

Codificando Rongcheng: Modelagem de Crédito Social via Redes Complexas e Simulação Baseada em Agentes (ABM)

Aluno: Pedro Medina Lara Silva¹,

Orientador: Martín Gómez Ravetti¹

Tipo: Científica e Tecnológica

¹Departamento de Ciência da Computação, Universidade Federal de Minas Gerais

Abstract

This work develops a modular computational tool for stochastic simulation of social credit systems, using the Rongcheng case as an empirical reference for parameterization. The architecture implemented in Python combines an Agent-Based Monte Carlo simulation engine with an interactive graphical interface (PySide6), enabling controlled experimentation on emergent dynamics in social networks. The system supports multiple network topologies (Watts–Strogatz and Barabási–Albert), parameterizable stochastic events, and social influence mechanisms, offering features such as state export, batch execution, and real-time visualization. The implementation includes performance optimizations (JIT compilation via Numba, network structure caching) and an automated experimental pipeline with non-parametric statistical validation. Experiments reveal that network topological structure and event distribution modulate the collective evolution of scores, demonstrating the tool's potential for analyzing algorithmic governance scenarios and distributed reputation systems.

Keywords: Agent-Based Simulation; Monte Carlo; Complex Networks; Social Credit Systems; Algorithmic Governance; Rongcheng; Computational Modeling.



Figura 1: Mural público em Rongcheng exibindo pontuações de crédito social atribuídas a cidadãos, ilustrando a visibilidade pública dessas classificações e seu uso como forma de incentivo e sinalização social. Fonte: <https://izzso.com/chinas-social-credit-system-and-its-penalties-on-citizens/>.

Sumário

Listas	3
1 Introdução	4
1.1 Contextualização	4
1.2 Objetivos e Hipóteses	4
1.3 Estrutura do Trabalho	5
2 Trabalhos Relacionados	5
2.1 Estudos Empíricos e Sociológicos do SCS	5
2.2 Modelagem via Teoria dos Jogos	5
2.3 Sistemas de Reputação e Simulação Social	6
2.4 Parametrização Baseada em Rongcheng	6
3 Arquitetura do Sistema	6
3.1 Justificativa da Implementação Ad-Hoc	6
3.2 Visão Geral Conceitual	7
3.3 Interface Gráfica	8
4 Modelo de Simulação	9
4.1 Notação Matemática e Definições Preliminares	9
4.2 Inicialização de Agentes e Demografia	9
4.3 Topologias de Rede	10
4.4 Modelagem de Eventos	10
4.5 Calibração e Estabilização do Sistema	11
4.6 Cálculo de Probabilidades de Eventos	12
4.7 Integração dos Modificadores e Fluxo de Simulação	14
5 Metodologia Experimental	15
5.1 Dois Modos de Uso: Exploratório e Experimental	15
5.2 Parâmetros Computacionais	16
5.3 Cenários Implementados	17
5.4 Métricas de Estabilidade e Convergência	17
6 Resultados e Análise	18
6.1 Impacto da Estrutura da Rede	18
6.2 Resultados Estatísticos	20
6.3 Síntese dos Resultados	21
7 Discussão	21
7.1 Análise dos Cenários Experimentais	22
7.2 Limitações do Modelo e Considerações sobre Transferibilidade Cultural	22
7.3 Nota sobre Uso de Ferramentas de IA	23
8 Conclusão	23
8.1 Principais Contribuições	23
8.2 Validação das Hipóteses	24
8.3 Trabalhos Futuros	24
8.4 Considerações Finais	25
A Sistema de Pontuação de Rongcheng	28
B Exemplos de Estrutura JSON de Eventos	29
C Sistema de Gravação de Dados em Tempo Real	30
D Otimizações Computacionais	30

Listas

Lista de Figuras

1	Mural público em Rongcheng exibindo pontuações de crédito social atribuídas a cidadãos, ilustrando a visibilidade pública dessas classificações e seu uso como forma de incentivo e sinalização social. Fonte: https://izzso.com/chinas-social-credit-system-and-its-penalties-on-citizens/	1
2	Interface gráfica do simulador. Painel esquerdo: controles de parâmetros e configuração. Centro: visualização da rede social. Painel inferior: estatísticas, gráficos de evolução temporal e distribuição de pontuações.	8
3	Capturas de tela da interface do simulador: (a) Janela de informações do agente exibindo ID, pontuação atual, idade, renda e grau na rede; (b) Janela de histórico de eventos do agente; (c) Janela de edição de cenário permitindo configuração de intervenções.	9
4	Boxplot comparativo do índice de Gini entre topologias (experimento de comparação).	18
5	Série temporal da média de pontuação: Barabási–Albert.	19
6	Série temporal da média de pontuação: Watts–Strogatz.	19
7	Mapa de calor composto mostrando métricas por topologia e cenário. A figura apresenta heatmaps para duas topologias (Barabási–Albert e Watts–Strogatz) e quatro cenários (“baseline”, “bad_influences”, “cluster_shock”, “structural_rewrite”) avaliando quatro indicadores: Gini, Moran’s I, P90–P10 e tempo até estabilidade (“stability_time”). Cada painel foi normalizado por coluna para facilitar a comparação entre cenários.	21

Lista de Tabelas

1	Parâmetros principais e valores padrão utilizados nas simulações.	16
2	Comparação estatística (Mann-Whitney U) entre as métricas das topologias Barabási–Albert e Watts–Strogatz, com tamanho de efeito (Cohen’s d). Valores em negrito indicam diferença estatisticamente significativa ($p < 0.05$).	20
3	Resumo do sistema de pontuação de crédito social de Rongcheng por nível de classificação. Dados complementares disponíveis em [34].	28

Lista de Abreviaturas e Siglas

Sigla	Significado
ABM	Agent-Based Modeling (Modelagem Baseada em Agentes)
API	Application Programming Interface
BA	Barabási–Albert (modelo de rede de escala livre)
CSV	Comma-Separated Values
ER	Erdős–Rényi (modelo de grafo aleatório)
IC95	Intervalo de Confiança de 95%
JIT	Just-In-Time (Compilação em Tempo de Execução)
JSON	JavaScript Object Notation
LLM	Large Language Model (Modelo de Linguagem de Grande Escala)
RMB	Renminbi (moeda chinesa)
SCS	Sistema de Crédito Social
UI	User Interface (Interface do Usuário)
UFMG	Universidade Federal de Minas Gerais
WS	Watts–Strogatz (modelo de rede de mundo pequeno)

1 Introdução

Os Sistemas de Crédito Social (SCS) representam um dos fenômenos mais ambiciosos de engenharia social e governança algorítmica do século XXI, combinando tecnologia de big data, vigilância digital e controle comportamental em escala populacional. Esta seção contextualiza o problema de pesquisa, apresenta os objetivos e hipóteses, e delineia a estrutura do documento.

1.1 Contextualização

Diferente da concepção ocidental focada no histórico financeiro, o projeto chinês busca quantificar a “confiabilidade” (*chengxin*) integral do cidadão, integrando dados administrativos, criminais, contratuais e comportamentais (ver Tabela 3). O sistema não opera como uma entidade monolítica nacional, mas como um ecossistema fragmentado de iniciativas locais (“pilotos”) e plataformas comerciais que alimentam mecanismos centralizados de recompensa e sanção, notadamente as “Listas Vermelhas” (comportamentos exemplares) e “Listas Negras” (ofensores graves), aplicadas tanto a empresas quanto a indivíduos [1]. Uma representação pública dessa visibilidade está ilustrada em Figura 1, que mostra um mural em Rongcheng expondo pontuações de moradores como forma de incentivo e alerta público. Pesquisas indicam que, apesar das preocupações ocidentais sobre vigilância [2], há níveis surpreendentemente altos de aprovação do sistema entre cidadãos chineses [3], refletindo diferenças culturais e contextuais na percepção de privacidade e governança.

Teoricamente, o SCS pode ser interpretado sob a ótica de Gilles Deleuze em seu *Post-Scriptum sobre as Sociedades de Controle* [4]. Deleuze argumenta que as sociedades modernas transicionaram da “disciplina” (confinamento em moldes fixos: escola, fábrica, prisão) para o “controle” (modulação contínua e comunicação instantânea). Neste paradigma, o indivíduo torna-se um “divíduo” — uma massa de dados, senhas e pontuações que variam continuamente. O SCS é a materialização dessa teoria: a governança não ocorre pelo confinamento físico, mas pela regulação do acesso através de códigos digitais e pontuações flutuantes. Essa transformação intensifica questões sobre a construção do “eu transparente” sob perfilamento massivo de dados [5], onde a privacidade individual é renegociada no contexto de sistemas de pontuação comportamental abrangentes.

Um dos pilares deste sistema é o Mecanismo de Punição Conjunta, onde a infração em um domínio (por exemplo, não pagar uma multa de trânsito) gera consequências em outros domínios (restrição de viagens de avião ou acesso a crédito bancário). Entre as implementações municipais, Rongcheng, na província de Shandong, destaca-se por sua sofisticação, documentação detalhada e sistema de pontuação explícito [1, 6]. Este trabalho desenvolve um simulador computacional experimental que modela a dinâmica desse sistema. Ao invés de focar apenas nas regras estáticas, a simulação investiga os padrões emergentes (como desigualdade e segregação) que surgem quando essas regras de pontuação interagem com a estrutura complexa das redes sociais humanas.

O trabalho combina modelagem computacional e análise de políticas tecnológicas, focando na dinâmica de redes complexas aplicada a sistemas de crédito social. A contribuição principal é a construção de uma ferramenta de simulação que torna essas discussões exploráveis e testáveis.

1.2 Objetivos e Hipóteses

O **objetivo geral** é desenvolver e analisar um simulador computacional de um sistema de crédito social inspirado em Rongcheng, que permita explorar como diferentes estruturas de rede e regras de eventos influenciam a distribuição de pontuações na população. Especificamente, busca-se: (i) implementar uma arquitetura modular em Python que separe motor de simulação, interface gráfica e pipeline de experimentos; (ii) modelar eventos e parâmetros demográficos a partir de documentação pública de Rongcheng; e (iii) realizar experimentos sistemáticos comparando as topologias de rede

de mundo pequeno (Watts–Strogatz) e escala livre (Barabási–Albert) sob cenários de intervenção distintos.

A partir desses objetivos, formulam-se as seguintes **hipóteses de trabalho**: (H1) redes de escala livre (Barabási–Albert) tendem a produzir maior desigualdade de pontuações (índice de Gini mais elevado) do que redes de mundo pequeno (Watts–Strogatz) sob o mesmo regime de eventos; e (H2) choques direcionados a subconjuntos estruturais da rede (como hubs ou clusters) têm impacto mais duradouro e instável em topologias de escala livre do que em topologias de mundo pequeno.

