Simulação: Pseudoaleatoriedade, um estudo sobre o método do meio do quadrado

João Ferreira da Silva Júnior[1], **Sérgio Francisco Tavares de Oliveira Mendonça**[1], **Edson Alves de Carvalho Júnior**[2]

[1]Unidade Acadêmica de Garanhuns, Universidade Federal Rural de Pernambuco (UFRPE)
Av. Bom Pastor, s/n, Boa Vista – 55.292-270 – Garanhuns – PE – Brasil

[2]Departamento de Estatística e Informática, Universidade Federal Rural de Pernambuco (UFRPE)

joaoferreirape@gmail.com, sftom@uag.ufrpe.br, edacjr@msn.com

Resumo. *Área da matemática estreitamente relacionada com os métodos computacionais, a geração de números pseudoaleatórios é bastante discutida e tem aplicação em várias situações como por exemplo na análise de algoritmos e na criptografia. Neste resumo é discutido o método dos meios quadrados para a geração de números pseudoaleatórios.*

Abstract. *Field mathematics closely related to the computational methods of generating pseudorandom numbers is widely debated and has application in various situations for example in the analysis of algorithms and encryption. This is discussed in the summary of the half-square method to generate pseudorandom number.*

1. Introdução

Durante o estudo do funcionamento de sistemas, encontramos algumas questões relevantes que ditam regras de inferência. Como por exemplo, pode ser extremamente dispendioso do ponto de vista financeiro replicar algum sistema para estudo prático, ou este pode ser impossível de ser copiado ou mesmo antiético. Em situações como estas torna-se necessário buscar alternativas que possibilitem o estudo e que sejam validadas pelo método científico. A partir de um modelo bem definido, que represente adequadamente o sistema, simulamos o funcionamento deste, de modo que nos permita observar seu comportamento por meio de algumas características.

No modelo de sistemas, estas características são chamadas de variáveis, e possuem comportamento aleatório com intuito de simular comportamentos bem próximos aos encontrados na natureza. Tais variáveis recebem o nome de números pseudoaleatórios. São utilizados quando é necessário que não existam relações entre eventos independentes, e é determinado como uma série numérica na qual não é possível prever o próximo número a partir de membros anteriores.

2. Números pseudoaleatórios

Computacionalmente existem várias formas de se gerar números aleatórios. Entretanto, iremos nos direcionar ao estudo do método do meio do quadrado.

Proposto por John Von Neumann o método do meio do quadrado utiliza como entrada um número composto de quatro dígitos, denominado semente, este número é então elevado ao quadrado e deste são extraídos os quatro algarismos do centro que irão formar a semente da próxima iteração, daí novamente eleva-se a semente ao quadrado e assim por diante, conforme ilustrado na "*Figura 1. Método do meio do quadrado*".

