UNIVERSIDADE CATÓLICA DE PELOTAS
CENTRO DE CIÊNCIAS SOCIAIS E TECNOLÓGICAS
MESTRADO EM ENGENHARIA ELETRÔNICA E COMPUTAÇÃO

MURILO DANIEL BRISOLARA CRUZ

# Um Novo Algoritmo de Criptografia Caótica Utilizando Técnicas de Ordenação a Partir da Medida Natural

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Engenharia Eletrônica e Computação

Orientador: Prof. Dr. Everton Granemann Souza

Pelotas
2020

**CIP — CATALOGAÇÃO NA PUBLICAÇÃO**

Cruz, Murilo Daniel Brisolara

Um Novo Algoritmo de Criptografia Caótica Utilizando Técnicas de Ordenação a Partir da Medida Natural / Murilo Daniel Brisolara Cruz. – Pelotas: 2020.

36 f.: il.

Dissertação (mestrado) – Universidade Católica de Pelotas. 2020. Orientador: Everton Granemann Souza.

1. Caos. 2. Mapa Logístico. 3. Criptografia. 4. Algoritmos de Ordenação. I. Souza, Everton Granemann. II. Título.

## RESUMO

A criptografia caótica é uma maneira eficiente de criptografar textos, porque utiliza recursos de embaralhamento intrínsecos do atrator. Neste trabalho, são apresentados algoritmos de encriptação que utilizam o mapa logístico, a estrutura é baseada no algoritmo de Baptista. Para encriptar utilizando o caos, Baptista associa partições da medida natural a caracteres alfanuméricos de forma fixa. Os algoritmos propostos utilizam dois métodos de ordenação, Bubblesort e Quicksort, para ordenar os intervalos da medida natural e os caracteres em uma tabela de frequência, para então associá-los, diminuindo assim o tempo de encriptação. Simulações mostram que a implementação de métodos de ordenação tornam o algoritmo de encriptação mais rápido que o algoritmo de Baptista. Os testes mostraram que a encriptação é segura, de acordo com os critérios do teste de Wald-Wolfowitz, e se mostra eficaz para mensagens longas.



**Palavras-chave:** Caos. Mapa Logístico. Criptografia. Algoritmos de Ordenação.

**New Chaotic Cryptography Protocol Based on the Logistic Map**


## ABSTRACT

Chaotic cryptography is an efficient way to encrypt text, since it uses the attractor's intrinsic shuffling features. In this work, encryption algorithms that use the logistic map are presented, the structure is based on the Baptista algorithm. To encrypt using chaos, Baptista associates partitions of natural measure with alphanumeric characters fixedly. The proposed algorithms use two methods of sorting, Bubblesort and Quicksort, to sort the intervals of the natural measure and the characters in a frequency table, to then associate them, thus reducing the encryption time. Simulations show that the implementation of sorting methods makes the encryption algorithm faster than the Baptista algorithm. Tests have shown that encryption is secure, according to the criteria of the Wald-Wolfowitz test, and proves effective for long messages.



**Keywords:** Chaos, Logistic Map, Cryptography, Sorting Algorithms.

## LISTA DE FIGURAS

## LISTA DE TABELAS

**SUMÁRIO**

# 1 INTRODUÇÃO

Criptografia é um método de proteger e dar autenticidade a comunicação através do uso de algoritmos, de forma que apenas aquele a quem a informação se destina seja capaz de interpretá-la (MENEZES; OORSCHOT; VANSTONE, 1996).

A criptografia atual ainda apresenta fraquezas em relação a segurança da transmissão de informações, sendo vulnerável a ataques de força bruta (SILVA; SOUSA, 2010). Por causa disso, uma alternativa aos protocolos de criptografia existentes se faz necessária, e a criptografia caótica se tornou uma alternativa promissora.

Pecora (PECORA; CARROLL, 1990) mostrou em seu trabalho a possibilidade de encriptação utilizando caos. Desde então, diversos trabalhos surgiram no esforço de criar um algoritmo de encriptação caótico eficiente (SILVA, 1996), (HAYES C. GREBOGI; MARK, 1994), (JOVIC, 2011). Dentre eles, Baptista (BAPTISTA, 1998) se destacou propondo um algoritmo que utiliza o comportamento caótico do mapa logístico para encriptar mensagens, tendo como fator de segurança a sensibilidade às condições iniciais. Seu trabalho se tornou conhecido e recebeu a contribuição de inúmeros autores na área da criptografia caótica na tentativa de otimizar o algoritmo e torná-lo mais rápido e seguro.

Jakimoski (JAKIMOSKI; KOCAREV, 2001), Fraser (FRASER; YU; LOOKMAN, 2002) e Álvarez (ÁLVAREZ F. MONTOYA; PASTOR, 2003) apresentaram importantes análises do algoritmo de Baptista, propondo alguns pontos que devem ser observados na elaboração da criptografia como, por exemplo, tornar a geração das chaves um processo mais randômico, utilização única das chaves, padronizar uma possível implementação em hardware e a realização de análises de segurança dos algoritmos propostos.

Algumas das contribuições mais importantes foram o encurtamento da cifra-texto (WONG; LEE; WONG, 2001), (WONG; YUNG, 2003), (WONG; YUEN, 2008), a diminuição no número de iterações do mapa logístico para encriptação (WEI et al., 2006), (WEI; WONG, 2006), (WANG; WANG, 2011) e modificações na equação do mapa logístico para aumentar o espaço de parâmetros (SMAOUI; KANSO, 2009), (PANDE; ZAMBRENO, 2011), (SAN-UM; KETTHONG, 2014). Apesar da importância das otimizações citadas, não foram realizadas análises comparando a velocidade e segurança dos algoritmos de encriptação propostos em relação ao protocolo de Baptista.

Nesse trabalho, propomos uma otimização do algoritmo de Baptista de forma a tornar a encriptação mais rápida. Para isso, analisamos o histograma de visitações do intervalo caótico, chamado de medida natural (ALLIGOOD; SAUER; YORKE, 1996), para propor um método

mais eficiente de associação entre os caracteres da mensagem que será encriptada e a medida natural. No algoritmo de Baptista, a medida natural é particionada em intervalos equidistantes, os caracteres ASCII são associados fixamente aos intervalos de forma que, dependendo do parâmetro de controle escolhido, os caracteres com maior frequência de aparição na mensagem podem ser associados a intervalos com baixa visitação, tornando o algoritmo lento.

Para realizar uma associação eficiente entre caracteres e medida natural, foi considerado a utilização de métodos de ordenação para associar os caracteres alfanuméricos e os intervalos da medida natural em ordem de frequência. Duas técnicas de ordenação foram testadas: O Bubblesort e o Quicksort. Assim, foi possível criar uma tabela de associação baseado na frequência de aparição dos caracteres e na frequência de visitação dos intervalos da medida natural e, consequentemente, otimizar a associação e diminuir o tempo de encriptação da mensagem.

Além dos testes de velocidade, foi realizado testes de segurança. O teste de Wald-Wolfowitz (WALD; WOLFOWITZ, 1940), ou *runs test*, mostrou o nível de aleatoriedade das séries, indicando que as mensagens encriptadas são seguras. Com os dados de regiões rápidas e seguras, foi possível criar um gráfico que indique as regiões próprias para encriptação, ou seja, regiões onde a razão de tempo dos algoritmos propostos é menor que 1 em relação ao algoritmo de Baptista, e a segurança da cifra-texto é assegurada pelo *runs test*. Também foi criado um gráfico para razão de tempo menor que 0,5, indicando as regiões onde o algoritmo proposto codifica a mensagem pelo menos 2 vezes mais rápido que o algoritmo de Baptista.

Os resultados mostram que a implementação dos métodos de ordenação tornaram os algoritmos mais rápidos que o algoritmo original de Baptista. Os algoritmos Bubblesort e Quick-Sort foram mais eficientes, tanto em regiões onde a medida natural apresenta uma geometria irregular quanto em regiões onde a medida natural é uniforme. Os resultados também mostraram níveis de aleatoriedade na cifra-texto para as regiões do espaço de parâmetros de controle analisados, a qual podemos considerar segura.

## 1.1 Motivação

A criptografia caótica se mostrou promissora e flexível, de forma que é possível modificá-la e torná-la mais eficiente em relação aos algoritmos originais. Diversos autores propuseram métodos de encriptação para otimizar a criptografia caótica que, apesar de ser considerada segura, ainda não se mostra competitiva em questão de tempo consumido com as criptografias utilizadas atualmente, como AES (do inglês *Advanced Encryption Standard*), 3DES (do inglês *Triple Data Encryption Standard*), etc. A implementação da criptografia caótica em aplicações do dia-a-dia se mostra um desafio a ser superado.

## 1.2 Objetivo

O objetivo desse trabalho é otimizar o método de encriptação proposto por Baptista e apresentar resultados de implementação que proporcionem uma base de comparação para futuros algoritmos. Foi programado em linguagem Fortran 90 três algoritmos utilizando diferentes métodos de encriptação, sendo um deles o algoritmo proposto por Baptista e outros dois são algoritmos propostos pelos autores. Os três algoritmos são comparados a fim de definir se os métodos de ordenação escolhidos podem exibir o melhor resultado no quesito tempo de encriptação.

## 1.3 Contribuições do Trabalho

Esse trabalho está centrado em uma proposta de otimização do algoritmo de Baptista, assim como analisar as características de velocidade e segurança dos algoritmos propostos. São apresentadas as seguintes contribuições desse trabalho:

- Aumento na velocidade de encriptação de mensagens com o mapa logístico através da utilização de métodos de ordenação;
- Otimização da associação entre caracteres ASCII e órbita caótica através da medida natural (histograma de frequência);
- Análise de segurança através do Teste de Wald-Wolfowitz;
- Encriptação da mensagem $2,5$ vezes mais rápida em comparação ao algoritmo original.

