COBENGE 2014
ENGENHARIA: Múltiplos saberes e atuações
16 a 19 de setembro | Juiz de Fora - MG

DESENVOLVIMENTO DE UMA FERRAMENTA COMPUTACIONAL DE GERAÇÃO E TESTE VISUAL DE NÚMEROS ALEATÓRIOS PARA AUXÍLIO NO ENSINO DE ESTATÍSTICA E PROBABILIDADE

Thiago R. F. Mendonça – rfmthi@gmail.com

João P. B. S. Duarte – joao.duarte@engenharia.ufjf.br

Igor A. Costa – igorabritta@yahoo.com.br

Daniel D. Silveira –danieldsilveira@engenharia.ufjf.br
Universidade Federal de Juiz de Fora
Rua José Lourenço Kelmer, s/n - Campus Universitário
36036-900– Juiz de Fora – Minas Gerais

Resumo: *Este artigo apresenta a pesquisa e o desenvolvimento de uma ferramenta computacional didática livre e de código aberto para auxiliar no ensino da disciplina de Estatística e Probabilidade. Apesar da importância da Estatística e o número crescente de aplicações nas pesquisas científicas e em empresas, existe uma carência de recursos práticos e de baixo custo que auxiliem no ensino e entendimento dessa matéria, especificamente na teoria de números aleatórios e como gerá-los. Observando-se essa lacuna foi desenvolvido um 'software', utilizando a plataforma Visual C# Express 2010 que, entre outras funções, tem como principal objetivo a geração de séries de números pseudo-aleatórios. Esta ferramenta conta com a possibilidade de manipulação dos parâmetros dos métodos e tem como principal diferencial a geração de uma imagem da sequência com a intenção de demonstrar aos usuários se a série criada realmente é aleatória ou se pode ser encontrado algum padrão em sua aleatoriedade e, consequentemente, efetuar a comparação entre os métodos analisados. Com isso, espera-se munir os docentes desta área com um novo instrumento que lhes deem recursos para lecionar o tópico de números, bem como conquistar um maior interesse dos discentes para com a disciplina e suas inúmeras possibilidades de aplicações em diversas áreas do conhecimento.*

Palavras-chave:*Probabilidade, Estatística, Números aleatórios, Ensino, Ferramenta Computacional;*

1. INTRODUÇÃO

O ensino de estatística e probabilidade é de extrema importância para formação em engenharia, independente da área de atuação.A estatística é uma ciência, de caráter multidisciplinar, que fornece ferramentas para uma análise mais profunda de dados de diversos ramos, como por exemplo, economia, agronomia, química, geologia, matemática, biologia, sociologia e psicologia(MARCELINO, 2010).Na pesquisa científica, aestatística tem

sido muito utilizada nos procedimentos de otimização de recursos econômicos, aumento da qualidade e produtividade, na otimização em análise de decisões e em previsões(RAO, 1999).

Muitos conceitos trabalham com a ideia de quantidade de experimentos tendendo ao infinito, tornando a experimentação impraticável e consequentemente dificultando o aprendizado. Visando contornar essa dificuldade, muitos educadores utilizam ferramentas computacionais para poder simular experimentos que possibilitem a visualização do problema pelos alunos. Suponha um exemplo simples no qual se deseja comprovar a distribuição uniformedos resultados de lançamento de um dado de seis faces. Em uma tentativa prática de comprovação, um aluno faz dez lançamentos, obtendo os resultados: 1, 1, 3, 2, 5, 1, 6, 1, 5, 6, parecendo tendencioso e difícil de acreditar que possuem a mesma probabilidade de ocorrência. Contudo, só é possível visualizar quando a quantidade de experimentos for muito elevada, exigindo uso computacional para comprovação.

Nesse contexto, pode-se inferir que a geração de sequências aleatórias é a base para simulação de problemas estatísticos. No entanto, esse tema possui pouca importância no ensino atualmente, seja em disciplinas de graduação ou pós-graduação, e mesmo para os interessados, é difícil encontrar em livros de métodos numéricos temas relacionados a geração de números aleatórios (ZIEGEL, E. ET AL., 1987). Existem alguns *softwares*que são utilizados no ensino da probabilidade e estatística, tais como o *Matlab, Scilab e Octave*, sendo os dois últimos livres, porém nenhum deles expõe o assunto de como são gerados os seus números aleatórios abertamente em sua documentação de ajuda, disponibilizando apenas uma biblioteca de código fechado para realização dessa tarefa.

