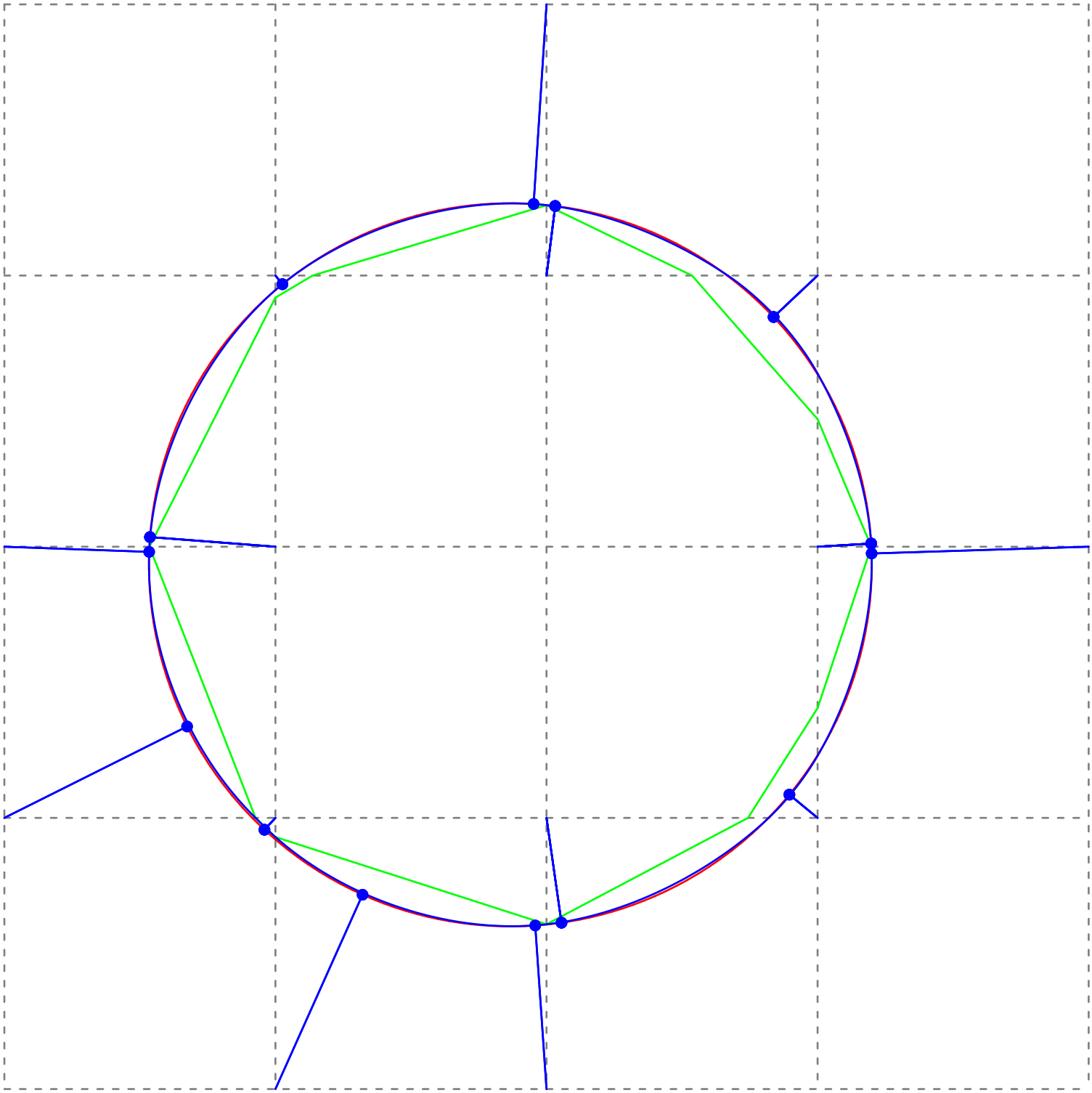
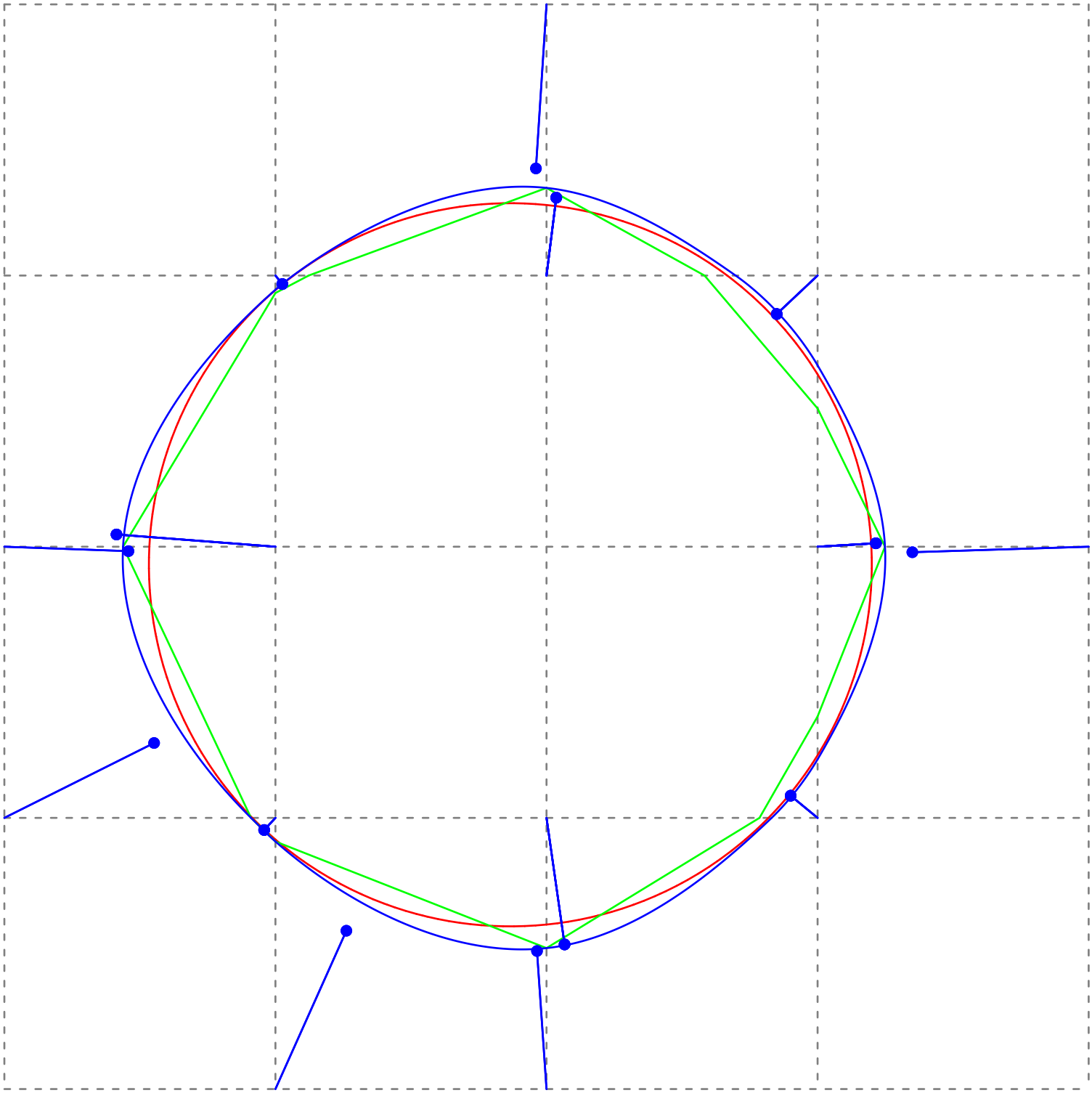
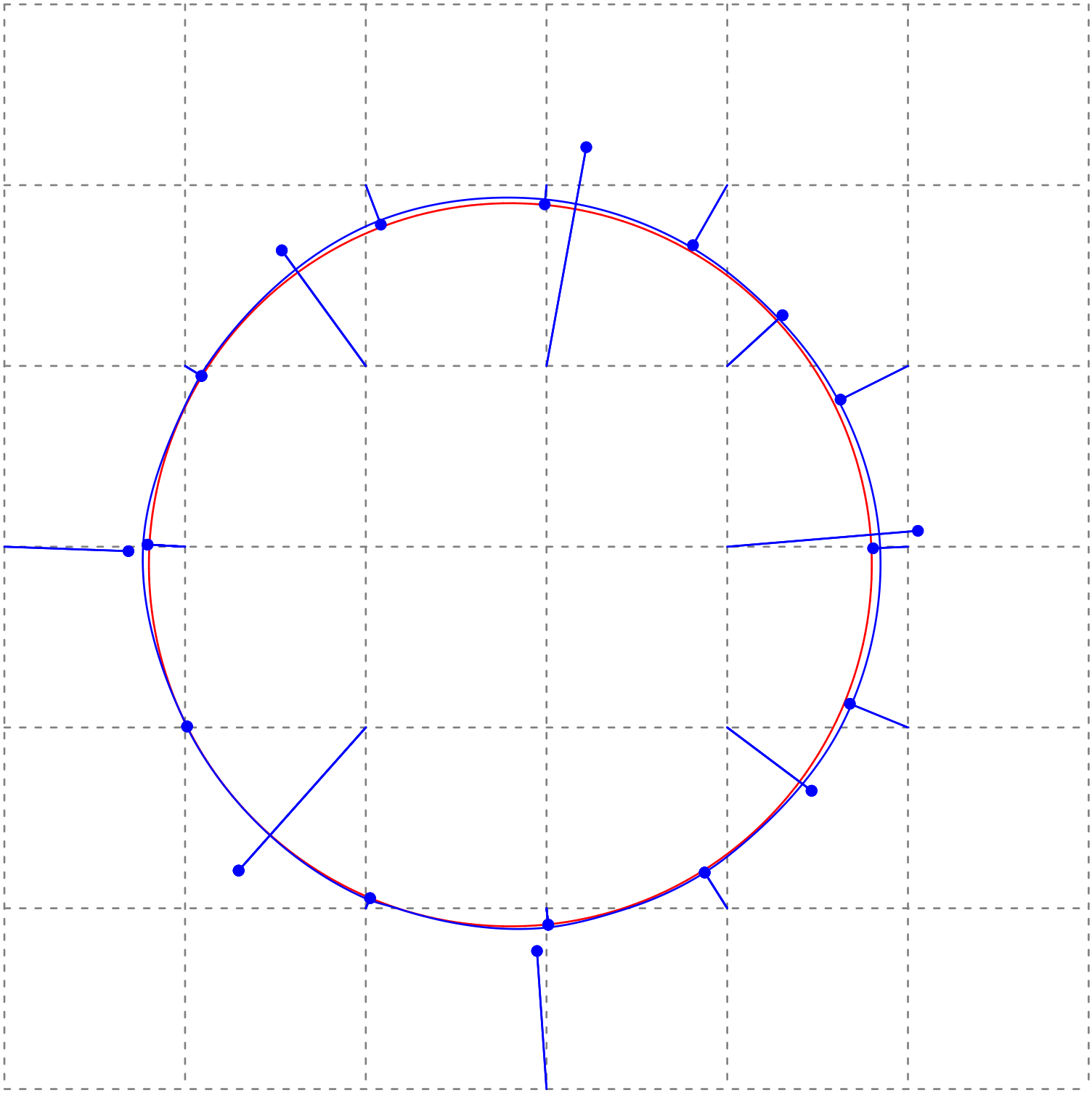


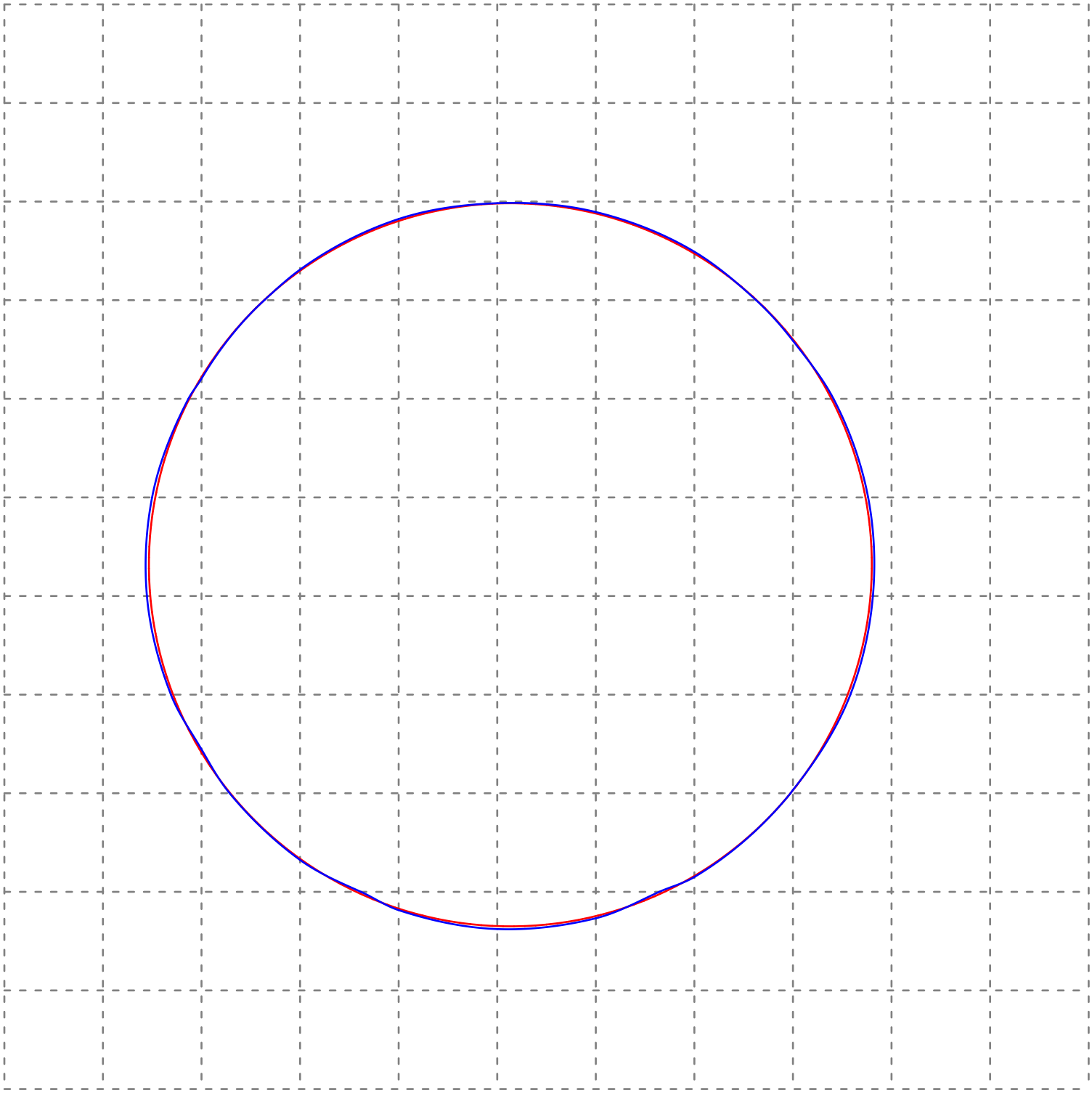
```
euclidean_distance = false
cells = 4
offsets = [-0.5, 0.5]
onsets = [-0.5, 0.5]

corner = [-1.4, -1.55]
bbox_edge = 3.0
ellipse = [1, 1]
curve_original = p -> sqrt((p[1]/ellipse[1])^2 + (p[2]/ellipse[2])^2) - 1

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red    green      blue      yellow   pink      cyan
