Trabalho A1: Curvas Offset

Iara Cristina e Janaina Neres

Introdução

Para este trabalho, escolhemos trabalhar com Curvas Offset matematicamente e computacionalmente. A importância de Curvas offset se dá pelas aplicações em diversas áreas, como usinagem, modelagem geométrica, computação gráfica e robótica. Podemos citar alguns exemplos: criar caminhos para ferramentas de corte, criar superfícies com certa espessura, planejar trajetórias de movimento e definir regiões de tolerância em peças mecânicas.

O estudo das propriedades geométricas e topológicas de curvas offset pode levar a novos insights matemáticos que contribuem para o desenvolvimento de novas técnicas para sua manipulação e uso em aplicações práticas. Vamos começar definindo alguns conceitos de curvas que serão utilizados e discutidos:

• Para uma curva parametrizada regular $\alpha : I \to \mathbb{R}^2$ representada por $\alpha(t) = (x(t), y(t))$, sua derivada é $\alpha'(t) = (x'(t), y'(t))$

• Para uma curva parametrizada regular $\alpha : I \to \mathbb{R}^3$ representada por $\alpha(t) = (x(t), y(t), z(t))$, sua derivada é $\alpha'(t) = (x'(t), y'(t), z'(t))$

• Uma curva é regular, se $\alpha'(t) \neq 0 \; \forall \; t \in I$

• Uma curva é unit speed, se $||\alpha'(t)|| = 1$

• O vetor tangente T é definido por $\frac{\alpha}{||\alpha'(t)||}$ no caso geral.

• O vetor normal N é definido por $\frac{\alpha''(t)}{||\alpha''(t)||}$, ou também, $\frac{T'(t)}{|T'(t)|}$.

• O vetor binormal B é definido por $T(t) \times N(t)$

O que é uma curva offset?

A curva offset [4] é representada pela a seguinte expressão:

$$O_d(t) = C(t) + d * n(t)$$

• $O_d(t)$ é a curva offset resultante, que está a uma distância d da curva original $C(t)$.
• $C(t)$ é a curva original em sua forma parametrizada, que define a forma do objeto que se deseja fabricar.
• d é a distância fixa entre a curva original e a curva offset. Por exemplo, em um processo de usinagem, essa distância pode ser determinada pelo diâmetro da ferramenta de corte usada para cortar ao longo da curva original.
• $n(t)$ é o vetor normal unitário à curva original $C(t)$ na posição t. Este vetor é perpendicular à curva e indica a direção para a qual a curva se afasta da curva original. Multiplicando por d, obtemos um vetor que indica a direção e a magnitude da distância entre as duas curvas para cada valor de t.

Curvas Offset x Curvas Paralelas

Até aqui, deve ter surgido o questionamento de uma curva offset ser ou não o mesmo que uma curva paralela. Pode-se dizer que a curva offset é um conceito mais específico da curva paralela pelo seu contexto em desenho assistido por computador (CAD), mas na prática, os termos ”curvas offset”e ”curvas paralelas”são usados de forma intercambiável,uma vez que o objetivo é criar uma geometria que seja paralela a uma forma dada a uma distância fixa.

Singularidades de Curvas Offset

Existem dois tipos de singularidades nas curvas offset: pontos irregulares e auto-interseções.

Pontos irregulares

• Pontos isolados: Este ponto ocorre quando a curva progenitora com raio R é um círculo e o deslocamento é $d = -R$.
• Cusps [2]: Este ponto ocorre em um ponto t onde o vetor tangente desaparece, ou seja, a curvatura $\kappa(t)$ da curva é:

$$\kappa(t) = -\frac{1}{d}$$

Auto-interseções

As auto-interseções ocorrem se a distância da curva original exceder o raio de curvatura mínimo (curvatura) da curva que está sendo compensada. Por exemplo, se uma curva tiver um raio de 5 cm e você tentar compensar mais de 5 cm para dentro, a curva offset se cruzará e criará um loop.

Na usinagem, o raio do cortador não deve exceder o menor raio principal côncavo de curvatura da superfície para evitar goivagem local. Como a densidade dos caminhos da ferramenta é alta, a eficiência diminuirá significativamente.