Como resposta, o presente trabalho propõe o desenvolvimento de uma ferramenta computacional livre, de fácil instalação e interface intuitiva, como objetivo de auxiliar nesta questão curiosa na área de probabilidade e estatística e que é pouco explorada e, de certa forma, tentar procurar o interesse do aluno com o entendimento da origem destas sequências numéricas. Com a utilização dessa ferramenta, o ensino desse tópico poderá ser lecionado sem demandar tempo excessivo, enriquecendo a experiência dos alunos sem prejudicar o conteúdo programático previsto.

2. METODOLOGIA

É importante inicialmente definir aleatoriedade. Aleatoriedade ocorre quando não se pode prever o resultado de determinado experimento (YNOGUTI, 2011). Suponha por exemplo o sinal de um ruído AWGN (*Additive White GaussianNoise*): mesmo sabendo que sua média é zero, não é possível dizer qual será o seu próximo valor. A aleatoriedade ocorre em alguns processos naturais, por exemplo, o ruído atmosférico, tempo entre estalos de um contador Geiger de medição de radioatividade, entre outros. Mas como gerar números aleatórios em uma das ferramentas mais precisa e determinística já construída pelo homem, o computador, é um questionamento que certamente deve surgir durante o estudo dessa disciplina.

Existem dois tipos de dados aleatórios, classificados como sendo verdadeiramente aleatório ou pseudoaleatório. A geração de sequências verdadeiramente aleatória utiliza-se de *hardwares* que coletam dados físicos imprevisíveis para alimentar determinado algoritmo. Já a geração de sequências pseudoaleatória baseia-se em algoritmos matemáticos recursivos completamente implementados em computadores (sem alimentação de dados coletados externamente).Portanto, conhecendo-se os parâmetros da expressão matemática, a sequência

passa a ser previsível. Diversas bibliotecas alteram um dos parâmetros sem conhecimento do usuário, tornando a sequência imprevisível para o mesmo. Atualmente, existem diversos algoritmos capazes de gerar sequências que se aproximam da aleatoriedade real. Porém são periódicas, ou seja, após determinada quantidade de elementos, a sequência passa a se repetir.Contudo, a periodicidade de sequências pseudo-aleatórias pode chegar a valores da ordem de 2^{19936}- 1 (com o algoritmo de Mersennetwister(MATSUMOTO e NISHIMURA, 1998), por exemplo), que é um valor extremamente grande e suficiente em qualquer aplicação.

A geração de números realmente aleatórios é mais trabalhosa, e baseia em hardwares específicos capazes de captar sinais externos de processos naturais conforme já mencionado. No entanto, em diversas aplicações, pode não se ter disponível tal estrutura, e por isso é importante o conhecimento dos métodosde geração de números pseudoaleatórios.

Quando se constrói um algoritmo para essa finalidade, deve-se ter em mente que a distribuição deve ser uniforme, ou seja, a chance de sortear determinado valor deve ser a mesma para qualquer valor quando se considera uma quantidade de experimentos tendendo ao infinito. Portanto, se uma sequência é representada por: 1 1 1 1 1 1 1 1 1 1, outra sequência representada por: 1 0 1 0 1 0 1 0 1 0 e outra por: 1 0 1 1 0 0 1 1 0 1. Como saber qual é mais aleatória? Todas são igualmente prováveis de ocorrer, mas intuitivamente preferimos acreditar que a terceira opção é a mais aleatória. Essa comparação para saber o quão aleatório é determinada sequência pode ser feita por meio de testes estatísticos. Os testes mais utilizados e reconhecidos são: Testes de Diehard e sequência de testes da *NIST – NationalInstituteof Standards and Technology*(RUKHIN, BASSHAM, *et al.*, 2010). A batelada de testesnão são triviais de compreender nem de serem implementadas, sendo estudadas e elaboradas por especialistas na área. Portanto, não são ensinadas nos cursos de graduação nem de pós-graduação em áreas de engenharia, deixando para os cursos focados em estatísticas.