## 1.4 Organização do Trabalho

O trabalho é organizado da seguinte forma. No capítulo 2 é realizado uma revisão histórica da literatura e o estado da arte da criptografia caótica. No capítulo 3 são introduzidos os conceitos matemáticos de criptografia caótica e são apresentados os algoritmos propostos. O capítulo 4 apresenta e discute os resultados, avaliando as vantagens dos algoritmos propostos em relação ao algoritmo de Baptista. No capítulo 5 é concluído o trabalho.

# 2 TRABALHOS RELACIONADOS

## 2.1 Histórico e Trabalhos Recentes na Criptografia Caótica

As primeiras ideias a respeito da criptografia caótica surgiram com Shannon (SHANNON, 1949). Em seu artigo, Shannon discute sobre a possibilidade de utilizar chaves longas para elaborar uma criptografia robusta e enviar mensagens de forma mais segura.

May (MAY, 1974) começa a elaborar uma equação análoga a equação conhecida na literatura como diferencial logística, que mais tarde seria o mapa logístico (MAY, 1976). May elaborou essa equação para descrever o crescimento biológico de uma população, a qual leva sempre a um ponto de equilíbrio. Anos mais tarde, o mapa logístico se tornou uma importante ferramenta para a encriptação através do caos.

Tang (TANG; CHUA, 1983) observa que a saída de circuitos caóticos é composto de uma soma de sinais periódicos inter-modulados. É proposto que, apesar de ser pouco presente na literatura, é possível sincronizar osciladores caóticos. O autor conclui que pequenas variações na condição inicial pode levar a grandes variações nas condições futuras.

Pecora (PECORA; CARROLL, 1990) estuda a forma como duas oscilações caóticas podem ser sincronizadas através de um sinal comum, conforme apontou (TANG; CHUA, 1983). O autor observou que, devido as características dos sinais caóticos, a possibilidade de sincronizar sistemas não-lineares e sistemas caóticos pode abrir oportunidades interessantes para a aplicação do caos na comunicação.

Ott e Grebogi (OTT; YORKE, 1990) realizam o controle de sinais em circuitos eletrônicos caóticos através de pequenas perturbações. O objetivo era controlar órbitas caóticas, tornando elas em órbitas estáveis e periódicas. Posteriormente, os autores realizaram novos trabalhos ((HAYES; OTT, 1993) e (HAYES C. GREBOGI; MARK, 1994)) que fundamentaram o controle de órbitas caóticas para esconder mensagens.

Baseado na possibilidade de comunicação segura utilizando sincronização de oscilações caóticas, conforme apresentado por (PECORA; CARROLL, 1990) e (HAYES C. GREBOGI; MARK, 1994), Baptista propôs um protocolo baseado na ergodicidade de sistemas caóticos, capaz de utilizar da sensibilidade às condições iniciais do mapa logístico e da entropia para encriptar mensagens de forma segura. A mensagem é criptografada transformando caracteres alfanuméricos em números inteiros de iterações do mapa logístico. A medida natural é dividida em partes e os caracteres são associados a essas partições. Essa tabela é igual para o emissor e para o receptor, os quais vão compartilhar a condição inicial e o parâmetro de controle (cha-

ves privadas longas). O mapa logístico será iterado a partir das chaves privadas e, quando os caracteres da mensagem são encontrados, a mensagem é encriptada.

Jakimoski (JAKIMOSKI; KOCAREV, 2001) analisa os algoritmos de Baptista e Alvarez (ALVAREZ et al., 1999) e verifica que os dois podem ter sua criptografia facilmente quebradas através do ataque de texto conhecido (método de ataque conhecido como *known-plaintext*, ou ataque de texto conhecido). Esse trabalho incentiva a comunidade a encontrar novas formas de encriptar mensagens utilizando caos como, por exemplo, o proposto por (WONG; LEE; WONG, 2001), que elaborou um algoritmo de encriptação onde o alfabeto é distribuído na medida natural por um número aleatório de iterações.

Fraser (FRASER; YU; LOOKMAN, 2002) considera alguns pontos para uma criptografia segura. A condição inicial e o parâmetro de controle não são conhecidos pelo interceptador, que possui apenas o método de encriptação e um texto conhecido como ferramentas. O autor ressalta que toda a segurança da criptografia está na geração das chaves. Assim, os autores do trabalho propõem uma modificação na forma como elas são geradas. Fraser também propõe uma padronização no hardware que executa a encriptação, de forma que a utilização de chaves muito sensíveis a oscilação não se tornem um obstáculo para hardwares com baixa precisão.

Álvarez (ÁLVAREZ F. MONTOYA; PASTOR, 2003) declara que deve ser feito análises mais rigorosas em relação a criptografia de Baptista, pois até então foram feitas poucas análises críticas desses algoritmos, como Jakimoski (JAKIMOSKI; KOCAREV, 2001), que fez apenas um tipo de ataque à criptografia caótica. São propostos quatro tipos de ataques contra a criptografia ergódica de Baptista e seus derivados: ataque de cifra única, ataque de entropia, ataque de recuperação de chave e estimativa de parâmetro de controle e condição inicial. Foram encontradas fraquezas no algoritmo de Baptista. No teste de cifra única, Álvarez descobriu que a chave, uma vez usada, deve ser descartada, pois a reutilização enfraquece a cifra, tornando vulnerável ao ataque de entropia, dentre outras vulnerabilidades. O autor ressalta que a aplicação do algoritmo de Baptista deve considerar esses pontos antes de uma possível aplicação.

Wong (WONG; YUNG, 2003) discute a possibilidade de cifras-texto mais curtas, que até então são até 2 vezes maiores que a mensagem original. Além disso, Wong ressalta que a criptografia é lenta para encriptar arquivos de multimídia grandes através da internet e propõe combinar o método de encriptação tradicional com uma tabela que referencia dinamicamente a posição de cada letra na medida natural. Assim, em vez de enviar um número de iterações que normalmente é muito maior que o caractere da mensagem, é apenas enviado o índice do caractere na tabela. A cifra-texto será composta pela mensagem encriptada que é do mesmo tamanho que a original e mais uma pequena sequência de índices, de forma que a mensagem

encriptada seja ligeiramente maior que a mensagem original.

Li (LI et al., 2004) estuda o algoritmo de Baptista e encontra quatro defeitos: a distribuição da cifra-texto não é uniforme, é necessário no mínimo 250 iterações para cada caractere, a cifra-texto é maior que a mensagem original, e é inseguro contra alguns tipos de ataques, conforme mostrou Jakimoski e Álvares. Li aponta que, apesar das modificações propostas em trabalhos anteriores, a segurança ainda não é suficiente para aplicações práticas. Li ainda afirma que seu trabalho anterior (LI et al., 2003) resolve parcialmente a falha de segurança mostrada por Jakimoski, mas tem erros possíveis de acontecer. No mesmo trabalho, o autor apresenta uma modificação de sua criptografia de forma a contornar esse erro.

Álvarez (ALVAREZ; LI, 2006) elabora uma lista de requerimentos que podem ser usados como guia na construção de novos métodos de criptografia caótica. Esses requerimentos podem, potencialmente, aumentar a segurança da implementação da criptografia, mas não devem barrar novas possibilidades de aplicação que possam diferir dos requerimentos apresentados.

Wei propõe em dois trabalhos (WEI et al., 2006), (WEI; WONG, 2006), novas criptografias. O primeiro trabalho é o melhoramento do esquema de encriptação do algoritmo de Baptista que, após uma investigação dos problemas encontrados no protocolo, foi possível diminuir o número de iterações necessárias para encriptar a mensagem original, assim como aumentar a segurança da cifra-texto. O segundo trabalho é um algoritmo de criptografia que precisa de menos iterações para encriptar a mensagem, ao mesmo tempo que aumenta a segurança da encriptação, associando as iterações a índices, de forma que a cifra-texto não exponha o número de iterações do mapa logístico.

Arroyo (ARROYO; ALVAREZ; FERNANDEZ, 2008) demonstra, através de análises das características dinâmicas, que o mapa logístico não é tão adequado para a encriptação, sendo inconveniente seu uso em aplicações que objetivam a segurança. É enfatizado as desvantagens do mapa logístico para aplicações criptográficas, assim como é explicado o problema da sua aplicação no contexto da criptografia caótica. Arroyo recomenda o mapa linear por partes como uma possível alternativa ao mapa logístico.

Wong (WONG; YUEN, 2008) propõe um algoritmo que associa partições frequentes com as letras mais prováveis, diminuindo a cifra-texto e tornando a encriptação mais rápida. Diferente do algoritmo de Baptista, que associa indiferentemente as letras às partições, Wong associa as letras mais frequentes a um maior número de partições, e as letras menos frequentes a um menor número de partições. O autor mostra que a mensagem encriptada foi comprimida a um tamanho satisfatório e a cifra-texto é sensível o suficiente à pequenas alterações, tornando a encriptação segura.

Novos algoritmos são propostos para a comunidade (SMAOUI; KANSO, 2009), (PANDE; ZAMBRENO, 2011), (WANG; WANG, 2011), (SAN-UM; KETTHONG, 2014), os quais trabalham com modificações do mapa logístico. Também são elaboradas novas análises (WANG; QIN, 2012) na tentativa de melhorar os aspectos já apresentados da criptografia caótica. A partir de então, a comunidade voltou-se aos estudos do mapa logístico, na busca de entender a dinâmica caótica. Khaleque (KHALEQUE; SEN, 2015) estuda o efeito de um parâmetro de controle variável (no intervalo de 1 a 4) no comportamento do mapa logístico. Maritz (MARTIZ, 2020) demonstra uma solução para dois parâmetros de controle do mapa logístico e propõe uma equação que pode mostrar detalhes de qualquer parte do diagrama de bifurcação, englobando todo o processo iterativo em uma única função. O autor ainda releva que o mapa logístico não é estudado de forma satisfatória, afirmando que os estudos são concentrados nas ilhas de estabilidade e a duplicação de período.

No estágio atual, são necessários estudos mais aprofundados sobre o mapa logístico, que até então recebeu diversas aplicações em criptografia, mas apresenta poucos estudos que podem ser críticos para a implementação eficiente na criptografia caótica.