```



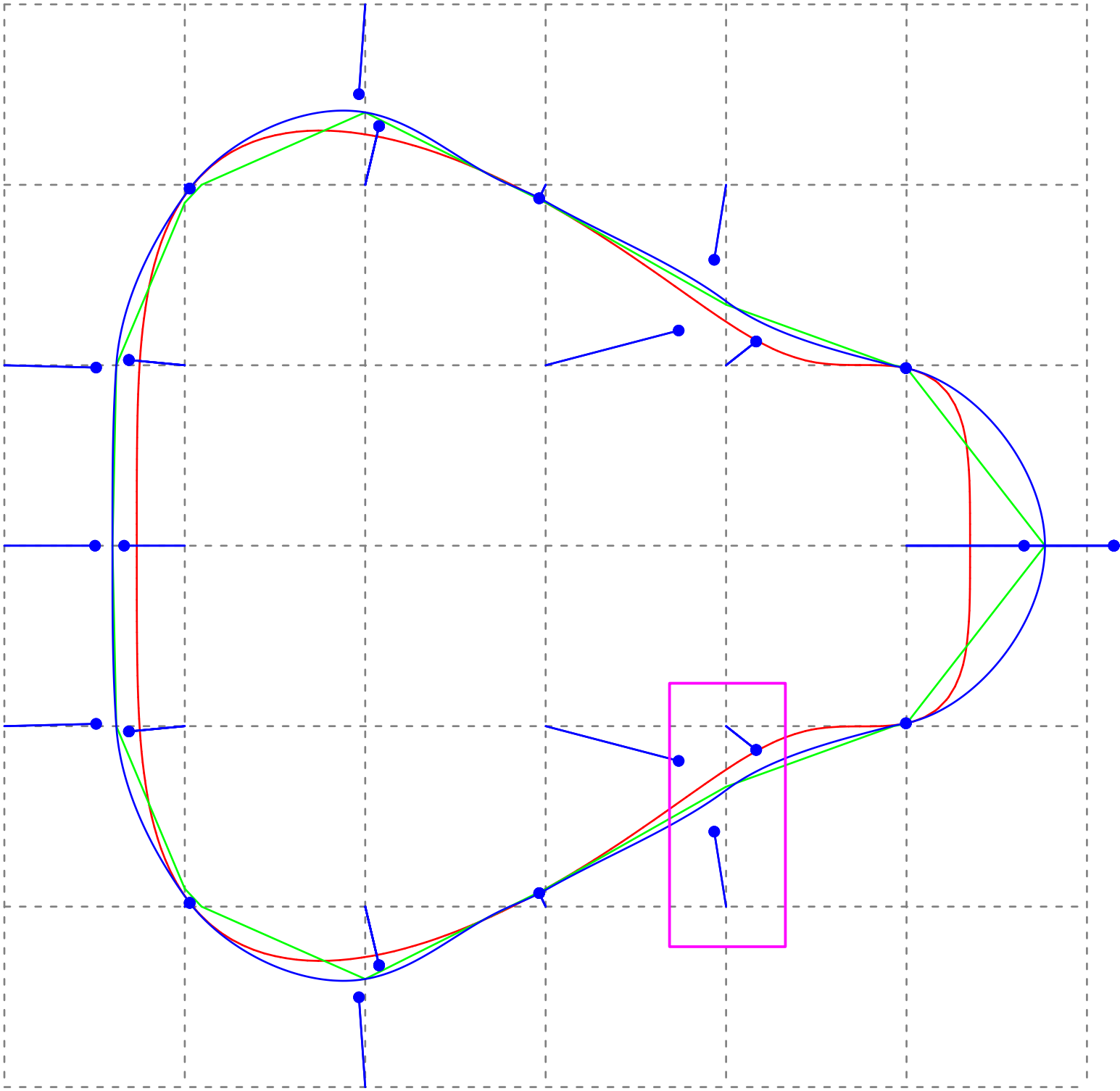
```
euclidean_distance = false
cells = 4
offsets = [-0.5, 0.5]
onsets = [-0.5, 0.5]

corner = [-1.4, -1.55]
bbox_edge = 3.0
ellipse = [1, 1]
curve_original = p -> (p[1]/ellipse[1])^2 + (p[2]/ellipse[2])^2 - 1

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red    green      blue      yellow   pink      cyan
```



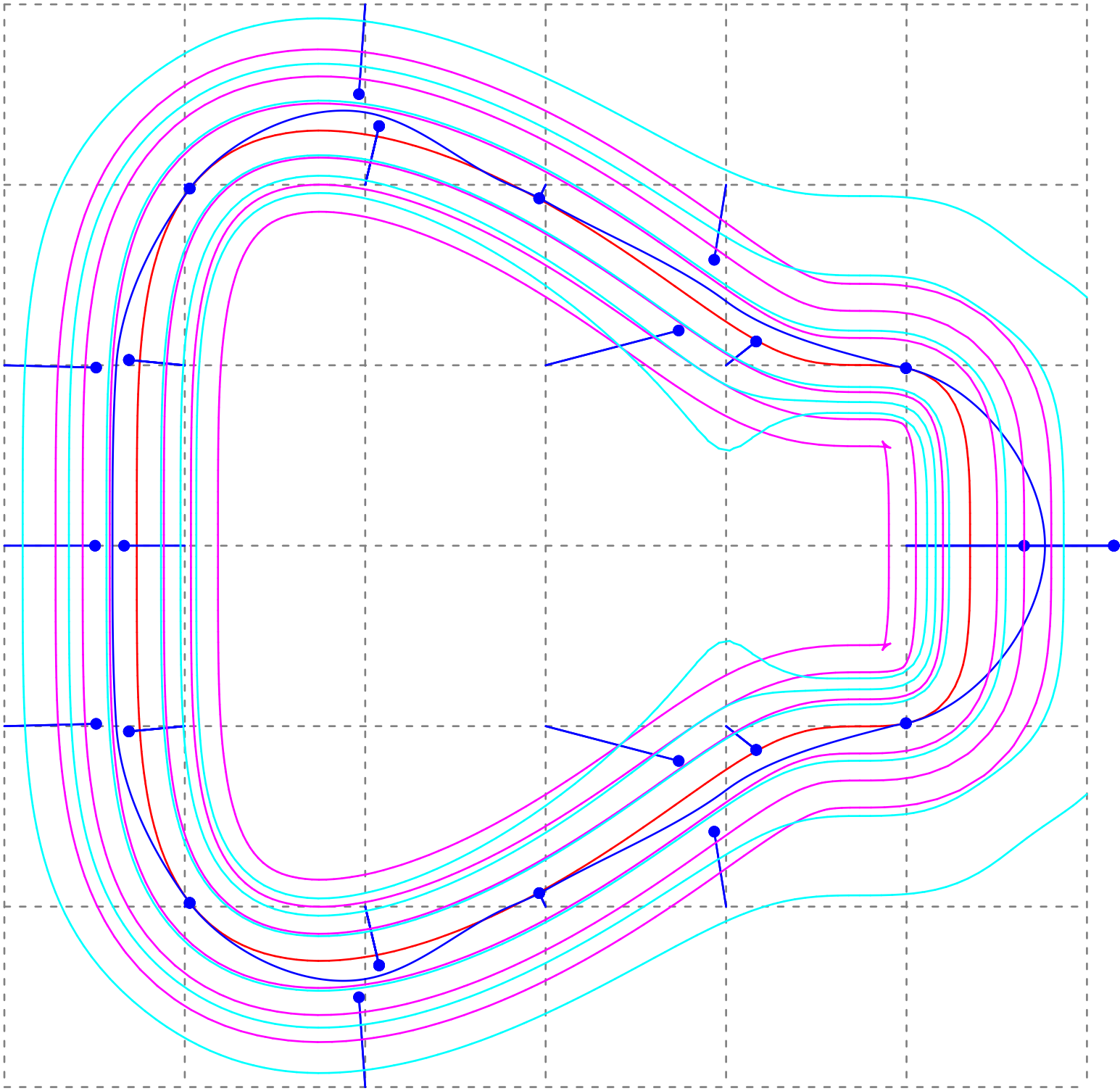
```
euclidean_distance = false
cells = 6