Uma forma interessante, porém ineficiente, para que um aluno não especializado em estatística possa se basear para comparar aleatoriedade é a forma visual, ou seja, gera-se uma imagem de duas dimensões com os dados da sequência aleatória representando intensidade de cor em determinada escala. Embora essa abordagem visual não deva ser utilizada como um teste oficial para mensurar aleatoriedade, é uma forma interativa e didática para que possa perceber o quão aleatório é determinada sequência.A ferramenta computacional apresentada no presente trabalho permite essa forma de comparação.

3. DESENVOLVIMENTO

O objetivo da ferramenta computacional discutida no presente trabalho é de facilitar a compreensão da geração de sequências pseudoaleatórias e possibilidade de comparação da aleatoriedade, temas até então discutidos apenas em cursos dedicados a estatística.

O programa foi desenvolvido na plataforma gratuita, *Visual C# Express* da *Microsoft*. Essa plataforma permite a criação de executáveis de fácil instalação em computadores com sistema operacional *Windows*. A ideia é disponibilizar a ferramenta para que possa ser usada e estudada por qualquer interessado.

A Figura 1 apresenta a tela principal, mostrando alguns métodos congruentes lineares. A primeira versão do programa conta com três algoritmos para comparação, representação gráfica do sinal aleatório gerado, representação por histograma a fim de comprovar a distribuição uniforme de determinado método, além da representação do sinal gerado em

forma de imagem bidimensional permitindo validação visual da aleatoriedade. Pretende-sedivulgar a ferramenta com código aberto, portanto o usuário poderá realizar as alterações que julgar necessário.

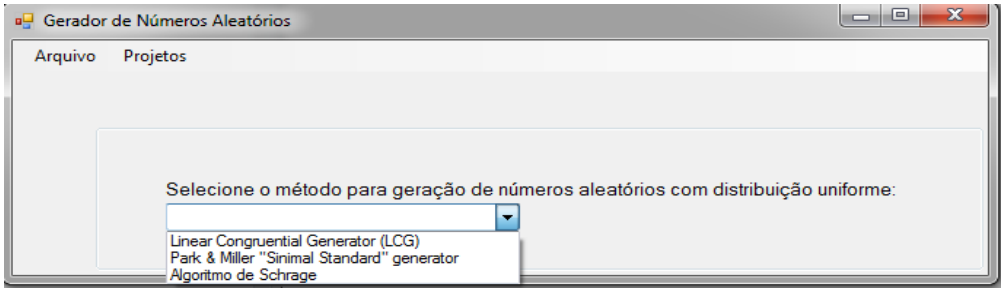

Figura 1 Tela inicial do programa desenvolvido. Em destaque três algoritmos implementados.

A seguir será descrito a ideia principal do algoritmo de gerador linear congruente misto, e sua variação, gerador linear congruente multiplicativo, sendo estes os algoritmos mais básicos, ainda assim muito utilizados(ZIEGEL, E. ET AL., 1987).

3.1. Gerador Linear Congruente Misto.

Este algoritmo utiliza uma fórmula de recorrência com parâmetros específicos, que geram a cada iteração um inteiro I_1, I_2, ... , I_n onde n é o ultimo termo do período. A expressão recursiva é apresentada na equação (1), onde*a* é denominado de multiplicador, *c* o incremento e *m* o módulo.

$$I_j = aI_{j-1} + c(mod\ m) \quad (1)$$

Pela expressão, concluímos que a sequência terá período no máximo igual ao módulo *m*. É preciso especificar parâmetros *a*, *c* e *m* de tal forma a conseguir maximizar o tamanho da sequência antes de começar a repetição. Observe que é preciso atribuir um valor para o primeiro inteiro I_o, este é o chamado de "semente" da sequência. Ainda segundo (Ziegel. et al., 1987), a escolha dos parâmetros dependem de uma profunda análise matemática, que foge do escopo do trabalho. Em(PARK e MILLER, 1988), toda a abordagem teórica é discutida, bem como revisão de alguns métodos que precedem o seu. Em suma, o período máximo é obtido se e somente se *m* for potência de dois; $c > 0$ e ímpar; $a - 1$ é múltiplo de quatro.

O valor definido para a semente simplesmente altera o ponto inicial da sequência. Em bibliotecas que implementam esse tipo de algoritmo, o valor para a semente é definido baseado em valores do relógio do computador, de forma que sempre forneça um valor diferente de semente, obtendo então sequências diferentes (com amostras menores que o período máximo da sequência).