# 3 MÉTODOS

## 3.1 Encriptação com o Mapa Logístico

Um algoritmo de criptografia é um conjunto de instruções e processos que, para prover privacidade, autenticidade e segurança em uma comunicação, definem como o emissor e o receptor vão encriptar e recuperar os dados (BELLARE; ROGAWAY, 2005).

Na criptografia caótica, o algoritmo de Baptista baseia-se na sensibilidade às condições iniciais do mapa logístico para encriptar os dados, particionando de maneira fixa e equidistante o atrator caótico e atribuindo um único caractere para cada intervalo.

O mapa logístico é um mapa real unidimensional que, quando iterado, gera séries temporais periódicas ou caóticas (MAY, 1976). Sua equação é descrita abaixo,

$$x_{n+1} = r x_n (1 - x_n) \tag{3.1}$$

onde $r$ é o parâmetro de controle que varia no domínio de $[0; 4]$ e $x$ representa a variável de estado para o tempo $n$ no intervalo de $[0; 1]$.

Nesse modelo, cada intervalo dentro do atrator caótico particionado tem um tamanho $\epsilon$, que é calculado como a razão entre os extremos do intervalo selecionado pelo número total de caracteres do alfabeto ASCII.

$$\epsilon = \frac{[x_{min} - x_{max}]}{S} \tag{3.2}$$

Segundo Baptista, não há restrição para a escolha do intervalo, contanto que seja em uma região caótica. Ainda, quando a medida natural não é uniforme, a chance de ocorrer a visitação de uma região do espaço de fase é maior do que em outras (OTT; YORKE, 1990), tornando o algoritmo de Baptista lento, já que a associação entre caractere e cifra é feita fixamente e caracteres com maior frequência de aparição podem ser associados a intervalos com baixa visitação. Na figura 3.1 é ilustrado a diferença na forma da medida natural para dois parâmetros de controle. No quadro (a), com $r = 3,78$, temos um caso típico de medida natural irregular. Já em (b), com $r = 4,00$, a medida natural é a mais uniforme possível dentre as regiões caóticas do mapa logístico.

Para contornar essa situação, foi utilizado um algoritmo para ler a mensagem e contar a frequência de cada caractere, assim como a frequência dos intervalos da medida natural. Os caracteres e os intervalos são então ordenados em uma tabela em ordem decrescente, associando

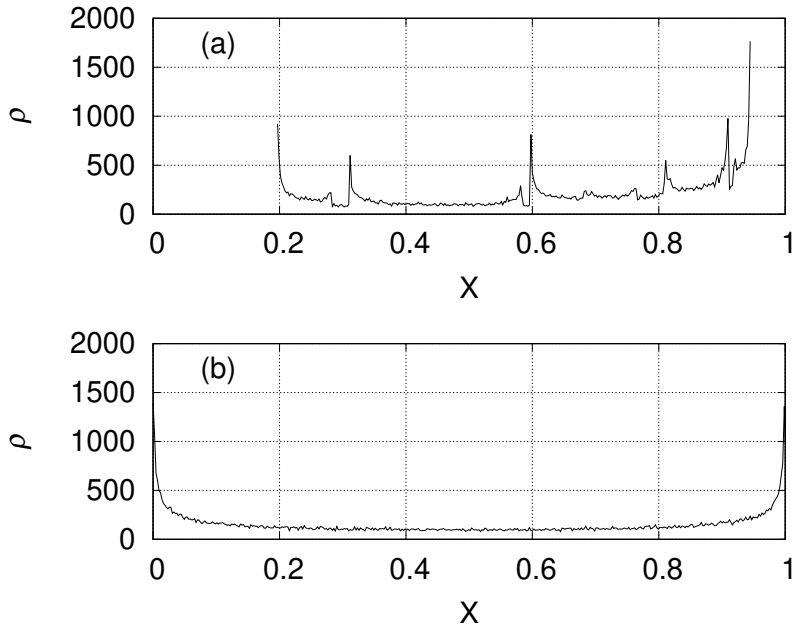


Figura 3.1 – Medida natural ($\rho$) para $r = 3,78$ (a) e $r = 4,00$ (b). Observe como a visitação dos intervalos de $x$ em $r = 3,78$ possui muito mais picos do que em $r = 4,00$.

a letra mais frequente no texto ao intervalo mais visitado. Esse recurso utiliza as irregularidades da medida natural para aumentar a velocidade da criptografia. Para tal tarefa, dois métodos de ordenação conhecidos foram utilizados, conforme descreve a seção a seguir.

## 3.2 Métodos de ordenação

Para ordenar os caracteres da mensagem de texto, bem como a frequência de visitação da medida natural, dois algoritmos de ordenação foram selecionados: o Bubble Sort e o Quick Sort.

O Bubble Sort utiliza a troca de posição dos elementos (caracteres) de uma lista, de modo que todos estes estejam ordenados de forma decrescente no final do processo (CORMEN C. E. LEISERSON; STEIN, 2009). Assim, dado um vetor $A_j$ que deseja-se ordenar em ordem decrescente, com $j = 1, \ldots, n$, onde $n$ é o comprimento do vetor, o algoritmo compara seus elementos um a um, e troca a posição dos elementos em $j + 1$ e $j$, caso $A_{j+1} > A_j$.

No melhor caso, quando o vetor já está em ordem decrescente, são realizadas $n$ operações de ordenação. Caso o vetor esteja em ordem crescente, o pior caso, são efetuadas $n^2$ operações. Esse procedimento é eficaz, principalmente, quando trabalha-se com vetores de tamanho pequeno, e exponencialmente lento, quando utilizado em vetores grandes.

Já o Quick Sort separa o vetor em partes, criando um pivô cada vez que o vetor é dividido

(BIGGAR; GREGG, 2005). Quando o subvetor tiver 7 elementos, é utilizado o Insert Sort para ordenar os caracteres em ordem de frequência (NUMERICAL..., 1986-1992).

Matematicamente, para um dado vetor $A$, este é dividido em dois subvetores ($A[i \dots j-1]$ e $A[j+1 \dots k]$) de forma que $A[i \dots j-1]$ seja menor que o seu pivô $A[j]$. Os dois subvetores são então ordenados através de substituições entre o pivô e o elemento de comparação. Para uma ordem decrescente, se $A[k] > A[j]$, $A[k]$ assume a posição do pivô. Na sequência são comparados os outros elementos, sempre começando do último elemento do vetor que foi trocado.

Se os subvetores possuem o mesmo tamanho, no caso ideal, são necessárias $n \log n$ operações. No pior caso, quando todos os elementos do subvetor são maiores ou menores que o pivô, são criados subvetores com tamanhos $0$ e $n-1$. No próximo passo serão criados subvetores de tamanhos $0$ e $n-2$ e assim sucessivamente até o último subvetor.

## 3.3 Algoritmos de Encriptação Propostos

Conforme mencionado na última seção, foram utilizados dois algoritmos de ordenação para os algoritmos emissores propostos, o Bubblesort e o Quicksort. É através da ordenação que é possível associar a frequência de ocorrência das letras no texto a ser criptografado com a frequência de visitação das partições do atrator, sendo esta, apenas uma etapa da encriptação. De maneira geral, o algoritmo emissor constitui-se de dois procedimentos principais:

- **Subrotinas textuais:** Faz o pré-processamento dos dados como: leitura da mensagem a ser enviada, contagem dos seus caracteres, determinação da frequência de visitação no espaço de fases (medida natural) para o parâmetro de controle escolhido e associação dos caracteres em formato ASCII com a medida natural;
- **Iteração do Mapa Logístico:** É responsável por executar a encriptação da mensagem, utilizando a tabela de associação criada no procedimento anterior para otimizar a velocidade de encriptação.

O diagrama da figura 3.2 mostra o esquema de encriptação utilizado no algoritmo do emissor. No primeiro estágio (esquerda do diagrama), é feita a leitura do texto, são contabilizados os caracteres e cria-se um vetor com a frequência das letras. Depois, o mapa logístico é iterado para obter a medida natural (direita do diagrama) para um determinado parâmetro de controle $r$. A medida natural é particionada e cada intervalo terá tamanho $\epsilon$, conforme equação 3.2, sendo atribuído a cada partição sua respectiva frequência. Nessa parte, cada emissor vai

realizar a ordenação dos vetores utilizando um dos métodos apresentados na subseção 3.2. Por fim, os vetores são associados, criando-se uma tabela em que o caractere mais frequente seja vinculado a partição mais visitada.

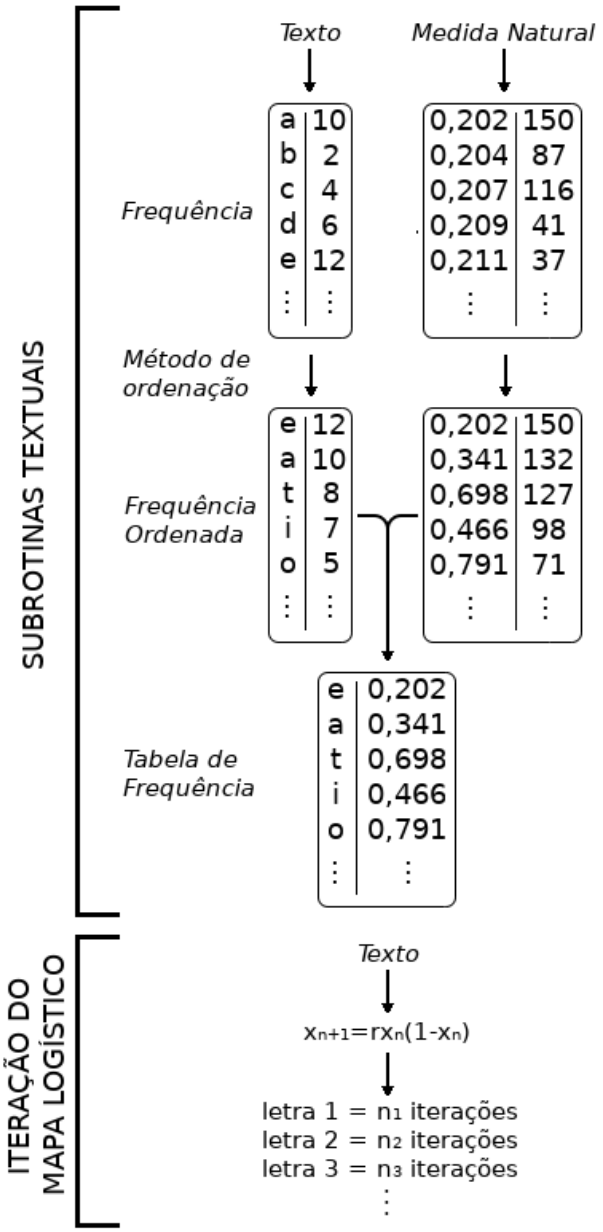


Figura 3.2 – Diagrama de encriptação do emissor.

