δx1 + Δx2],
    {x + δs, {Δ + Δs, dΔ + δds}, iterationconstants}, {x + {δx1, δx2},
     {Δ + A + B + 2 Sign[δx1] Sign[δx2] C + Δx1 + Δx2, dΔ + δd1 + δd2}, iterationconstants}]]]
```

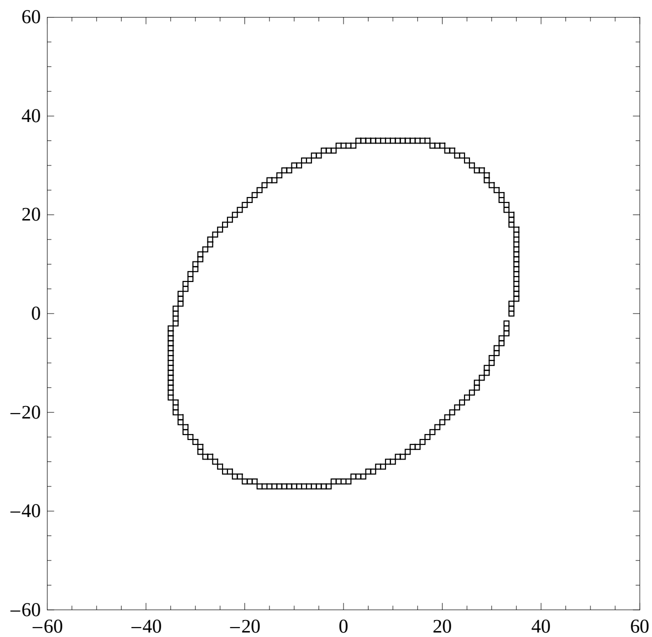
```
ellipsefunction[{30, 0}, {1, 0}, {30, 40}]
```

```
{{30, 0}, {0, {96 000, 0}}, {1600, 900, 0}}
```

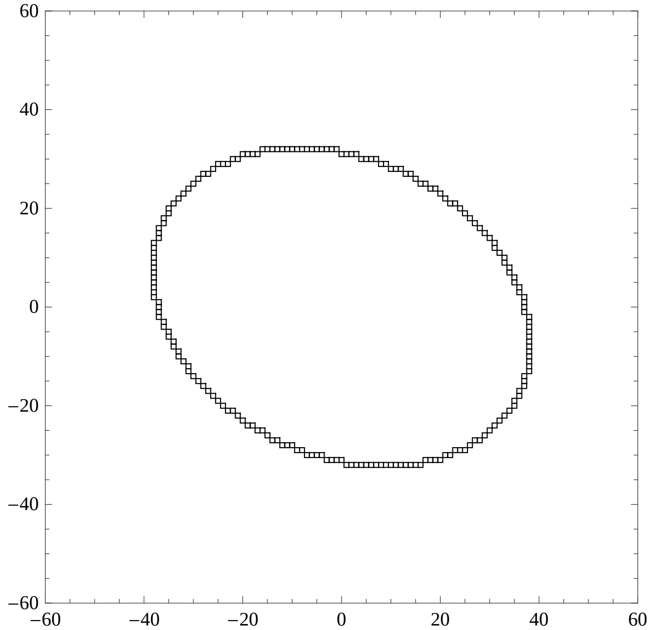
```
NestList[EllipseDiagonalIteration @@ # &, ellipsefunction[{30, 0}, {1, 0}, {30, 40}], 199];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}],
 Graphics[{Hue[0], Circle[{0, 0}, {30, 40}]}, Frame → True, AspectRatio → 1,
  PlotRange → 60 {{-1, 1}, {-1, 1}}], DisplayFunction → $DisplayFunction]
```



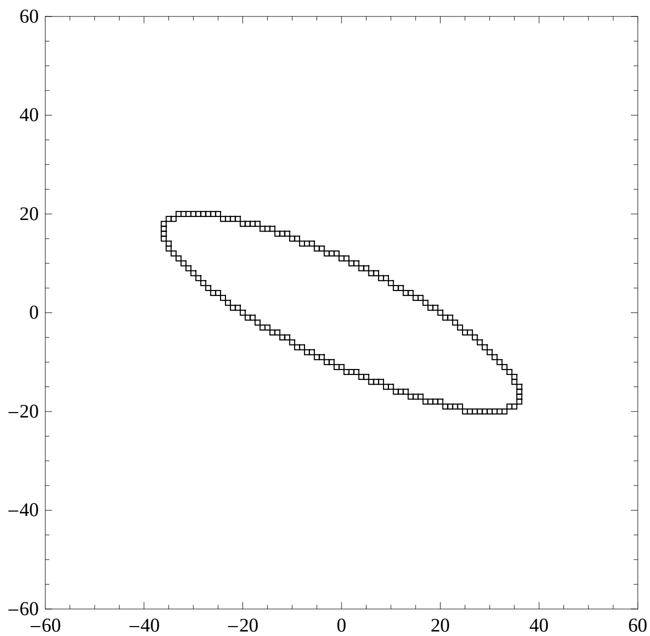

```
NestList[EllipseDiagonalIteration @@ # &, ellipsefunction[{34, 0}, {1, 1}, {40, 30}], 196];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



```
NestList[EllipseDiagonalIteration @@ # &, ellipsefunction[{37, 0}, {-2, 1}, {40, 30}], 197];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```




```
NestList[EllipseDiagonalIteration @@ # &, ellipsefunction[{20, 0}, {-2, 1}, {40, 10}], 151];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



Starting and stopping conditions

Finding a parametrization that will set the initial point onto an ellipse arc to be drawn, and that will tell the ellipse iteration program when to stop drawing, is not entirely straightforward.
The most practical approach seems to be to define the initial and final tangent directions (because we happen to keep track of the partial derivatives of the discriminant Δ, anyway). Since every tangent direction occurs twice on the ellipse arc, we need to keep track of a direction, too, to resolve the ambiguity.

The tangent direction $\begin{pmatrix} t \\ u \end{pmatrix}$ is perpendicular to the gradient of the discriminant, i.e., $0 = \left\langle \begin{matrix} t & u \end{matrix} \,\middle|\, \begin{matrix} \frac{\partial\Delta}{\partial x} & \frac{\partial\Delta}{\partial y} \end{matrix} \right\rangle = t\,\frac{\partial\Delta}{\partial x} + u\,\frac{\partial\Delta}{\partial y}$.

For the orientation of the direction, we require that initially, $0 < \left\langle \begin{matrix} t & u \end{matrix} \,\middle|\, \begin{matrix} 0 & 1 \\ -1 & 0 \end{matrix} \,\middle|\, \begin{matrix} \frac{\partial\Delta}{\partial x} & \frac{\partial\Delta}{\partial y} \end{matrix} \right\rangle = t\,\frac{\partial\Delta}{\partial y} - u\,\frac{\partial\Delta}{\partial x}$.

Stopping condition

Initially, the dot product $\left\langle \begin{matrix} t & u \end{matrix} \,\middle|\, \begin{matrix} \frac{\partial\Delta}{\partial x} & \frac{\partial\Delta}{\partial y} \end{matrix} \right\rangle$ will go positive, i.e., the transition we are looking for as termination condition is a transition from negative to positive.
If we want to be able to use the same tangent vector as starting and stopping condition, we must start testing for the stopping condition *after* the first iteration.

Starting condition

Problem:
Find the point on a discrete grid closest to the point that satisfies
$0 == A\,x^2 + B\,y^2 + 2\,C\,x\,y - D$ subject to $0 == t(A\,x + C\,y) + u(C\,x + B\,y)$ and $0 < t(C\,x + B\,y) - u(A\,x + C\,y)$.

Note the negaive sign, such that $D > 0$, contrary to the definition further above.

```
FpPeS[FullSimplify[Solve[{0 == A x^2 + B y^2 + 2 C x y - D, 0 == t (A x + C y) + u (C x + B y)}, {x, y}]]]
```

$$\left\{\left\{x \to -\frac{\sqrt{D}\,(C\,t + B\,u)}{\sqrt{A\,B - C^2}\,\sqrt{A\,t^2 + u\,(2\,C\,t + B\,u)}},\ y \to \frac{\sqrt{D}\,(A\,t + C\,u)}{\sqrt{A\,B - C^2}\,\sqrt{A\,t^2 + u\,(2\,C\,t + B\,u)}}\right\},\right.$$
$$\left.\left\{x \to \frac{\sqrt{D}\,(C\,t + B\,u)}{\sqrt{A\,B - C^2}\,\sqrt{A\,t^2 + u\,(2\,C\,t + B\,u)}},\ y \to -\frac{\sqrt{D}\,(A\,t + C\,u)}{\sqrt{A\,B - C^2}\,\sqrt{A\,t^2 + u\,(2\,C\,t + B\,u)}}\right\}\right\}$$

$$\sqrt{\frac{D}{(A\,B - C^2)(A\,t^2 + B\,u^2 + 2\,C\,t\,u)}}\begin{pmatrix} \pm(C\,t + B\,u) \\ \mp(A\,t + C\,u) \end{pmatrix}$$

$$\{A \to b^2\,\xi^2 + a^2\,\upsilon^2,\ B \to a^2\,\xi^2 + b^2\,\upsilon^2,\ C \to (-a^2 + b^2)\,\xi\,\upsilon,\ D \to a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

$$\text{Simplify}\left[\sqrt{\frac{D}{(A\,B - C^2)\,(A\,t^2 + B\,u^2 + 2\,C\,t\,u)}}\ \text{/. \%}\right]$$

```
Simplify[{C t + B u, A t + C u} /. %%]
```

$$\{A \to b^2\,\xi^2 + a^2\,\upsilon^2,\ B \to a^2\,\xi^2 + b^2\,\upsilon^2,\ C \to (-a^2 + b^2)\,\xi\,\upsilon,\ D \to a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

$$\sqrt{\frac{1}{(\xi^2 + \upsilon^2)\,\left(a^2\,(u\,\xi - t\,\upsilon)^2 + b^2\,(t\,\xi + u\,\upsilon)^2\right)}}$$

$$\{a^2\,\xi\,(u\,\xi - t\,\upsilon) + b^2\,\upsilon\,(t\,\xi + u\,\upsilon),\ a^2\,\upsilon\,(-u\,\xi + t\,\upsilon) + b^2\,\xi\,(t\,\xi + u\,\upsilon)\}$$

$$\text{Collect}\left[\left(a^2\,(u\,\xi - t\,\upsilon)^2 + b^2\,(t\,\xi + u\,\upsilon)^2\right),\ \{u,\ t\}\right]$$

$$t\,u\,(-2\,a^2\,\xi\,\upsilon + 2\,b^2\,\xi\,\upsilon) + t^2\,(b^2\,\xi^2 + a^2\,\upsilon^2) + u^2\,(a^2\,\xi^2 + b^2\,\upsilon^2)$$

$$\{A \to b^2\,\xi^2 + a^2\,\upsilon^2,\ B \to a^2\,\xi^2 + b^2\,\upsilon^2,\ C \to (-a^2 + b^2)\,\xi\,\upsilon,\ D \to a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

$$\text{Simplify}\left[\sqrt{\frac{D}{A\,B - C^2}}\ \text{/. \%}\right]$$

$$\{A \to b^2\,\xi^2 + a^2\,\upsilon^2,\ B \to a^2\,\xi^2 + b^2\,\upsilon^2,\ C \to (-a^2 + b^2)\,\xi\,\upsilon,\ D \to a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

$$\sqrt{\frac{1}{\xi^2 + \upsilon^2}}$$

$$\frac{1}{\sqrt{(\xi^2+\upsilon^2)\,(A\,t^2+B\,u^2+2\,C\,t\,u)}}\begin{pmatrix}\pm(C\,t + B\,u)\\ \mp(A\,t + C\,u)\end{pmatrix}$$