Exemplos

Para uma primeira impressão, vamos ilustrar como as curvas offset se comportam para alguns exemplos em diferentes parâmetros e intervalos.

• Parábola

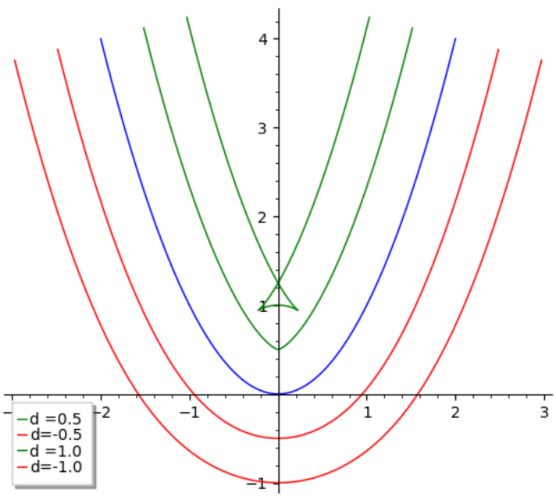

Figura 1: Exemplo de Curvas offset para $C(t) = (t, t^2)$

Para $d = -1$, ocorre uma **auto-interseção**.

• Involutas
A involuta de uma curva pode ser definida como a trajetória de um ponto que está preso em um fio que é enrolado na curva original. Visto isso, podemos representar a involuta como uma função de arco s e um parâmetro a que representa a distância do ponto inicial no fio à curva original: $\vec{C}_a(s) = \vec{c}(s) - \vec{c}'(s)(s-a)$, temos que $\vec{C}'_a(s) \cdot \vec{c}'(s) = 0$. Então as involutas são curvas paralelas, devido a $\vec{C}_a(s) = \vec{C}_0(s) + a\vec{c}'(s)$, onde a é a distância e $\vec{c}'(s)$ é $N(t)$ em $\vec{C}_0(s)$.

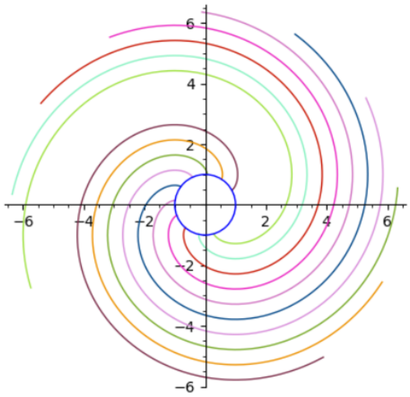

Figura 2: 10 involutas para $C(t) = (sen(t), cos(t))$

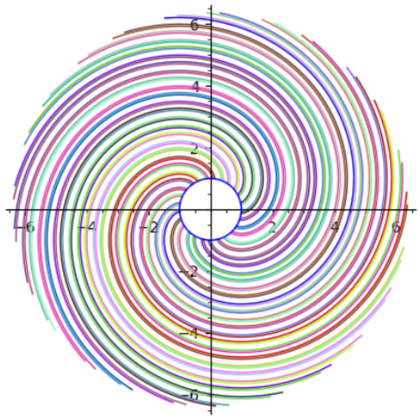

Figura 3: 100 involutas para $C(t) = (sen(t), cos(t))$

• Evolutas
A definição de Evolutas é o lugar geométrico de todos os seus centros de curvatura. Uma definição alternativa é tratar a evoluta de C como o lugar geométrico das "cusps"das suas curvas paralelas.