No segundo estágio, o mapa logístico é iterado $n$ vezes até encontrar a partição referente a letra desejada, conforme a regra de associação criada na tabela de frequência. Após criptografar a primeira letra, o último valor da variável de estado $x$ é utilizado como nova condição inicial para a encriptação do caractere seguinte da mensagem, de modo que a encriptação do caractere posterior dependa deterministicamente da encriptação do caractere anterior. Devido

a dependência sensível as condições iniciais, esse procedimento aumenta a dificuldade de um ataque por força bruta.

Para reobter a mensagem original, o receptor (um clone do emissor, com a tabela de frequência) recebe a chave privada simétrica, composta da primeira condição inicial $x_0$ e o parâmetro de controle $r$. Dessa forma, o mapa logístico pode ser iterado até encontrar os valores numéricos associados aos caracteres da mensagem.

## 3.4 Teste de Segurança

O teste de Wald-Wolfowitz (WALD; WOLFOWITZ, 1940) (ou simplesmente *Runs test*) é um teste não paramétrico, para checar a hipótese de que uma sequência origina-se de um processo randômico. Para tanto, testa-se a hipótese nula ($H_0$) de que uma determinada sequência é randômica. Caso $H_0$ seja rejeitada, a hipótese alternativa ($H_1$) é aceita, e a sequência é dita não-randômica.

O teste de Wald-Wolfowitz considera uma sequência de dados dicotomizada, onde é atribuído o valor $0$ para valores abaixo da referência e $1$ para valores acima da referência, para as séries encriptadas, a mediana é tida como referência. Quando há uma sequência de valores iguais ou valores isolados alternados (seja $0$ ou $1$), na série dicotomizada, formando uma oscilação, define-se um *run*.

É contabilizado o número de *runs* que ocorrem na sequência dicotomizada e o teste estatístico $Z$ é calculado, conforme a equação 3.3

$$Z = \frac{R - \bar{R}}{s_R}, \tag{3.3}$$

onde $R$ é o número de *runs*, $\bar{R}$ é o valor esperado de *runs* e $s_R$ é o desvio padrão do número de *runs*. O valor esperado do número de *runs* ($\bar{R}$) e o desvio padrão do número de *runs* ($s_R$) são calculados conforme as equações 3.4 e 3.5

$$\bar{R} = \frac{2n_1n_2}{n_1 + n_2} + 1 \tag{3.4}$$

$$s_R^2 = \frac{2n_1n_2(2n_1n_2 - n_1 - n_2)}{(n_1 + n_2)^2(n_1 + n_2 - 1)}, \tag{3.5}$$

onde $n_1$ é o número de $1$ e $n_2$ o número de $0$ na série dicotomizada.

Quando o número de *runs* é grande ($n_1 > 10$ e $n_2 > 10$), como no caso das séries criptografadas, o valor de $Z$ é comparado com o valor crítico ($Z_{1-\alpha/2}$) da distribuição normal

(FILLIBEN, 2018), levando em conta o nível de significância $\alpha$, que se refere ao nível de confiança do teste. Dessa forma, se o teste estatístico $Z$ for menor que o valor crítico ($Z_{1-\alpha/2} > Z$), a hipótese nula não é rejeitada e a série é randômica.

# 4 RESULTADOS E DISCUSSÃO

Para mensurar a eficácia dos algoritmos de ordenação em relação ao protocolo de Baptista, foi utilizado trechos do livro (ALLIGOOD; SAUER; YORKE, 1996) e gerado um arquivo de texto com $11197$ caracteres. Todos os algoritmos foram executados em um processador Intel Quad Core i7-4510U, de 4ª geração com $3.1$ GHz.

A tabela 4.1 exibe o tempo consumido pelos algoritmos propostos para cada um dos procedimentos discutidos no capítulo anterior. Além disso, compara os resultados com o algoritmo original de Baptista. Os valores são obtidos utilizando um parâmetro de controle $r = 3,78$ a partir de uma condição $x_0 = 0,232323$ (os mesmos utilizados por Baptista).

Percebe-se que o tempo de execução das subrotinas textuais é $10$ vezes menor em comparação a iteração do mapa logístico, o que se deve ao processo de iteração exigir mais processamento do que as subrotinas textuais.

Tabela 4.1 – Tempo consumido pelo algoritmo para cada técnica.

| | Baptista | Bubble Sort | Quick Sort |
|---|---|---|---|
| Subrotinas Textuais | 0,0079329s | 0,004652s | 0,0048195s |
| Iteração do Mapa Logístico | 0,0726929s | 0,0330327s | 0,0325506s |
| Total | 0,0806258s | 0,0376847s | 0,0373701s |

No entanto, os emissores Bubblesort e Quicksort são 53,26% e 53,74%, respectivamente, mais rápidos que o emissor Baptista.

Para comparar a velocidade dos algoritmos para as outras regiões caóticas, foi criado um mapa de razão de tempo com intervalos de tamanho $re$, dado por:

$$re = \frac{r_{final} - r_{inicial}}{R} \tag{4.1}$$

onde $r_{inicial}$ é o início do espaço de parâmetros, $r_{final}$ o fim do espaço de parâmetros e $R$ é a quantidade de intervalos utilizados. Para as figuras que seguem foi considerado $R = 100$ dentro de um intervalo de $r = [3,56; 4,00]$.

As figuras 4.1 e 4.2 mostram a razão dos tempos de criptografia entre os algoritmos Bubblesort/Baptista e Quicksort/Baptista, respectivamente. Um valor menor que $1$ (branco, verde ou azul na escala de cores) corresponde a uma região do espaço de parâmetros onde os algoritmos propostos são mais rápidos que o algoritmo de Baptista. Para os algoritmos emissores propostos, todas as regiões são azul, brancas ou verdes (menores que $1$), oferecendo portanto, uma otimização no tempo de encriptação para todo o espaço de parâmetros.

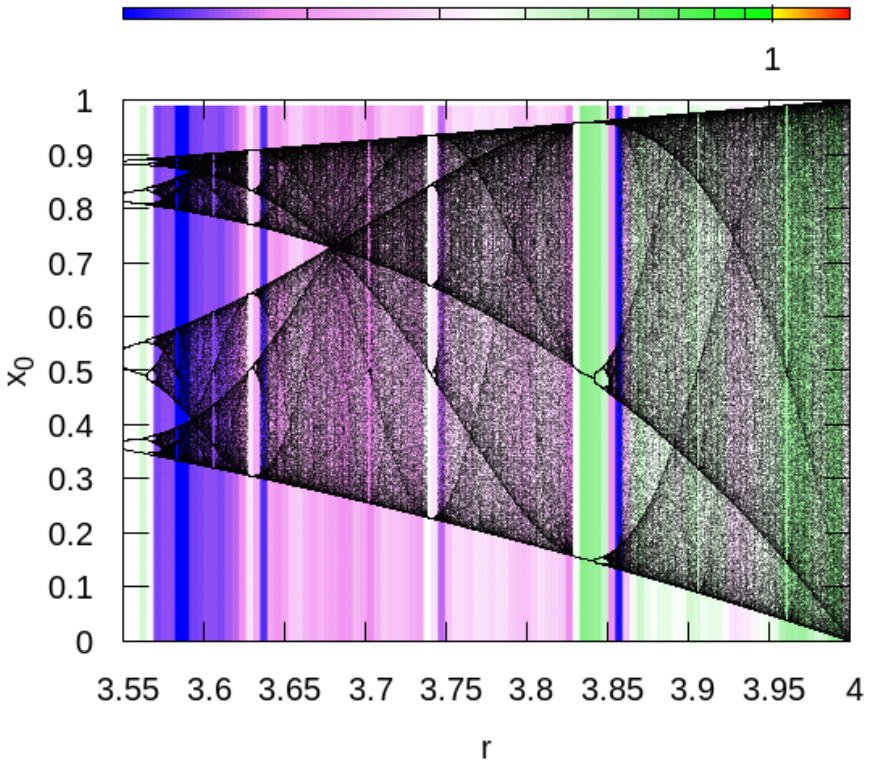


Figura 4.1 – Mapa de razão do tempo de encriptação. As regiões entre azul, branco e verde identificam onde o emissor Bubblesort é mais rápido que o protocolo de Baptista.