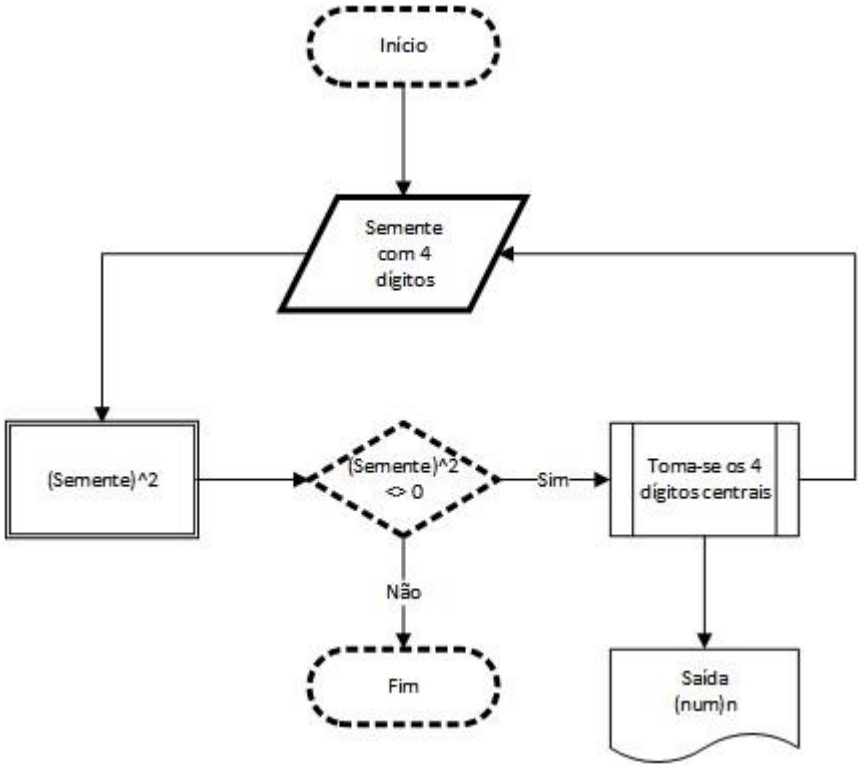

Figura 1. Método do meio do quadrado

Este é um algoritmo determinístico, pois sempre que a semente for repetida a sequência de números gerada será a mesma.

2.1. Implementação do código

O código abaixo foi implementado pelo autor para este estudo a partir da definição do algoritmo de Von Neumann e escrito na linguagem de programação Python:

```
#!/usr/local/bin/python
# -*- encoding: utf-8 -*-
'''
Geração de números aleatórios pelo método do meio do quadrado
'''
seed = int(raw_input(u"Semente inicial com quatro dígitos xxxx: "))
def meiodoquadrado(seed):
    if len(str(seed)) < 4:
        print u"Semente inválida!\r\nMenos de 4 dígitos..."
        return 0
    else:
        seed = seed ** 2
        return str(seed)[((len(str(seed))/2) - 2):((len(str(seed))/2) - 2) + 4]
i = 1
```

```
while True:
    seed = meiodoquadrado(int(seed))
        if seed == 0:
            break
        else:
            print u"#%i: %s" % (i, seed)
            i += 1
```

3. Problemática

Este algoritmo traz diversas desvantagens: a primeira é que a sequência de números gerada tende a se repetir após poucas iterações, a segunda se dá no caso quando o meio do quadrado coincidir com uma sequência de zeros, nessa situação teríamos a parada do algoritmo.

4. Solução proposta e otimização

Como proposta de solução, a cada iteração do algoritmo, deveremos somar à semente o número sequencial desta iteração elevado ao quadrado. Assim, teremos um deslocamento exponencial do alcance numérico gerado para a semente.

4.1. Investigação e testes de otimização do algoritmo

O código a seguir atende à proposta de solução. Nele omitimos a saída que informa a validade da semente quanto ao seu tamanho, ao invés de informar em tela a saída de erro, a implementação retorna para o laço onde gera nova semente e segue com o algoritmo.

```
#!/usr/local/bin/python
# -*- encoding: utf-8 -*
from math import sqrt
'''
Geração de números aleatórios pelo método do meio do quadrado
'''
seed = int(raw_input(u"Semente inicial com quatro dígitos xxxx: "))
def meiodoquadrado(seed):
    if len(str(seed)) < 4:
        return 0
    else:
        seed = seed ** 2
        return str(seed)[((len(str(seed))/2) - 2):((len(str(seed))/2) - 2) + 4]
i = 1
while True:
    seed = meiodoquadrado((int(seed) + i) ** 2)
    print u"#%i: %s" % (i, seed)
    i += 1
```

4.2. Gráfico de dispersão

Observamos que a dispersão da amostra tende a ficar uniforme conforme aumentamos o número de iterações. Nas ilustrações "*Figura 2. Dispersão da amostra 1k iterações*" e "*Figura 3. Dispersão da amostra 10k iterações*" exibimos tal efeito.

Isto se dá pelo fato de que o universo de saída da amostra é delimitado, no caso analisado, dadas as características do algoritmo dos meios dos quadrados, são gerados números entre 1.000 e 10.000, e isto faz com que após 9.000 interações haja ao menos uma repetição.