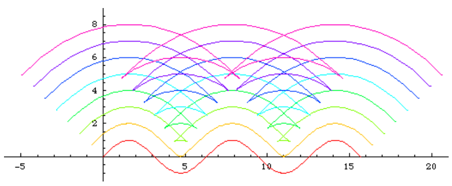

Figura 4: $C(t) = (t, sin(t))$ e offsets

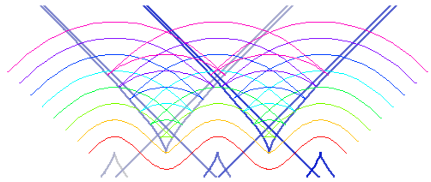

Figura 5: Sobreposição de $C(t) = (t, sin(t))$ e offsets com suas Evolutas

NC Machining & Ball-End Cutter

As curvas offset são importantes em "Numericall Controlled Machining"ou "Usinagem Controlada Numericamente", onde definem a forma do corte feito por uma ferramenta de corte redonda de uma máquina de dois eixos que é transladada da trajetória do cortador por uma distância constante na direção normal à trajetória do cortador em cada ponto. Uma dessas ferramentas de corte é a "Ball-End Nose Cutter", que é usada em Controle Numérico Computadorizado (CNC) que tem uma ponta esférica para criar superfícies curvas em objetos tridimensionais. Segue a função que gera uma animação para alguns exemplos:

```
def ball_end(C, d, t_range, x_range, y_range):
    T = diff(C, t) /diff(C, t).norm() # vetor tangente
    N = vector((- T[1], T[0])) # vetor normal
    Od = C + d*N # curva offset

    frames = [parametric_plot(Od, (t, t_range[0], t_range[1]), color="black") +
          parametric_plot(C, (t, t_range[0], t_range[1]), color = "blue") +
          plot(N.subs(t=i)*2*d, start = C.subs(t=i) ,  color = "red" ) +
          plot(circle(Od.subs(t=i), d, color = "blue")) for i in srange(t_range
    [0], t_range[1], 0.5)]
    return animate(frames,figsize=10,xmin=x_range[0], xmax=x_range[1], ymin=
    y_range[0], ymax=y_range[1])
```

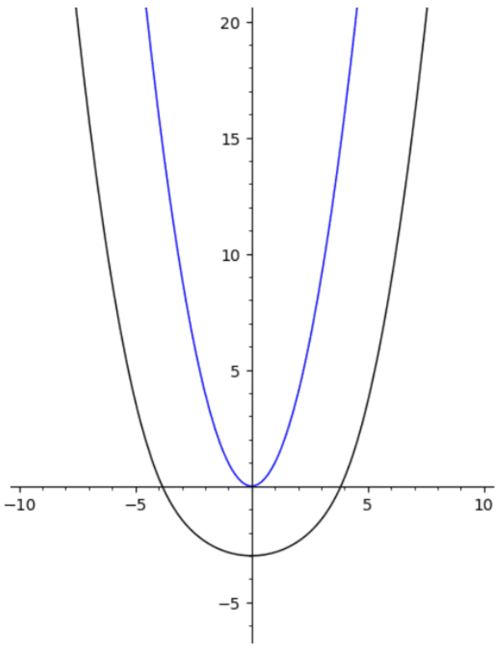

Figura 6: "ball-end curve cutter"ao longo do offset de $C(t) = (t, t^2)$

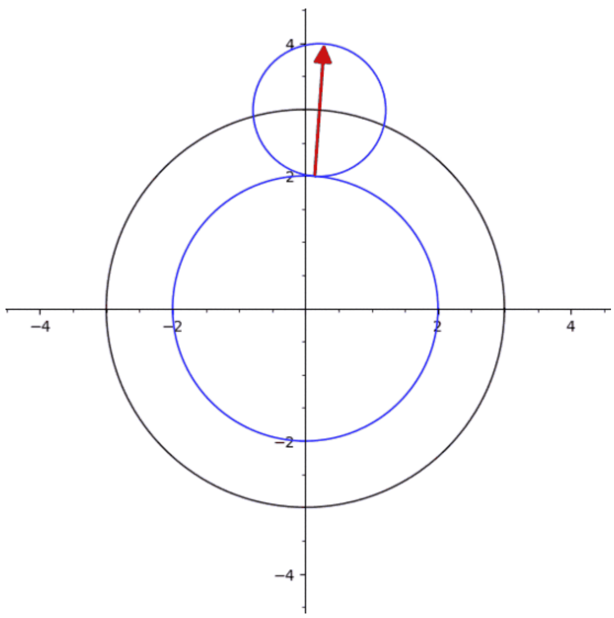

Figura 7: "ball-end curve cutter"ao longo do offset de $C(t) = (2\cos(t), 2\sin(t))$

Curvas Offset em Funções Implícitas?