offsets = [-0.5, 0.5]
onsets = [-0.5, 0.5]

corner = [-1.4, -1.55]
bbox_edge = 3.0
ellipse = [1, 1]
curve_original = p -> (p[1]/ellipse[1])^2 + (p[2]/ellipse[2])^2 - 1

show_types = [:real :nolinear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow   pink      cyan
```



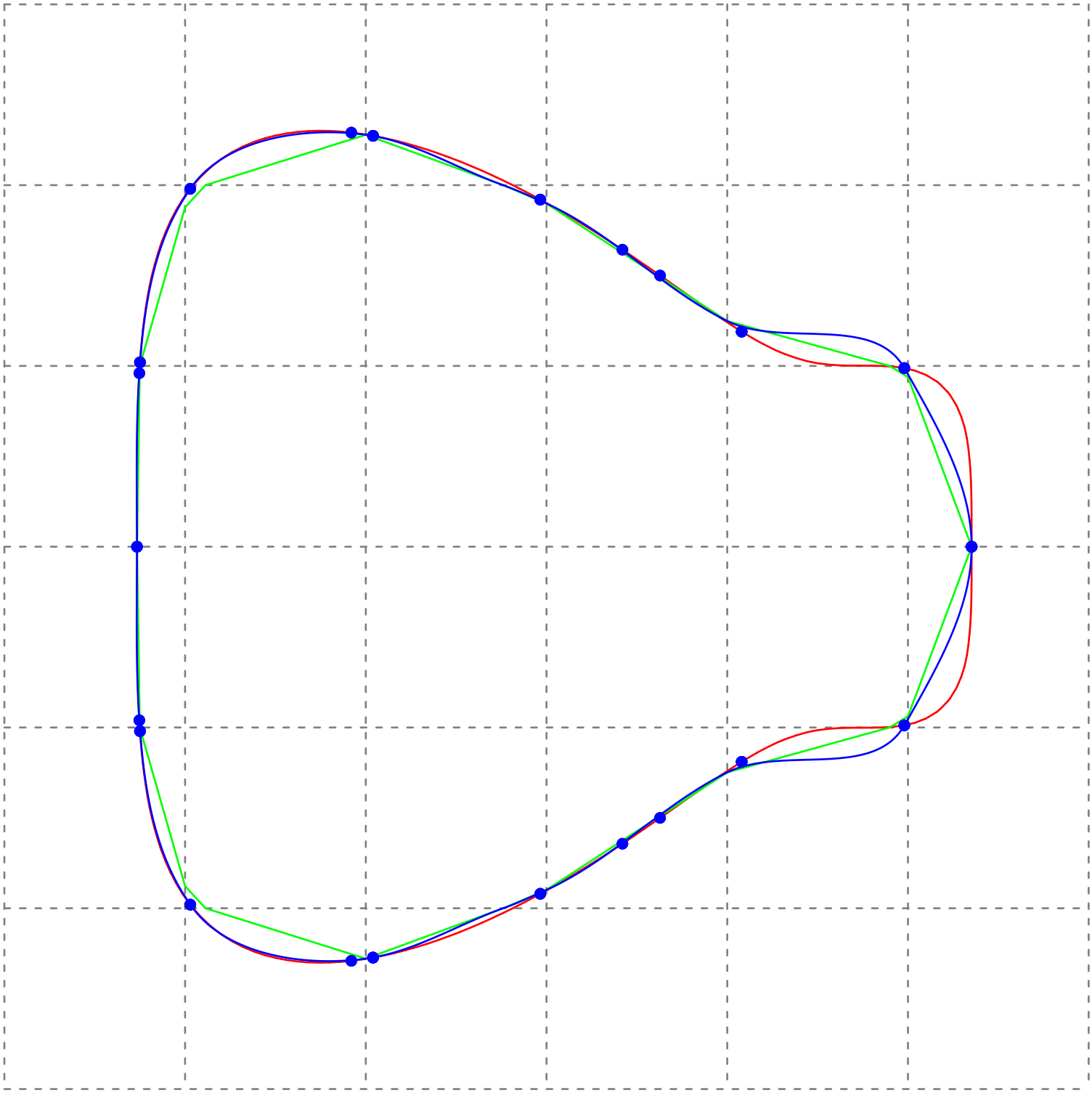
```
euclidean_distance = false
cells = 11
offsets = [-0.5, 0.5]
onsets = [-0.5, 0.5]

corner = [-1.4, -1.55]
bbox_edge = 3.0
ellipse = [1, 1]