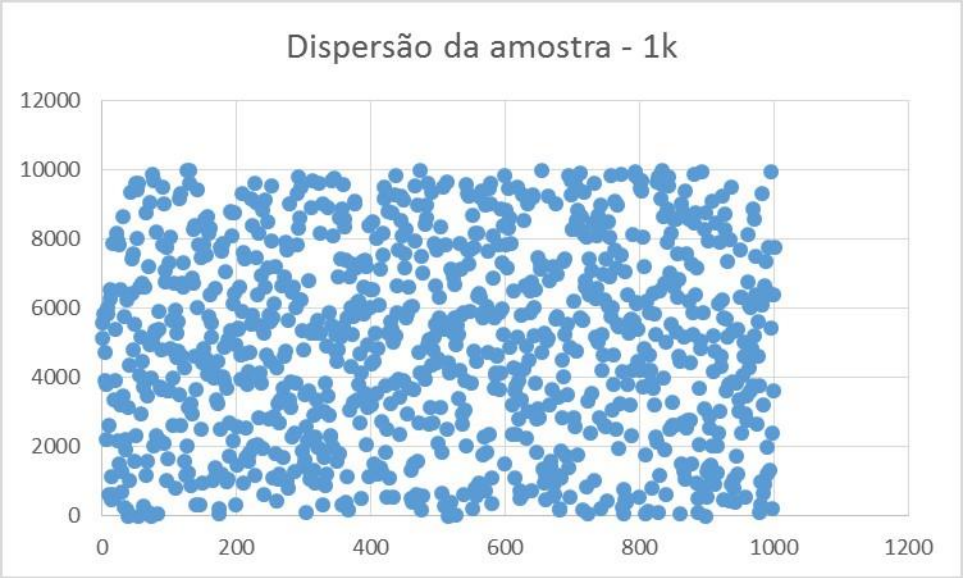

Figura 2. Dispersão da amostra 1k iterações

Quando estendemos a amostra para 10 mil iterações o efeito se torna mais evidente.

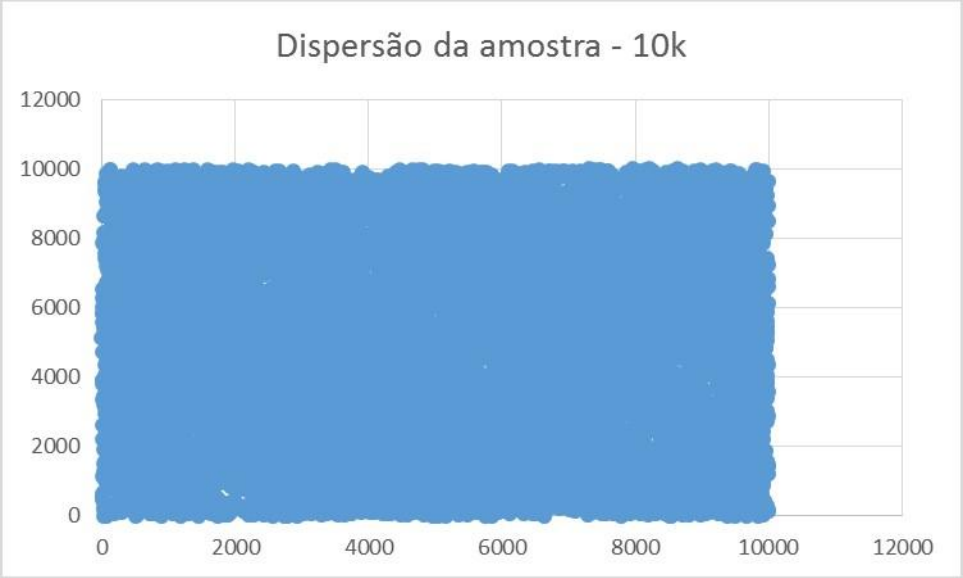

Figura 3. Dispersão da amostra 10k iterações

5. Conclusão

O trabalho está em fase de estudos. Até aqui pudemos observar que é possível, a partir de poucas modificações no algoritmo original, gerar uma grande margem de números pseudoaleatórios pelo método de Von Neumann.

Referências

Moreira, Laís Aparecida, and Rausley A. A. de Souza. "Métodos Computacionais para Geração de Sinais Aleatórios Aplicados a Sistema de Transmissão Digital." (2012).

Chwif, Leonardo, and Afonso Celso Medina. “Modelagem e simulação de eventos discretos.” Afonso C. Medina, (2006).

da Rosa, Fernando Henrique Ferraz Pereira, and Vagner Aparecido Pedro Junior, and Eduardo Colli. "Gerando números aleatórios." (2002).