Após investigar sobre o caso de curvas offset em funções implícitas, concluímos que geralmente a representação analítica de uma curva paralela de uma curva implícita não é possível. Mas existe uma forma de representar a distância orientada analiticamente. Para isso calculamos:

- Grad: O gradiente da função C.
- n: O gradiente da função C normalizado.
- N(x,y): Produto de cada componente de Grad com seu respectivo componente em n.
- Offset: Curva offset representada por $O(x, y) = C(x, y) + d * N(x, y)$

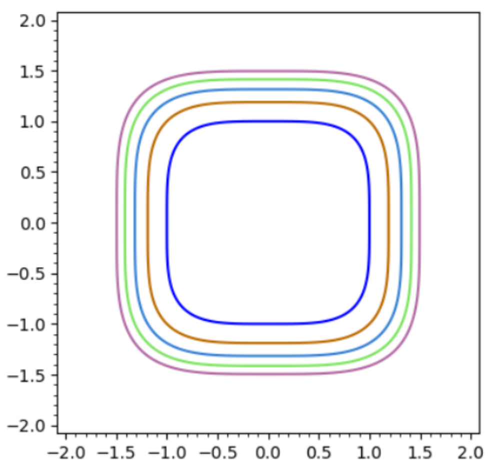

Figura 8: $F(x,y) = sen^4(x) + cos^4(y) - 1$

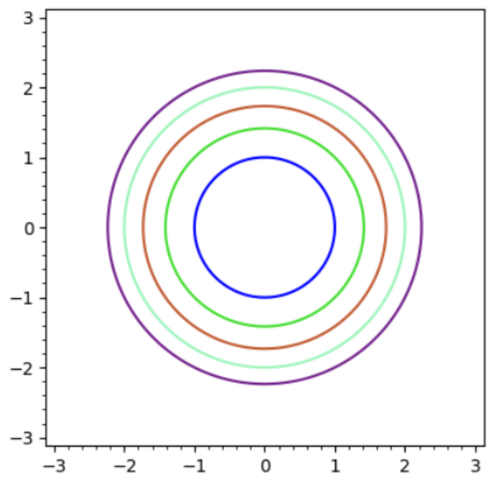

Figura 9: $F(x,y) = sen(x)^2 + cos(x)^2 - 1$

Apesar de obter os resultados desejados, não foi possível restringir a distância d como gostaríamos, e nem sempre obteremos curvas paralelas. Para mais detalhes [5].

Curvas 3D

A situação das curvas offset pode ser generalizada pro plano 3d. É possível mover a curva ao longo do vetor normal, mas também ao longo do vetor binormal para obter uma "curva binomial paralela", representada pela expressão:

$$C_d(t) = C(t) + d * B(t)$$

Podemos criar a curva offset normal e binormal de uma curva 3D através da função abaixo:

```
def normal_binormal_offsets(C, d, t_range):
    Ta = diff(alpha, t) / norm(diff(alpha, t))
    Na = diff(Ta, t) / norm(diff(Ta, t))
    Ba = Ta.cross_product(Na)

    # Principal Parallel Curve
    Alpha_N = alpha + d*Na

    # Binormal Parallel Curve
    Alpha_B = alpha + d*Ba

    #Plots 3d
    figures = []
    figures.append(parametric_plot(alpha, (t, -2*pi, 2*pi), color='blue'))
    figures.append(parametric_plot(Alpha_N, (t, -2*pi, 2*pi), color='black'))
    figures.append(parametric_plot(Alpha_B, (t, -2*pi, 2*pi), color='red'))

    show(sum(figures),frame=False)
```