Numeric range considerations

We need to make sure that the quantities used in the computations do not overflow.

```
ellipsediscriminant
```

$$2\,(-a^2 + b^2)\,x\,y\,\xi\,\upsilon - a^2\,b^2\,(\xi^2 + \upsilon^2) + x^2\,(b^2\,\xi^2 + a^2\,\upsilon^2) + y^2\,(a^2\,\xi^2 + b^2\,\upsilon^2)$$

$$\text{ellipsematrixeqns} = \{A == b^2\,\xi^2 + a^2\,\upsilon^2,\ B == a^2\,\xi^2 + b^2\,\upsilon^2,\ C == (-a^2 + b^2)\,\xi\,\upsilon,\ D == a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

$$\{A == b^2\,\xi^2 + a^2\,\upsilon^2,\ B == a^2\,\xi^2 + b^2\,\upsilon^2,\ C == (-a^2 + b^2)\,\xi\,\upsilon,\ D == a^2\,b^2\,(\xi^2 + \upsilon^2)\}$$

```
Collect[D[ellipsediscriminant, #], {x, y}, Simplify] & /@ {x, y}
D[%, #] & /@ {x, y}
```

$$\{2\,(-a^2 + b^2)\,y\,\xi\,\upsilon + 2\,x\,(b^2\,\xi^2 + a^2\,\upsilon^2),\ 2\,(-a^2 + b^2)\,x\,\xi\,\upsilon + 2\,y\,(a^2\,\xi^2 + b^2\,\upsilon^2)\}$$

$$\{\{2\,(b^2\,\xi^2 + a^2\,\upsilon^2),\ 2\,(-a^2 + b^2)\,\xi\,\upsilon\},\ \{2\,(-a^2 + b^2)\,\xi\,\upsilon,\ 2\,(a^2\,\xi^2 + b^2\,\upsilon^2)\}\}$$

```
{C t + B u, A t + C u} /. LSolve[ellipsematrixeqns, {A, B, C, D}]
```

$$\{-(a^2 - b^2)\,t\,\xi\,\upsilon + u\,(a^2\,\xi^2 + b^2\,\upsilon^2),\ -(a^2 - b^2)\,u\,\xi\,\upsilon + t\,(b^2\,\xi^2 + a^2\,\upsilon^2)\}$$

```
A t^2 + B u^2 + 2 C t u /. LSolve[ellipsematrixeqns, {A, B, C, D}]
```

$-2\,(a^2-b^2)\,t\,u\,\xi\,\upsilon + t^2\,(b^2\,\xi^2 + a^2\,\upsilon^2) + u^2\,(a^2\,\xi^2 + b^2\,\upsilon^2)$

In the final iterations, we must scale the matrix elements A, B, C, and D such that the iteration states can't overflow.

- **Using normalization to alleviate dynamic range and roundoff problems**

To help restricting the numerical range of quantities, it helps to normalize the vectors describing the principal axis $\begin{pmatrix}\xi\\ \upsilon\end{pmatrix}$ and the tangent direction $\begin{pmatrix}t\\ u\end{pmatrix}$ to unit magnitude.

```
ellipsematrixeqns /. {ξ^2 + υ^2 → 1}
```

$\{A == b^2\,\xi^2 + a^2\,\upsilon^2,\ B == a^2\,\xi^2 + b^2\,\upsilon^2,\ C == (-a^2 + b^2)\,\xi\,\upsilon,\ D == a^2\,b^2\}$

■ Implementation of ellipse drawing with starting and stopping conditions

This version uses $\{x, \Xi, ab\}$, where
x is the 2-dimensional initial coordinate vector
Ξ is the 2-dimensional direction of the first principal axis
ab is the 2-dimensional vector of principal axis lengths

```
ellipsebegin = Function[{T, Ξ, ab}, Block[{τ = T, ξ = Ξ, A, B, C, D, x, Δ, dΔ},
   A = ab[[2]]^2 ξ[[1]]^2 + ab[[1]]^2 ξ[[2]]^2; B = ab[[1]]^2 ξ[[1]]^2 + ab[[2]]^2 ξ[[2]]^2; C = (ab[[2]]^2 - ab[[1]]^2) ξ[[1]] ξ[[2]];
   D = (ab[[1]] ab[[2]])^2 (ξ.ξ); x = Round[{{0, 1}, {-1, 0}}.{{A, C}, {C, B}}.τ / Sqrt[(ξ.ξ) (τ.{{A, C}, {C, B}}.τ)]];
   {x, {Δ = A x[[1]]^2 + B x[[2]]^2 + 2 C x[[1]] x[[2]] - D, dΔ = 2 {A x[[1]] + C x[[2]], C x[[1]] + B x[[2]]}}, {A, B, C}}]]
```

```
Function[{T, Ξ, ab}, Block[{τ = T, ξ = Ξ, A, B, C, D, x, Δ, dΔ},
  A = ab[[2]]^2 ξ[[1]]^2 + ab[[1]]^2 ξ[[2]]^2; B = ab[[1]]^2 ξ[[1]]^2 + ab[[2]]^2 ξ[[2]]^2; C = (ab[[2]]^2 - ab[[1]]^2) ξ[[1]] ξ[[2]];
  D = (ab[[1]] ab[[2]])^2 ξ.ξ; x = Round[{{0, 1}, {-1, 0}}.{{A, C}, {C, B}}.τ / Sqrt[ξ.ξ τ.{{A, C}, {C, B}}.τ]];
  {x, {Δ = A x[[1]]^2 + B x[[2]]^2 + 2 C x[[1]] x[[2]] - D, dΔ = 2 {A x[[1]] + C x[[2]], C x[[1]] + B x[[2]]}}, {A, B, C}}]]
```

This version iterates $\{x$, iterationvariables, iterationconstants$\}$, where
x is the 2-dimensional coordinate vector
iterationvariables $\equiv \{\Delta, d\Delta\}$, where
Δ is the discriminant
$d\Delta \equiv \left\{\frac{\partial\Delta}{\partial x}, \frac{\partial\Delta}{\partial y}\right\}$
iterationconstants are the matrix elements $\{A, B, C\}$

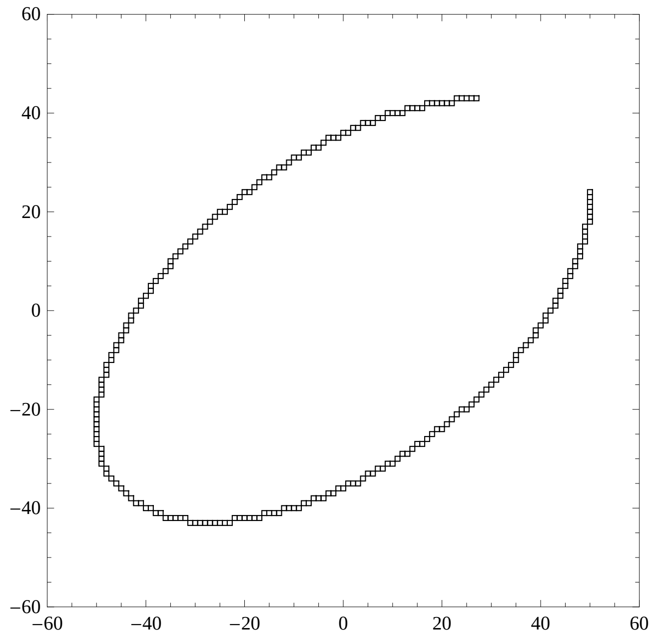
Now put everything together

```
EllipseDraw = Function[{τ1, τ2, Ξ, ab},
  NestWhileList[EllipseDiagonalIteration @@ # &,
   ellipsebegin[τ1, Ξ, ab], ! ((τ2.#1[[2, 2]] < 0) && (τ2.#2[[2, 2]] ≥ 0)) &, 2]
 ]
```

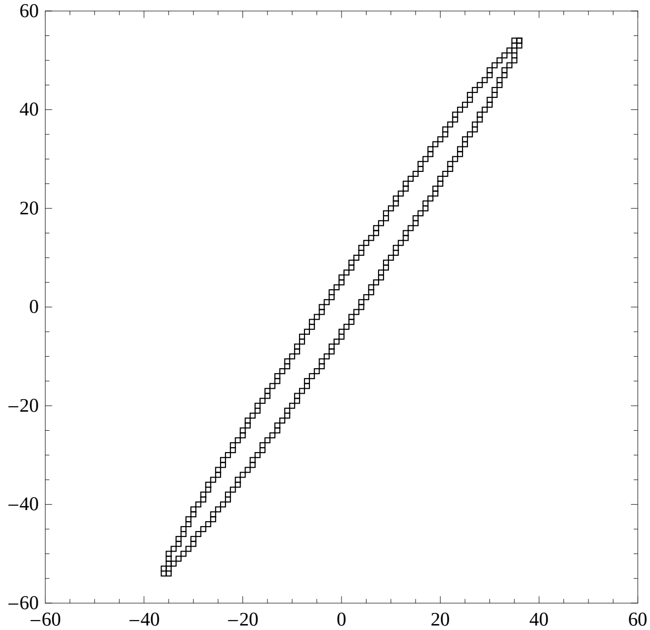
```
Function[{τ1, τ2, Ξ, ab}, NestWhileList[EllipseDiagonalIteration @@ #1 &,
  ellipsebegin[τ1, Ξ, ab], ! (τ2.#1[[2, 2]] < 0 && τ2.#2[[2, 2]] ≥ 0) &, 2]]
```

Examples

```
EllipseDraw[{-1, 0}, {0, 1}, {4, 3}, {58, 31}];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```

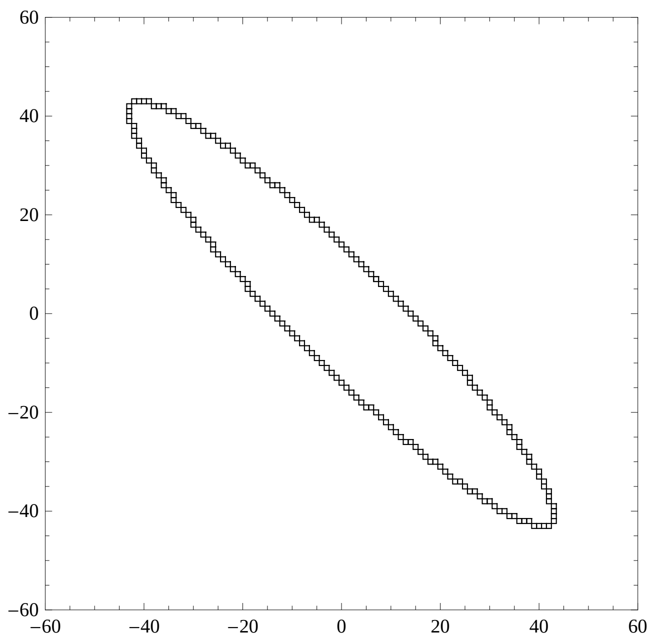


```
EllipseDraw[{-3, 2}, {-3, 2}, {2, 3}, {65, 3}];
Show[Graphics[unitsquare[First[#]] & /@ %,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



■ Verification

```
ell60 = EllipseDraw @@ {{-1, 1}, {-1, 1}, {1, -1}, {60, 10}};
Show[Graphics[unitsquare[First[#]] & /@ ell60,
  Frame → True, AspectRatio → 1, PlotRange → 60 {{-1, 1}, {-1, 1}}]]
```



