Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 7, n. 1, 2020.

Trabalho apresentado no XXXIX CNMAC, Uberlândia - MG, 2019.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Um Algoritmo para Construção de Conjuntos de Julia

Alisson Lucas de Souza[1]
Departamento de Matemática - UEM, Maringá - PR
Beatriz Aparecida Leopoldino da Silva[2], Bruna Alves da Silva[3], Paulo Henrique Rodrigues[4], Michele Cristina Valentino[5]
Departamento de Matemática, UTFPR, Câmpus Cornélio Procópio - PR

1 Introdução

O conjunto de Julia de uma função $f : \mathbb{C} \to \mathbb{C}$ consiste, informalmente, dos pontos cujo comportamento, em um período, sob repetidas iterações de f não muda drasticamente sob pequenas pertubações. Os algoritmos clássicos de construção dos conjuntos de Julia baseiam-se no estudo das órbitas confinadas da função complexa $f(z) = z^2 + c$, isto é, no estudo do conjunto de pontos ficam limitados em um conjunto, mesmo sendo iterados sucessivamente pela função f [1].

O objetivo deste trabalho é estudar a teoria de sistemas dinâmicos complexos e analisar algoritmos para construção gráfica dos conjuntos de Julia. No decorrer desta análise, verificamos que um algoritmo proposto pelos autores em [2] possui um grande esforço computacional e propusemos uma alternativa para melhorar este problema. O algoritmo aqui proposto foi implementado no *software* MATLAB.

2 Construção Gráfica do Conjunto de Julia

Seja $f : \mathbb{C} \to \mathbb{C}$ uma função contínua de variável complexa. Um sistema dinâmico complexo é uma sequência de pontos $z_k \in \mathbb{C}$ a partir de um ponto z_0 da função f, ou seja, cada ponto é a imagem do precedente pela função f:

$$z_k = f(z_{k-1}) = f \circ f \circ \ldots \circ f(z_0) = f^k(z_0).$$

Em um sistema dinâmico, as órbitas que se desenvolvem em uma região limitada, ou seja, que não escapam para o infinito, denominam-se órbitas confinadas. Por outro lado, as órbitas que escapam para o infinito recebem o nome de órbitas fugitivas. O conjunto

[1] alissonazuos@outlook.com
[2] beatrizsilva@alunos.utfpr.edu.br
[3] bsilva.2016@alunos.utfpr.edu.br
[4] prodrigues@alunos.utfpr.edu.br
[5] valentino@utfpr.edu.br

dos pontos que dão origem a órbitas fugitivas constitui a bacia de atração do infinito B_∞. Por outro lado, o conjunto dos pontos que dão origem a órbitas confinadas é chamado de conjunto de confinamento K. O conjunto de Julia é a fronteira dos conjuntos B_∞ e K. Neste trabalho, estamos principalmente interessados no estudo do conjunto de Julia da família de funções quadráticas da forma $f(z) = z^2 + c$, onde $c = p + qi$ (p e q reais).

Levando em consideração a definição do conjunto de Julia e o algoritmo apresentado em [2], denominado "Método dos pontos vizinhos com atratores distintos", exploramos o *software* MATLAB para determinar os pontos $z \in \mathbb{C}$ que estão nas fronteiras do conjunto de confinamento K e da bacia de atração do infinito B_∞. A estratégia dos autores em [2] baseia-se na análise de uma vizinhança próxima de cada ponto na tela gráfica, verificando se existem pontos com órbitas diferentes dentro dessa vizinhança. Dessa forma, se essa verificação for positiva, o ponto será de Julia, pois estará na fronteira de K e B_∞.

A cada ponto $z = (x, y)$ analisado, define-se valores dx e dy, como sendo os lados de um pixel cujo centro é o ponto z. Com isso, a vizinhança escolhida pelos autores em [2] consitui-se de 4 pontos nos extremos do pixel de lados $\frac{dx}{2}$ e $\frac{dy}{2}$. O algoritmo, então, verifica-se se todos esses pontos tem comportamento igual, ou se tem ao menos um ponto com comportamento distinto dos demais.

No decorrer da análise e implementação deste algoritmo, nos confrontamos com a dificuldade de calcular, em todos os pontos da tela gráfica, todos os 4 pontos de cada vizinhança. Para contornar este problema, desenvolvemos uma estratégia diferente dos autores: definimos a vizinhança de análise como sendo os extremos do pixel original, de lados dx e dy. Essa escolha tem como objetivo principal utilizar as informações dos extremos de um pixel de um ponto com seu ponto vizinho, que compartilham um lado do pixel. Desta forma, no primeiro ponto analisa-se os 4 pontos da vizinhança, porém no segundo ponto analisa-se somente 2 pontos da vizinhança. A partir do segundo ponto da segunda linha da tela gráfica, o algoritmo analisa somente um ponto da vizinhança, pois a informação sobre os outros 3 pontos da vizinhança já estará salva.

Conclusão e Perspectivas Futuras

Após implementação em MATLAB do algoritmo com a estratégia proposta neste trabalho, foi possível identificar melhorias significativas de custo computacional, o que pode ser justificado pelo fato de que o novo algoritmo possui um número menor de operações para calcular as órbitas que são analisadas. Além disso, para trabalhos futuros, pretende-se buscar outros algoritmos do gênero a fim de comparação do custo computacional com o proposto neste trabalho.

Referências

[1] G. JULIA. Mémoire sur la iteration des fonctions rationelles, *Journal de Mathématiques Pures et Appliquées*, 8:47-245, 1918.

[2] C. P. SERRA and E. W. KARAS. *Fractais gerados por sistemas dinâmicos complexos, 1a. edição.* Champagnat, Curitiba, 1997.