Delimitação de Escopo: É fundamental ressaltar que este trabalho não propõe um modelo preditivo do comportamento humano real, dada a complexidade de variáveis psicológicas e culturais não modeladas. O simulador deve ser compreendido como um **laboratório computacional** para análise de dinâmicas emergentes e regimes sistêmicos, permitindo isolar variáveis estruturais e testar a lógica dos algoritmos em um ambiente controlado.

1.3 Estrutura do Trabalho

O restante deste trabalho está organizado da seguinte forma: a Seção 2 contextualiza o trabalho em relação à literatura existente e apresenta a fundamentação empírica baseada em Rongcheng. A Seção 3 descreve a arquitetura do sistema simulador, incluindo a interface gráfica e o pipeline de experimentos. A Seção 4 detalha o modelo de simulação, formalizando a dinâmica de eventos, influência social e cálculo de probabilidades. A Seção 5 expõe a metodologia experimental, definindo parâmetros, cenários e métricas de análise. A Seção 6 apresenta os resultados estatísticos obtidos nos experimentos. A Seção 7 discute as implicações teóricas e práticas dos achados, relacionando-os a casos reais e políticas de governança digital. A Seção 8 sintetiza as contribuições do trabalho e propõe direções para pesquisas futuras. Apêndices complementares detalham otimizações computacionais e o sistema de gravação de dados.

2 Trabalhos Relacionados

A modelagem computacional de sistemas como o Sistema de Crédito Social (SCS) chinês situa-se na interseção de três áreas: os estudos empíricos sobre governança digital, a teoria dos jogos aplicada à conformidade e a computação social baseada em agentes. Esta seção revisa como a literatura aborda esses temas e identifica a contribuição do presente trabalho: a simulação de dinâmicas de pontuação sob a influência da estrutura da rede social.

2.1 Estudos Empíricos e Sociológicos do SCS

A literatura acadêmica sobre o SCS oferece a base descritiva para a modelagem. Trabalhos como os de Creemers [7] e Kostka [3] detalham os mecanismos de recompensa, punição e a percepção pública do sistema, fornecendo dados para a parametrização do simulador (como visto na Tabela 3). Enquanto Engelmann et al. [8] analisa as disparidades na aplicação de sanções, esses estudos são predominantemente qualitativos, carecendo de ferramentas computacionais que permitam prever a evolução temporal das pontuações em cenários contrafactuais (“o que aconteceria se...”).

2.2 Modelagem via Teoria dos Jogos

Uma vertente quantitativa relevante aplica a Teoria dos Jogos para analisar a estabilidade de estratégias. Pesquisadores como Li e Liu [9] modelam o SCS como um jogo tripartite entre governo, empresas e cidadãos, buscando identificar pontos de equilíbrio onde a supervisão é eficaz. Essas abordagens, porém, frequentemente assumem populações homogêneas e comportamento estritamente racional. Este trabalho adota a Simulação Baseada em Agentes (ABM) [10], que permite modelar a

heterogeneidade individual (idade, renda) e comportamentos complexos, como a influência social involuntária — aspectos que dificilmente são capturados por modelos analíticos puros.

2.3 Sistemas de Reputação e Simulação Social

Na Ciência da Computação, modelos de confiança e reputação são estudados para garantir a segurança em sistemas distribuídos. Sabater e Sierra [11] classificam esses sistemas focando em como a reputação emerge da agregação de feedbacks locais em ambientes como comércio eletrônico e redes P2P. O presente trabalho estende esses conceitos para o domínio da **Simulação Social** (Social Simulation), utilizando agentes computacionais como “laboratórios virtuais” para testar hipóteses sociológicas.

Ao contrário dos sistemas de reputação clássicos, que buscam otimizar a confiança entre pares para transações, o simulador *Codificando Rongcheng* investiga a **estratificação social** resultante de um sistema de pontuação centralizado. A inovação consiste em combinar a lógica burocrática rígida do sistema de pontuação com a topologia orgânica das redes sociais, mostrando como a estrutura de conexões (quem conhece quem) afeta o destino dos agentes.

2.4 Parametrização Baseada em Rongcheng

O sistema implementado em Rongcheng opera através de uma lógica de pontuação base, embora a documentação oficial não especifique a pontuação inicial atribuída aos residentes [6]. Para fins de simulação, adotou-se uma pontuação inicial média $\mu_{\text{init}} = 750$ com variabilidade estocástica (distribuição normal $\mathcal{N}(750, 250)$), representando heterogeneidade inicial da população. Este valor médio situa-se abaixo do limite inferior do nível B (850 a 959), permitindo que os agentes transitem tanto para níveis superiores quanto inferiores durante a dinâmica estocástica. A partir desse baseline distribuído, as pontuações flutuam dinamicamente.

As pontuações são ajustadas através de uma coleta de dados que, para fins de modelagem na simulação, foram sintetizados em três categorias representativas [12]:

- **Setor Comercial:** Cumprimento de contratos, pagamento de empréstimos, produção de bens de qualidade.
- **Gestão Social:** Voluntariado, doação de sangue, serviços comunitários.
- **Cumprimento Legal:** Infrações de trânsito, adesão a políticas de planejamento familiar, respeito a normas éticas profissionais.

A pontuação final não é apenas um número, mas um determinante de classe social algorítmica. Cidadãos são classificados em níveis (de D a AAA), e essa classificação determina o acesso a bens públicos e oportunidades de mercado. Estudos apontam que, enquanto as sanções são claramente definidas e aplicadas de forma rigorosa, as recompensas tendem a ser mais vagas e menos tangíveis [8], criando uma assimetria entre punições e incentivos positivos. Modelar Rongcheng justifica-se pela sua **relevância social** — é um dos sistemas mais estudados pela literatura acadêmica [7, 13] — e pela **disponibilidade de dados** sobre a magnitude das punições e recompensas, permitindo calibrar os parâmetros da simulação de forma fundamentada.

3 Arquitetura do Sistema

3.1 Justificativa da Implementação Ad-Hoc

A escolha por desenvolver um motor de simulação proprietário, em detrimento da utilização de *frameworks* consolidados de Modelagem Baseada em Agentes (ABM), fundamenta-se em requisitos de desempenho e flexibilidade de interface que limitariam o uso de ferramentas de prateleira.

No cenário acadêmico, o **NetLogo** [14] é a ferramenta de referência para prototipagem rápida devido à sua linguagem acessível e visualização integrada. Entretanto, sua arquitetura fechada dificulta a integração com bibliotecas científicas avançadas do ecossistema Python (como SciPy e NumPy) e impõe restrições severas à personalização da interface gráfica (UI), impedindo a criação da experiência de “painel de controle governamental” moderna desejada para este projeto.

Frameworks em Python como o **Mesa** [15] oferecem a flexibilidade da linguagem, mas frequentemente enfrentam gargalos de desempenho em simulações com grande número de agentes ou lógicas de interação complexas, devido ao *overhead* da orientação a objetos pura do Python [16].

A decisão de implementar uma arquitetura *ad-hoc* permitiu:

1. **Otimização via JIT:** O controle sobre o laço de simulação possibilitou o uso do compilador *Just-In-Time* (Numba), convertendo funções críticas de Python para código de máquina otimizado. Isso viabilizou a execução de 15 anos de simulação (5.475 passos) em segundos, uma performance difícil de replicar em frameworks genéricos sem modificações profundas.
2. **Desacoplamento da Interface:** Ao contrário do NetLogo, onde lógica e visualização são entrelaçadas, a arquitetura proposta separa estritamente o modelo (backend) da interface (PySide6). Isso permitiu construir uma UI reativa profissional com gráficos em tempo real e controles complexos de cenários, sem impactar a velocidade do motor de simulação.
3. **Vetorização de Regras:** A implementação direta permitiu traduzir as equações de probabilidade do modelo de Rongcheng diretamente para operações vetorizadas sobre arrays NumPy, oferecendo maior eficiência computacional do que a abordagem iterativa agente-a-agente comum em frameworks padrão.

Essa abordagem “construída do zero” exigiu esforço de engenharia considerável, mas resultou em uma ferramenta que equilibra a profundidade analítica de bibliotecas científicas com a interatividade de softwares desktop nativos.

3.2 Visão Geral Conceitual

Com a justificativa técnica estabelecida, apresenta-se agora a arquitetura conceitual do sistema, seus componentes modulares e os princípios de design que orientaram a implementação.

O simulador implementa uma **Simulação Baseada em Agentes** (ABM) [17] combinada com **Monte Carlo** [18], onde cada cidadão é representado como um nó em uma rede social. A arquitetura modular é organizada em componentes principais:

- **Core:** Gerenciamento de configurações (JSON), constantes e utilitários.
- **Network:** Criação de topologias (NetworkX [19]), cálculo de influência e cache de vizinhos.
- **Events:** Processamento de eventos, aplicação de cooldowns e cálculo de probabilidades.
- **Simulation:** Motor de simulação, lógica de passos temporais com JIT (Numba), e gravação de métricas.
- **UI:** Interface gráfica (PySide6 [20]), visualizações (Matplotlib [21]), e controles interativos.
- **Experiments:** Framework para execução de experimentos batch, análise estatística e geração de relatórios.
- **Scenarios:** Definição e aplicação de intervenções experimentais.
- **Analysis:** Pós-processamento de dados e geração de visualizações.

Princípios de design adotados:

- **Separação Model-View:** Lógica de simulação independente da UI.
- **Configurabilidade:** Parâmetros e eventos em JSON, sem recompilação.
- **Reprodutibilidade:** Seeds explícitas em todos os componentes estocásticos.

Detalhes sobre as otimizações computacionais implementadas (compilação JIT, cache de estruturas, vetorização e análise de complexidade) estão disponíveis no Apêndice D.

3.3 Interface Gráfica

A interface gráfica, desenvolvida com PySide6, permite o controle interativo dos parâmetros de simulação (velocidade, execução passo a passo e seleção de cenários) e a visualização em tempo real da topologia da rede e das métricas estatísticas relevantes (ver Figura 2). O simulador suporta exportação/importação do estado completo da simulação, modo batch (execução sem interface para experimentos em larga escala) e cenários configuráveis via arquivos JSON, mantendo a separação entre a lógica de simulação e a visualização.

Para manter interface responsiva durante simulação, utilizou-se:

- **Blitting:** Redesenho parcial do canvas Matplotlib, atualizando apenas nós que mudaram de cor/tamanho, resultando em melhoria de desempenho.
- **Edge Color Cache:** Pré-computação de cores de arestas (destaque de vizinhos) armazenada e reutilizada, resultando em melhoria de desempenho.
- **Update Throttling:** Atualização visual a cada $N = 10$ dias ao invés de diariamente, reduzindo overhead de rendering.