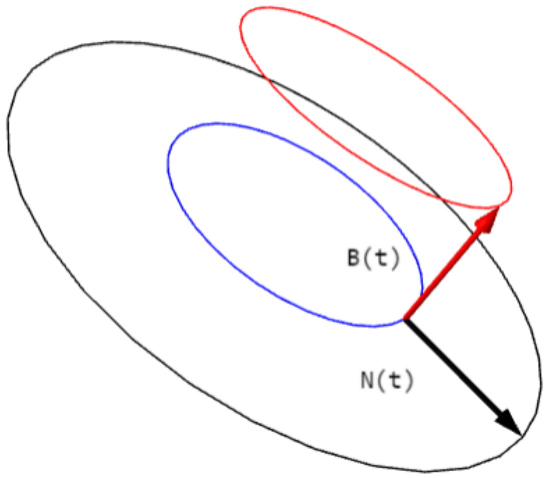

Figura 10: Exemplo $C = ((4/5) * cos(t), 1 - sin(t), (-3/5) * cos(t))$

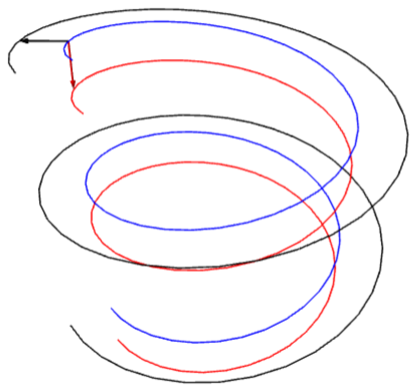

Figura 11: Exemplo $C = (3 * cos(2t), 3 * sin(2t), t)$

Para mais detalhes [1].

Exercícios

Os exercícios resolvidos são propostos na página 108 do livro [3] "Applied Geometry for Computer Graphics and CAD"do Duncan Marsh.

Começamos criando duas funções, a primeira retorna a equação da curva offset, a partir da curva original e o valor "d", enquanto a segunda gera os plots de múltiplas curvas offsets para vários valores de d.

```
def get_offset_curve(alpha, d):
    #---- INPUT ------
    # alpha: curva
    # d: distancia entre a curva original e offset
    #-----------------
    C(t) = alpha
    C_d(t) = diff(C(t), t)
    v_n(t) = (- C_d(t)[1], C_d(t)[0]) # vetor normal
    n(t) = v_n(t) / v_n(t).norm() # vetor normal unitario
    offset = C(t) + d*n(t) # aplica a formula da curva offset

    return offset
```

```
def plot_multiple_d(alpha, d, k, intervalo, size):
    figures = []
```

```
    curve = figures.append(parametric_plot(alpha, (t, intervalo[0], intervalo[1]), figsize=size, color="blue"))
    for i in range(1,k+1):
        d_k = d*i
        positive_offset = get_offset_curve(alpha, d_k)
        negative_offset = get_offset_curve(alpha, -d_k)

        if k < 4:
            figures.append(parametric_plot((positive_offset), (t, intervalo[0], intervalo[1]), figsize=size, color="green", legend_label=f"d ={float(str(d_k).rstrip('0').rstrip('.'))}"))
            figures.append(parametric_plot((negative_offset), (t, intervalo[0], intervalo[1]), figsize=size, color="red", legend_label=f"d ={float(str(-d_k).rstrip('0').rstrip('.'))}"))
        else:
            figures.append(parametric_plot((positive_offset), (t, intervalo[0], intervalo[1]), figsize=size, color="green"))
            figures.append(parametric_plot((negative_offset), (t, intervalo[0], intervalo[1]), figsize=size, color="red"))
    return sum(figures)
```

Questões

5.15. Determine a offset de $\mathbf{C}(t) = \left(1 - 3t + 3t^2, 3t^2 - 2t^3\right)$ em uma distância d. Plote a curva e sua offcet em uma distância $d = 1$.

5.16. Determine as offsets à uma distance d para as seguintes curvas:
(a) $(c\cosh(t/c), t)$
(b) $\left(e^{bt}\cos t, e^{bt}\sin t\right)$.
(c) $(\cos t + t\sin t, \sin t - t\cos t)$.