- **Data set of ellipse drawn**

$$\left\{\{x, y\}, \left\{\Delta, \left\{\frac{\partial\Delta}{\partial x}, \frac{\partial\Delta}{\partial y}\right\}\right\}, \{A, B, C\}\right\}$$

```
ell60
```

```
{{{7, 7}, {-14400, {100800, 100800}}, {3700, 3700, 3500}},
 {{6, 8}, {-14000, {100400, 101200}}, {3700, 3700, 3500}},
 {{5, 9}, {-12800, {100000, 101600}}, {3700, 3700, 3500}},
 {{4, 10}, {-10800, {99600, 102000}}, {3700, 3700, 3500}},
 {{3, 11}, {-8000, {99200, 102400}}, {3700, 3700, 3500}},
 {{2, 12}, {-4400, {98800, 102800}}, {3700, 3700, 3500}},
 {{1, 13}, {0, {98400, 103200}}, {3700, 3700, 3500}},
 {{0, 14}, {5200, {98000, 103600}}, {3700, 3700, 3500}},
 {{-1, 15}, {11200, {97600, 104000}}, {3700, 3700, 3500}},
 {{-2, 16}, {18000, {97200, 104400}}, {3700, 3700, 3500}},
 {{-3, 17}, {25600, {96800, 104800}}, {3700, 3700, 3500}},
 {{-4, 18}, {34000, {96400, 105200}}, {3700, 3700, 3500}},
 {{-5, 19}, {43200, {96000, 105600}}, {3700, 3700, 3500}},
 {{-6, 19}, {-49100, {88600, 98600}}, {3700, 3700, 3500}},
 {{-7, 20}, {-38700, {88200, 99000}}, {3700, 3700, 3500}},
 {{-8, 21}, {-27500, {87800, 99400}}, {3700, 3700, 3500}},
 {{-9, 22}, {-15500, {87400, 99800}}, {3700, 3700, 3500}},
 {{-10, 23}, {-2700, {87000, 100200}}, {3700, 3700, 3500}},
 {{-11, 24}, {10900, {86600, 100600}}, {3700, 3700, 3500}},
 {{-12, 25}, {25300, {86200, 101000}}, {3700, 3700, 3500}},
 {{-13, 26}, {40500, {85800, 101400}}, {3700, 3700, 3500}},
 {{-14, 26}, {-41600, {78400, 94400}}, {3700, 3700, 3500}},
 {{-15, 27}, {-25200, {78000, 94800}}, {3700, 3700, 3500}},
 {{-16, 28}, {-8000, {77600, 95200}}, {3700, 3700, 3500}},
 {{-17, 29}, {10000, {77200, 95600}}, {3700, 3700, 3500}},
 {{-18, 30}, {28800, {76800, 96000}}, {3700, 3700, 3500}},
 {{-19, 30}, {-44300, {69400, 89000}}, {3700, 3700, 3500}},
 {{-20, 31}, {-24300, {69000, 89400}}, {3700, 3700, 3500}},
 {{-21, 32}, {-3500, {68600, 89800}}, {3700, 3700, 3500}},
 {{-22, 33}, {18100, {68200, 90200}}, {3700, 3700, 3500}},
 {{-23, 34}, {40500, {67800, 90600}}, {3700, 3700, 3500}},
 {{-24, 34}, {-23600, {60400, 83600}}, {3700, 3700, 3500}},
 {{-25, 35}, {0, {60000, 84000}}, {3700, 3700, 3500}},
 {{-26, 36}, {24400, {59600, 84400}}, {3700, 3700, 3500}},
 {{-27, 36}, {-31500, {52200, 77400}}, {3700, 3700, 3500}},
 {{-28, 37}, {-5900, {51800, 77800}}, {3700, 3700, 3500}},
 {{-29, 38}, {20500, {51400, 78200}}, {3700, 3700, 3500}},
 {{-30, 38}, {-27200, {44000, 71200}}, {3700, 3700, 3500}},
 {{-31, 39}, {400, {43600, 71600}}, {3700, 3700, 3500}},
 {{-32, 40}, {28800, {43200, 72000}}, {3700, 3700, 3500}},
 {{-33, 40}, {-10700, {35800, 65000}}, {3700, 3700, 3500}},
 {{-34, 41}, {18900, {35400, 65400}}, {3700, 3700, 3500}},
 {{-35, 41}, {-12800, {28000, 58400}}, {3700, 3700, 3500}},
 {{-36, 42}, {18000, {27600, 58800}}, {3700, 3700, 3500}},
 {{-37, 42}, {-5900, {20200, 51800}}, {3700, 3700, 3500}},
 {{-38, 42}, {-22400, {12800, 44800}}, {3700, 3700, 3500}},
```

```
{{-39, 43}, {10 000, {12 400, 45 200}}, {3700, 3700, 3500}},
{{-40, 43}, {1300, {5000, 38 200}}, {3700, 3700, 3500}},
{{-41, 43}, {0, {-2400, 31 200}}, {3700, 3700, 3500}},
{{-42, 43}, {6100, {-9800, 24 200}}, {3700, 3700, 3500}},
{{-43, 42}, {6100, {-24 200, 9800}}, {3700, 3700, 3500}},
{{-43, 41}, {0, {-31 200, 2400}}, {3700, 3700, 3500}},
{{-43, 40}, {1300, {-38 200, -5000}}, {3700, 3700, 3500}},
{{-43, 39}, {10 000, {-45 200, -12 400}}, {3700, 3700, 3500}},
{{-42, 38}, {-22 400, {-44 800, -12 800}}, {3700, 3700, 3500}},
{{-42, 37}, {-5900, {-51 800, -20 200}}, {3700, 3700, 3500}},
{{-42, 36}, {18 000, {-58 800, -27 600}}, {3700, 3700, 3500}},
{{-41, 35}, {-12 800, {-58 400, -28 000}}, {3700, 3700, 3500}},
{{-41, 34}, {18 900, {-65 400, -35 400}}, {3700, 3700, 3500}},
{{-40, 33}, {-10 700, {-65 000, -35 800}}, {3700, 3700, 3500}},
{{-40, 32}, {28 800, {-72 000, -43 200}}, {3700, 3700, 3500}},
{{-39, 31}, {400, {-71 600, -43 600}}, {3700, 3700, 3500}},
{{-38, 30}, {-27 200, {-71 200, -44 000}}, {3700, 3700, 3500}},
{{-38, 29}, {20 500, {-78 200, -51 400}}, {3700, 3700, 3500}},
{{-37, 28}, {-5900, {-77 800, -51 800}}, {3700, 3700, 3500}},
{{-36, 27}, {-31 500, {-77 400, -52 200}}, {3700, 3700, 3500}},
{{-36, 26}, {24 400, {-84 400, -59 600}}, {3700, 3700, 3500}},
{{-35, 25}, {0, {-84 000, -60 000}}, {3700, 3700, 3500}},
{{-34, 24}, {-23 600, {-83 600, -60 400}}, {3700, 3700, 3500}},
{{-34, 23}, {40 500, {-90 600, -67 800}}, {3700, 3700, 3500}},
{{-33, 22}, {18 100, {-90 200, -68 200}}, {3700, 3700, 3500}},
{{-32, 21}, {-3500, {-89 800, -68 600}}, {3700, 3700, 3500}},
{{-31, 20}, {-24 300, {-89 400, -69 000}}, {3700, 3700, 3500}},
{{-30, 19}, {-44 300, {-89 000, -69 400}}, {3700, 3700, 3500}},
{{-30, 18}, {28 800, {-96 000, -76 800}}, {3700, 3700, 3500}},
{{-29, 17}, {10 000, {-95 600, -77 200}}, {3700, 3700, 3500}},
{{-28, 16}, {-8000, {-95 200, -77 600}}, {3700, 3700, 3500}},
{{-27, 15}, {-25 200, {-94 800, -78 000}}, {3700, 3700, 3500}},
{{-26, 14}, {-41 600, {-94 400, -78 400}}, {3700, 3700, 3500}},
{{-26, 13}, {40 500, {-101 400, -85 800}}, {3700, 3700, 3500}},
{{-25, 12}, {25 300, {-101 000, -86 200}}, {3700, 3700, 3500}},
{{-24, 11}, {10 900, {-100 600, -86 600}}, {3700, 3700, 3500}},
{{-23, 10}, {-2700, {-100 200, -87 000}}, {3700, 3700, 3500}},
{{-22, 9}, {-15 500, {-99 800, -87 400}}, {3700, 3700, 3500}},
{{-21, 8}, {-27 500, {-99 400, -87 800}}, {3700, 3700, 3500}},
{{-20, 7}, {-38 700, {-99 000, -88 200}}, {3700, 3700, 3500}},
{{-19, 6}, {-49 100, {-98 600, -88 600}}, {3700, 3700, 3500}},
{{-19, 5}, {43 200, {-105 600, -96 000}}, {3700, 3700, 3500}},
{{-18, 4}, {34 000, {-105 200, -96 400}}, {3700, 3700, 3500}},
{{-17, 3}, {25 600, {-104 800, -96 800}}, {3700, 3700, 3500}},
{{-16, 2}, {18 000, {-104 400, -97 200}}, {3700, 3700, 3500}},
{{-15, 1}, {11 200, {-104 000, -97 600}}, {3700, 3700, 3500}},
{{-14, 0}, {5200, {-103 600, -98 000}}, {3700, 3700, 3500}},
{{-13, -1}, {0, {-103 200, -98 400}}, {3700, 3700, 3500}},
{{-12, -2}, {-4400, {-102 800, -98 800}}, {3700, 3700, 3500}},
{{-11, -3}, {-8000, {-102 400, -99 200}}, {3700, 3700, 3500}},
{{-10, -4}, {-10 800, {-102 000, -99 600}}, {3700, 3700, 3500}},
{{-9, -5}, {-12 800, {-101 600, -100 000}}, {3700, 3700, 3500}},
{{-8, -6}, {-14 000, {-101 200, -100 400}}, {3700, 3700, 3500}},