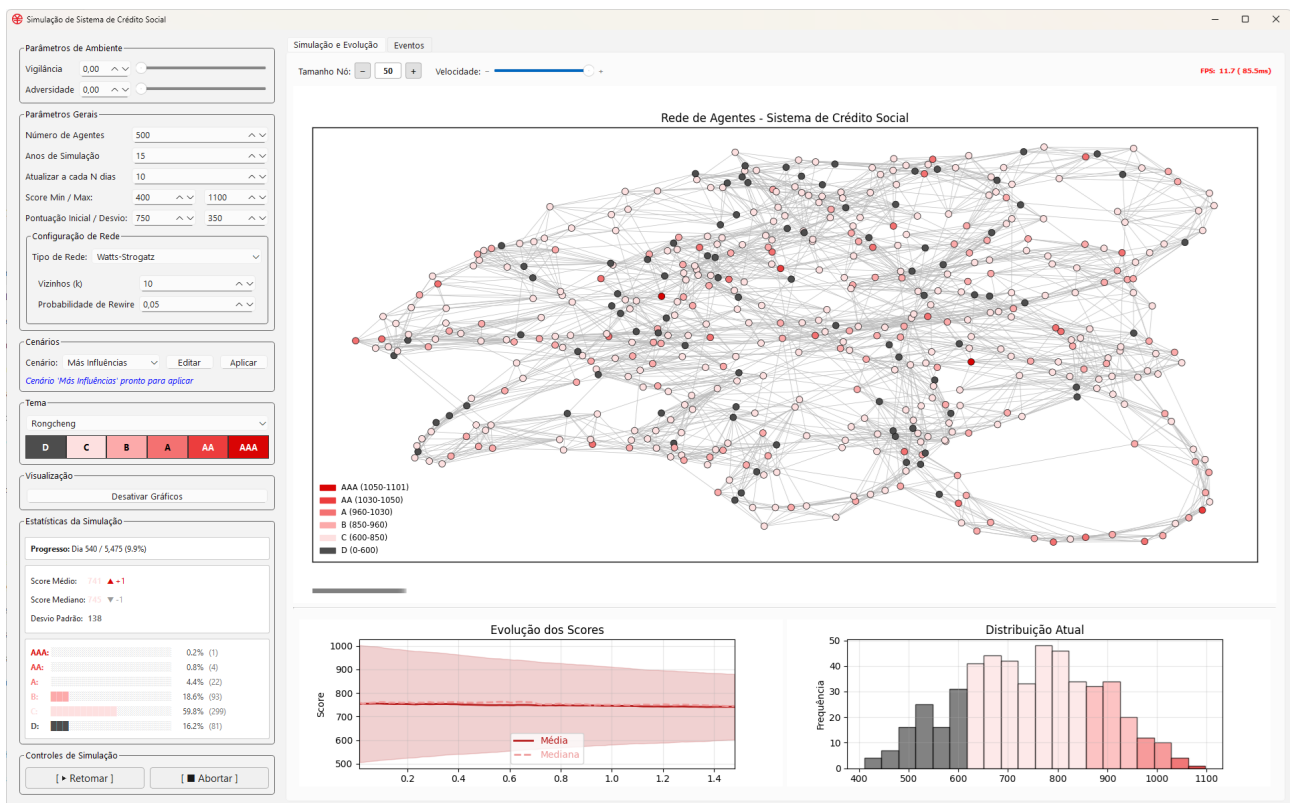


Figura 2: Interface gráfica do simulador. Painel esquerdo: controles de parâmetros e configuração. Centro: visualização da rede social. Painel inferior: estatísticas, gráficos de evolução temporal e distribuição de pontuações.

Capturas adicionais da interface estão apresentadas em Figura 3.



Figura 3: Capturas de tela da interface do simulador: (a) Janela de informações do agente exibindo ID, pontuação atual, idade, renda e grau na rede; (b) Janela de histórico de eventos do agente; (c) Janela de edição de cenário permitindo configuração de intervenções.

Para análise detalhada e experimentos em larga escala, o simulador implementa um sistema de gravação de dados em tempo real. Detalhes sobre os dados capturados e a arquitetura do sistema de gravação estão disponíveis no Apêndice C.

4 Modelo de Simulação

Com a arquitetura do sistema definida, esta seção formaliza o modelo de simulação que governa a dinâmica de pontuações dos agentes. A cada passo temporal (dia), eventos estocásticos podem ocorrer para cada agente, modificando sua pontuação. A probabilidade de cada evento é modulada por múltiplos fatores: influência social dos vizinhos, parâmetros globais do sistema, e características individuais (idade e renda).

4.1 Notação Matemática e Definições Preliminares

Antes de descrever os componentes do modelo, definem-se as convenções de notação utilizadas.

Conjuntos de Vizinhança: Para um agente i em um grafo $G = (A, E)$, define-se:

- $\mathcal{N}_1(i) = \{j : (i, j) \in E\}$ – vizinhança direta (1º grau): conjunto de todos os vizinhos diretos do agente i .
- $\mathcal{N}_2(i) = \{k : k \notin \mathcal{N}_1(i), \exists j \in \mathcal{N}_1(i) \text{ tal que } (j, k) \in E\}$ – vizinhança indireta (2º grau): conjunto de agentes através de dois passos em G a partir de i .

Notação Temporal: Utiliza-se $s_i(t)$ para denotar a pontuação do agente i no dia t , representando uma série temporal discreta. Quando o contexto deixa claro a referência ao estado em um dado instante, omite-se o argumento temporal e escreve-se simplesmente s_i .

Operador de Truncamento: Define-se o operador $[x]_a^b$ como

$$[x]_a^b = \min(\max(x, a), b),$$

que limita o valor x ao intervalo $[a, b]$. Este operador é equivalente à função `clamp` em linguagens de programação. Como caso especial, $[x]_0^1$ normaliza x para o intervalo $[0, 1]$.

4.2 Inicialização de Agentes e Demografia

A população de agentes é inicializada com distribuições demográficas realistas de **idade** e **renda**, fundamentadas em dados censitários e econômicos recentes da China. A distribuição etária segue faixas populacionais estimadas a partir dos dados do Sétimo Censo Nacional [22], ajustadas para focar na população economicamente ativa (excluindo menores de 14 anos): 14 a 17 anos (12%), 18 a

24 anos (16 %), 25 a 54 anos (45 %), 55 a 64 anos (14 %), e 65 a 90 anos (13 %). Cada agente recebe uma idade amostrada uniformemente dentro da faixa sorteada. A distribuição de **renda** é gerada através de uma distribuição log-normal com parâmetros configuráveis ($\mu = 10.0$, $\sigma = 1.0$), posteriormente normalizada no intervalo $[0, 1]$. Para refletir a estratificação socioeconômica real, os agentes são classificados em três classes (Baixa, Média, Alta). No modelo, as proporções populacionais resultantes (Baixa $\approx 53\%$, Média $\approx 37\%$, Alta $\approx 10\%$) foram derivadas de uma composição de indicadores econômicos do **Banco Mundial** e dados demográficos do **NBS** [23, 22], harmonizando diferentes critérios de estratificação de renda. Tais percentis determinam a probabilidade diferenciada de eventos conforme a classe social do agente.

Embora classes socioeconômicas (Baixa, Média, Alta) sejam descritas para simplificar a exposição demográfica, a implementação computacional utiliza renda normalizada contínua $I_i \in [0, 1]$. As classes são atribuídas apenas como rótulos descritivos (via percentis da distribuição), sendo a renda contínua a variável mecânica que afeta as probabilidades de eventos através do modificador de renda (Eq. do modificador de renda). Em termos práticos, isso significa que o sistema enxerga renda como um eixo contínuo para modular eventos, evitando cortes artificiais entre classes.

4.3 Topologias de Rede

O simulador implementa três topologias clássicas de redes complexas:

- **Watts–Strogatz (WS)** [24]: Modelo de mundo pequeno com alto coeficiente de agrupamento e caminho médio curto. Parâmetros: $k = 10$ vizinhos iniciais, $p = 0,05$ probabilidade de reconexão. Valores escolhidos por serem padrão na literatura para redes sociais realistas.
- **Barabási–Albert (BA)** [25]: Rede de escala livre com crescimento preferencial, produzindo distribuição de grau em lei de potência. Parâmetro: $m = 3$ arestas por novo nó, valor comum para redes sociais reais.
- **Erdős–Rényi (ER)** [26]: Grafo aleatório onde cada aresta existe com probabilidade $p = 0,03$, resultando em grau médio similar às outras topologias.

Os experimentos reportados neste trabalho focaram nas topologias Watts–Strogatz e Barabási–Albert, que são mais representativas de estruturas sociais reais: a primeira captura a propriedade de mundo pequeno (clustering alto, caminho curto) [24] e a segunda reproduz distribuição heterogênea de graus (lei de potência) [25], ambas observadas em redes sociais empíricas [27]. Embora a topologia Erdős–Rényi esteja implementada e disponível na interface gráfica para experimentação exploratória, ela não foi incluída nos lotes experimentais automatizados devido ao seu caráter puramente aleatório, que oferece menor relevância para modelagem de redes sociais reais.

4.4 Modelagem de Eventos

Embora a documentação oficial de Rongcheng catalogue 404 itens de informação para cálculo de crédito [6], a simulação implementada adota uma abordagem simplificada selecionando 30 eventos representativos (15 positivos e 15 negativos).¹ A seleção priorizou comportamentos generalizáveis e de ocorrência comum dentro das categorias principais — setor comercial, gestão social e cumprimento legal — capturando a dinâmica de incentivos e punições do sistema sem a complexidade computacional da lista exaustiva original. Cada evento modelado possui os seguintes atributos configuráveis:

¹ A extração e processamento dos 404 itens originais envolveu: (i) obtenção da tabela HTML traduzida por China Law Translate; (ii) parsing automatizado com Pandas; (iii) conversão para formato JSON estruturado; e (iv) seleção manual de 30 eventos representativos para reduzir complexidade computacional mantendo diversidade de categorias e magnitudes.

- **Mudança de pontos (Δs):** Entre -100 e 50 pontos, seguindo a magnitude dos valores documentados [6].
- **Probabilidade base diária (p_{base}):** Frequência de ocorrência ajustada via calibração.
- **Cooldown:** Período mínimo de latência entre ocorrências repetidas do mesmo evento para um agente específico, variando por tipo de evento (30 a 365 dias).²
- **Sensibilidade à influência (α_{inf}):** Fator de modulação pela pressão social (0,7 a 1,08).
- **Filtros etários:** Restrições de aplicação por faixa etária (idade mínima/máxima) com sensibilidade ajustável (α_{age}).
- **Viés de renda:** Modificador de probabilidade baseado na classe socioeconômica (β_{income}) com sensibilidade configurável (α_{income}).