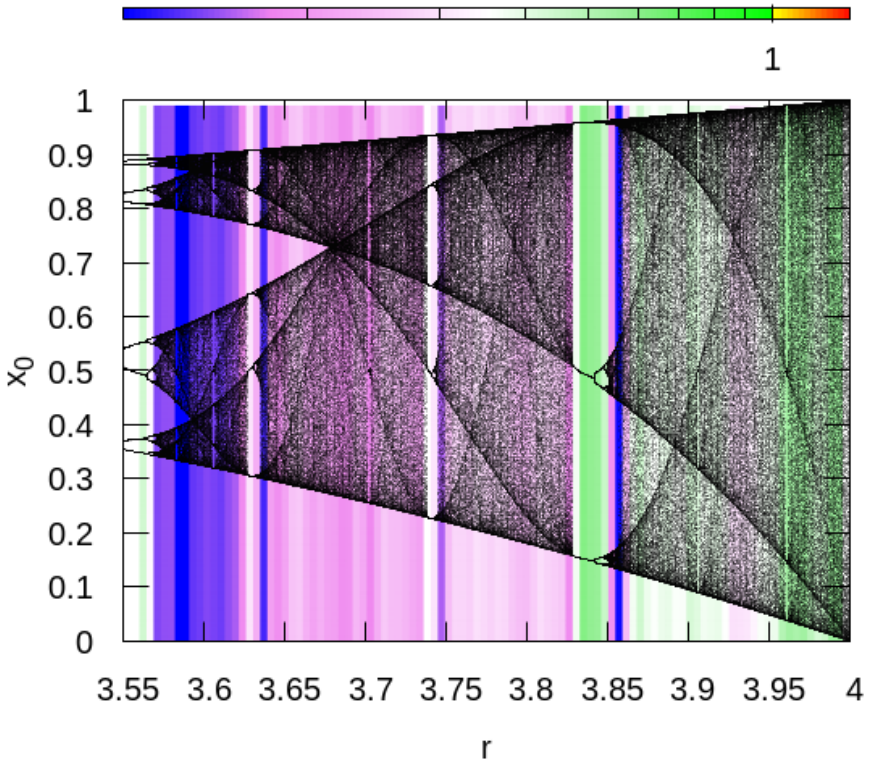


Figura 4.2 – Mapa da razão do tempo de encriptação. As regiões entre branco e verde identificam onde o emissor Quicksort é mais rápido que o protocolo de Baptista.

A média do tempo de execução dos algoritmos sobre os parâmetros de controle do mapa (100 valores no intervalo $r = [3,56; 4,0]$) nos fornece o tempo médio de encriptação. Na tabela 4.2 é exibido o tempo médio, para cada algoritmo, para uma frase com $11197$ caracteres. Dessa forma, podemos visualizar qual algoritmo, na média, é mais rápido em relação aos outros.

Tabela 4.2 – Tempo médio consumido pelos algoritmos.

| | Baptista | Bubble Sort | Quick Sort |
|---|---|---|---|
| Velocidade Média | 1,07798673s | 0,4879576s | 0,49285718s |

Os mapas mostrados anteriormente são dependentes do tipo e do tamanho das frases encriptadas. Dessa forma, é importante avaliar esses parâmetros a fim de obter uma ideia geral da eficiência de todos os protocolos estudados. A estratégia adotada é aumentar o tamanho da frase gradativamente, pois à medida que a frase encriptada aumenta de tamanho, a frequência de ocorrência das letras na frase se aproxima mais da frequência média de ocorrência dessas letras no alfabeto. Assim, frases longas possuem uma maior similaridade média entre si do que frases curtas.

Para se ter uma estimativa da dependência dos resultados em relação ao tamanho da frase, pode-se também avaliar os valores médios do tempo de criptografia, para o intervalo de $r = [3,56; 4,0]$, variando o número de caracteres da frase. Esse resultado é exibido na Figura 4.3, que mostra os resultados da Tabela 4.2 para $9$ tamanhos de frases.

Através da média de velocidade exibida no gráfico, podemos perceber a melhora no tempo de encriptação dos algoritmos que utilizam métodos de ordenação. Para uma mensagem com $11197$ caracteres, a redução no tempo foi de $1,12$ segundo para $0,49$ segundos, aproximadamente $56,25\%$ de vantagem quando comparamos os tempos do emissor Quicksort e do emissor Baptista.

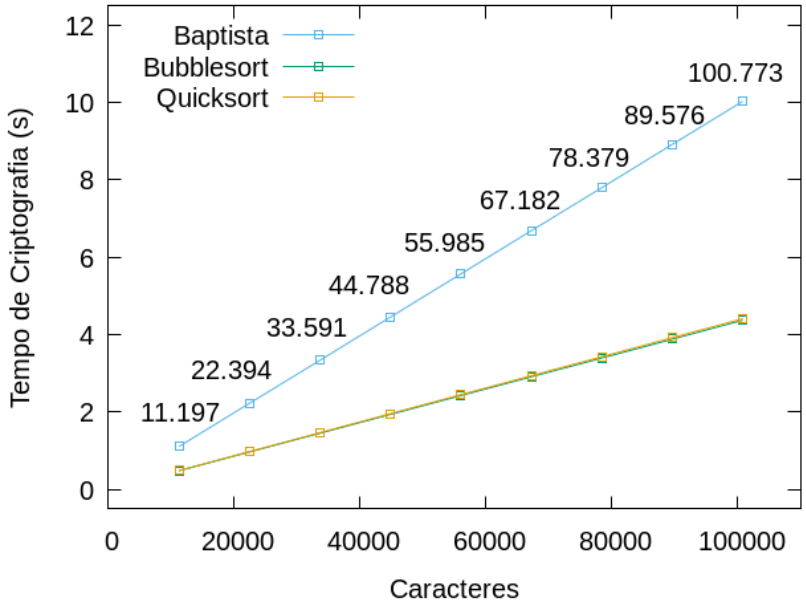


Figura 4.3 – Velocidade de encriptação média para os algoritmos analisados de acordo com o tamanho da frase. No testes, foram utilizados 9 tamanhos de texto, encriptados a partir da condição inicial $x_0 = 0,232323$.

## 4.1 Teste de Wald-Wolfowitz

Para os testes de aleatoriedade de Wald-Wolfowitz foram escolhidas 5 regiões do atrator caótico com dinâmicas distintas: $r = 3,60$ (próxima do *onset* do caos), $r = 3,78$ (região caótica com medida natural não-uniforme), $r = 3,82$ (antes de uma janela periódica, $r = 3,93$ (em uma região caótica ligeiramente uniforme) e $r = 4,00$ (região caótica com medida natural quase uniforme). Para dicotomizar e contabilizar o número de *runs* das séries temporais criptografadas, considerou-se a mediana como valor de referência.

Para os testes, optou-se por uma significância $\alpha = 0,05$, ou seja, $95\%$ de nível de confiança. Assim, quando $p < \alpha \rightarrow h = 1$ e a hipótese nula, de que as séries temporais são randômicas, é rejeitada. Em contrapartida, as séries são consideradas randômicas quando a hipótese nula não é rejeitada, ou seja, quando $p > \alpha \rightarrow h = 0$. A Tabela 4.3 indica os valores obtidos nos testes de Wald-Wolfowitz para os 5 valores de $r$ pré-determinados. A aleatoriedade da mensagem encriptada utilizando os parâmetros de controle escolhidos foi confirmada para todos os emissores, portanto, os algoritmos de Baptista e Bubblesort mostraram alguns pontos inseguros, como em $r = 3,60$ e em $r = 3,93$, respectivamente.

Para definir as regiões onde a mensagem encriptada é considerada segura, aplicamos o teste de Wald-Wolfowitz para cada parâmetro de controle no intervalo de $r = 3,56; 4,00$. O teste indica com um erro de de $\pm 1\%$ as regiões onde a encriptação da mensagem resulta

Tabela 4.3 – Teste de Wald-Wolfowitz para alguns valores aleatórios de $r$. Para cada um dos quatro algoritmos abaixo, é indicado o valor de h obtido no teste, com o seu respectivo valor de $p$, entre parênteses.

| r | Baptista | Bubblesort | Quicksort |
|---|---|---|---|
| 3,60 | 1 (0,00013) | 0 (0,56365) | 0 (0,37806) |
| 3,78 | 0 (0,19849) | 0 (0,66300) | 0 (0,46039) |
| 3,82 | 0 (1,00000) | 0 (0,09039) | 0 (0,93964) |
| 3,93 | 0 (0,53890) | 1 (0,04579) | 0 (0,70462) |
| 4,00 | 0 (0,64302) | 0 (0,26453) | 0 (0,56406) |

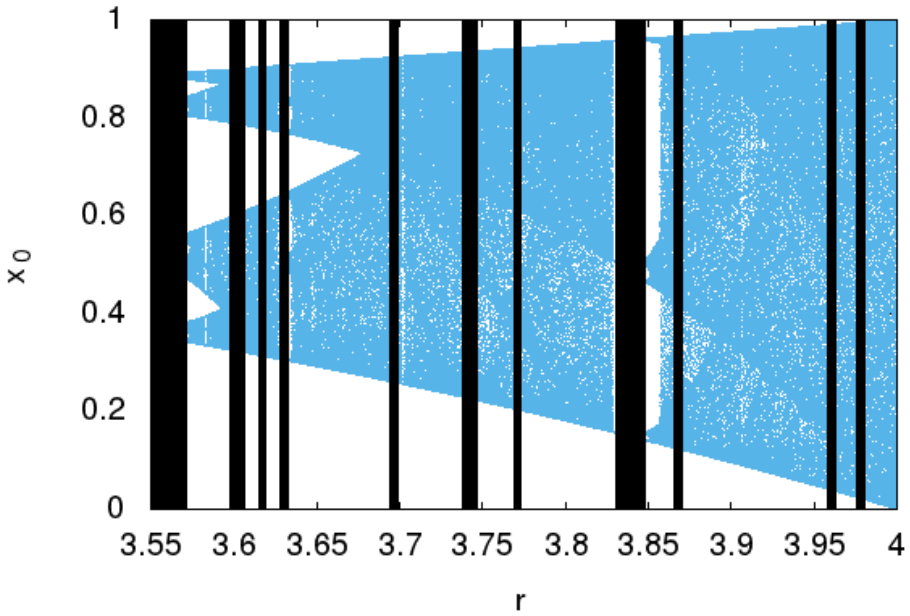


Figura 4.4 – Mapa de Wald-Wolfowitz para séries encriptadas pelo emissor Baptista. Regiões em claro indicam h=0, enquanto regiões escuras indicam h=1.

em séries randômicas o suficiente para serem consideradas seguras. As figuras 4.4, 4.5 e 4.6 exibem as regiões do espaço de parâmetros onde as séries são consideradas aleatórias pelo *runs* teste para os emissores Baptista, Bubblesort e Quicksort, respectivamente. As regiões claras são onde a hipótese nula não é rejeitada, já as regiões em escuro são regiões onde a hipótese nula é rejeitada e a série é considerada insegura.