curve_original = p -> (p[1]/ellipse[1])^2 + (p[2]/ellipse[2])^2 - 1

show_types = [:real :nolinear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow   pink      cyan
```



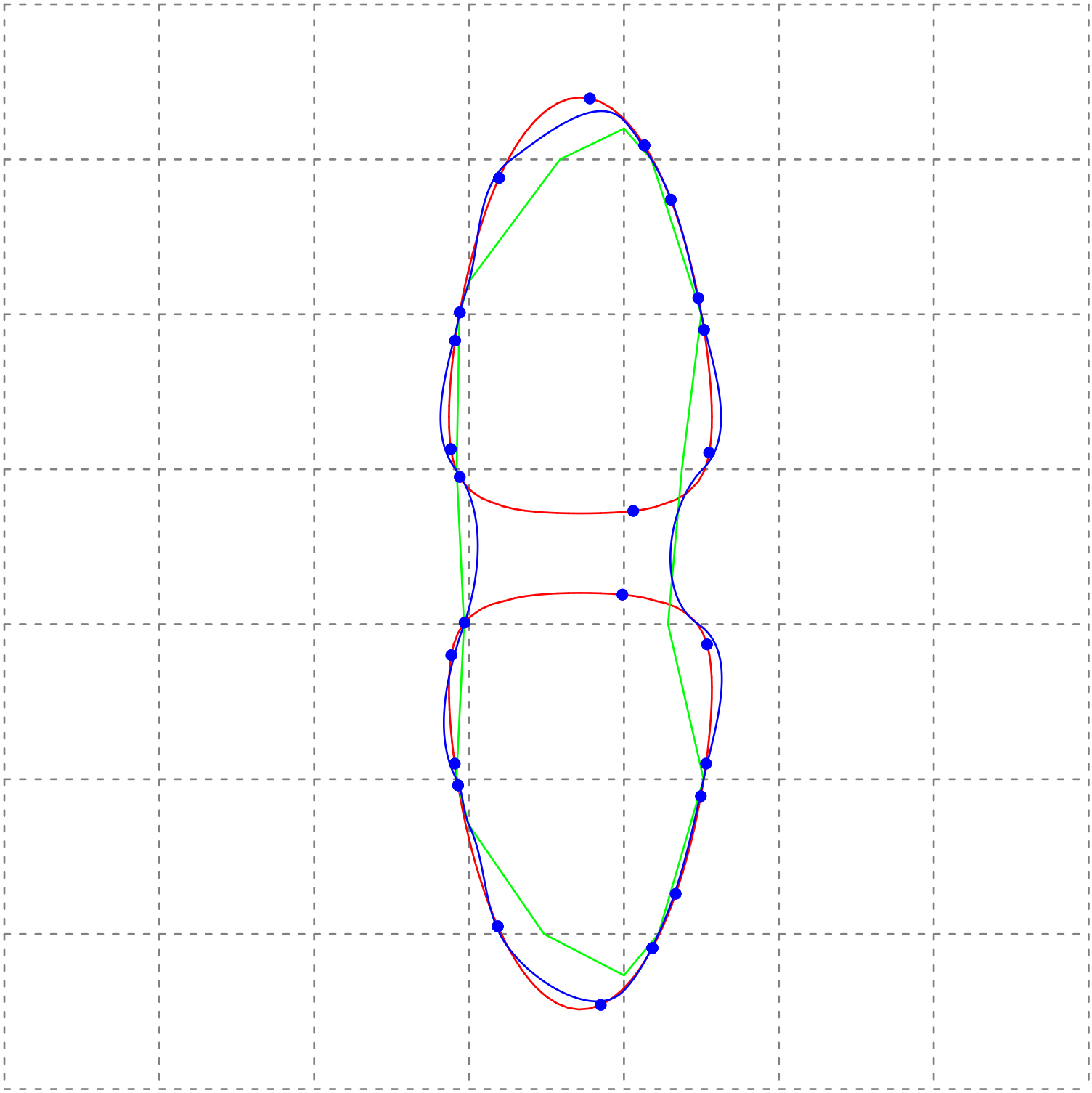
```
euclidean_distance = false
cells = 6
offsets = [-0.5, 0.5]
onsets = [-0.5, 0.5]

corner = [-9.5, -6]
bbox_edge = 12.0
curve_original = p -> p[1]^4 + 8p[1]^3 + p[2]^4 - 16

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#               red   green      blue      yellow   pink      cyan