Exemplos concretos da estrutura JSON dos eventos, ilustrando a diversidade de parâmetros e suas interpretações sociológicas, estão disponíveis no Apêndice B.

4.5 Calibração e Estabilização do Sistema

Para garantir que a dinâmica do sistema não fosse dominada por uma tendência de crescimento ou colapso das pontuações (deriva), implementou-se uma rotina de calibração iterativa automatizada. O objetivo foi encontrar um conjunto de multiplicadores para as probabilidades de eventos positivos (f_{pos}) e negativos (f_{neg}) que estabilizasse a média das pontuações ao longo do tempo. O algoritmo de calibração opera através de um método de **correção de erro baseado em sensibilidade**. O processo segue os seguintes passos (detalhado no Algoritmo 1):

1. **Medição:** Executa-se um lote de simulações (paralelizadas) nas topologias Watts-Strogatz e Barabási-Albert para medir a deriva média diária (\bar{d}) das pontuações.
2. **Cálculo de Sensibilidade:** Estima-se o impacto teórico agregado dos eventos positivos e negativos na pontuação global.
3. **Correção:** Compara-se a deriva observada com o alvo (zero). Calcula-se então, através de um sistema linear simplificado, o ajuste necessário nas probabilidades para anular essa diferença.
4. **Atualização:** Aplica-se os novos multiplicadores às probabilidades base dos eventos e repete-se o processo até que a deriva esteja dentro da margem de tolerância (ϵ).

Essa abordagem permite que o sistema encontre um ponto de equilíbrio robusto, minimizando a discrepância de comportamento entre diferentes estruturas de rede.

O Algoritmo 1 descreve o processo iterativo de calibração das probabilidades de eventos.

² Os períodos de cooldown não são especificados nos documentos oficiais de Rongcheng e foram definidos arbitrariamente para a simulação.

Algoritmo 1: Calibração Iterativa de Probabilidades**Data:** Eventos E , Tolerância ϵ **Result:** Probabilidades calibradas**while** $|deriva| > \epsilon$ **do** Executar simulações de teste e medir deriva média \bar{d} ; **if** $\bar{d} \approx 0$ **then** **break** (Sistema calibrado); **end**

Calcular fator de correção baseado no peso dos eventos;

if $\bar{d} > 0$ (*Pontuação subindo*) **then** Reduzir prob. de eventos positivos ($f_{pos} < 1$); Aumentar prob. de eventos negativos ($f_{neg} > 1$); **end** **else** Aumentar prob. de eventos positivos ($f_{pos} > 1$); Reduzir prob. de eventos negativos ($f_{neg} < 1$); **end** Atualizar probabilidades p_e de todos os eventos;**end**

Validação da calibração: Para aferir a convergência, executou-se o algoritmo de verificação de deriva em cada topologia principal (50 réplicas). Estatísticas agregadas:

- **Watts–Strogatz (n=50):** $\bar{d} = 0.00347$ pontos/dia; $\sigma = 0.00335$; min = -0.00287 , max = 0.01347 .
- **Barabási–Albert (n=50):** $\bar{d} = -0.00472$ pontos/dia; $\sigma = 0.01245$; min = -0.02415 , max = 0.03451 .

As médias residuais indicam pequenas derivações do objetivo nulo: a topologia Watts–Strogatz apresenta uma leve tendência positiva (cerca de $3,47 \times 10^{-3}$ pontos/dia) enquanto a Barabási–Albert apresenta tendência negativa (cerca de $-4,72 \times 10^{-3}$ pontos/dia). Tais magnitudes são reduzidas em relação à escala de pontuação usada ($s_{init} = 750$) e, considerando os desvios-padrão observados ($\sigma = 0,003.35$ para WS e $\sigma = 0,012.45$ para BA), são compatíveis com flutuações estocásticas inerentes ao processo de Monte Carlo e à variabilidade entre diferentes realizações (réplicas) da simulação. Em outras palavras, as pequenas derivas médias observadas situam-se dentro da faixa esperada de variação aleatória e não indicam viés sistemático. Portanto, a calibração é considerada adequada para estudos comparativos, embora ajustes finos possam ser aplicados para reduzir ainda mais a deriva média.

4.6 Cálculo de Probabilidades de Eventos

A probabilidade de ocorrência de cada evento é modulada por múltiplos fatores que capturam diferentes aspectos da dinâmica social e individual. Cada fator é representado por um modificador que ajusta a probabilidade base do evento. As subseções seguintes descrevem cada um desses modificadores em detalhe.

Modificador de Influência Social O código implementado utiliza um modelo aditivo baseado na diferença de pontuações entre o agente e sua vizinhança, considerando tanto vizinhos diretos quanto indiretos. O cálculo do modificador de influência M_{inf} segue três etapas sequenciais.

Primeiro, calcula-se a **pontuação média ponderada da vizinhança** ($\bar{s}_{viz}(i, t)$). O algoritmo considera tanto vizinhos diretos (1º grau) quanto indiretos (2º grau), ponderados pela capacidade de influência (c_j) de cada vizinho. Para um agente i , a vizinhança média ponderada no tempo t é dada por:

$$\bar{s}_{\text{viz}}(i, t) = \frac{\sum_{j \in \mathcal{N}_1(i)} s_j(t) \cdot c_j + \sum_{k \in \mathcal{N}_2(i)} s_k(t) \cdot (0.5 \cdot c_k)}{\sum_{j \in \mathcal{N}_1(i)} c_j + \sum_{k \in \mathcal{N}_2(i)} 0.5 \cdot c_k} \quad (1)$$

Onde $\mathcal{N}_1(i)$ são os vizinhos diretos (1º grau) do agente i e $\mathcal{N}_2(i)$ os vizinhos de segundo grau, ambos formalmente definidos na Subseção 4.1. Note que a influência dos vizinhos de segundo grau é atenuada por um fator de 0.5, refletindo um decaimento de influência com a distância topológica.

Em seguida, calcula-se um **fator base** a partir da diferença entre a média da vizinhança e a pontuação atual do agente $s_i(t)$, normalizada por um parâmetro de escala fixo (120 pontos):

$$\text{Base}(i, t) = 1 - \frac{\bar{s}_{\text{viz}}(i, t) - s_i(t)}{120} \quad (2)$$

O denominador fixo de 120 pontos baseia-se na estrutura de classes de Rongcheng, onde a transição da classe modal (B) para a classe superior (A) exige um ganho de 110 pontos ($\Delta_{B \rightarrow A} = 960 - 850$). O valor de 120 pontos incorpora uma **margem de segurança de cerca de 10 %** sobre o limiar teórico de transição, evitando saturação imediata da função de influência em casos de fronteira.

O fator base é posteriormente truncado ao intervalo $[0.1, 4.0]$: $[\text{Base}(i, t)]_{0.1}^{4.0}$, limitando a magnitude da influência social. Se a vizinhança tem pontuação maior que o agente ($\bar{s}_{\text{viz}}(i, t) > s_i(t)$), o fator base será menor que 1. Em consequência, o modificador de influência tende a ficar abaixo de 1, reduzindo a probabilidade de eventos negativos; o efeito contrário ocorre quando a vizinhança tem pontuação inferior. Por fim, aplica-se a **resistência individual** (r_i) do agente para determinar o quanto ele é suscetível a essa pressão social. O modificador final de influência social para o agente i no tempo t é obtido ajustando o desvio do fator Base em relação à neutralidade (1.0):

$$M_{\text{inf}}(i, t) = 1 + \underbrace{\left(\frac{1}{1 + r_i} \right)}_{\text{Suscetibilidade}} \cdot ([\text{Base}(i, t)]_{0.1}^{4.0} - 1) \quad (3)$$

Onde $[\text{Base}(i, t)]_{0.1}^{4.0}$ denota o truncamento do fator base ao intervalo $[0.1, 4.0]$. Este modificador é então aplicado à probabilidade do evento através da **sensibilidade específica do evento** (α_e):

$$p_{e,i}^{\text{influenciada}}(t) = p_{e,i}^{\text{base}}(t) \times \begin{cases} M_{\text{inf}}(i, t)^{\alpha_e} & \text{se evento negativo} \\ M_{\text{inf}}(i, t)^{-\alpha_e} & \text{se evento positivo} \end{cases} \quad (4)$$

Este modelo implica que agentes com alta resistência (r_i) mantêm M_{inf} próximo de 1.0, ignorando a pressão dos vizinhos, enquanto agentes com baixa resistência são fortemente puxados pela média local. O parâmetro α_e controla o quanto cada tipo de evento é sensível à influência social: valores altos ($\alpha_e \approx 1$) fazem a probabilidade variar fortemente com M_{inf} , enquanto valores baixos ($\alpha_e \approx 0$) tornam o evento praticamente imune à pressão social.

Modificador de Vigilância No código implementado, a vigilância ($\theta_{\text{vig}} \in [0, \infty)$) opera através de dois mecanismos simultâneos: dissuasão de crimes e aumento da eficiência de notificação. Primeiro, aplica-se a **dissuasão** para eventos negativos marcados como sensíveis à vigilância. O parâmetro atua reduzindo a probabilidade base em até 50%, representando o efeito inibidor da fiscalização:

$$M_{\text{vig},1}(i, t) = \begin{cases} [1 - 0.5 \cdot \theta_{\text{vig}}]_0^1 & \text{se } \Delta s_e < 0 \text{ e sensível} \\ 1 & \text{caso contrário} \end{cases} \quad (5)$$

Segundo, aplica-se um ajuste global de **notificação**, simulando que um sistema mais vigiado registra mais ocorrências. Este efeito aumenta a probabilidade de todos os eventos, mas favorece desproporcionalmente o registro de ações positivas (20% por unidade de vigilância) em relação às negativas (5%):

$$M_{\text{vig},2}(t) = \begin{cases} 1 + 0.05 \cdot \theta_{\text{vig}} & \text{se } \Delta s_e < 0 \\ 1 + 0.20 \cdot \theta_{\text{vig}} & \text{se } \Delta s_e > 0 \end{cases} \quad (6)$$

A probabilidade final é multiplicada por ambos os fatores: $P_{e,i}(t) \cdot M_{\text{vig},1}(i, t) \cdot M_{\text{vig},2}(t)$. Em termos práticos, isso significa que sistemas mais vigiados tendem a registrar menos infrações graves sensíveis à fiscalização, mas ao mesmo tempo tornam mais provável que tanto boas ações quanto pequenas infrações sejam efetivamente capturadas nos registros.