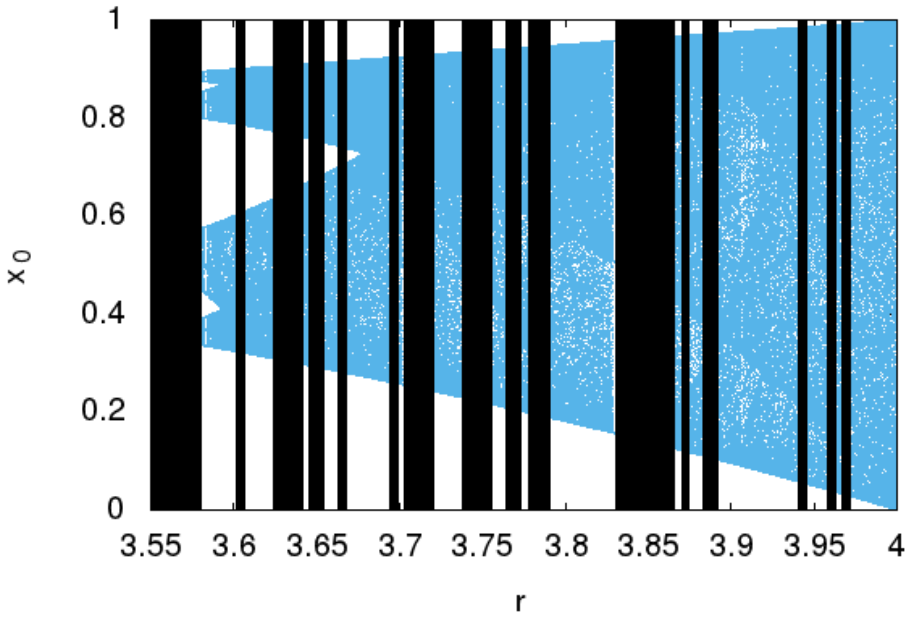


Figura 4.5 – Mapa de Wald-Wolfowitz para séries encriptadas pelo emissor Bubblesort. Regiões em claro indicam h=0, enquanto regiões escuras indicam h=1.

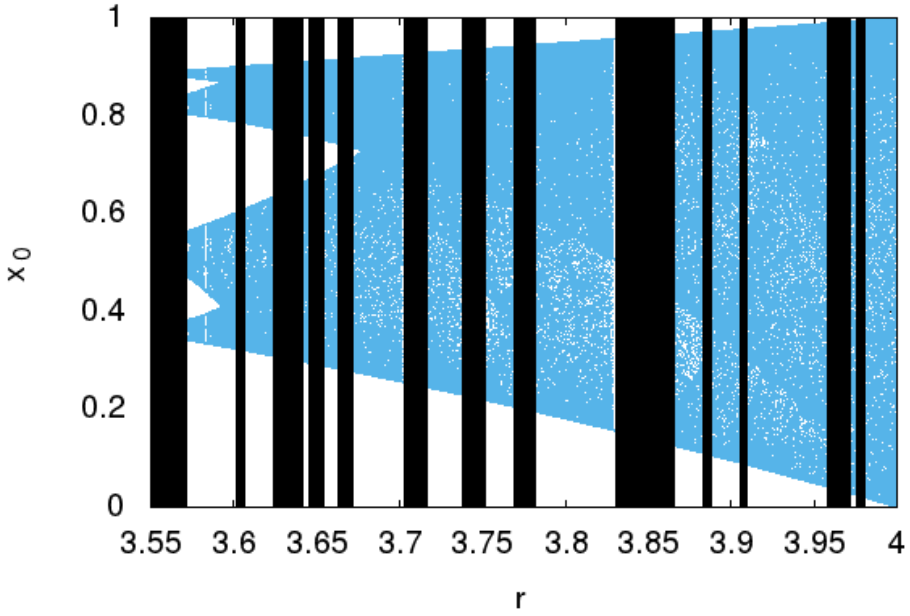


Figura 4.6 – Mapa de Wald-Wolfowitz para séries encriptadas pelo emissor Quicksort. Regiões em claro indicam h=0, enquanto regiões escuras indicam h=1.

## 4.2 Regiões Próprias para Encriptação

O mapa de razão de tempo compara o tempo de encriptação dos algoritmos através da razão entre eles, quanto menor a razão, mais rápido é o emissor proposto em relação ao emissor de Baptista. Já o teste de Wald-Wolfowitz permite avaliar se uma determinada série é randômica o suficiente para ser considerada segura.

O mapa de regiões próprias para encriptação une essas duas características em um único mapa, exibindo apenas os parâmetros de controle ideais para realizar uma encriptação rápida e segura.

### 4.2.1 Gráfico de Razão de Tempo < 1

O gráfico de regiões próprias para encriptação, com Razão de Tempo menor que 1 permite avaliar a disponibilidade de regiões seguras e mais rápidas que o emissor de Baptista. O emissor Quicksort obteve vantagem no número de regiões próprias para encriptação, assim como a média da razão do tempo de encriptação das regiões se mostrou ligeiramente menor que o emissor Bubblesort. Isto indica que, para o emissor Quicksort, a maioria das regiões com razão de tempo menor que 1 e consideradas seguras, de acordo com o teste de Wald-Wolfowitz, são mais rápidas em relação ao emissor Bubblesort, conforme podemos ver na tabela 4.4.

Tabela 4.4 – Tabela de Regiões Próprias para Encriptação para Razão de Corte = 1.

| Emissor | Bubblesort | Quicksort |
|---|---|---|
| Número de Regiões Próprias para Encriptação | 53 | 62 |
| Razão de tempo média dos intervalos (s) | 0,39288541 | 0,36523559 |

Podemos observar as regiões escuras no Mapa de Regiões Próprias para Encriptação do emissor Bubblesort na Figura 4.7, indicando que essas regiões não são próprias para uma encriptação mais rápida que o emissor Baptista, ou são regiões com pouca segurança para encriptação.

O emissor Quicksort apresenta o mesmo padrão, com intervalos claros e escuros bem distribuídos sobre o espaço de parâmetros, como vemos na Figura 4.8. Com 62 regiões próprias para encriptação, o emissor Quicksort apresenta maior quantidade de intervalos próprios para encriptação que o emissor Bubblesort. A média da Razão de Tempo do emissor Quicksort em relação ao emissor Baptista mostra um algoritmo ligeiramente mais rápido, comparado a média de Razão de Tempo do emissor Bubblesort.

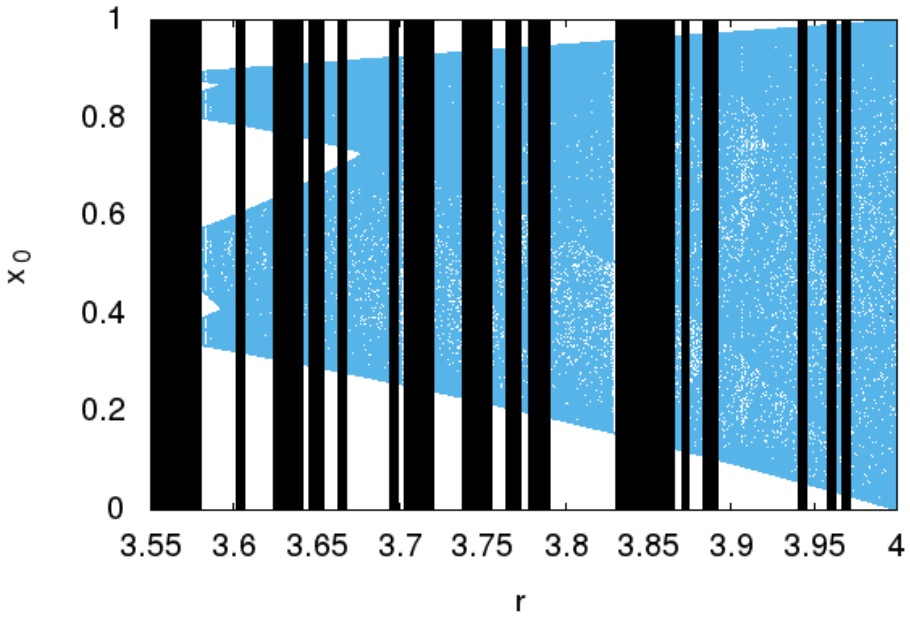


Figura 4.7 – Mapa de Regiões Próprias para Encriptação com razão de tempo menor que 1 e h=0 para o emissor Bubblesort.

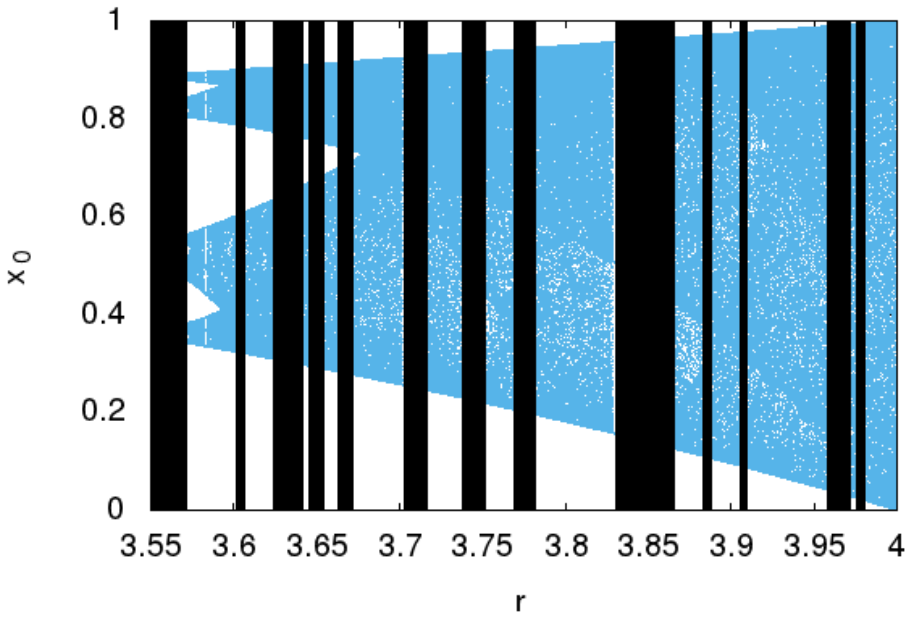


Figura 4.8 – Mapa de Regiões Próprias para Encriptação com razão de tempo menor que 1 e h=0 para o emissor Quicksort.