Para ilustrar melhor o algoritmo, considere o caso em que $m = 8, a = 5$e $c = 1$, os 16 primeiros termos da sequência são: 1 6 7 4 5 2 3 0 1 6 7 4 5 2 3 0 . Observe o período igual a 8 elementos, com distribuição uniforme, ou seja cada número tem mesma chance de aparecer. Para limitar os valores entre 0 e 1, basta dividir por *m* – 1, que é o maior valor, obtendo assim:

0,29 0,43 0 0,143 0,86 1 0,57 0,71. Esse formato normalizado é exigido para alterar a distribuição da sequência posteriormente. Essas comparações são facilitadas pelo *software*,permitindo ao aluno testar essas e outras condições para seleção de parâmetros.

A grande vantagem desse método é sua velocidade, demandando pouco esforço computacional, podendo ser utilizada inclusive em aplicações embarcadas que precisem de poucas amostras aleatórias (sendo limitada pelo valor utilizado para os parâmetros, já que se utilizar, por exemplo, valores de 32 bits, o resultado será de 64, o que pode não estar disponível no microcontrolador).

A Figura 2 contém o fluxograma do funcionamento desse algoritmo.

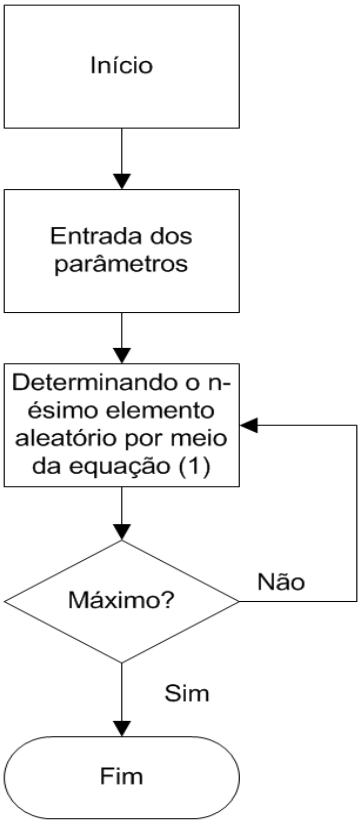

Figura 2 Algoritmo utilizado para implementação.

3.2. Gerador Linear Congruente Multiplicativo

Esse método é uma alteração feita no método anterior. Após uma série de considerações, Park e Miller apresentam juntamente com uma vasta revisão teórica alguns geradores inadequados(PARK e MILLER, 1988). Propuseram uma alteração na equação (1) tornando c = 0, levando a equação (2).

$$I_{j+1} = aI_j \,(mod\ m) \qquad (2)$$

Note que ainda assim é preciso definir a semente. A proposta é baseada na escolha dos parâmetros como sendo:

- $a = 7^5 = 16807$ e $m = 2^{31} - 1 = 2147483647$

Um problema relacionado com essa abordagem é que, conforme já mencionado, o maior valor possível de ser obtido é $m - 1$, então com esses valores propostos, a multiplicação de a por $m - 1$ resultará em um inteiro maior que 32 bits. Outras abordagens como a de Schrage(ZIEGEL, E. ET AL., 1987)permite a operação com inteiros de 32 bits e módulo também de 32 bits sem precisar utilizar 64 bits, permitindo a utilização desse gerador com menos de 64 bits, abrangendo essencialmente qualquer linguagem de programação atual. Não convém detalhar a metodologia do algoritmo de Schrage no presente trabalho. O fluxograma para essa abordagem é semelhante ao apresentado na Figura 2, alterando apenas a expressão recursiva, excluindo o parâmetro *c*.

A grande vantagem computacionalmente é que essa abordagem possui uma soma a menos por iteração, exigindo menor tempo de execução. Um detalhe é que o valor da semente nunca poderá igualar a zero já que, pela equação 2, o resultado será sempre zero.

De acordo com (ZIEGEL, E. ET AL., 1987), para esse método apenas alguns valores podem ser utilizados como parâmetros, os quais são apresentados na Tabela 1, considerando o valor de $m = 2^{31} - 1$.

Tabela 1: Possíveis valores para multiplicador no gerador congruente linear multiplicativo.