Modificador de Adversidade O modificador de adversidade ($M_{\text{adv}}(t)$) é independente da pontuação ou características individuais do agente, funcionando como um *stress test* global que amplifica uniformemente a probabilidade de eventos negativos. Este design reflete a hipótese de que crises exógenas (econômicas, sanitárias) afetam toda a população proporcionalmente. Se o evento é negativo, sua probabilidade é aumentada linearmente pelo parâmetro global de adversidade $\theta_{\text{adv}} \in [0, \infty)$ ponderado pela sensibilidade específica do evento α_{adv} :

$$M_{\text{adv}}(t) = \begin{cases} 1 + \theta_{\text{adv}} \cdot \alpha_{\text{adv}} & \text{se } \Delta s_e < 0 \\ 1 & \text{se } \Delta s_e \geq 0 \end{cases} \quad (7)$$

Note que $M_{\text{adv}}(t)$ não depende de nenhum atributo individual ($s_i(t)$, r_i , c_i , a_i , I_i), garantindo uniformidade do choque. Este modelo simula cenários de crise externa onde a frequência de incidentes negativos aumenta uniformemente para toda a população. Em outras palavras, M_{adv} funciona como um “botão de crise” global que intensifica eventos negativos para todos ao mesmo tempo, sem distinguir entre perfis de agentes.

Filtros Etários Eventos possuem faixas etárias relevantes definidas por $[\text{idade}_{\min}, \text{idade}_{\max}]$. Para agentes cuja idade a_i está dentro dessa faixa, $M_{\text{age}}(i) = 1$ (sem penalidade). Para agentes fora da faixa, o modificador depende do parâmetro de rigidez $\alpha_{\text{age}} \in [0, 1]$:

$$M_{\text{age}}(i) = \begin{cases} 1 & \text{se } \text{idade}_{\min} \leq a_i \leq \text{idade}_{\max} \\ [1 - \alpha_{\text{age}}]_0^1 & \text{se } a_i < \text{idade}_{\min} \text{ ou } a_i > \text{idade}_{\max} \end{cases} \quad (8)$$

O operador de truncamento $[x]_0^1 = \min(\max(x, 0), 1)$ garante que $M_{\text{age}}(i) \in [0, 1]$. Quando $\alpha_{\text{age}} = 0$, o filtro é completamente suavizado ($M_{\text{age}}(i) = 1$ sempre). Quando $\alpha_{\text{age}} = 1$, agentes fora da faixa etária têm evento completamente desabilitado ($M_{\text{age}}(i) = 0$).

Modificador de Renda O viés de renda $\beta_{\text{income}} \in [-1, 1]$ favorece ou desfavorece agentes de maior renda:

$$M_{\text{income}}(i) = \max(0, 1 + \beta_{\text{income}} \cdot (I_i - 0.5) \cdot \alpha_{\text{income}}) \quad (9)$$

Aqui $I_i \in [0, 1]$ é a renda normalizada contínua do agente i , α_{income} é a sensibilidade do evento ao viés de renda, e $\beta_{\text{income}} \in [-1, 1]$ controla a direção e magnitude do viés.

4.7 Integração dos Modificadores e Fluxo de Simulação

A probabilidade final de um evento e ocorrer para um agente i no dia t é calculada através da composição multiplicativa de todos os fatores aplicados à probabilidade base:

$$P_{e,i}(t) = p_{\text{base}} \cdot M_{\text{inf}}(i, t) \cdot M_{\text{vig}}(t) \cdot M_{\text{adv}}(t) \cdot M_{\text{age}}(i) \cdot M_{\text{income}}(i) \quad (10)$$

onde cada modificador $M \in [0, \infty)$ ajusta a probabilidade baseado em diferentes fatores. O Algoritmo 2 descreve o fluxo principal da simulação, mostrando como esses componentes se integram no processo estocástico diário.

Algoritmo 2: Fluxo principal da Simulação Estocástica Diária**Data:** Grafo G , Agentes A , Lista de Eventos E **Result:** Estados atualizados dos agentes após período Δt

```

foreach dia  $t \in [1, T]$  do
  foreach agente  $i \in A$  do
    Calcula Influência Social:  $M_{\text{inf}}(i, t)$  (Eq. 3);
    foreach evento  $e \in E$  do
      if evento  $e$  em cooldown para agente  $i$  then
        continue;
      end
      Calcular probabilidade final:  $P_{e,i}(t)$  (Eq. 10);
      Gerar número aleatório  $u \sim U[0, 1]$ ;
      if  $u < P_{e,i}(t)$  then
        Atualizar Pontuação:  $s_i(t+1) \leftarrow s_i(t) + \Delta s_e$ ;
        Registrar Evento:  $(e, i, t, P_{e,i}(t), \Delta s_e)$  em log;
        Definir Cooldown( $e, i, t$ );
      end
    end
  end
end

```

5 Metodologia Experimental

Esta seção descreve a metodologia experimental utilizada para investigar as hipóteses de pesquisa. São apresentados os dois modos de uso do simulador (exploratório e experimental), os parâmetros computacionais adotados, os cenários de intervenção implementados e as métricas de avaliação utilizadas.

5.1 Dois Modos de Uso: Exploratório e Experimental

O simulador foi projetado para suportar dois paradigmas distintos de utilização, adequados a diferentes necessidades de pesquisa e comunicação:

Modo Exploratório (Interface Gráfica Interativa):

- Pesquisador ou educador ajusta parâmetros em tempo real via sliders e campos de entrada na UI.
- Visualiza topologia da rede com código de cores de pontuação, gráficos em tempo real (séries temporais de média, distribuição de níveis).
- Pode pausar simulação em qualquer dia, inspecionar agentes individuais (histórico de eventos, vizinhos, fatores de influência), retroceder ou avançar.
- Permite “e se...?” exploração intuitiva: “e se aumentar vigilância para 5?”, “e se a rede for totalmente aleatória?”
- Ideal para: educação, comunicação pública, validação de hipóteses iniciais, descoberta de comportamentos emergentes.

Modo Experimental (Execução Batch Paralela):

- Executa N repetições (padrão: 100) com múltiplas configurações (topologias, cenários, parâmetros) em paralelo usando multiprocessing.

- Sem interface gráfica, otimizado para performance computacional.
- Gera outputs estruturados: CSVs com resumos agregados (média \pm IC95%), logs de eventos brutos, arquivos de configuração reproduzíveis.
- Aplica testes estatísticos rigorosos (Mann-Whitney U, Kruskal-Wallis) via scripts de análise pós-processamento.
- Ideal para: pesquisa rigorosa, publicação, validação de hipóteses estatísticas, sensibilidade paramétrica.

Integração Técnica: Ambos os modos utilizam o mesmo motor de simulação, garantindo que descobertas no modo exploratório podem ser rigorosamente validadas no modo experimental, e vice-versa.

5.2 Parâmetros Computacionais

Os parâmetros experimentais foram definidos considerando limitações computacionais e robustez estatística.

- **Número de agentes:** $N = 500$. Escolhido como compromisso entre realismo (redes sociais suficientemente grandes) e tempo computacional. Experimentos piloto mostraram que $N \geq 500$ produz resultados qualitativamente similares a $N = 1.000$, mas com metade do tempo de execução.
- **Anos simulados:** $T = 15$ anos (5.475 dias). Período suficiente para observar convergência de métricas (média, mediana, Gini) e dinâmicas de longo prazo, sem custo proibitivo.
- **Repetições:** $n = 100$ runs por cenário (baseline, bad_influences, cluster_shock, structural_rewrite), executadas em paralelo no modo experimental. Cada repetição usa seed independente para garantir variabilidade estocástica enquanto permite reprodutibilidade.

Os parâmetros principais estão listados na Tabela 1.

Parâmetro	Valor padrão	Descrição
N	500	Número de agentes na população.
T	15 anos (5.475 dias)	Duração da simulação (período temporal).
n_{baseline}	100	Repetições para os cenários principais (base, más influências, choque em cluster, reconfiguração estrutural).
s_{init}	750	Pontuação inicial atribuída a cada agente (com distribuição normal, $\mu = 750$, $\sigma = 250$).
Topologia WS	$k = 10$, $p = 0,05$	Parâmetros do modelo Watts–Strogatz (grau inicial e prob. de reconexão).
Topologia BA	$m = 3$	Parâmetro do modelo Barabási–Albert (arestas por nó novo).
Topologia ER	$p = 0,03$	Probabilidade de aresta no modelo Erdős–Rényi.
α (capacidade)	1,0	Expoente usado no cálculo da capacidade c_i .
γ (resistência)	0,6	Expoente usado no cálculo da resistência r_i .
c_{min}	0,3	Valor mínimo permitido para capacidade.
r_{min}	0,4	Valor mínimo permitido para resistência.
α_e	0,5 a 0,95	Faixa típica da sensibilidade de eventos à influência (varia por evento).
θ_{adv}	0,0	Parâmetro global de adversidade (aumenta prob. de eventos negativos)*.
θ_{vig}	0,0	Intensidade de vigilância*.

Tabela 1: Parâmetros principais e valores padrão utilizados nas simulações.

*Parâmetros com valor 0,0 estão desativados nas execuções reportadas.

Robustez estatística (valores reportados): a partir dos resumos agregados das 100 repetições do cenário base (baseline) obtiveram-se os seguintes valores para o índice de Gini ao final da simulação (média \pm CI95):

- Watts–Strogatz: $\overline{\text{Gini}} = 0,055.8 \pm 0,000.74$ ($n = 100$)

² Simulações executadas em computador pessoal com processador Intel i5-14600K e 32 GB de RAM.

- Barabási–Albert: $\overline{\text{Gini}} = 0,055.9 \pm 0,001.90$ ($n = 100$)

Esses intervalos foram calculados como intervalo de confiança de 95% para a média.

5.3 Cenários Implementados

Quatro cenários experimentais foram criados para investigar diferentes aspectos do sistema:

1. **Cenário Base (Baseline):** Cenário padrão sem intervenções externas. Serve como referência para comparações. Todos os 30 eventos ativos com probabilidades balanceadas.
2. **Cenário de Más Influências (Bad Influences):** Intervenção pontual no dia 1.095 (ano 3), onde os 5 % de agentes com maior número de conexões sofrem dedução de 200 pontos. Simula um choque assimétrico nos nós mais conectados da rede, testando resiliência do sistema.
3. **Choque em Cluster (Cluster Shock):** Choque espacial concentrado em 5 % dos nós formando um cluster conectado, com dedução de 200 pontos no dia 1.095. Investiga propagação de choques localizados através da estrutura de rede.
4. **Reconfiguração Estrutural (Structural Rewire):** Rewire estrutural aplicado no dia configurado (por padrão: reconexão de 25 % das arestas). Este cenário reconecta uma fração das arestas segundo o método configurado (por exemplo: random ou preference_to_high_degree) e permite analisar como alterações topológicas imediatas afetam desigualdade e estabilidade do sistema.