{{-7, -7}, {-14 400, {-100 800, -100 800}}, {3700, 3700, 3500}},
```

```
{{-6, -8}, {-14 000, {-100 400, -101 200}}, {3700, 3700, 3500}},
{{-5, -9}, {-12 800, {-100 000, -101 600}}, {3700, 3700, 3500}},
{{-4, -10}, {-10 800, {-99 600, -102 000}}, {3700, 3700, 3500}},
{{-3, -11}, {-8000, {-99 200, -102 400}}, {3700, 3700, 3500}},
{{-2, -12}, {-4400, {-98 800, -102 800}}, {3700, 3700, 3500}},
{{-1, -13}, {0, {-98 400, -103 200}}, {3700, 3700, 3500}},
{{0, -14}, {5200, {-98 000, -103 600}}, {3700, 3700, 3500}},
{{1, -15}, {11 200, {-97 600, -104 000}}, {3700, 3700, 3500}},
{{2, -16}, {18 000, {-97 200, -104 400}}, {3700, 3700, 3500}},
{{3, -17}, {25 600, {-96 800, -104 800}}, {3700, 3700, 3500}},
{{4, -18}, {34 000, {-96 400, -105 200}}, {3700, 3700, 3500}},
{{5, -19}, {43 200, {-96 000, -105 600}}, {3700, 3700, 3500}},
{{6, -19}, {-49 100, {-88 600, -98 600}}, {3700, 3700, 3500}},
{{7, -20}, {-38 700, {-88 200, -99 000}}, {3700, 3700, 3500}},
{{8, -21}, {-27 500, {-87 800, -99 400}}, {3700, 3700, 3500}},
{{9, -22}, {-15 500, {-87 400, -99 800}}, {3700, 3700, 3500}},
{{10, -23}, {-2700, {-87 000, -100 200}}, {3700, 3700, 3500}},
{{11, -24}, {10 900, {-86 600, -100 600}}, {3700, 3700, 3500}},
{{12, -25}, {25 300, {-86 200, -101 000}}, {3700, 3700, 3500}},
{{13, -26}, {40 500, {-85 800, -101 400}}, {3700, 3700, 3500}},
{{14, -26}, {-41 600, {-78 400, -94 400}}, {3700, 3700, 3500}},
{{15, -27}, {-25 200, {-78 000, -94 800}}, {3700, 3700, 3500}},
{{16, -28}, {-8000, {-77 600, -95 200}}, {3700, 3700, 3500}},
{{17, -29}, {10 000, {-77 200, -95 600}}, {3700, 3700, 3500}},
{{18, -30}, {28 800, {-76 800, -96 000}}, {3700, 3700, 3500}},
{{19, -30}, {-44 300, {-69 400, -89 000}}, {3700, 3700, 3500}},
{{20, -31}, {-24 300, {-69 000, -89 400}}, {3700, 3700, 3500}},
{{21, -32}, {-3500, {-68 600, -89 800}}, {3700, 3700, 3500}},
{{22, -33}, {18 100, {-68 200, -90 200}}, {3700, 3700, 3500}},
{{23, -34}, {40 500, {-67 800, -90 600}}, {3700, 3700, 3500}},
{{24, -34}, {-23 600, {-60 400, -83 600}}, {3700, 3700, 3500}},
{{25, -35}, {0, {-60 000, -84 000}}, {3700, 3700, 3500}},
{{26, -36}, {24 400, {-59 600, -84 400}}, {3700, 3700, 3500}},
{{27, -36}, {-31 500, {-52 200, -77 400}}, {3700, 3700, 3500}},
{{28, -37}, {-5900, {-51 800, -77 800}}, {3700, 3700, 3500}},
{{29, -38}, {20 500, {-51 400, -78 200}}, {3700, 3700, 3500}},
{{30, -38}, {-27 200, {-44 000, -71 200}}, {3700, 3700, 3500}},
{{31, -39}, {400, {-43 600, -71 600}}, {3700, 3700, 3500}},
{{32, -40}, {28 800, {-43 200, -72 000}}, {3700, 3700, 3500}},
{{33, -40}, {-10 700, {-35 800, -65 000}}, {3700, 3700, 3500}},
{{34, -41}, {18 900, {-35 400, -65 400}}, {3700, 3700, 3500}},
{{35, -41}, {-12 800, {-28 000, -58 400}}, {3700, 3700, 3500}},
{{36, -42}, {18 000, {-27 600, -58 800}}, {3700, 3700, 3500}},
{{37, -42}, {-5900, {-20 200, -51 800}}, {3700, 3700, 3500}},
{{38, -42}, {-22 400, {-12 800, -44 800}}, {3700, 3700, 3500}},
{{39, -43}, {10 000, {-12 400, -45 200}}, {3700, 3700, 3500}},
{{40, -43}, {1300, {-5000, -38 200}}, {3700, 3700, 3500}},
{{41, -43}, {0, {2400, -31 200}}, {3700, 3700, 3500}},
{{42, -43}, {6100, {9800, -24 200}}, {3700, 3700, 3500}},
{{43, -42}, {6100, {24 200, -9800}}, {3700, 3700, 3500}},
{{43, -41}, {0, {31 200, -2400}}, {3700, 3700, 3500}},
{{43, -40}, {1300, {38 200, 5000}}, {3700, 3700, 3500}},
{{43, -39}, {10 000, {45 200, 12 400}}, {3700, 3700, 3500}},
{{42, -38}, {-22 400, {44 800, 12 800}}, {3700, 3700, 3500}},
```

```
{{42, -37}, {-5900, {51 800, 20 200}}, {3700, 3700, 3500}},
{{42, -36}, {18 000, {58 800, 27 600}}, {3700, 3700, 3500}},
{{41, -35}, {-12 800, {58 400, 28 000}}, {3700, 3700, 3500}},
{{41, -34}, {18 900, {65 400, 35 400}}, {3700, 3700, 3500}},
{{40, -33}, {-10 700, {65 000, 35 800}}, {3700, 3700, 3500}},
{{40, -32}, {28 800, {72 000, 43 200}}, {3700, 3700, 3500}},
{{39, -31}, {400, {71 600, 43 600}}, {3700, 3700, 3500}},
{{38, -30}, {-27 200, {71 200, 44 000}}, {3700, 3700, 3500}},
{{38, -29}, {20 500, {78 200, 51 400}}, {3700, 3700, 3500}},
{{37, -28}, {-5900, {77 800, 51 800}}, {3700, 3700, 3500}},
{{36, -27}, {-31 500, {77 400, 52 200}}, {3700, 3700, 3500}},
{{36, -26}, {24 400, {84 400, 59 600}}, {3700, 3700, 3500}},
{{35, -25}, {0, {84 000, 60 000}}, {3700, 3700, 3500}},
{{34, -24}, {-23 600, {83 600, 60 400}}, {3700, 3700, 3500}},
{{34, -23}, {40 500, {90 600, 67 800}}, {3700, 3700, 3500}},
{{33, -22}, {18 100, {90 200, 68 200}}, {3700, 3700, 3500}},
{{32, -21}, {-3500, {89 800, 68 600}}, {3700, 3700, 3500}},
{{31, -20}, {-24 300, {89 400, 69 000}}, {3700, 3700, 3500}},
{{30, -19}, {-44 300, {89 000, 69 400}}, {3700, 3700, 3500}},
{{30, -18}, {28 800, {96 000, 76 800}}, {3700, 3700, 3500}},
{{29, -17}, {10 000, {95 600, 77 200}}, {3700, 3700, 3500}},
{{28, -16}, {-8000, {95 200, 77 600}}, {3700, 3700, 3500}},
{{27, -15}, {-25 200, {94 800, 78 000}}, {3700, 3700, 3500}},
{{26, -14}, {-41 600, {94 400, 78 400}}, {3700, 3700, 3500}},
{{26, -13}, {40 500, {101 400, 85 800}}, {3700, 3700, 3500}},
{{25, -12}, {25 300, {101 000, 86 200}}, {3700, 3700, 3500}},
{{24, -11}, {10 900, {100 600, 86 600}}, {3700, 3700, 3500}},
{{23, -10}, {-2700, {100 200, 87 000}}, {3700, 3700, 3500}},
{{22, -9}, {-15 500, {99 800, 87 400}}, {3700, 3700, 3500}},
{{21, -8}, {-27 500, {99 400, 87 800}}, {3700, 3700, 3500}},
{{20, -7}, {-38 700, {99 000, 88 200}}, {3700, 3700, 3500}},
{{19, -6}, {-49 100, {98 600, 88 600}}, {3700, 3700, 3500}},
{{19, -5}, {43 200, {105 600, 96 000}}, {3700, 3700, 3500}},
{{18, -4}, {34 000, {105 200, 96 400}}, {3700, 3700, 3500}},
{{17, -3}, {25 600, {104 800, 96 800}}, {3700, 3700, 3500}},
{{16, -2}, {18 000, {104 400, 97 200}}, {3700, 3700, 3500}},
{{15, -1}, {11 200, {104 000, 97 600}}, {3700, 3700, 3500}},
{{14, 0}, {5200, {103 600, 98 000}}, {3700, 3700, 3500}},
{{13, 1}, {0, {103 200, 98 400}}, {3700, 3700, 3500}},
{{12, 2}, {-4400, {102 800, 98 800}}, {3700, 3700, 3500}},
{{11, 3}, {-8000, {102 400, 99 200}}, {3700, 3700, 3500}},
{{10, 4}, {-10 800, {102 000, 99 600}}, {3700, 3700, 3500}},
{{9, 5}, {-12 800, {101 600, 100 000}}, {3700, 3700, 3500}},
{{8, 6}, {-14 000, {101 200, 100 400}}, {3700, 3700, 3500}},
{{7, 7}, {-14 400, {100 800, 100 800}}, {3700, 3700, 3500}}}
```