Possíveis valores de *a*
16807
48271
69621

3.3.Sobre o software.

Esses dois métodos discutidos foram implementados na ferramenta proposta, podendo-se alterar os parâmetros e visualizar a sequência, tanto em forma de gráfico quanto em forma de imagem. Outra característica importante do aplicativo é que este direciona o usuário curioso para algumas referências bibliográficas específicas sobre o tópico, além de explicar cada parâmetro e fornecer valores predeterminados (como os mencionados na Tabela 1) para testes.A Figura 3 ilustra as opções informativas sobre os diferentes métodos implementados. A Figura 4 ilustra um exemplo do menu de ajuda do *software* proposto e traz informações a respeito do valor predeterminado do parâmetro *m*.

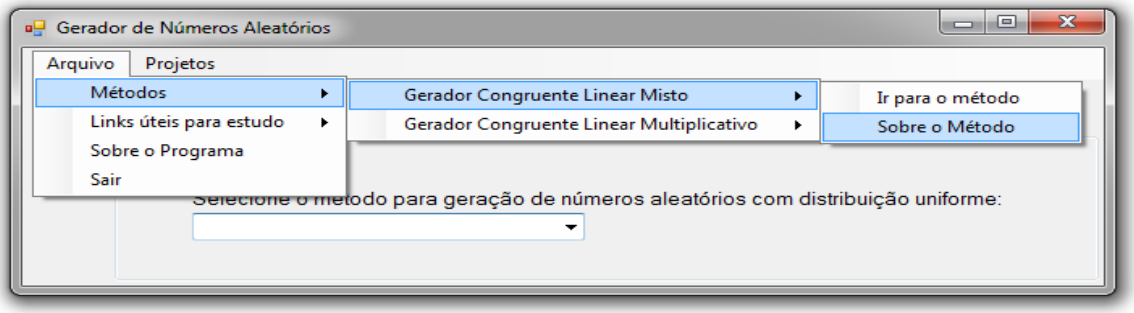

Figura 3: Opções informativas sobre diferentes métodos implementados.

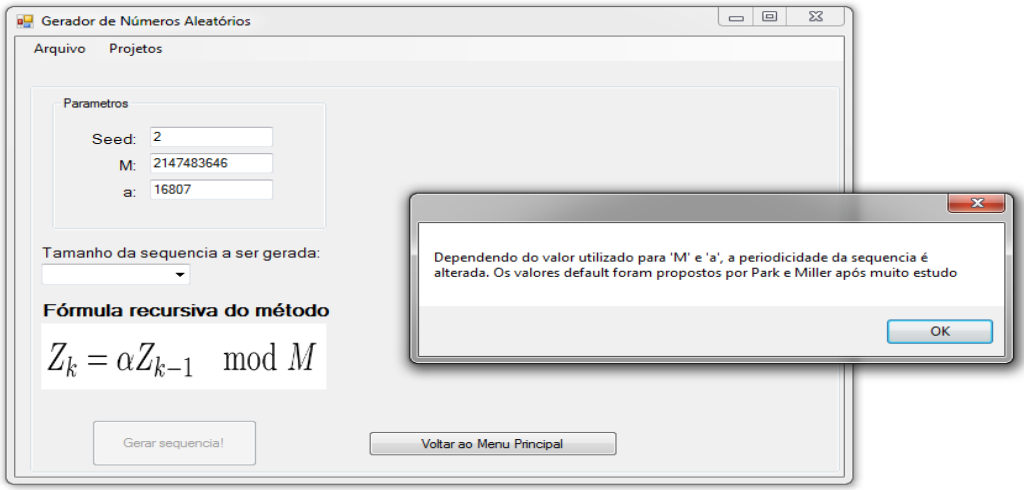

Figura 4: Exemplo do *menu* de ajuda do *software* proposto.

Após selecionar os parâmetros e definir o tamanho da sequência, clica-se no botão "Gerar sequência!" e aparecerão à direita da tela, dentro de uma caixa de texto, os valores da sequência, que poderá ser exportada para algum outro programa como Excel ou *Matlab* para alguma aplicação. Será permitido ao usuário rodar o teste de aleatoriedade visual mencionado além de gerar os gráficos do sinal, bem como o seu histograma. A Figura 5 apresenta a tela "Gerador Congruente Linear Multiplicativo" com os parâmetros m=2147483646 e a=16807. Para visualizar o histograma, bastaria clicar no botão "Histograma".