A escolha desses cenários permite análise de **robustez** (choques pontuais), **propagação espacial** (estrutura de rede), e **contribuição de componentes** (remoção ou alteração de elementos do sistema).

5.4 Métricas de Estabilidade e Convergência

Para avaliar a dinâmica do sistema, usa-se a **amplitude relativa** das pontuações médias (\bar{s}), verificando se a oscilação dos valores permanece dentro de uma margem de tolerância de $\tau = 1\%$ (0,01).

Para uma dada janela temporal W , define-se a Amplitude Relativa (A_{rel}) como:

$$A_{rel}(W) = \frac{\max(\bar{s}_W) - \min(\bar{s}_W)}{\text{média}(\bar{s}_W)} \quad (11)$$

A partir disso, calcula-se uma pontuação de qualidade $Q(W) \in [0, 1]$, que penaliza linearmente oscilações que se aproximam ou excedem a tolerância:

$$Q(W) = \max\left(0, 1 - \frac{A_{rel}(W)}{\tau}\right) \quad (12)$$

As métricas finais são definidas como:

- **Qualidade de Convergência (Q_{conv}):** Corresponde ao valor de Q calculado apenas sobre o último ano da simulação (W_{final}):

$$Q_{conv} = Q(W_{final}) \quad (13)$$

- **Índice de Estabilidade (I_{stab}):** Corresponde à média aritmética das pontuações $Q(W_k)$ calculadas em K janelas deslizantes distribuídas ao longo de todo o período simulado:

$$I_{stab} = \frac{1}{K} \sum_{k=1}^K Q(W_k) \quad (14)$$

Esta métrica captura a robustez histórica, penalizando cenários que sofreram instabilidades transitórias antes de convergirem.

6 Resultados e Análise

A presente seção apresenta os resultados obtidos através dos experimentos batch executados com 100 repetições por cenário. Primeiro, são analisados os impactos da estrutura topológica da rede sobre as métricas de desigualdade e estabilidade. Em seguida, são apresentados os testes estatísticos que quantificam as diferenças observadas entre as topologias Watts–Strogatz e Barabási–Albert nos quatro cenários experimentais.

6.1 Impacto da Estrutura da Rede

As figuras abaixo apresentam boxplots e séries temporais comparativas: Figura 4 (boxplot comparativo do índice de Gini), Figura 5 (série temporal da média — Barabási–Albert), Figura 6 (série temporal da média — Watts–Strogatz) e Figura 7 (mapa de calor composto).

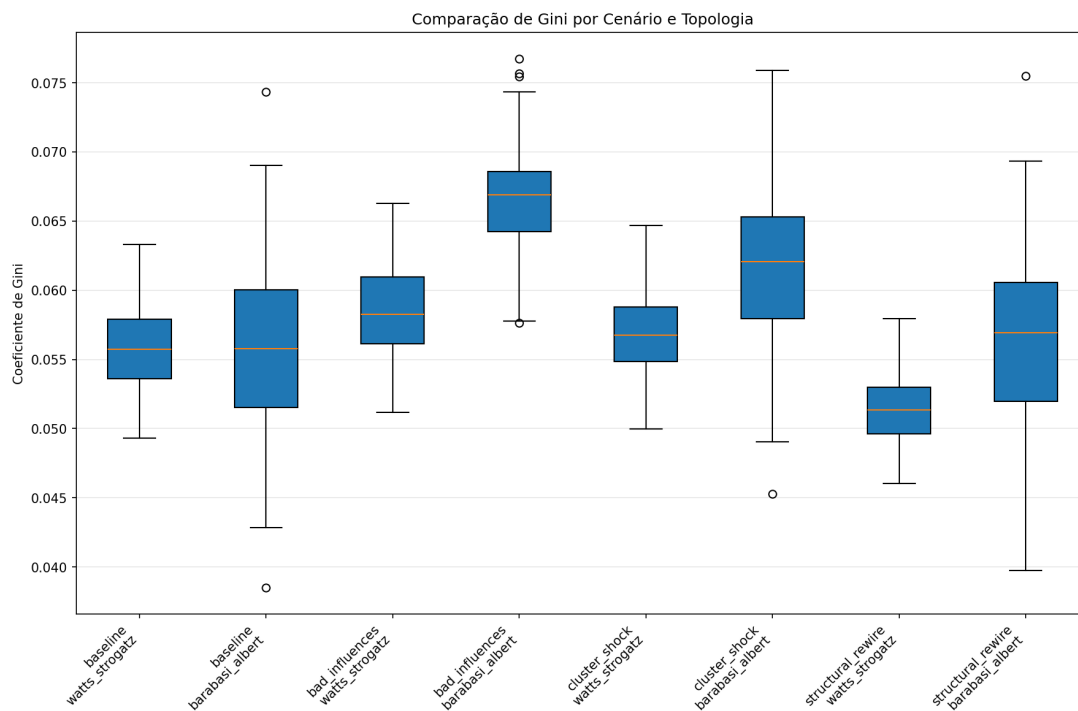


Figura 4: Boxplot comparativo do índice de Gini entre topologias (experimento de comparação).

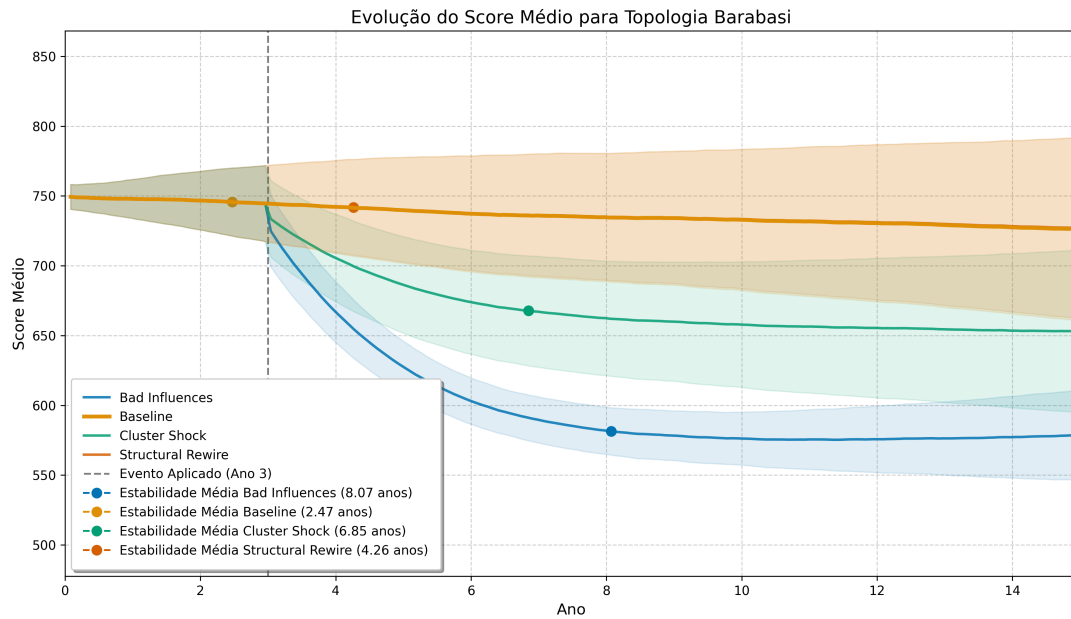


Figura 5: Série temporal da média de pontuação: Barabási–Albert.

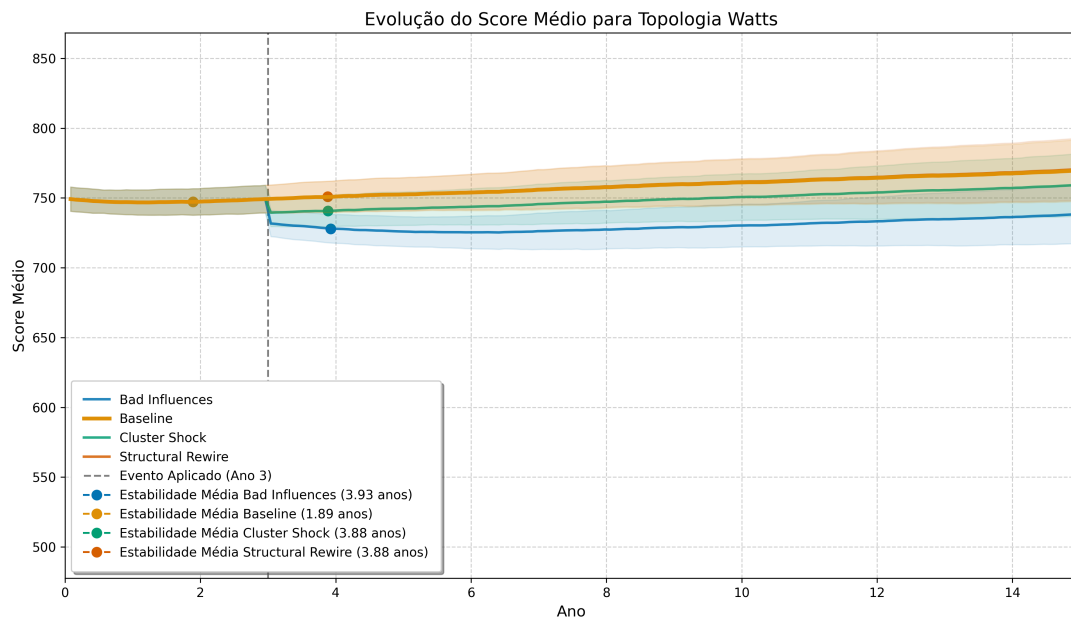


Figura 6: Série temporal da média de pontuação: Watts–Strogatz.

Observações qualitativas:

- **Barabási–Albert** apresenta desigualdade (Gini) equivalente a Watts–Strogatz no cenário base, mas maior desigualdade em todos os cenários de intervenção, revelando que hubs concentram e amplificam impactos de choques exógenos.
- **Watts–Strogatz** demonstra maior resiliência no cenário de Choque em Cluster, uma consequência direta de seu alto coeficiente de agrupamento que favorece a estabilidade local frente a perturbações.

6.2 Resultados Estatísticos

As diferenças observadas entre as topologias foram quantificadas através do teste não paramétrico de **Mann-Whitney U** [28], comparando os resultados das simulações entre as redes Barabási-Albert (BA) e Watts-Strogatz (WS) para cada cenário.