### 4.2.2 Gráfico de Razão de Tempo < 0.5

Para o gráfico de Razão de Tempo menor que $0,5$ são esboçados os mesmos dados para regiões seguras, portanto utilizando a metade da razão de tempo, ou seja, regiões onde os algoritmos propostos são, peo menos, $2$ vezes mais rápidos que o algoritmo de Baptista. São destacadas as regiões super rápidas.

Podemos perceber a vantagem do Quicksort em relação ao emissor Bubblesort no número de regiões próprias para encriptação, sendo que o Quicksort apresenta 49 regiões próprias para encriptação contra 39 regiões próprias para encriptação do Bubblesort, conforme é mostrado na tabela 4.5.

Tabela 4.5 – Tabela de Regiões Próprias para Encriptação para Razão de Corte = 0,5

| Emissor | Bubblesort | Quicksort |
|---|---|---|
| Número de Regiões Próprias para Encriptação | 39 | 49 |
| Razão de tempo média dos intervalos (s) | 0,27512967 | 0,28166033 |

O número de Regiões Próprias para Encriptação diminui conforme a Razão de Tempo diminui, como mostrado na Figura 4.9. O emissor Bubblesort ocultou algumas regiões caóticas ao utilizar a razão de tempo menor que $0,5$.

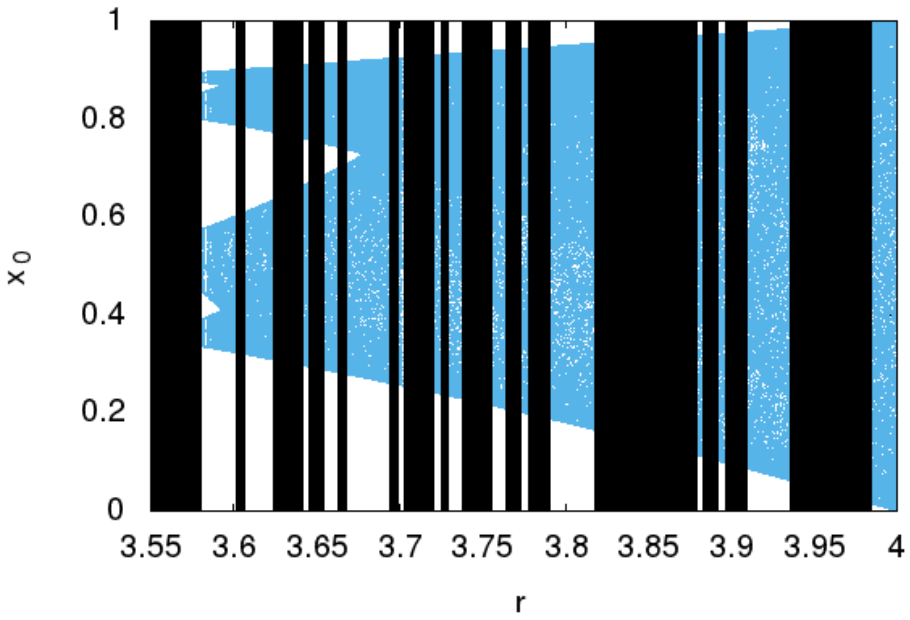


Figura 4.9 – Mapa de Regiões Próprias para Encriptação com razão de tempo menor que 0,5 e h=0 para o emissor Bubblesort.

É mostrado na Figura 4.10 o mapa de regiões rápidas e seguras do emissor Quicksort, que apresenta um grande número de regiões próprias para encriptação.

A partir das análises de velocidade e segurança, podemos destacar os seguintes quesitos, englobando as vantagens e desvantagens dos algoritmos de encriptação propostos:

**Segurança:**

Bubblesort: Redução de 28,05% no número de Regiões Seguras;

Quicksort: Redução de 20,73% no número de Regiões Seguras;

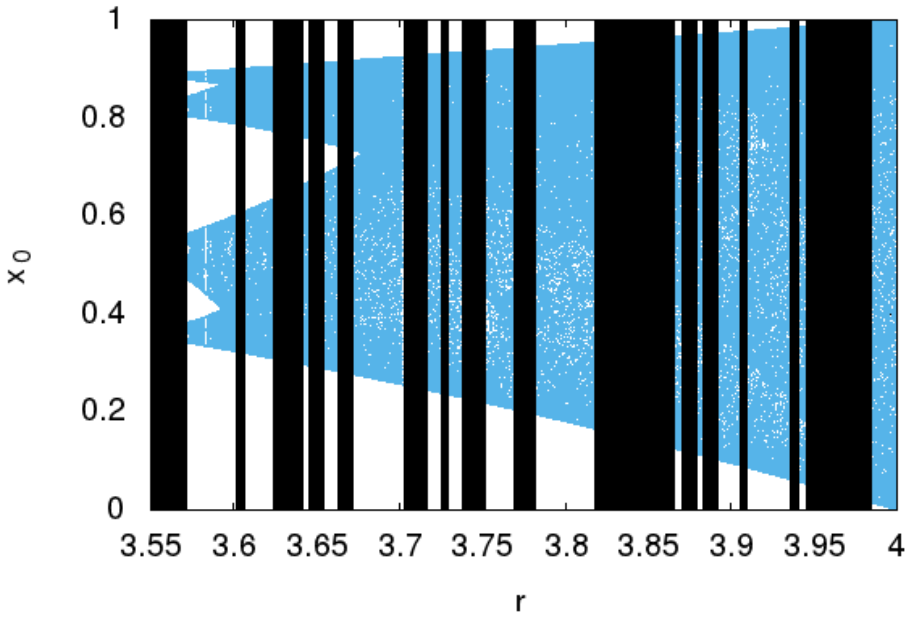


Figura 4.10 – Mapa de Regiões Próprias para Encriptação com razão de tempo menor que 0,5 e h=0 para o emissor Quicksort.

**Razão de Tempo < 1:**

Bubblesort: Aumento na velocidade média em 63,34%;

Quicksort: Aumento na velocidade média em 65,14%;

**Razão de Tempo < 0,5:**

Bubblesort: Aumento na velocidade média em 47,59%;

Quicksort: Aumento na velocidade média em 44,83%;

## 4.3 Teste de Velocidade para Mensagens Diferentes

Os algoritmos Bubblesort e Quicksort mostraram que é possível encriptar com velocidade maior que o algoritmo de Baptista. Foram realizados testes para encriptar outras mensagens, abrangendo diferentes áreas do conhecimento e da literatura para avaliar a velocidade de encriptação ao mudar a mensagem e, consequentemente, a frequência dos caracteres. Os textos escolhidos para encriptar possuem exatamente 120.000 caracteres.

Para criar os arquivos de texto, foram escolhidos 10 livros de diversas áreas. As mensagens de texto foram nomeadas com a abreviação do nome do livro. Sendo assim, abaixo é listado o livro correspondente a cada mensagem:

- **bio-cor**: Biology: The Core, de Eric Simon (SIMON, 2019);
- **hol-bib**: The Holy Bible, da editora Crossway Bibles (BIBLES, 2011);

- **lin-alg**: Linear algebra: concepts and methods, de Martin Anthony e Michele Harvey (ANTHONY; HARVEY, 2012).
- **reb-bac**: Rebellion in the Backlands, de Euclides da Cunha (CUNHA, 2010);
- **sto-art**: The story of art, de Ernst Gombrich (GOMBRICH, 1995);
- **wor-geo**: World Regional Geography: Global Patterns, Local Lives, de Lydia Pulsipher e Alex Pulsipher (PULSIPHER; PULSIPHER, 2013);
- **cha-int**: Chaos: An Introduction for Applied Mathematicians, de Andrew Fowler e Mark McGuinness (FOWLER; MCGUINNESS, 2020);
- **ifi-ble**: If It Bleeds, de Stephen King (KING, 2020);
- **man-che**: The Complete Manual of Positional Chess: The Russian Chess School 2.0, Volume 2: Middlegame Structures and Dynamics, de Sakaev Konstantin e Landa Konstantin (KONSTANTIN, 2017);
- **con-sys**: Control systems engineering, de Norman Nise (NISE, 2020).

O gráfico 4.11 mostra o tempo de encriptação médio para todos os parâmetros de controle em $r = [3,56; 4,00]$, desconsiderando as regiões periódicas. Os algoritmos propostos se mostraram mais rápido em todas as mensagens, sendo mais eficientes para encriptação em comparação ao algoritmo de Baptista.

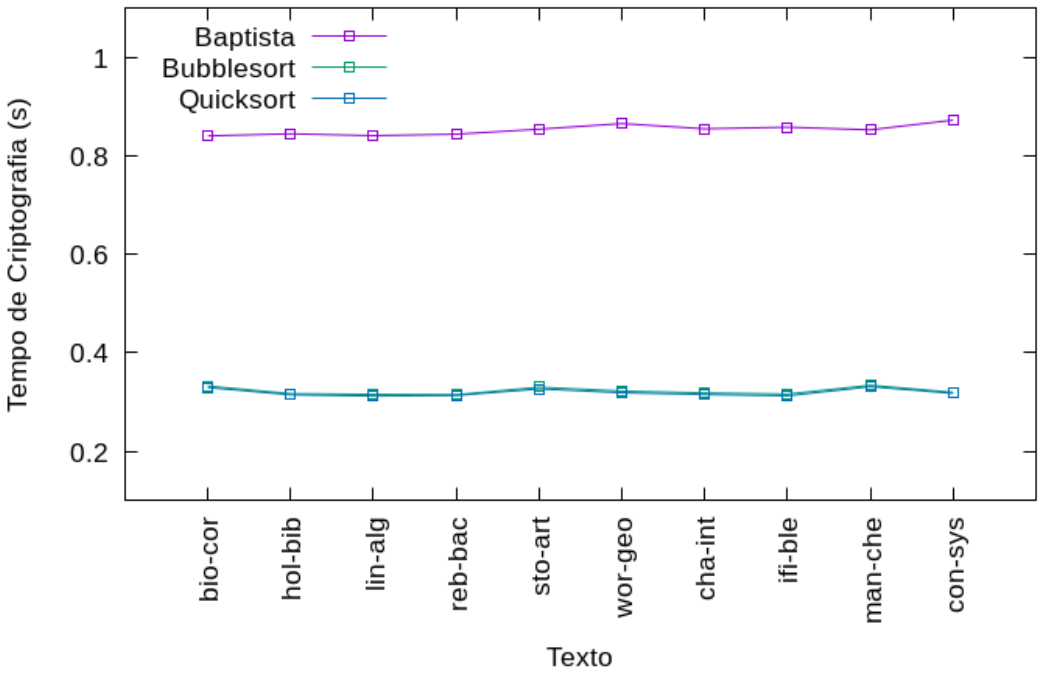


Figura 4.11 – Tempo de encriptação para 10 mensagens diferentes de tamanho fixo de 120.000 caracteres.