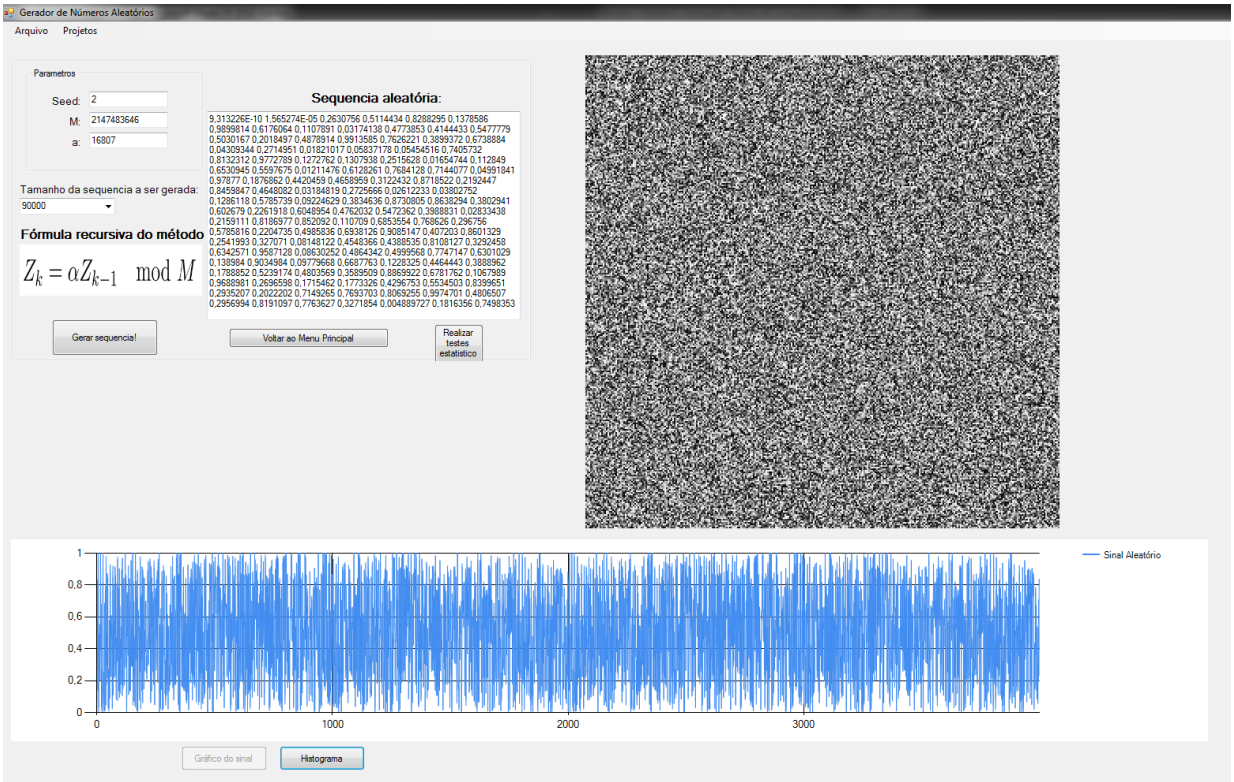

Figura 5: Tela do software apresentado no presente trabalho.

4. RESULTADOS E CONCLUSÃO:

Desenvolveu-se uma ferramenta computacional capaz de gerar séries de números pseudoaleatórios com dois algoritmos diferentes, com mensagens informativas que auxiliam na utilização do mesmo, além de direcionar para algumas referências especialistas no tema em questão.

O *software* contém funções para geração do gráfico do sinal aleatório, Histograma e imagem, permitindo a análise e comparação entre os diferentes algoritmos. Para exemplificar a comparação de ruídos por uma abordagem visual, a Figura 6 (a) apresenta a transformação de um ruído realmente aleatório em imagem, enquanto a Figura 6 (b) apresenta o ruído transformado em imagem gerado a partir do algoritmo gerador congruente linear misto, com parâmetros iguais a: m = 6075, a = 106 e c = 1283 que, conforme visto no tópico 3.1, são mal selecionados. Os dados para construção da Figura 6 (a) foram extraídos de um gerador aleatório baseado no sinal de ruído atmosférico, o qual é um processo natural e não gerado via expressão matemática(RANDOMNESS AND INTEGRITY SERVICES LTD, 2009). Já para a construção da Figura 6 (b), foi utilizado o programa em desenvolvimento descrito no presente trabalho.