A análise revela três padrões estruturais que diferenciam as topologias: (1) redes BA não geram desigualdade inerente, mas **amplificam impactos** quando perturbadas — no cenário base, Gini é idêntico ($p = 0,897$), mas sob intervenções BA apresenta desigualdade sistematicamente maior ($p < 0,001$); (2) redes WS mantêm **autocorrelação espacial** elevada (Moran's I) devido ao clustering, exceto quando sua estrutura é destruída por recabeamento; (3) redes BA exibem **recuperação mais lenta** após choques, mesmo sem intervenções (baseline: 2,29 vs 1,89 anos), indicando volatilidade estrutural inerente às redes de escala livre.

A Tabela 2 apresenta os resultados completos, incluindo médias, p-valores e tamanhos de efeito (Cohen's d).

Tabela 2: Comparação estatística (Mann-Whitney U) entre as métricas das topologias Barabási-Albert e Watts-Strogatz, com tamanho de efeito (Cohen's d). Valores em negrito indicam diferença estatisticamente significativa ($p < 0.05$).

Cenário	Métrica	Média BA	Média WS	p-valor	Cohen's d
Baseline	Gini	0.0559	0.0558	0.897	0.0085
	Moran's I	0.0604	0.2014	<0.001	-2.9688***
	P90-P10	175.47	189.46	<0.001	-1.3527***
	Estabilidade (anos)	2.29	1.89	<0.001	0.5984**
	Índ. Estabilidade	0.327	0.501	<0.001	-2.0080***
	Amplitude Rel.	0.008	0.005	0.002	0.5650**
	Qual. Convergência	0.346	0.459	0.005	-0.4675*
Bad Influences	Gini	0.0665	0.0586	<0.001	2.1157***
	Moran's I	0.0460	0.2084	<0.001	-3.5771***
	P90-P10	168.56	190.43	<0.001	-2.3523***
	Estabilidade (anos)	7.91	3.90	<0.001	6.7325***
	Índ. Estabilidade	0.153	0.480	<0.001	-4.6128***
	Amplitude Rel.	0.009	0.006	<0.001	0.8423***
	Qual. Convergência	0.253	0.438	<0.001	-0.8285***
Cluster Shock	Gini	0.0616	0.0569	<0.001	0.9803***
	Moran's I	0.0601	0.2085	<0.001	-3.1180***
	P90-P10	173.67	190.85	<0.001	-1.6102***
	Estabilidade (anos)	6.58	3.87	<0.001	3.3442***
	Índ. Estabilidade	0.209	0.494	<0.001	-3.3855***
	Amplitude Rel.	0.008	0.006	0.0001	0.6825**
	Qual. Convergência	0.308	0.447	0.0002	-0.5952**
Structural Rewire	Gini	0.0567	0.0514	<0.001	1.1327***
	Moran's I	0.0670	0.0496	0.0003	0.5287**
	P90-P10	177.89	173.16	0.0018	0.4917*
	Estabilidade (anos)	4.14	3.87	<0.001	0.6220**
	Índ. Estabilidade	0.336	0.500	<0.001	-1.8483***
	Amplitude Rel.	0.007	0.005	0.0054	0.5327**
	Qual. Convergência	0.347	0.462	0.0049	-0.4774*

* pequeno ($0.2 \leq |d| < 0.5$); ** médio ($0.5 \leq |d| < 0.8$); *** grande ($|d| \geq 0.8$)

Os tamanhos de efeito confirmam a relevância prática dessas diferenças. Destaca-se o cenário *Bad Influences* com os maiores efeitos: Gini ($d = 2,12$) e tempo de estabilização ($d = 6,73$) evidenciam que penalizar hubs em redes BA produz cascatas de desigualdade e instabilidade muito superiores ao impacto difuso observado em WS. No cenário *Structural Rewire*, o recabeamento destrói vantagens estruturais de WS, homogeneizando parcialmente o comportamento entre topologias (Moran's I inverte: $d = 0,53$).

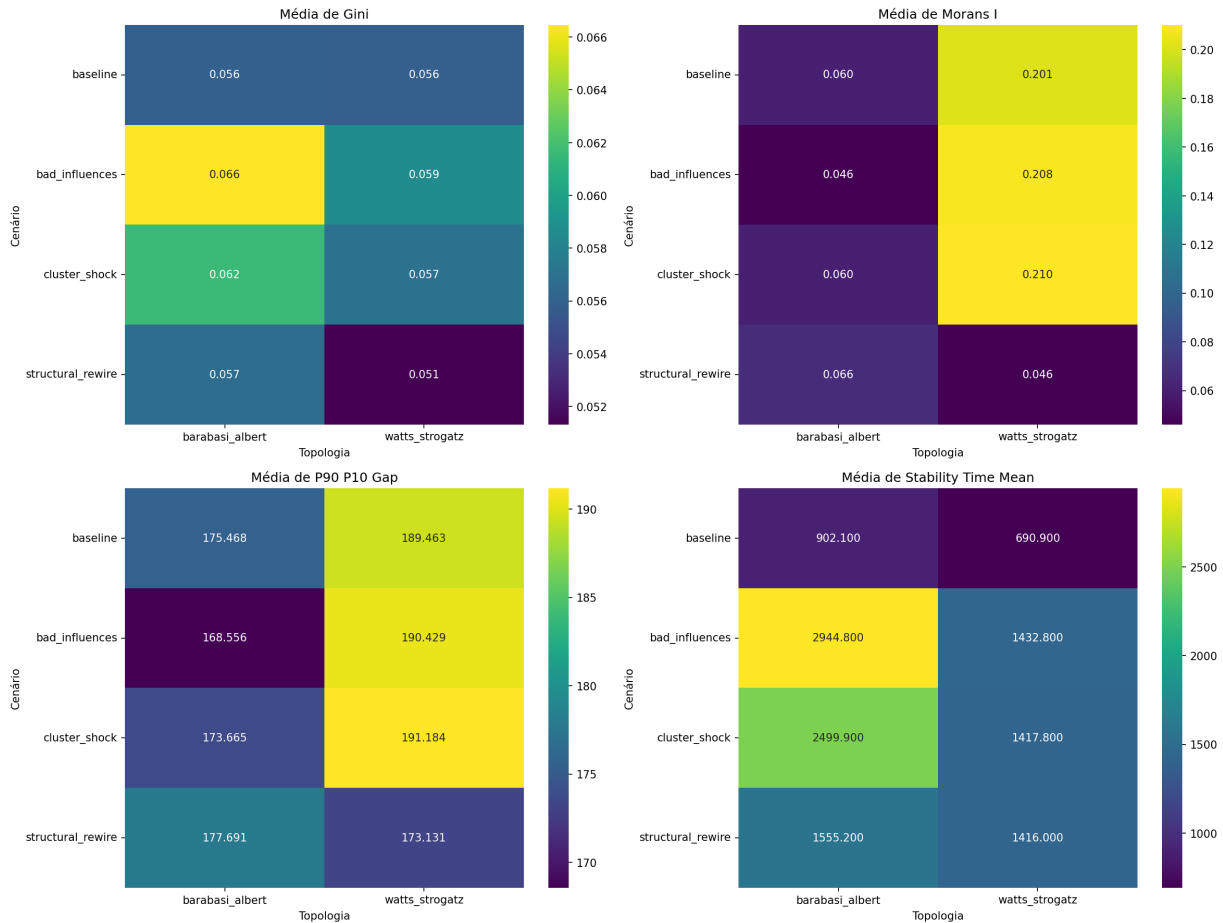


Figura 7: Mapa de calor composto mostrando métricas por topologia e cenário. A figura apresenta heatmaps para duas topologias (Barabási–Albert e Watts–Strogatz) e quatro cenários (“baseline”, “bad_influences”, “cluster_shock”, “structural_rewire”) avaliando quatro indicadores: Gini, Moran’s I, P90–P10 e tempo até estabilidade (“stability_time”). Cada painel foi normalizado por coluna para facilitar a comparação entre cenários.

6.3 Síntese dos Resultados

Em síntese, os experimentos demonstram um padrão estrutural claro:

- As redes de escala livre (Barabási–Albert) atuam como **amplificadores de choques**, convertendo perturbações locais em instabilidade sistêmica e aumento de desigualdade;
- O alto coeficiente de agrupamento das redes de mundo pequeno (Watts–Strogatz) confere **resiliência local**, amortecendo a propagação de penalidades;
- A topologia da rede, por si só, não dita o destino dos agentes, mas modula drasticamente a intensidade da resposta a intervenções governamentais.

7 Discussão

Os resultados obtidos através do simulador confirmam a hipótese de que a estrutura da rede influencia a dinâmica de pontuação e permitem traçar paralelos teóricos com a governança algorítmica contemporânea.

Ao conectar com casos reais, é importante frisar que o modelo não afirma que isso ocorre exatamente assim na realidade, mas sugere que ele pode revelar mecanismos plausíveis e gerar hipóteses que merecem investigação empírica.

7.1 Análise dos Cenários Experimentais

Para conectar os resultados experimentais com fenômenos reais de governança algorítmica, analisam-se três casos correspondentes aos cenários de intervenção: o escândalo dos bancos de Henan (choque em cluster), as políticas de gestão em grade (reconfiguração estrutural) e as campanhas de “limpeza da internet” (más influências).

O Caso de Henan e o Choque em Clusters O cenário `cluster_shock` (choque localizado) mostrou que redes com alto coeficiente de agrupamento (Watts–Strogatz) absorvem melhor punições direcionadas do que redes de escala livre. Reportagens investigativas sobre a implementação real do sistema em Rongcheng [29] revelam como a vigilância se integra ao cotidiano dos cidadãos. Esse cenário tem paralelo direto no escândalo dos bancos rurais de Henan em 2022, onde autoridades usaram o sistema de “Código de Saúde” para restringir a mobilidade de correntistas, alterando seus códigos para vermelho [30]. Assim como no evento real, o simulador **sugere que** a punição algorítmica pode ser instrumentalizada para conter subgrupos específicos conectados por interesses comuns (clusters), isolando a insatisfação antes que ela se propague para o restante da rede.

Gestão em Grade e a Reconfiguração Estrutural O cenário `structural_rewire` simula a política de “Gestão em Grade” (*Wanggehua Guanli*) e o controle informacional da *Great Firewall*. Na prática, a governança chinesa busca atomizar a sociedade, quebrando laços orgânicos fortes (sindicatos, grupos religiosos) e substituindo-os por conexões monitoradas e verticais [31]. A simulação mostra que a reconfiguração forçada das arestas (25 % das conexões reconectadas aleatoriamente) altera significativamente a estrutura da rede: em redes Watts-Strogatz, o Índice de Moran cai de 0,20 (baseline) para 0,05, indicando redução drástica da autocorrelação espacial e dissolução dos clusters naturais. Este resultado quantifica o efeito de políticas que visam fragmentar estruturas sociais coesas, um tema central nas discussões sobre governança digital autoritária.