5.17. Mostre que a offset em uma distância d de um círculo de raio r é um círculo de raio $r + d$.

Soluções

5.15

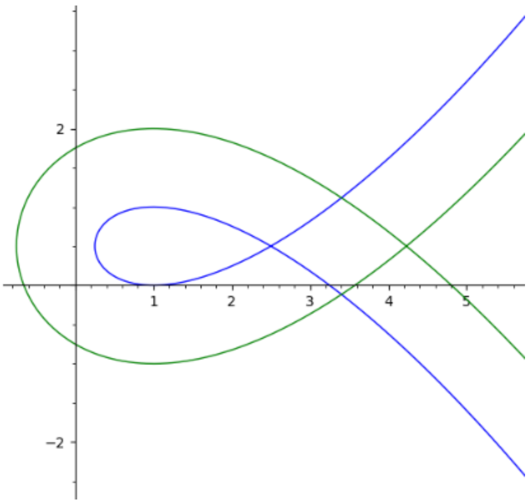

Figura 12: Exercício 5.15, d = 1

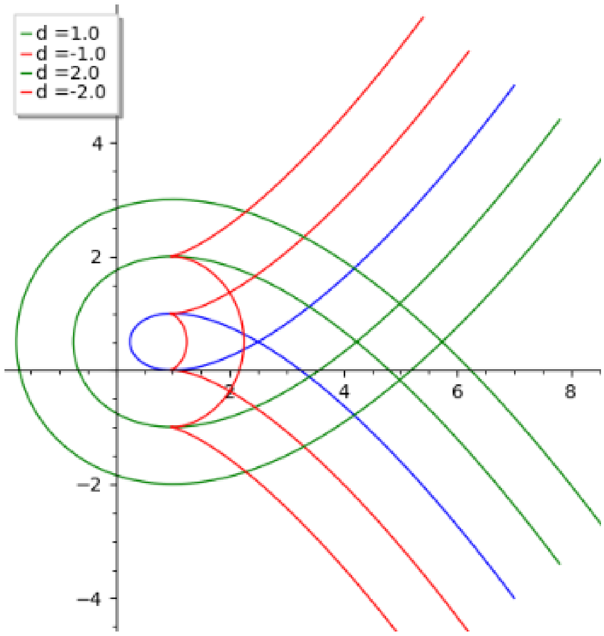

Figura 13: Exercício 5.15, outros valores de d

5.16, a)

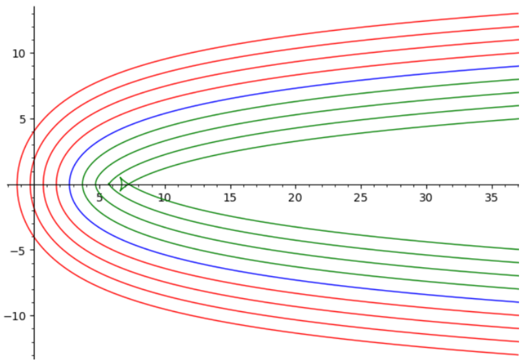

Figura 14: $C(t) = (c\cosh(t/c), t)$, $c = e$

b)

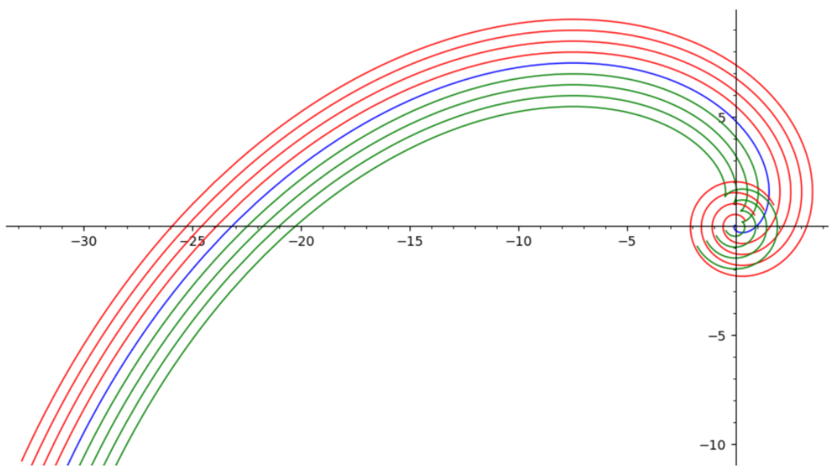

Figura 15: $C(t) = \left(e^{bt}\cos t, e^{bt}\sin t\right)$

c)