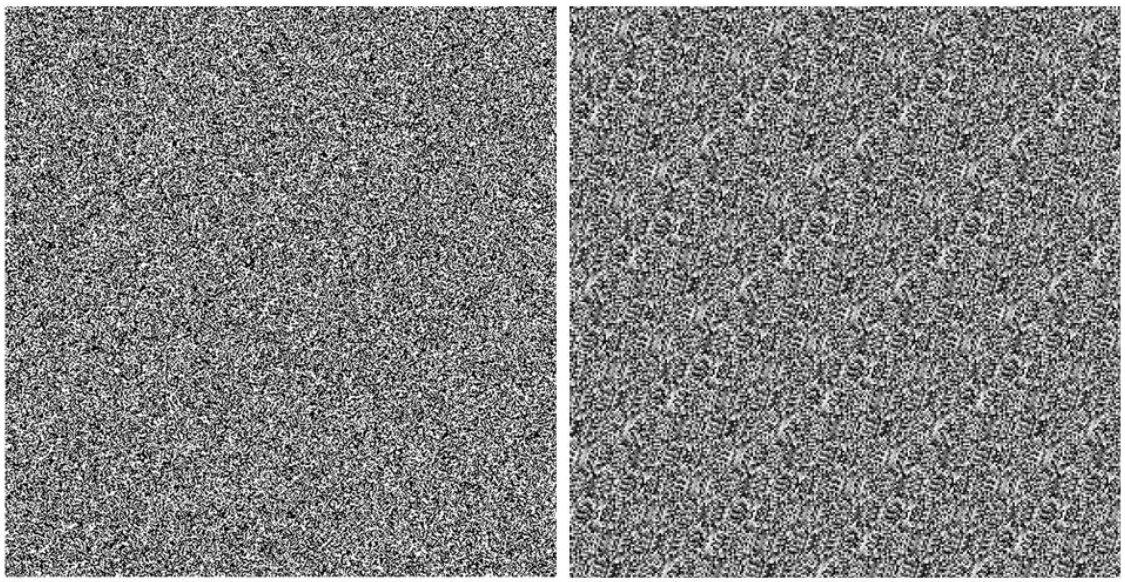

Figura 6: Visualização do sinal aleatório (a) e pseudoaleatório (b) em forma de imagem.

A Figura 6 (b) possui uma resolução de 300 x 300, ou seja, representa um sinal de 90000 amostras. A Figura 7apresenta o mesmo sinal sendo observado em forma de gráfico. Note que é muito mais difícil reconhecer padrões, além de limitar a quantidade de pontos a representar (essa mesma figura foi construída com apenas 2000 pontos), tornando a representação por imagens uma alternativa muito mais interessante. A Figura 8mostra o histograma da sequência gerada. Note que possui uma distribuição uniforme, o que não impede de possuir padrão no sinal.

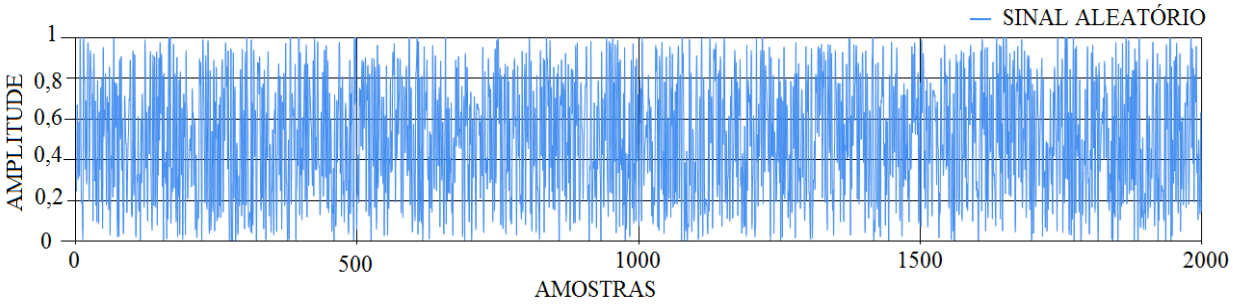

Figura 7: Gráfico gerado no *software* proposto.