# 5 CONCLUSÕES

Neste trabalho, comparou-se o protocolo de Baptista com outros algoritmos: Bubblesort e Quicksort. Através das análises apresentadas, evidencia-se os benefícios da implementação dos algoritmos de ordenação na associação de caracteres à medida natural, o que oferece um método de otimizar a velocidade do protocolo originalmente proposto por Baptista.

Foi feito testes de velocidade para todo o espaço de parâmetros de controle, embasando a conclusão que o emissor Quicksort e o emissor Bubblesort tem potencial de otimizar a velocidade de encriptação caótica. Através dos testes de aleatoriedade de Wald-Wolfowitz, concluímos que os emissores propostos encriptam a mensagem de forma segura, do ponto de vista randômico.

Os mapas de regiões próprias para encriptação forneceram uma visualização das melhores regiões para encriptar utilizando cada algoritmo. Ao implementar otimizações de velocidade de encriptação, deve-se atentar a redução do número de regiões seguras, equilibrando racionalmente o custo-benefício entre segurança e velocidade.

Os algoritmos Bubblesort e Quicksort cumpriram o objetivo de encriptar mais rápido que o algoritmo original de Baptista, em última análise, os algoritmos propostos são capazes de encriptar até $2,5$ vezes mais rápido que o algoritmo de Baptista, conforme visualizado no gráfico 4.11.

## 5.1 Trabalhos Futuros

Implementar técnicas de redução do tamanho da cifra-texto, diminuindo o tamanho do arquivo que será enviado para o receptor;

Aplicar técnicas otimizadas de segurança, como a substituição do número de iterações por índices e realizar novos testes de segurança, aplicando ataques conhecidos na criptografia, conforme sugerido por (ÁLVAREZ F. MONTOYA; PASTOR, 2003);

Elaboração de um protocolo de criptografia caótica independente para o compartilhamento das chaves.

# REFERÊNCIAS

ALLIGOOD, K. T.; SAUER, T. D.; YORKE, J. A. **CHAOS: An Introduction to Dynamical Systems**. [S.l.]: Springer-Verlag, 1996.

ALVAREZ, E. et al. New approach to chaotic encryption. **Physics Letters A**, v. 263, 1999.

ALVAREZ, G.; LI, S. Some basic cryptographic requirements for chaos-based cryptosystems. **International Journal of Bifurcation and Chaos**, v. 16, 2006.

ANTHONY, M.; HARVEY, M. **Linear algebra: concepts and methods**. [S.l.]: Cambridge University Press, 2012.

ARROYO, D.; ALVAREZ, G.; FERNANDEZ, V. On the inadequacy of the logistic map for cryptographic applications. Instituto de Física Aplicada, 2008.

BAPTISTA, M. S. Cryptography with chaos. **Physics Letters A**, 1998.

BELLARE, M.; ROGAWAY, P. **Introduction to Modern Cryptography**. [S.l.]: Ucsd Cse, 2005.

BIBLES, C. **The Holy Bible, Englih Standard Version**. [S.l.]: Crossway Bibles, 2011.

BIGGAR, P.; GREGG, D. **Sorting in the Presence of Branch Prediction and Caches**. [S.l.: s.n.], 2005.

CORMEN C. E. LEISERSON, R. L. R. T.; STEIN, C. **Introduction to Algorithms**. [S.l.: s.n.], 2009.

CUNHA, E. da. **Rebellion in the Backlands**. [S.l.]: University of Chicago Press, 2010.

FILLIBEN, J. J. Runs test for detecting non-randomness. NIST/SEMATECH, 2018. Available from Internet: <https://www.itl.nist.gov/div898/handbook/eda/section3/eda35d.htm>.

FOWLER, A.; MCGUINNESS, M. **Chaos: An Introduction for Applied Mathematicians**. [S.l.]: Springer Nature, 2020.

FRASER, B.; YU, P.; LOOKMAN, T. Steps towards improving the security of chaotic encryption. **Physical Review E**, v. 66, 2002.

GOMBRICH, E. H. **The story of art**. [S.l.]: London: Phaidon, 1995.

HAYES, C. G. S.; OTT, E. Communicating with chaos. **Physical Review Letters**, v. 70, 1993.

HAYES C. GREBOGI, E. O. S.; MARK, A. Experimental control of chaos for communication. **Physical Review Letters.**, v. 73, 1994.

JAKIMOSKI, G.; KOCAREV, L. Analysis of some recently proposed chaos-based encryption algorithms. **Physics Letters A**, v. 291, 2001.

JOVIC, B. **Synchronization Techniques for Chaotic Communication Systems**. [S.l.]: Springer, 2011.

KHALEQUE, A.; SEN, P. Effect of randomness in logistic maps. **International Journal of Modern Physics C**, v. 26, 2015.

KING, S. **If It Bleeds**. [S.l.]: Scribner, 2020.

KONSTANTIN, L. K. S. **The Complete Manual of Positional Chess: The Russian Chess School 2.0, Volume 2: Middlegame Structures and Dynamics**. [S.l.]: New in Chess, 2017.

LI, S. et al. Baptista-type chaotic cryptosystems: problems and countermeasures. **Physics Letters A**, v. 332, 2004.

LI, S. et al. Performance analysis of jakimoski-kocarev attack on a class of chaotic cryptosystems. IIPSEIE - Xi'an Jiaotong University, 2003.

MARTIZ, M. F. A note on exact solutions of the logistic map. **Chaos**, v. 30, 2020.

MAY, R. M. Biological populations obeying difference equations: Stable points, stable cycles, and chaos. **Journal of Theoretical Biology**, v. 51, 1974.

MAY, R. M. Simple mathematical models with very complicated dynamics. **Nature**, v. 261, 1976.

MENEZES, A. J.; OORSCHOT, P. van; VANSTONE, S. **Handbook of Appl. Cryptography**. [S.l.]: CRC Press, 1996.

NISE, N. S. **Control systems engineering**. [S.l.]: John Wiley & Sons, 2020.

NUMERICAL Recipes in Fortran 77: The Art of Scientific Computing. [S.l.]: Cambridge University Press, 1986–1992.

OTT, C. G. E.; YORKE, J. A. Controlling chaos. **Physical Review Letters**, v. 64, 1990.

PANDE, A.; ZAMBRENO, J. A chaotic encryption scheme for real-time embedded systems: design and implementation. **Springer Telecommun Syst**, 2011.

PECORA, L. M.; CARROLL, T. L. Synchronization in chaotic system. **Physical Review Letters**, v. 64, 1990.

PULSIPHER, L. M.; PULSIPHER, A. **World Regional Geography: Global Patterns, Local Lives**. [S.l.]: Macmillan Higher Education, 2013.

SAN-UM, W.; KETTHONG, P. The generalization of mathematically simple and robust chaotic maps with absolute value nonlinearity. IEEE, 2014.

SHANNON, C. E. Communication theory of secrecy systems. **Bell System Technical Journal**, 1949.

SILVA, C. P. A survey of chaos and its applications. **IEEE MTT-S International Microwave Symposium Digest**, v. 3, 1996.

SILVA, F. C.; SOUSA, J. J. New comparative study between des, 3des and aes within nine factors. **Journal of Computing**, v. 2, 2010.

SIMON, E. J. **Biology: The Core**. [S.l.]: Pearson Education Inc., 2019.

SMAOUI, N.; KANSO, A. Cryptography with chaos and shadowing. **Chaos, Solitons and Fractals**, v. 42, 2009.

TANG, A. I. M. Y. S.; CHUA, L. O. Synchronization and chaos. **IEEE Transactions on Circuits and Systems**, v. 30, 1983.

WALD, A.; WOLFOWITZ, J. On a test whether two samples are from the same population. **Ann. Math. Statist.**, v. 11, 1940.

WANG, X.; WANG, X. A new chaotic encryption algorithm based on the ergodicity of chaos. **International Journal of Modern Physics B**, v. 25, 2011.

WANG, Y. X. X.; QIN, X. Cryptanalysis of an ergodic chaotic encryption algorithm. **Chin. Phys. B**, v. 21, 2012.

WEI, J. et al. Analysis and improvement for the performance of baptista's cryptographic scheme. **Physics Letters A**, v. 354, 2006.

WEI, X. L. J.; WONG, T. X. K. A new chaotic cryptosystem. **Chaos, Solitons and Fractals**, v. 30, 2006.

WONG, K.; YUEN, C. Embedding compression in chaos-based cryptography. **IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS**, v. 55, 2008.

WONG, S. H. K.; YUNG, C. A chaotic cryptography scheme for generating short ciphertext. **Physics Letters A**, v. 310, 2003.

WONG, W.; LEE, L.; WONG, K. A modified chaotic cryptographic method. **Computer Physics Communications**, v. 138, 2001.

ÁLVAREZ F. MONTOYA, M. R. G.; PASTOR, G. Cryptanalysis of an ergodic chaotic cipher. **Physics Letters A**, v. 311, 2003.