Verify that the iterated Δ and the Δ computed from the coordinates match:

```
ellipsediscriminant
disc60 = ellipsediscriminant /. {ξ → 1, υ → -1, a → 60, b → 10}
```

$$2\left(-a^2+b^2\right) x\, y\, \xi\, \upsilon - a^2 b^2 \left(\xi^2+\upsilon^2\right) + x^2 \left(b^2 \xi^2 + a^2 \upsilon^2\right) + y^2 \left(a^2 \xi^2 + b^2 \upsilon^2\right)$$

$$-720\,000 + 3700\, x^2 + 7000\, x\, y + 3700\, y^2$$

$\{x, y\}$ $\left\{\Delta_{\text{computed}}, \Delta_{\text{iterated}}\right\}$

```
TableForm[{#[[1]], {disc60 /. {x → #[[1, 1]], y → #[[1, 2]]}, First[#[[2]]]}} & /@ ell60, TableDepth → 2]
```

```
{7, 7}      {-14400, -14400}
{6, 8}      {-14000, -14000}
{5, 9}      {-12800, -12800}
{4, 10}     {-10800, -10800}
{3, 11}     {-8000, -8000}
{2, 12}     {-4400, -4400}
{1, 13}     {0, 0}
{0, 14}     {5200, 5200}
{-1, 15}    {11200, 11200}
{-2, 16}    {18000, 18000}
{-3, 17}    {25600, 25600}
{-4, 18}    {34000, 34000}
{-5, 19}    {43200, 43200}
{-6, 19}    {-49100, -49100}
{-7, 20}    {-38700, -38700}
{-8, 21}    {-27500, -27500}
{-9, 22}    {-15500, -15500}
{-10, 23}   {-2700, -2700}
{-11, 24}   {10900, 10900}
{-12, 25}   {25300, 25300}
{-13, 26}   {40500, 40500}
{-14, 26}   {-41600, -41600}
{-15, 27}   {-25200, -25200}
{-16, 28}   {-8000, -8000}
{-17, 29}   {10000, 10000}
{-18, 30}   {28800, 28800}
{-19, 30}   {-44300, -44300}
{-20, 31}   {-24300, -24300}
{-21, 32}   {-3500, -3500}
{-22, 33}   {18100, 18100}
{-23, 34}   {40500, 40500}
{-24, 34}   {-23600, -23600}
{-25, 35}   {0, 0}
{-26, 36}   {24400, 24400}
{-27, 36}   {-31500, -31500}
{-28, 37}   {-5900, -5900}
{-29, 38}   {20500, 20500}
{-30, 38}   {-27200, -27200}
{-31, 39}   {400, 400}
{-32, 40}   {28800, 28800}
{-33, 40}   {-10700, -10700}
{-34, 41}   {18900, 18900}
{-35, 41}   {-12800, -12800}
{-36, 42}   {18000, 18000}
{-37, 42}   {-5900, -5900}
```

```
{-38, 42}  {-22 400, -22 400}
{-39, 43}  {10 000, 10 000}
{-40, 43}  {1300, 1300}
{-41, 43}  {0, 0}
{-42, 43}  {6100, 6100}
{-43, 42}  {6100, 6100}
{-43, 41}  {0, 0}
{-43, 40}  {1300, 1300}
{-43, 39}  {10 000, 10 000}
{-42, 38}  {-22 400, -22 400}
{-42, 37}  {-5900, -5900}
{-42, 36}  {18 000, 18 000}
{-41, 35}  {-12 800, -12 800}
{-41, 34}  {18 900, 18 900}
{-40, 33}  {-10 700, -10 700}
{-40, 32}  {28 800, 28 800}
{-39, 31}  {400, 400}
{-38, 30}  {-27 200, -27 200}
{-38, 29}  {20 500, 20 500}
{-37, 28}  {-5900, -5900}
{-36, 27}  {-31 500, -31 500}
{-36, 26}  {24 400, 24 400}
{-35, 25}  {0, 0}
{-34, 24}  {-23 600, -23 600}
{-34, 23}  {40 500, 40 500}
{-33, 22}  {18 100, 18 100}
{-32, 21}  {-3500, -3500}
{-31, 20}  {-24 300, -24 300}
{-30, 19}  {-44 300, -44 300}
{-30, 18}  {28 800, 28 800}
{-29, 17}  {10 000, 10 000}
{-28, 16}  {-8000, -8000}
{-27, 15}  {-25 200, -25 200}
{-26, 14}  {-41 600, -41 600}
{-26, 13}  {40 500, 40 500}
{-25, 12}  {25 300, 25 300}
{-24, 11}  {10 900, 10 900}
{-23, 10}  {-2700, -2700}
{-22, 9}   {-15 500, -15 500}
{-21, 8}   {-27 500, -27 500}
{-20, 7}   {-38 700, -38 700}
{-19, 6}   {-49 100, -49 100}
{-19, 5}   {43 200, 43 200}
{-18, 4}   {34 000, 34 000}
{-17, 3}   {25 600, 25 600}
```

```
{-16, 2}   {18000, 18000}
{-15, 1}   {11200, 11200}
{-14, 0}   {5200, 5200}
{-13, -1}  {0, 0}
{-12, -2}  {-4400, -4400}
{-11, -3}  {-8000, -8000}
{-10, -4}  {-10800, -10800}
{-9, -5}   {-12800, -12800}
{-8, -6}   {-14000, -14000}
{-7, -7}   {-14400, -14400}
{-6, -8}   {-14000, -14000}
{-5, -9}   {-12800, -12800}
{-4, -10}  {-10800, -10800}
{-3, -11}  {-8000, -8000}
{-2, -12}  {-4400, -4400}
{-1, -13}  {0, 0}
{0, -14}   {5200, 5200}
{1, -15}   {11200, 11200}
{2, -16}   {18000, 18000}
{3, -17}   {25600, 25600}
{4, -18}   {34000, 34000}
{5, -19}   {43200, 43200}
{6, -19}   {-49100, -49100}
{7, -20}   {-38700, -38700}
{8, -21}   {-27500, -27500}
{9, -22}   {-15500, -15500}
{10, -23}  {-2700, -2700}
{11, -24}  {10900, 10900}
{12, -25}  {25300, 25300}
{13, -26}  {40500, 40500}
{14, -26}  {-41600, -41600}
{15, -27}  {-25200, -25200}
{16, -28}  {-8000, -8000}
{17, -29}  {10000, 10000}
{18, -30}  {28800, 28800}
{19, -30}  {-44300, -44300}
{20, -31}  {-24300, -24300}
{21, -32}  {-3500, -3500}
{22, -33}  {18100, 18100}
{23, -34}  {40500, 40500}
{24, -34}  {-23600, -23600}
{25, -35}  {0, 0}
{26, -36}  {24400, 24400}
{27, -36}  {-31500, -31500}
{28, -37}  {-5900, -5900}
```

```
{29, -38}  {20 500, 20 500}
{30, -38}  {-27 200, -27 200}
{31, -39}  {400, 400}
{32, -40}  {28 800, 28 800}
{33, -40}  {-10 700, -10 700}
{34, -41}  {18 900, 18 900}
{35, -41}  {-12 800, -12 800}
{36, -42}  {18 000, 18 000}
{37, -42}  {-5900, -5900}
{38, -42}  {-22 400, -22 400}
{39, -43}  {10 000, 10 000}
{40, -43}  {1300, 1300}
{41, -43}  {0, 0}
{42, -43}  {6100, 6100}
{43, -42}  {6100, 6100}
{43, -41}  {0, 0}
{43, -40}  {1300, 1300}
{43, -39}  {10 000, 10 000}
{42, -38}  {-22 400, -22 400}
{42, -37}  {-5900, -5900}
{42, -36}  {18 000, 18 000}
{41, -35}  {-12 800, -12 800}
{41, -34}  {18 900, 18 900}
{40, -33}  {-10 700, -10 700}
{40, -32}  {28 800, 28 800}
{39, -31}  {400, 400}
{38, -30}  {-27 200, -27 200}
{38, -29}  {20 500, 20 500}
{37, -28}  {-5900, -5900}
{36, -27}  {-31 500, -31 500}
{36, -26}  {24 400, 24 400}
{35, -25}  {0, 0}
{34, -24}  {-23 600, -23 600}
{34, -23}  {40 500, 40 500}
{33, -22}  {18 100, 18 100}
{32, -21}  {-3500, -3500}
{31, -20}  {-24 300, -24 300}
{30, -19}  {-44 300, -44 300}
{30, -18}  {28 800, 28 800}
{29, -17}  {10 000, 10 000}
{28, -16}  {-8000, -8000}
{27, -15}  {-25 200, -25 200}
{26, -14}  {-41 600, -41 600}
{26, -13}  {40 500, 40 500}
{25, -12}  {25 300, 25 300}
```

```
{24, -11}  {10 900, 10 900}
{23, -10}  {-2700, -2700}
{22, -9}   {-15 500, -15 500}
{21, -8}   {-27 500, -27 500}
{20, -7}   {-38 700, -38 700}
{19, -6}   {-49 100, -49 100}
{19, -5}   {43 200, 43 200}
{18, -4}   {34 000, 34 000}
{17, -3}   {25 600, 25 600}
{16, -2}   {18 000, 18 000}
{15, -1}   {11 200, 11 200}
{14, 0}    {5200, 5200}
{13, 1}    {0, 0}
{12, 2}    {-4400, -4400}
{11, 3}    {-8000, -8000}
{10, 4}    {-10 800, -10 800}
{9, 5}     {-12 800, -12 800}
{8, 6}     {-14 000, -14 000}
{7, 7}     {-14 400, -14 400}
```

Generalization to muralizer

Basic equation

To draw a straight line, the muralizer spoolers need to follow a trajectory defined by $\begin{pmatrix} r^2 & s^2 \end{pmatrix} \cdot \begin{pmatrix} A & C \\ C & B \end{pmatrix} \cdot \begin{pmatrix} r^2 \\ s^2 \end{pmatrix} + \begin{pmatrix} r^2 & s^2 \end{pmatrix} \cdot \begin{pmatrix} D \\ E \end{pmatrix} + F == 0$, where r and s are the lengths of thread spooled.

Iterated variables

The discriminant and its gradient need not be computed from scratch for each new point, but can be iterated, exploiting the fact that for continuous lines, each of the coordinate can only change by 0 or ± 1.

▪ Discriminant

```
a x^2 + c x y + b y^2 + d x + e y + f /. {x → r^2, y → s^2}
TableForm[% /. {{r → r + 1}, {r → r + 1, s → s + 1}, {s → s + 1}, {r → r - 1, s → s + 1},
   {r → r - 1}, {r → r - 1, s → s - 1}, {s → s - 1}, {r → r + 1, s → s - 1}}]
TableForm[Collect[#, {a, b, c, d, e, f}, FullSimplify] & /@ (% - %%)]
TableForm[FullSimplify[ListConvolve[{1, -1}, %, 1]]]
TableForm[Simplify[{-1, 1, -1}.%%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}}]
```

$f + d\,r^2 + a\,r^4 + e\,s^2 + c\,r^2\,s^2 + b\,s^4$

$f + d\,(1+r)^2 + a\,(1+r)^4 + e\,s^2 + c\,(1+r)^2\,s^2 + b\,s^4$
$f + d\,(1+r)^2 + a\,(1+r)^4 + e\,(1+s)^2 + c\,(1+r)^2\,(1+s)^2 + b\,(1+s)^4$
$f + d\,r^2 + a\,r^4 + e\,(1+s)^2 + c\,r^2\,(1+s)^2 + b\,(1+s)^4$
$f + d\,(-1+r)^2 + a\,(-1+r)^4 + e\,(1+s)^2 + c\,(-1+r)^2\,(1+s)^2 + b\,(1+s)^4$
$f + d\,(-1+r)^2 + a\,(-1+r)^4 + e\,s^2 + c\,(-1+r)^2\,s^2 + b\,s^4$
$f + d\,(-1+r)^2 + a\,(-1+r)^4 + e\,(-1+s)^2 + c\,(-1+r)^2\,(-1+s)^2 + b\,(-1+s)^4$
$f + d\,r^2 + a\,r^4 + e\,(-1+s)^2 + c\,r^2\,(-1+s)^2 + b\,(-1+s)^4$
$f + d\,(1+r)^2 + a\,(1+r)^4 + e\,(-1+s)^2 + c\,(1+r)^2\,(-1+s)^2 + b\,(-1+s)^4$

$d\,(1+2r) + a\left(-r^4 + (1+r)^4\right) + c\,(1+2r)\,s^2$
$d\,(1+2r) + a\left(-r^4 + (1+r)^4\right) + e\,(1+2s) + c\,(1+r+s)\,(1+r+s+2rs) + b\left(-s^4 + (1+s)^4\right)$
$e\,(1+2s) + c\,r^2\,(1+2s) + b\left(-s^4 + (1+s)^4\right)$
$d\,(1-2r) + a\left((-1+r)^4 - r^4\right) + e\,(1+2s) + c\,(-1+r-s)\,(-1+r-s+2rs) + b\left(-s^4 + (1+s)^4\right)$
$d\,(1-2r) + a\left((-1+r)^4 - r^4\right) + c\,(1-2r)\,s^2$
$d\,(1-2r) + a\left((-1+r)^4 - r^4\right) + e\,(1-2s) + b\left((-1+s)^4 - s^4\right) - c\,(-1+r+s)\,(1-s+r\,(-1+2s))$
$e\,(1-2s) + c\,r^2\,(1-2s) + b\left((-1+s)^4 - s^4\right)$
$d\,(1+2r) + a\left(-r^4 + (1+r)^4\right) + e\,(1-2s) + b\left((-1+s)^4 - s^4\right) - c\,(1+r-s)\,(-1+s+r\,(-1+2s))$

$(-1+2s)\left(e + c\,(1+r)^2 + b\,(1+2\,(-1+s)\,s)\right)$
$(1+2s)\left(e + c\,(1+r)^2 + b\,(1+2s\,(1+s))\right)$
$-(1+2r)\left(d + a\,(1+2r\,(1+r)) + c\,(1+s)^2\right)$
$-(-1+2r)\left(d + a\,(1+2\,(-1+r)\,r) + c\,(1+s)^2\right)$
$-(1+2s)\left(e + c\,(-1+r)^2 + b\,(1+2s\,(1+s))\right)$
$-(-1+2s)\left(e + c\,(-1+r)^2 + b\,(1+2\,(-1+s)\,s)\right)$
$(-1+2r)\left(d + a\,(1+2\,(-1+r)\,r) + c\,(-1+s)^2\right)$
$(1+2r)\left(d + a\,(1+2r\,(1+r)) + c\,(-1+s)^2\right)$

$c\,(1+2r)\,(1+2s)$
$-c\,(-1+2r)\,(1+2s)$
$c\,(-1+2r)\,(-1+2s)$
$-c\,(1+2r)\,(-1+2s)$

Step in r direction: $a\left((r \pm 1)^4 - r^4\right) + \left(c\,s^2 + d\right)\left((r \pm 1)^2 - r^2\right)$

Step in s direction: $b\left((s \pm 1)^4 - s^4\right) + \left(c\,r^2 + e\right)\left((s \pm 1)^2 - s^2\right)$

Diagonal step: sum of orthogonal steps, $+c\left(r^2 - (r \pm 1)^2\right)\left(s^2 - (s \pm 1)^2\right)$

- **Partial derivative of discriminant w.r.t. r^2**

```
D[a x^2 + c x y + b y^2 + d x + e y + f, x] /. {x → r^2, y → s^2}
TableForm[% /. {{r → r + 1}, {r → r + 1, s → s + 1}, {s → s + 1}, {r → r - 1, s → s + 1},
   {r → r - 1}, {r → r - 1, s → s - 1}, {s → s - 1}, {r → r + 1, s → s - 1}}]
TableForm[Collect[#, {a, b, c, d, e, f}, FullSimplify] & /@ (% - %%)]
TableForm[FullSimplify[ListConvolve[{1, -1}, %, 1]]]
TableForm[Simplify[{-1, 1, -1}.%%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}}]
```

$d + 2 a r^2 + c s^2$

$d + 2 a (1 + r)^2 + c s^2$
$d + 2 a (1 + r)^2 + c (1 + s)^2$
$d + 2 a r^2 + c (1 + s)^2$
$d + 2 a (-1 + r)^2 + c (1 + s)^2$
$d + 2 a (-1 + r)^2 + c s^2$
$d + 2 a (-1 + r)^2 + c (-1 + s)^2$
$d + 2 a r^2 + c (-1 + s)^2$
$d + 2 a (1 + r)^2 + c (-1 + s)^2$