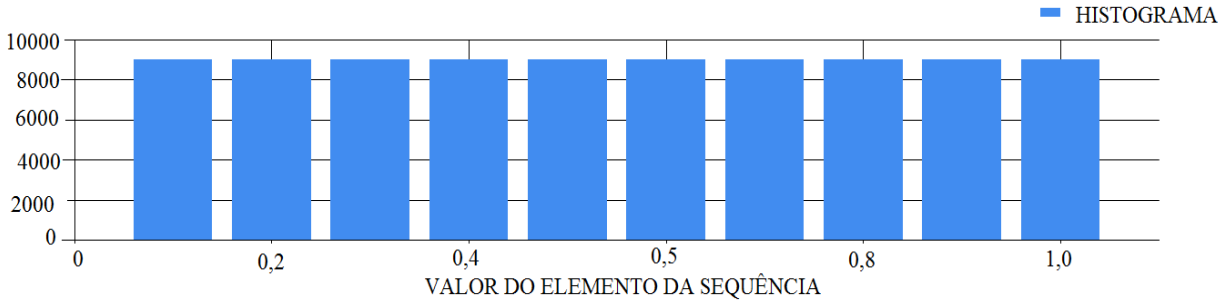

Figura 8: Histograma gerado no *software* proposto.

Apresentando essas três formas de analise da sequência, o aluno pode comparar e compreender melhor os conceitos relacionados à geração de números aleatórios e pseudoaleatórios, somado ao fato de o aplicativo ter sido desenvolvido em plataforma gratuita, e de código aberto, o que torna essa aplicação muito interessante para o ensino de estatística, probabilidade e programação em engenharia.

5. REFERÊNCIAS / CITAÇÕES

MARCELINO, K. J. **A importância da estatística na atualidade**. Instituto Superior de Línguas e Administração. [S.l.], p. 9. 2010.

MATSUMOTO, M.; NISHIMURA, T. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. **ACM Trans. on Modeling and Computer Simulation**, Janeiro 1998. 3-30.

PARK, S. K.; MILLER, K. W. RANDOM NUMBER GENERATORS : GOOD ONES ARE HARD TO FIND. **Communications of the ACM**, New York, Outubro 1988. 1192-1201.

RANDOMNESS AND INTEGRITY SERVICES LTD. True Random Number Generator. **RANDOM.ORG**, 2009. Disponivel em: <http://www.random.org/>. Acesso em: 3 Maio 2014.

RAO, P. K. The Economics of Global Climatic Change. **Environmental and Resource Economics**, v. 21, n. 1, p. 102-104, jan. 1999. ISSN 1573-1502.

RUKHIN, A. L. et al. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. **National Institute of Standards & Technology**, 2010.

YNOGUTI, C. A. **Probabilidade, Estatística e Processos Estocásticos**. [S.l.]: [s.n.], 2011.

ZIEGEL, E. ET AL. **Numerical Recipes:** The Art of Scientific Computing. [S.l.]: [s.n.], 1987.

DEVELOPMENT OF A COMPUTATIONAL TOOLFOR GENERATING RANDOM NUMBERS WITH VISUAL TESTS FOR SUPPORT THE TEACHING OF STATISTICS AND PROBABILITY

Abstract:*This paper presents the research and development of a free software tool and open code to assist in teaching the discipline of Statistics and Probability. Despite the importance of statistics and the growing number of applications in scientific research and companies, there is a lack of practical and low cost resources that help in teaching and understanding of this subject, as random numbers and how to generate them effectively. Observing this problem, we developed anopen source software, using Visual C # Express 2010 platform, which is designed for the generation of pseudo-random number series. This tool offers the possibility of manipulating the parameters of the method, and itsmajor feature is the generation of an image to demonstratethe true randomness of a specific series. It also offers the possibility to compare two images of different series, allowing the detection of hidden patterns in series. Thus, this article shows a new instrument for teaching, providing resources to teach the topic of random numbers and its applications in various areas of knowledge.*

Key-words:*Probability, Statistics, Random Numbers, Education, Computational Tool;*