```



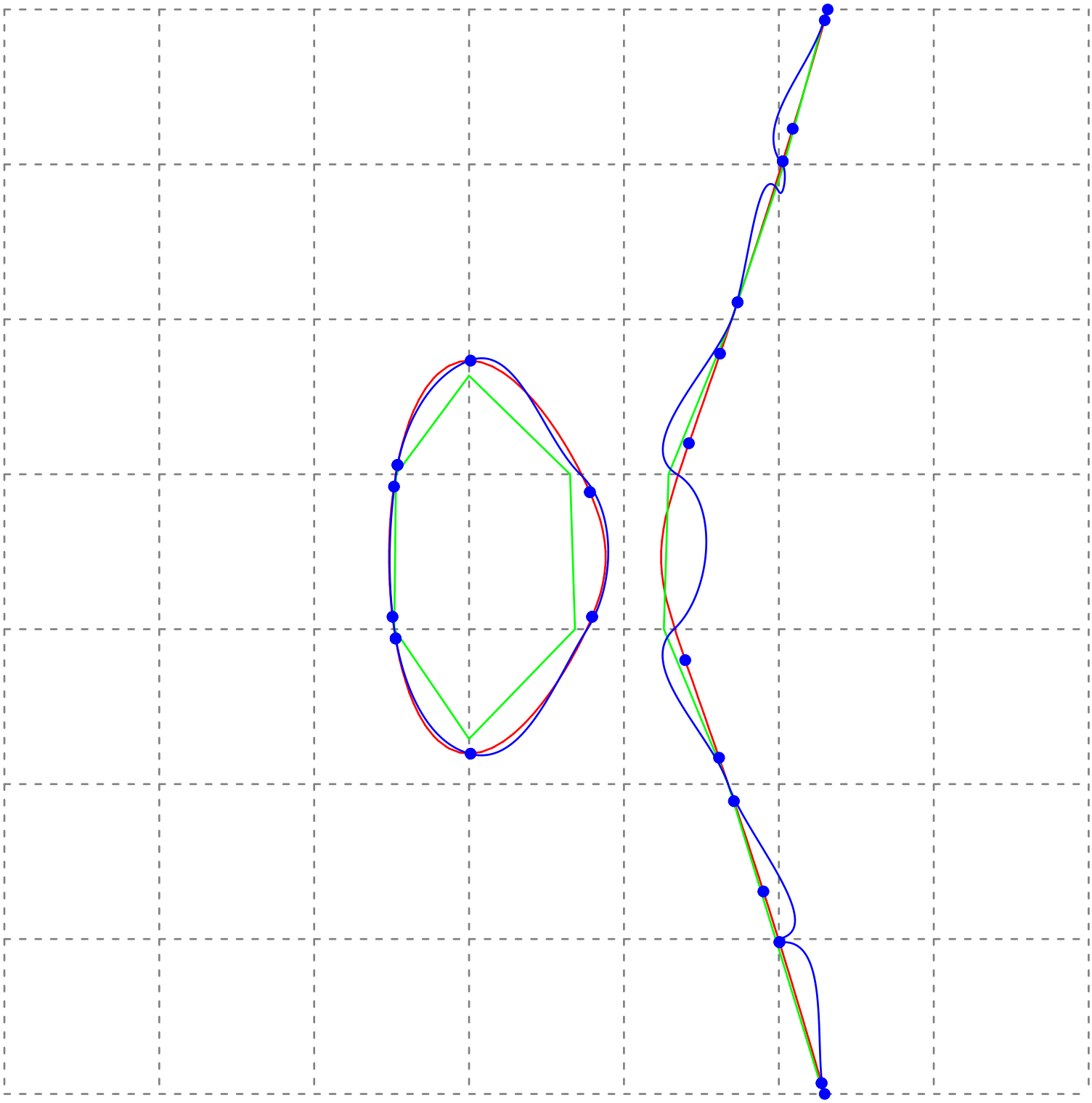
```
euclidean_distance = false
cells = 6
offsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]
onsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]

corner = [-9.5, -6]
bbox_edge = 12.0
curve_original = p -> p[1]^4 + 8p[1]^3 + p[2]^4 - 16

show_types = [:real :nolinear :liming_cubic :skip :offsets :onsets]
#              red     green      blue      yellow   pink    cyan
```