$a (2 + 4 r)$
$a (2 + 4 r) + c (1 + 2 s)$
$c (1 + 2 s)$
$a (2 - 4 r) + c (1 + 2 s)$
$a (2 - 4 r)$
$a (2 - 4 r) + c (1 - 2 s)$
$c (1 - 2 s)$
$a (2 + 4 r) + c (1 - 2 s)$

$c (-1 + 2 s)$
$c + 2 c s$
$-2 a (1 + 2 r)$
$a (2 - 4 r)$
$-c (1 + 2 s)$
$c - 2 c s$
$2 a (-1 + 2 r)$
$a (2 + 4 r)$

0
0
0
0

Partial derivative of discriminant w.r.t. s^2

```
D[a x^2 + c x y + b y^2 + d x + e y + f, y] /. {x → r^2, y → s^2}
TableForm[% /. {{r → r + 1}, {r → r + 1, s → s + 1}, {s → s + 1}, {r → r - 1, s → s + 1},
   {r → r - 1}, {r → r - 1, s → s - 1}, {s → s - 1}, {r → r + 1, s → s - 1}}]
TableForm[Collect[#, {a, b, c, d, e, f}, FullSimplify] & /@ (% - %%)]
TableForm[FullSimplify[ListConvolve[{1, -1}, %, 1]]]
TableForm[Simplify[{-1, 1, -1}.%%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}}]
```

$e + c\, r^2 + 2\, b\, s^2$

$e + c\, (1 + r)^2 + 2\, b\, s^2$
$e + c\, (1 + r)^2 + 2\, b\, (1 + s)^2$
$e + c\, r^2 + 2\, b\, (1 + s)^2$
$e + c\, (-1 + r)^2 + 2\, b\, (1 + s)^2$
$e + c\, (-1 + r)^2 + 2\, b\, s^2$
$e + c\, (-1 + r)^2 + 2\, b\, (-1 + s)^2$
$e + c\, r^2 + 2\, b\, (-1 + s)^2$
$e + c\, (1 + r)^2 + 2\, b\, (-1 + s)^2$

$c\, (1 + 2\, r)$
$c\, (1 + 2\, r) + b\, (2 + 4\, s)$
$b\, (2 + 4\, s)$
$c\, (1 - 2\, r) + b\, (2 + 4\, s)$
$c\, (1 - 2\, r)$
$c\, (1 - 2\, r) + b\, (2 - 4\, s)$
$b\, (2 - 4\, s)$
$c\, (1 + 2\, r) + b\, (2 - 4\, s)$

$2\, b\, (-1 + 2\, s)$
$b\, (2 + 4\, s)$
$-c\, (1 + 2\, r)$
$c - 2\, c\, r$
$-2\, b\, (1 + 2\, s)$
$b\, (2 - 4\, s)$
$c\, (-1 + 2\, r)$
$c + 2\, c\, r$

0
0
0
0

Gradient of discriminant

```
(D[a x^2 + c x y + b y^2 + d x + e y + f, #] & /@ {x, y}) /. {x → r^2, y → s^2}
Outer[ReplaceAll, %, {{r → r + 1}, {r → r + 1, s → s + 1}, {s → s + 1}, {r → r - 1, s → s + 1},
   {r → r - 1}, {r → r - 1, s → s - 1}, {s → s - 1}, {r → r + 1, s → s - 1}}, 1];
TableForm[Transpose[%], TableDepth → 1]
TableForm[
 Transpose[Map[Collect[#, {a, b, c, d, e, f}, FullSimplify] &, %% - %%%, {2}]], TableDepth → 1]
TableForm[Simplify[{-1, 1, -1}.%[[#]]] & /@ {{1, 2, 3}, {3, 4, 5}, {5, 6, 7}, {7, 8, 1}},
 TableDepth → 1]
```

$\{d + 2 a r^2 + c s^2, e + c r^2 + 2 b s^2\}$

$\{d + 2 a (1 + r)^2 + c s^2, e + c (1 + r)^2 + 2 b s^2\}$
$\{d + 2 a (1 + r)^2 + c (1 + s)^2, e + c (1 + r)^2 + 2 b (1 + s)^2\}$
$\{d + 2 a r^2 + c (1 + s)^2, e + c r^2 + 2 b (1 + s)^2\}$
$\{d + 2 a (-1 + r)^2 + c (1 + s)^2, e + c (-1 + r)^2 + 2 b (1 + s)^2\}$
$\{d + 2 a (-1 + r)^2 + c s^2, e + c (-1 + r)^2 + 2 b s^2\}$
$\{d + 2 a (-1 + r)^2 + c (-1 + s)^2, e + c (-1 + r)^2 + 2 b (-1 + s)^2\}$
$\{d + 2 a r^2 + c (-1 + s)^2, e + c r^2 + 2 b (-1 + s)^2\}$
$\{d + 2 a (1 + r)^2 + c (-1 + s)^2, e + c (1 + r)^2 + 2 b (-1 + s)^2\}$

$\{a (2 + 4 r), c (1 + 2 r)\}$
$\{a (2 + 4 r) + c (1 + 2 s), c (1 + 2 r) + b (2 + 4 s)\}$
$\{c (1 + 2 s), b (2 + 4 s)\}$
$\{a (2 - 4 r) + c (1 + 2 s), c (1 - 2 r) + b (2 + 4 s)\}$
$\{a (2 - 4 r), c (1 - 2 r)\}$
$\{a (2 - 4 r) + c (1 - 2 s), c (1 - 2 r) + b (2 - 4 s)\}$
$\{c (1 - 2 s), b (2 - 4 s)\}$
$\{a (2 + 4 r) + c (1 - 2 s), c (1 + 2 r) + b (2 - 4 s)\}$

$\{0, 0\}$
$\{0, 0\}$
$\{0, 0\}$
$\{0, 0\}$

Step in r direction: $\begin{pmatrix} \pm\left(4\, a\left((r \pm 1)^3 - r^3\right) + 2\left(c\, s^2 + d\right)\right) \\ 2\, c\, s\left((r \pm 1)^2 - r^2\right) \end{pmatrix}$

Step in s direction: $\begin{pmatrix} 2\, c\, r\left((s \mp 1)^2 - s^2\right) \\ \mp\left(4\, b\left((s \mp 1)^3 - s^3\right) + 2\left(c\, r^2 + e\right)\right) \end{pmatrix}$

Diagonal step: sum of orthogonal steps, $+\begin{pmatrix} \pm 2\, c\left((s \mp 1)^2 - s^2\right) \\ \mp 2\, c\left((r \pm 1)^2 - r^2\right) \end{pmatrix}$

Terms that need to be updated/recalculated per iteration

$\delta\mathrm{r4} \equiv (r \pm 1)^4 - r^4 = \delta\mathrm{r4c} \pm \delta\mathrm{r4d} \equiv 1 + 6\, r^2 \pm r\left(1 + r^2\right)$

```
Apart[(r + #)^4 - r^4 & /@ {1, -1}]
Simplify[#@@%/2 & /@ {Plus, Subtract}]
```

$\{1 + 4\,r + 6\,r^2 + 4\,r^3,\ 1 - 4\,r + 6\,r^2 - 4\,r^3\}$

$\{1 + 6\,r^2,\ 4\,(r + r^3)\}$

$$\delta\text{r3} \equiv (r \pm 1)^3 - r^3 = \delta\text{r3c} \pm \delta\text{r3d} \equiv 3\,r \pm (1 + 3\,r^2)$$

```
Apart[(r + #)^3 - r^3 & /@ {1, -1}]
Simplify[#@@%/2 & /@ {Plus, Subtract}]
```

$\{1 + 3\,r + 3\,r^2,\ -1 + 3\,r - 3\,r^2\}$

$\{3\,r,\ 1 + 3\,r^2\}$

■ Setting up initial conditions for the iterations

```
coeffrules = FullSimplify[MapThread[Rule, {
    {a, b, c, d, e, f},
    {δx^2 + δy^2,
     δx^2 + δy^2,
     -2 (δx^2 + δy^2),
     -2 (δx ΔX + δy ΔY)^2 - 4 (ΔX δy - δx ΔY) (δy (x - X0) - δx (y - Y0)),
     -2 (δx ΔX + δy ΔY)^2 + 4 (ΔX δy - δx ΔY) (δy (x - X0) - δx (y - Y0)),
    (ΔX^2 + ΔY^2) ((δx ΔX + δy ΔY)^2 + 4 (δy (x - X0) - δx (y - Y0))^2)}}] /. {δx → x2 - x1, δy → y2 - y1,
   x → (x1 + x2)/2, y → (y1 + y2)/2, ΔX → Xb - Xa, ΔY → Yb - Ya, X0 → (Xa + Xb)/2, Y0 → (Ya + Yb)/2}]
```

$\{a \to (x1 - x2)^2 + (y1 - y2)^2,\ b \to (x1 - x2)^2 + (y1 - y2)^2,\ c \to -2\,((x1 - x2)^2 + (y1 - y2)^2),$
$d \to -2\,((x1 - x2)\,(Xa - Xb) + (y1 - y2)\,(Ya - Yb))^2 + 2\,(-(Xa - Xb)\,(y1 - y2) + (x1 - x2)\,(Ya - Yb))$
$((Xa + Xb)\,(y1 - y2) + x1\,(2\,y2 - Ya - Yb) + x2\,(-2\,y1 + Ya + Yb)),$
$e \to -2\,((x1 - x2)\,(Xa - Xb) + (y1 - y2)\,(Ya - Yb))^2 + 2\,((Xa - Xb)\,(y1 - y2) - (x1 - x2)\,(Ya - Yb))$
$((Xa + Xb)\,(y1 - y2) + x1\,(2\,y2 - Ya - Yb) + x2\,(-2\,y1 + Ya + Yb)),$
$f \to ((Xa - Xb)^2 + (Ya - Yb)^2)\,(((x1 - x2)\,(Xa - Xb) + (y1 - y2)\,(Ya - Yb))^2 +$
$((Xa + Xb)\,(y1 - y2) + x1\,(2\,y2 - Ya - Yb) + x2\,(-2\,y1 + Ya + Yb))^2)\}$

```
InitIteration[refpoints_] := Function[{points},
  Block[{
    Xa = refpoints[[1, 1]],
    Ya = refpoints[[1, 2]],
    Xb = refpoints[[2, 1]],
    Yb = refpoints[[2, 2]],
    x1 = points[[1, 1]],
    y1 = points[[1, 2]],
    x2 = points[[2, 1]],
    y2 = points[[2, 2]],
    signum = If[# ≥ 0, 1, -1] &,
    a, b, c, d, e, f,
    finalPoint,
    discriminant,
    gradient,
    sign
   },
   finalPoint = Round[{Sqrt[(x2 - Xa)^2 + (y2 - Ya)^2], Sqrt[(x2 - Xb)^2 + (y2 - Yb)^2]}];
   {a, b, c, d, e, f} = ({a, b, c, d, e, f} /. coeffrules);
   discriminant = f + d r^2 + a r^4 + e s^2 + c r^2 s^2 + b s^4;
   gradient = ({d + 2 a r^2 + c s^2, e + c r^2 + 2 b s^2});
   sign = signum[(x1 - x2) (Xa - Xb) + (y1 - y2) (Ya - Yb)];
   {{r, s}, {discriminant, gradient}, {a, b, c, d, e, f, sign, finalPoint}} /.
    {r -> Round[Sqrt[(x1 - Xa)^2 + (y1 - Ya)^2]], s -> Round[Sqrt[(x1 - Xb)^2 + (y1 - Yb)^2]]}
  ]
 ]

InitIteration[{{-100, 0}, {100, 0}}][{{-70, -120}, {80, -20}}]

{{124, 208}, {-373 992 320 000, {-4 492 720 000, 892 720 000}},
 {32 500, 32 500, -65 000, -2 680 000 000, -920 000 000, 55 360 000 000 000, 1, {181, 28}}}
```

■ Identifying quadrants and octants

Since $r \geq 0$ and $s \geq 0$, it doesn't make a difference if the sign of the derivatives of the discriminant w.r.t. $\{r, s\}$ or $\{x, y\} \equiv \{r^2, s^2\}$ is computed.