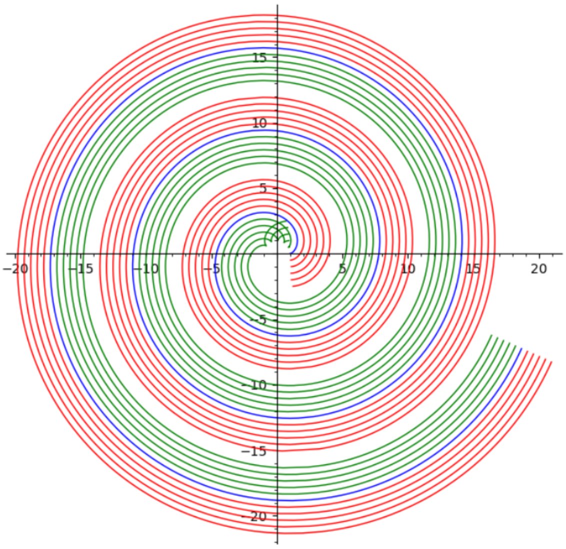

Figura 16: $(\cos t + t \sin t, \sin t - t \cos t)$

5.17

Para provar que a offset em uma distância d de um círculo de raio r é um círculo de raio $r+d$, seja a parametrização de um círculo de raio r, $C(t) = (r * cos(t), r * sin(t))$, calculamos a expressão da curva offset e mostramos que é a parametrização de um círculo de raio $(r + d)$

```
t,r, d = var("t r d")
assume(r > 0)

alpha(t) = (r*cos(t),r*sin(t))
offset = get_offset_curve(alpha, -d)
offset = vector((offset[0].full_simplify(), offset[1].full_simplify()))
show(offset)
```

$((d + r)\cos(t), (d + r)\sin(t))$

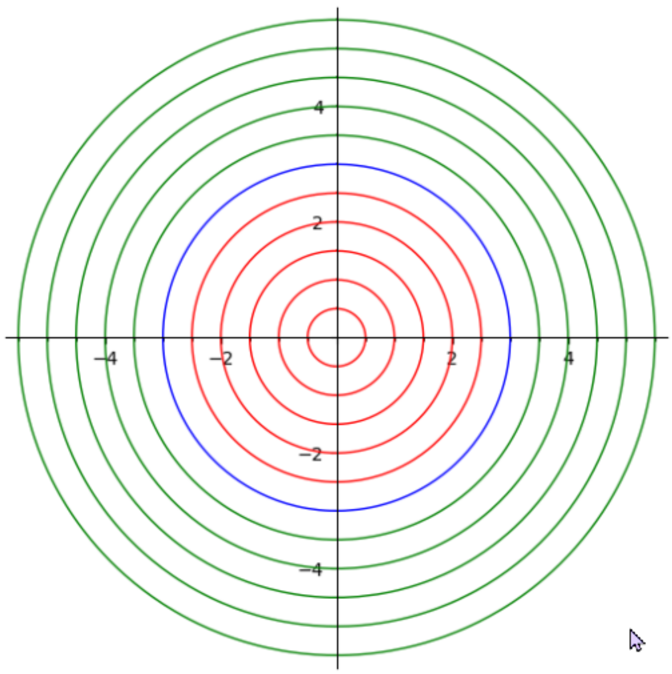

Figura 17: Exercício 5.17: para vários valores de d

Referências

[1] Brown. Ma106: Lab 3 - brown math. [Online; accessed 24-April-2023].

[2] Rida T Farouki and C Andrew Neff. Analytic properties of plane offset curves. *Computer aided geometric design*, 7(1-4):83–99, 1990.

[3] Erich Hartmann. Geometry and algorithms for computer aided design. *Darmstadt University of Technology*, 80, 2003.

[4] Duncan Marsh. *Applied geometry for computer graphics and CAD*. Springer Science & Business Media, 2005.

[5] Wikipedia contributors. Parallel curve — Wikipedia, the free encyclopedia, 2022. [Online; accessed 24-April-2023].