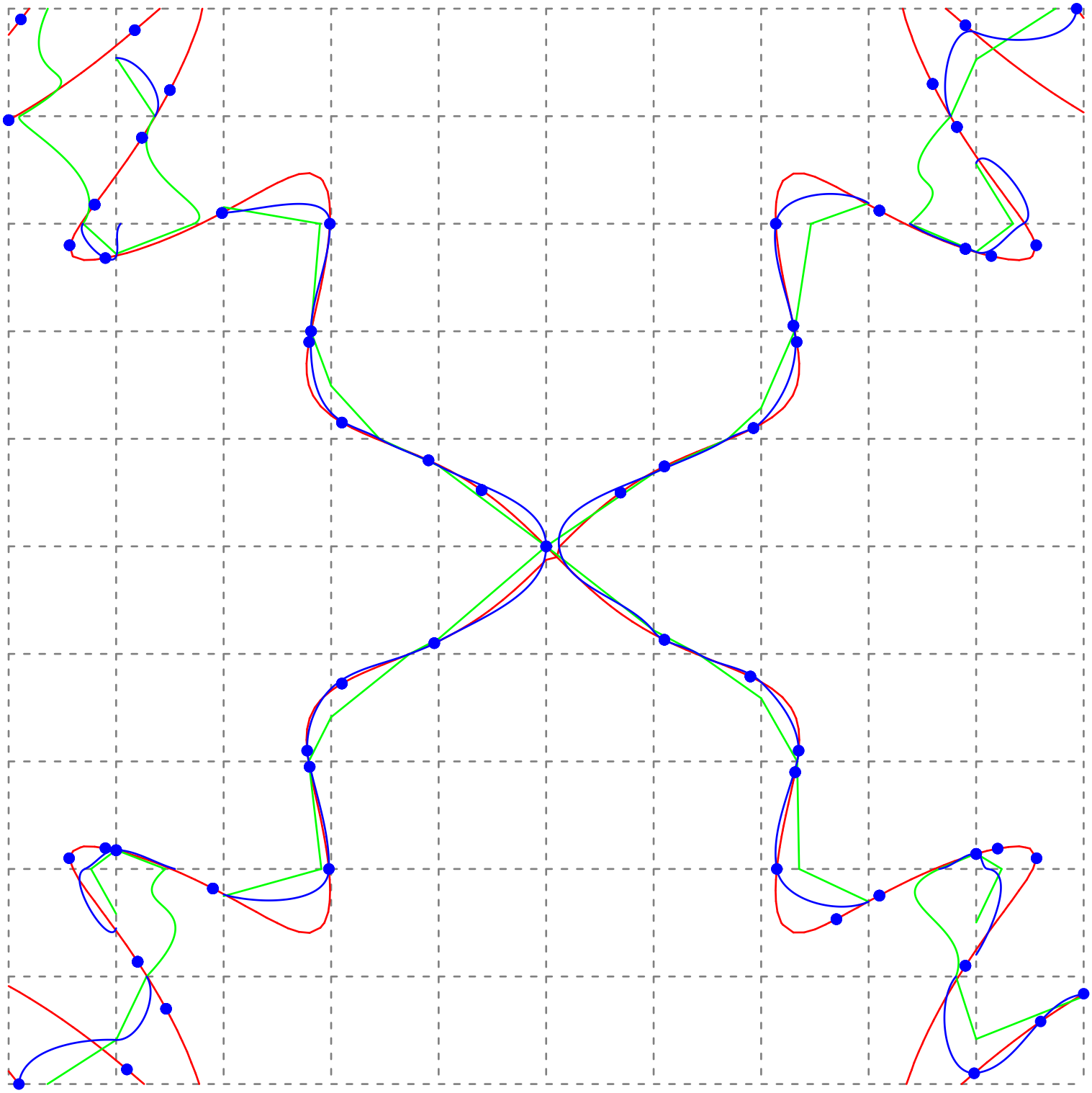
```
euclidean_distance = true
cells = 6
offsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]
onsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]

corner = [-9.5, -6]
bbox_edge = 12.0
curve_original = p -> p[1]^4 + 8p[1]^3 + p[2]^4 - 16

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow    pink      cyan
```



```
euclidean_distance = true
cells = 7
offsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]
onsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]

corner = [-4.05, -4.05]
bbox_edge = 8.0
curve_original = p -> -4p[1] + 10p[1]^2 + p[2]^(-2) + p[2]^2 - 11

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow    pink      cyan
```



```
euclidean_distance = true
cells = 7
offsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]
onsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]

corner = [-10.1, -10.1]
bbox_edge = 20.0
curve_original = p -> p[2]^2 - p[1]^3 + 7p[1] - 6

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow    pink      cyan
```



```
euclidean_distance = true
cells = 10
offsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]
onsets = [0.9, 0.6, 0.3, -0.3, -0.6, -0.9]

corner = [-4.05, -4.05]
bbox_edge = 8.0
curve_original = p -> p[1]^2 - p[2]^2 - p[1]*p[2]*sin(p[1]*p[2])

show_types = [:real :linear :liming_cubic :skip :nooffsets :noonsets]
#              red     green      blue      yellow    pink      cyan
```