■ Diagonal iteration

Underlying general rules:
$\delta x = -\sigma \operatorname{Sign}\left[\frac{\partial \Delta}{\partial y}\right]$, $\delta y = \sigma \operatorname{Sign}\left[\frac{\partial \Delta}{\partial x}\right]$, with $\sigma \equiv \operatorname{Sign}[(x_1 - x_2)(X_a - X_b) + (y_1 - y_2)(Y_a - Y_b)]$; the straight direction is the direction for which the change in Δ is smaller.

"Cheating" iteration: compute discriminant and gradient from scratch each time

```
MuralizerIterationCheat = Function[{position, iterationvariables, iterationconstants},
  Block[{r, s,
    dfunc = Function[{r, s}, f + d r^2 + a r^4 + e s^2 + c r^2 s^2 + b s^4],
    dgradfunc = Function[{r, s}, {d + 2 a r^2 + c s^2, e + c r^2 + 2 b s^2}],
    a, b, c, d, e, f, sign, finalPoint,
    signum = If[# ≥ 0, 1, -1] &,
    σr, σs,
    discriminant,
    discriminantGradient,
    orthogonalStep,
    diagonalStep
   },
   {r, s} = position;
   {a, b, c, d, e, f, sign, finalPoint} = iterationconstants;
   discriminant = dfunc[r, s];
   discriminantGradient = dgradfunc[r, s];
   {σr, σs} = sign {{0, -1}, {1, 0}}.(signum /@ discriminantGradient);
   orthogonalStep = If[Abs[r discriminantGradient[[1]]] < Abs[s discriminantGradient[[2]]],
     {σr, 0}, {0, σs}
    ];
   diagonalStep = {σr, σs};
   Append[
    If[Abs[dfunc @@ (position + orthogonalStep)] < Abs[dfunc @@ (position + diagonalStep)],
     {position + orthogonalStep,
      {dfunc @@ (position + orthogonalStep), dgradfunc @@ (position + orthogonalStep)}},
     {position + diagonalStep, {dfunc @@ (position + diagonalStep),
       dgradfunc @@ (position + diagonalStep)}}
    ],
    iterationconstants]
  ]
 ]

Function[{position, iterationvariables, iterationconstants},
 Block[{r, s, dfunc = Function[{r, s}, f + d r^2 + a r^4 + e s^2 + c r^2 s^2 + b s^4],
   dgradfunc = Function[{r, s}, {d + 2 a r^2 + c s^2, e + c r^2 + 2 b s^2}], a, b,
   c, d, e, f, sign, finalPoint, signum = If[#1 ≥ 0, 1, -1] &, σr, σs,
   discriminant, discriminantGradient, orthogonalStep, diagonalStep},
  {r, s} = position; {a, b, c, d, e, f, sign, finalPoint} = iterationconstants;
  discriminant = dfunc[r, s]; discriminantGradient = dgradfunc[r, s];
  {σr, σs} = sign {{0, -1}, {1, 0}}.signum /@ discriminantGradient; orthogonalStep =
   If[Abs[r discriminantGradient[[1]]] < Abs[s discriminantGradient[[2]]], {σr, 0}, {0, σs}];
  diagonalStep = {σr, σs}; Append[If[Abs[dfunc @@ (position + orthogonalStep)] <
      Abs[dfunc @@ (position + diagonalStep)], {position + orthogonalStep,
      {dfunc @@ (position + orthogonalStep), dgradfunc @@ (position + orthogonalStep)}},
     {position + diagonalStep, {dfunc @@ (position + diagonalStep),
       dgradfunc @@ (position + diagonalStep)}}], iterationconstants]]]
```

Real iteration: Discriminant and gradients are iterated

```
MuralizerIteration = Function[{position, iterationvariables, iterationconstants},
  Block[{r, s,
    discriminant,
    discriminantGradient,
    a, b, c, d, e, f, sign, finalPoint,
    signum = If[# ≥ 0, 1, -1] &,
    σr, σs,
    discriminantSteps,
    gradientSteps,
    gradients,
    orthogonalStep,
    diagonalStep,
    discriminantValues
   },
   {r, s} = position;
   {discriminant, discriminantGradient} = iterationvariables;
   {a, b, c, d, e, f, sign, finalPoint} = iterationconstants;
   {σr, σs} = sign {{0, -1}, {1, 0}}.(signum /@ discriminantGradient);
   discriminantSteps = {
     a (6 r^2 + 1 + σr 4 r (r^2 + 1)) + (c s^2 + d) (1 + σr 2 r),
     b (6 s^2 + 1 + σs 4 s (s^2 + 1)) + (c r^2 + e) (1 + σs 2 s),
     c (2 r + σr) (2 s + σs)
    };
   gradientSteps = {
     (1 + σr 2 r) {2 a, c},
     (1 + σs 2 s) {c, 2 b}
    };
   orthogonalStep = (* {position step, discriminant step, discriminant gradient step} *)
    If[Subtract @@ Abs[discriminantGradient {r, s}] < 0,
     {{σr, 0}, {discriminantSteps⟦1⟧, gradientSteps⟦1⟧}},
     {{0, σs}, {discriminantSteps⟦2⟧, gradientSteps⟦2⟧}}];
   diagonalStep = {
     {σr, σs},
     {Plus @@ discriminantSteps, Plus @@ gradientSteps}};
   discriminantValues = discriminant + {orthogonalStep⟦2, 1⟧, diagonalStep⟦2, 1⟧};
   Append[{position, iterationvariables} +
     If[Subtract @@ Abs[discriminantValues] < 0,
      orthogonalStep,
      diagonalStep
     ],
    iterationconstants]
  ]
 ]
```

```
Function[{position, iterationvariables, iterationconstants},
 Block[{r, s, discriminant, discriminantGradient, a, b, c, d, e, f,
   sign, finalPoint, signum = If[#1 ≥ 0, 1, -1] &, σr, σs, discriminantSteps,
   gradientSteps, gradients, orthogonalStep, diagonalStep, discriminantValues},
  {r, s} = position; {discriminant, discriminantGradient} = iterationvariables;
  {a, b, c, d, e, f, sign, finalPoint} = iterationconstants;
  {σr, σs} = sign {{0, -1}, {1, 0}}.signum /@ discriminantGradient;
  discriminantSteps = {a (6 r^2 + 1 + σr 4 r (r^2 + 1)) + (c s^2 + d) (1 + σr 2 r),
    b (6 s^2 + 1 + σs 4 s (s^2 + 1)) + (c r^2 + e) (1 + σs 2 s), c (2 r + σr) (2 s + σs)};
  gradientSteps = {(1 + σr 2 r) {2 a, c}, (1 + σs 2 s) {c, 2 b}};
  orthogonalStep = If[Subtract @@ Abs[discriminantGradient {r, s}] < 0,
    {{σr, 0}, {discriminantSteps[[1]], gradientSteps[[1]]}},
    {{0, σs}, {discriminantSteps[[2]], gradientSteps[[2]]}}];
  diagonalStep = {{σr, σs}, {Plus @@ discriminantSteps, Plus @@ gradientSteps}};
  discriminantValues = discriminant + {orthogonalStep[[2, 1]], diagonalStep[[2, 1]]};
  Append[{position, iterationvariables} + If[Subtract @@ Abs[discriminantValues] < 0,
     orthogonalStep, diagonalStep], iterationconstants]]]
```

Termination criterion

```
continueDrawing = Function[{position, iterationvariables, iterationconstants},
  Block[{
    gradient = iterationvariables[[2]],
    finalpoint = Last[iterationconstants],
    distance
   },
   distance = (position - finalpoint).{{0, -1}, {1, 0}}.gradient;
   2 distance^2 > gradient.gradient
  ]
 ]
```

```
Function[{position, iterationvariables, iterationconstants},
 Block[{gradient = iterationvariables[[2]], finalpoint = Last[iterationconstants], distance},
  distance = (position - finalpoint).{{0, -1}, {1, 0}}.gradient;
  2 distance^2 > gradient.gradient]]
```

- **testrun**

- **Inverse transformation (to check results)**

```
inversexform[refpoints_] := Function[{radii},
  Block[{ra = radii[[1]], rb = radii[[2]], xa = refpoints[[1, 1]],
    ya = refpoints[[1, 2]], xb = refpoints[[2, 1]], yb = refpoints[[2, 2]]},
   {(xa + xb)/2 + ((ra - rb) (ra + rb) (-xa + xb))/(2 ((-xa + xb)^2 + (-ya + yb)^2)) +
     ((-ya + yb) Sqrt[-(-(ra - rb)^2 + (-xa + xb)^2 + (-ya + yb)^2) (-(ra + rb)^2 + (-xa + xb)^2 + (-ya + yb)^2)])/(2 ((-xa + xb)^2 + (-ya + yb)^2)),
    (ya + yb)/2 + ((ra - rb) (ra + rb) (-ya + yb))/(2 ((-xa + xb)^2 + (-ya + yb)^2)) -
     ((-xa + xb) Sqrt[-(-(ra - rb)^2 + (-xa + xb)^2 + (-ya + yb)^2) (-(ra + rb)^2 + (-xa + xb)^2 + (-ya + yb)^2)])/(2 ((-xa + xb)^2 + (-ya + yb)^2))}
  ]
 ]
```

- **"cheat" transform**

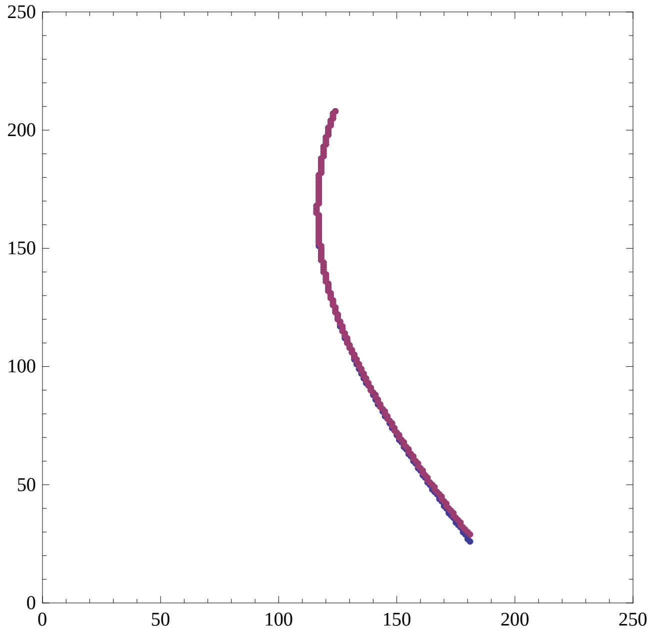
```
auxc = NestWhileList[MuralizerIterationCheat @@ # &, InitIteration[{{-100, 0}, {100, 0}}][
    {{-70, -120}, {80, -20}}], continueDrawing @@ # &, 1, 200];
```

- **real transform**

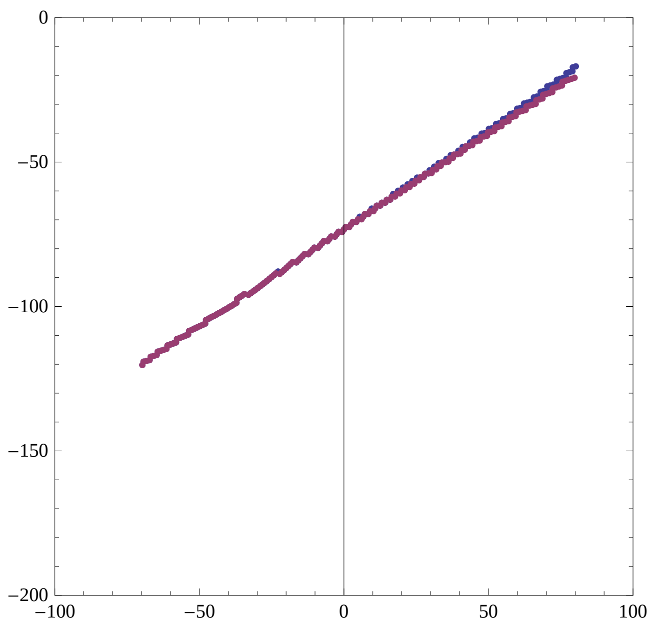
```
auxl = NestWhileList[MuralizerIteration @@ # &, InitIteration[{{-100, 0}, {100, 0}}][
    {{-70, -120}, {80, -20}}], continueDrawing @@ # &, 1, 200];
```

Evaluation

```
ListPlot[{First /@ auxl, First /@ auxc},
 Frame → True, PlotRange → {#, #} &[{0, 250}], AspectRatio → 1]
```



```
ListPlot[Map[inversexform[{{-100, 0}, {100, 0}}][First[#]] &, {auxl, auxc}, {2}],
 Frame → True, PlotRange → {{-100, 100}, {-200, 0}}, AspectRatio → 1]
```



```
Last[First[auxl]]
Last[First[auxc]]
```

{32 500, 32 500, -65 000, -2 680 000 000, -920 000 000, 55 360 000 000 000, 1, {181, 28}}

{32 500, 32 500, -65 000, -2 680 000 000, -920 000 000, 55 360 000 000 000, 1, {181, 28}}

```
Dimensions /@ {auxl, auxc}
```

```
{{183, 3}, {180, 3}}
```

■ several runs

```
coords = Table[{{Random[Real, {-100, 100}], Random[Real, {-200, 0}]},
    {Random[Real, {-100, 100}], Random[Real, {-200, 0}]}}, {n, 42}]
```

```
{{{-4.97228, -140.049}, {-79.0617, -24.8791}}, {{-55.363, -146.738}, {54.377, -64.9355}},
 {{-21.4175, -83.2669}, {-27.897, -197.281}}, {{87.927, -10.6215}, {13.7911, -156.958}},
 {{59.046, -4.45173}, {35.1657, -195.984}}, {{-71.9742, -6.67087}, {87.2992, -186.808}},
 {{32.9981, -66.6218}, {66.3609, -161.929}}, {{-11.6389, -119.883}, {-88.0161, -96.9935}},
 {{-90.2214, -36.6165}, {39.8809, -99.7124}}, {{-78.1484, -25.995}, {-73.9102, -142.754}},
 {{-37.1944, -21.5433}, {-9.07593, -146.77}}, {{-65.2202, -14.8724}, {3.62488, -159.962}},
 {{1.78176, -148.251}, {37.264, -198.033}}, {{-86.5793, -28.3672}, {25.2802, -101.04}},
 {{-96.3579, -191.751}, {85.3993, -1.32763}}, {{81.7904, -165.756}, {59.3095, -58.5735}},
 {{18.9848, -144.212}, {-31.6146, -111.803}}, {{-15.795, -129.34}, {64.7605, -151.841}},
 {{82.4232, -181.089}, {-72.5035, -153.807}}, {{69.0026, -152.722}, {2.21635, -52.7671}},
 {{65.3605, -160.971}, {16.8171, -51.4394}}, {{83.5701, -195.216}, {57.5076, -192.866}},
 {{-35.4147, -51.0035}, {-10.8778, -81.0629}}, {{80.3803, -121.664}, {24.3616, -129.222}},
 {{97.9571, -140.574}, {-3.1349, -175.415}}, {{-71.0455, -187.852}, {94.6487, -122.648}},
 {{-36.406, -26.8808}, {-22.1683, -71.2088}}, {{-19.976, -31.665}, {20.3241, -78.3429}},
 {{-84.5613, -180.662}, {-68.7981, -197.28}}, {{-64.9416, -58.9979}, {6.84032, -68.0576}},
 {{-62.8987, -118.424}, {-90.0248, -92.6423}}, {{-91.8532, -130.571}, {-84.6735, -169.994}},
 {{44.5528, -103.69}, {37.4948, -98.7853}}, {{-35.4712, -72.0254}, {-82.8293, -20.4425}},
 {{-50.9099, -91.3639}, {85.9688, -23.1625}}, {{-85.9683, -32.3661}, {-20.8715, -155.105}},
 {{76.9304, -113.943}, {-30.8467, -62.4626}}, {{68.7836, -183.371}, {-46.1732, -92.4685}},
 {{-75.7692, -79.6808}, {16.332, -193.683}}, {{59.702, -7.65538}, {-0.838781, -173.241}},
 {{10.6119, -116.291}, {13.1924, -150.078}}, {{-3.4198, -83.9254}, {-65.9361, -194.973}}}
```

```
cheats = NestWhileList[MuralizerIterationCheat @@ # &,
     InitIteration[{{-100, 0}, {100, 0}}][#], continueDrawing @@ # &, 1, 300] & /@ coords;
Dimensions /@
 cheats
```

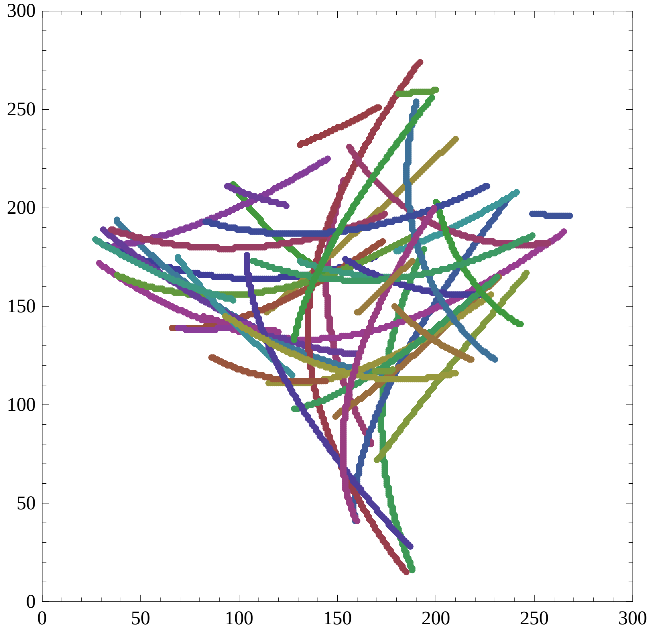
```
{{137, 3}, {135, 3}, {97, 3}, {164, 3}, {166, 3}, {237, 3}, {84, 3}, {58, 3},
 {136, 3}, {112, 3}, {108, 3}, {154, 3}, {62, 3}, {131, 3}, {260, 3}, {96, 3},
 {57, 3}, {71, 3}, {106, 3}, {114, 3}, {104, 3}, {20, 3}, {39, 3}, {40, 3}, {67, 3},
 {134, 3}, {37, 3}, {59, 3}, {20, 3}, {64, 3}, {31, 3}, {41, 3}, {10, 3}, {71, 3},
 {149, 3}, {140, 3}, {118, 3}, {143, 3}, {144, 3}, {160, 3}, {29, 3}, {124, 3}}
```

```
curves = NestWhileList[MuralizerIteration @@ # &,
     InitIteration[{{-100, 0}, {100, 0}}][#], continueDrawing @@ # &, 1, 300] & /@ coords;
Dimensions /@
 curves
```

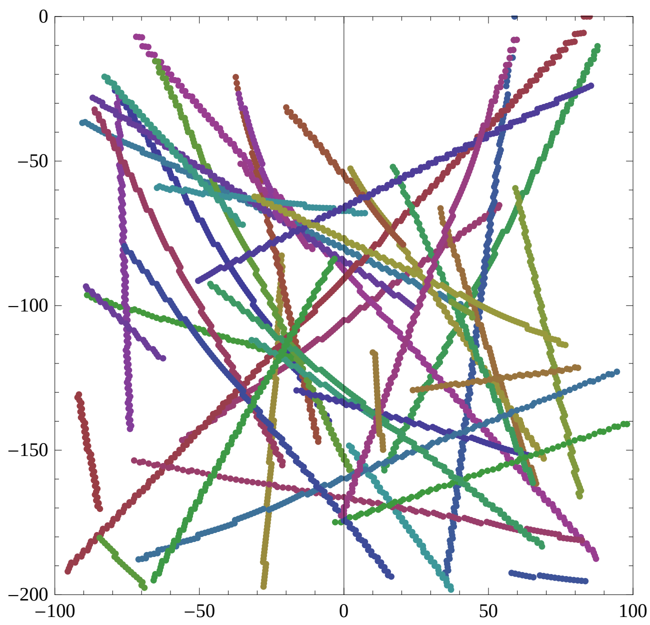
```
{{138, 3}, {135, 3}, {97, 3}, {164, 3}, {166, 3}, {237, 3}, {84, 3}, {58, 3},
 {136, 3}, {112, 3}, {108, 3}, {154, 3}, {62, 3}, {131, 3}, {263, 3}, {96, 3},
 {57, 3}, {71, 3}, {107, 3}, {114, 3}, {104, 3}, {20, 3}, {39, 3}, {40, 3}, {67, 3},
 {135, 3}, {37, 3}, {59, 3}, {20, 3}, {64, 3}, {31, 3}, {41, 3}, {10, 3}, {72, 3},
 {152, 3}, {140, 3}, {118, 3}, {143, 3}, {144, 3}, {160, 3}, {29, 3}, {124, 3}}
```

- **"cheat" function**

```
ListPlot[(First /@ #) & /@ cheats, Frame → True, PlotRange → {#, #} &[{0, 300}], AspectRatio → 1]
```

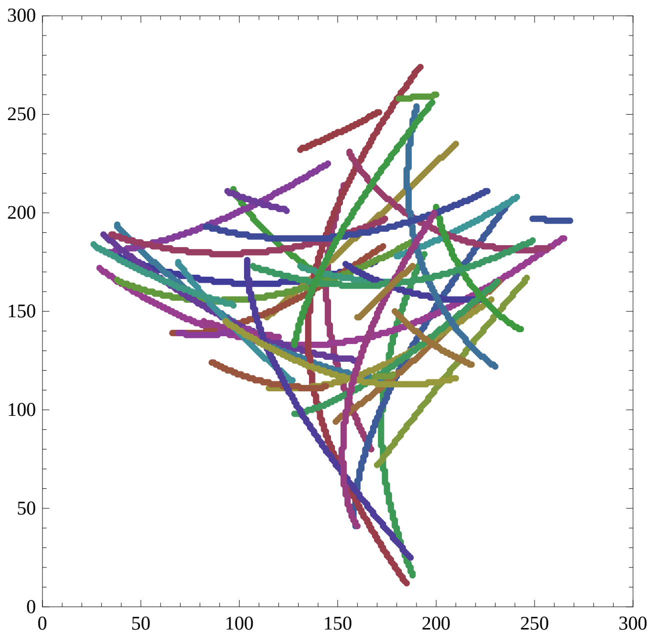



```
ListPlot[Map[inversexform[{{-100, 0}, {100, 0}}][First[#]] &, cheats, {2}],
 Frame → True, PlotRange → {{-100, 100}, {-200, 0}}, AspectRatio → 1]
```



- **function with iteration**

```
ListPlot[(First /@ #) & /@ curves, Frame → True, PlotRange → {#, #} &[{0, 300}], AspectRatio → 1]
```




```
ListPlot[Map[inversexform[{{-100, 0}, {100, 0}}][First[#]] &, curves, {2}],
 Frame → True, PlotRange → {{-100, 100}, {-200, 0}}, AspectRatio → 1]
```