Campanha Qinglang e Más Influências A intervenção `bad_influences`, onde agentes de alta pontuação são punidos, espelha as campanhas de “limpeza da internet” (*Qinglang*) conduzidas pela Administração do Ciberespaço da China [32]. Essas campanhas de limpeza da internet têm múltiplos objetivos, incluindo a proteção de menores online, o controle da cultura de fãs de celebridades e o combate a influenciadores que promovem conteúdo considerado “caótico”, ostentação de riqueza ou fraudes (um fenômeno global, similar à proliferação de golpes de apostas por influenciadores no Brasil). O modelo **indica que**, em redes Barabási–Albert (onde a influência é concentrada em hubs), atacar esses nós centrais gera um efeito desproporcional na rede inteira. Isso **oferece um substrato teórico para compreender** a estratégia estatal de usar punições exemplares em celebridades para modular o comportamento da massa.

7.2 Limitações do Modelo e Considerações sobre Transferibilidade Cultural

Diversas simplificações foram necessárias:

- **Rede estática:** Conexões sociais não evoluem ao longo do tempo. Em sociedades reais, laços sociais se formam e dissolvem dinamicamente, frequentemente baseados em homofilia (pessoas similares conectam-se mais). A ausência de dinâmica de rede pode subestimar efeitos de segregação e polarização.
- **Eventos independentes:** Não há encadeamento causal entre eventos (ex: inadimplência levando a perda de emprego). Sistemas reais frequentemente apresentam cascatas onde eventos negativos aumentam probabilidade de eventos negativos subsequentes, potencialmente levando a “espirais descendentes”.
- **Agentes passivos:** Ausência de estratégias ativas para maximizar pontuação. Cidadãos reais adaptam comportamento em resposta a incentivos, podendo “jogar o sistema” ou evitar

comportamentos de risco quando pontuação está baixa.

- **Mobilidade social:** Classes de renda fixas, sem mobilidade econômica. Pontuações de crédito social podem afetar oportunidades econômicas, criando feedback loop não capturado no modelo.
- **Modelo determinístico de influência:** A Equação 3 assume que influência é função determinística da média de pontuações dos vizinhos. Na realidade, influência social é processo mais complexo envolvendo confiança, autoridade percebida, contexto da relação e fatores psicológicos não quantificáveis.
- **Ausência de contexto temporal:** Eventos ocorrem de forma estocástica sem sazonalidade ou tendências históricas. Sistemas reais sofrem influência de ciclos econômicos, mudanças políticas e eventos históricos que afetam distribuição e frequência de eventos.
- **Escala populacional:** Foram simulados 500 agentes enquanto Rongcheng tem 740.000 habitantes. Embora análise de escalabilidade sugira comportamento qualitativo similar, efeitos emergentes em escala nacional podem diferir.

Considerações sobre Transferibilidade Cultural O sistema simulado é baseado em Rongcheng (contexto chinês), incorporando pressupostos culturais implícitos:

Coletivismo vs. Individualismo: A forte influência social modelada (Equação 3) reflete valores coletivistas prevalentes na sociedade chinesa, onde a conformidade social é valorizada. Em culturas individualistas (EUA, Europa Ocidental), a resistência à influência social (r_i) poderia ser sistematicamente maior, reduzindo efeitos de rede. Formas de vigilância e perfilamento comportamental também emergem em contextos ocidentais através do que Zuboff denomina “capitalismo de vigilância” [33], onde empresas privadas operam lógicas similares de quantificação e previsão comportamental, embora com diferentes mecanismos de governança.

Confiança institucional: O modelo assume aceitação do sistema de pontuação. Em sociedades com baixa confiança em instituições governamentais, poderiam surgir comportamentos de resistência ativa não capturados no modelo (subversão, boycotts).

Escolha de eventos e pontuações: Os 30 eventos implementados refletem prioridades documentadas em Rongcheng (conformidade legal, contribuição social, honestidade comercial). Diferentes culturas poderiam valorizar comportamentos distintos (ex: voluntarismo em comunidades religiosas, participação política em democracias liberais).

7.3 Nota sobre Uso de Ferramentas de IA

Ao longo do desenvolvimento deste trabalho, foram utilizados modelos de linguagem de grande porte (LLMs) como ferramenta auxiliar de programação. Especificamente, modelos das famílias GPT (OpenAI) e Claude (Anthropic), disponíveis via GitHub Copilot, auxiliaram em melhorias de responsividade da interface gráfica (layout, atualização de gráficos e uso de *blitting*), ajustes no processo de calibração das probabilidades de eventos e otimizações de desempenho (vetorização com NumPy, uso de Numba JIT e simplificação de estruturas de dados). As sugestões geradas foram avaliadas criticamente antes de sua incorporação ao projeto.

8 Conclusão

Este trabalho apresentou um simulador experimental de sistemas de crédito social inspirado em Rongcheng, implementando modelagem estocástica com influência social, eventos configuráveis baseados em legislação real, e análise sistemática através de experimentos computacionais.

8.1 Principais Contribuições

As principais contribuições deste trabalho são:

1. **Metodológica:** Framework de simulação que articula parametrização baseada em dados legislativos (Rongcheng), dinâmica de influência social em redes complexas e análise sistemática de diferentes topologias estruturais (Watts–Strogatz e Barabási–Albert).
2. **Empírica:** Demonstração de que a estrutura da rede social atua como um fator determinante na evolução da desigualdade (Gini) e na resiliência do sistema. Os resultados indicam que topologias de escala livre (Barabási–Albert) tendem a apresentar maior concentração de pontuações e desigualdade em comparação a redes de mundo pequeno (Watts–Strogatz), além de exibirem vulnerabilidades distintas a choques direcionados em *hubs*. Observou-se também que a presença de influência social atua consistentemente como um amplificador das disparidades iniciais em relação a modelos nulos.
3. **Prática:** Ferramenta open-source interativa com interface gráfica (PySide6) e framework de experimentos batch, permitindo pesquisadores e educadores explorarem hipóteses sobre governança algorítmica sem necessidade de implementação completa do zero. Otimizações computacionais (Numba JIT, caching e vetorização) reduzem o custo computacional, permitindo a execução prática de simulações com centenas de agentes por períodos que podem abranger anos.
4. **Teórica:** Evidência empírica de que sistemas de pontuação comportamental com influência social exibem dinâmicas de vantagem cumulativa, onde estrutura de rede determina quem se beneficia ou sofre desproporcionalmente. Conexão entre literatura de redes complexas, ABM e estudos de governança digital.
5. **Debate Público:** Demonstração acessível de como regras algorítmicas aparentemente neutras podem gerar resultados desiguais dependendo de estrutura social subjacente, contribuindo para debate informado sobre ética em IA e vigilância algorítmica.
6. **Caráter Lúdico e Exploratório:** O simulador transcende o uso estritamente acadêmico ao oferecer um ambiente de “sandbox” digital. A interface interativa permite que usuários não especializados explorem intuitivamente cenários “e se...” (ex: “e se a sociedade for muito vigilante?”, “e se houver uma crise econômica?”), fomentando o engajamento com o tema através da experimentação direta e da observação visual dos fenômenos emergentes.

8.2 Validação das Hipóteses

Os resultados obtidos permitem validar as hipóteses formuladas:

- **H1 (Estrutura e Desigualdade): Confirmada sob intervenção.** Embora o cenário base não apresente diferença significativa ($p = 0.897$), as redes de escala livre (Barabási–Albert) geraram índices de Gini sistematicamente maiores em todos os cenários de estresse ($p < 0.001$). Isso confirma que a topologia BA atua como um amplificador latente de desigualdades.
- **H2 (Resiliência a Choques): Confirmada integralmente.** A análise temporal demonstrou que choques direcionados (cenários *bad_influences* e *cluster_shock*) causam impactos mais profundos e recuperações mais lentas em redes BA do que em redes WS, validando a instabilidade estrutural inerente à dependência de *hubs*.

8.3 Trabalhos Futuros

1. **Redes dinâmicas:** Implementar formação e dissolução de arestas baseada em homofilia de pontuação, permitindo que agentes com scores similares formem novas conexões e que laços entre agentes muito distantes sejam enfraquecidos ou rompidos. Incorporar entrada e saída de agentes (nascimento/morte ou migração), tornando a população não estacionária e permitindo estudar a transmissão de normas entre gerações.
2. **Aprendizado por reforço:** Desenvolver agentes estratégicos capazes de adaptar seu comportamento ao longo do tempo para otimizar sua pontuação, seja por meio de aprendizado individual

ou observação de vizinhos bem-sucedidos, criando dinâmicas de jogo e potencial manipulação do sistema.

3. **Cascatas de eventos:** Modelar dependências causais entre eventos (ex: dívida → inadimplência → registro negativo), capturando trajetórias de degradação ou melhoria que refletem mais fielmente as interdependências temporais observadas em dados reais.
4. **Validação empírica:** Comparar distribuições simuladas com dados reais de sistemas de crédito social (se disponibilizados), utilizando técnicas de calibração e validação para ajustar parâmetros e avaliar a capacidade preditiva do modelo.
5. **Intervenções políticas:** Expandir o repertório de cenários de intervenção para simular efeitos de políticas públicas (ex: programas de perdão de dívidas, campanhas educacionais, subsídios para grupos vulneráveis), permitindo análise prospectiva de impacto antes da implementação real.

8.4 Considerações Finais

Este trabalho materializou conceitos abstratos de governança algorítmica em uma ferramenta computacional funcional que opera sobre dados reais extraídos da legislação de Rongcheng. A **arquitetura modular** desenvolvida em Python articula motor de simulação otimizado (Numba JIT), processador de eventos parametrizado por JSON, pipeline de experimentação automatizada e interface gráfica interativa (PySide6) — permitindo tanto análise rigorosa quanto exploração intuitiva de cenários.

As otimizações implementadas (vetorização NumPy, caching de estruturas, blitting para renderização) viabilizam a execução prática de experimentos com centenas de agentes por períodos prolongados, reduzindo tempos de simulação de horas para minutos e tornando viável a exploração iterativa de hipóteses com robustez estatística.

Entretanto, é fundamental reconhecer que **sistemas sociais reais excedem radicalmente qualquer modelo computacional**. Relações humanas, cultura, contextos históricos e sutilezas psicológicas envolvem milhões de variáveis interdependentes que não podem ser plenamente capturadas em equações ou grafos. O simulador não pretende “replicar” Rongcheng ou prever comportamentos sociais com precisão — ele oferece um **ambiente controlado para isolar mecanismos específicos** (estrutura de rede, distribuição de eventos, influência social) e observar suas interações em condições simplificadas.

Em última análise, a principal contribuição deste trabalho é oferecer um laboratório computacional robusto para analisar como regras algorítmicas e a estrutura da rede social interagem na produção de desigualdade e estabilidade sistêmica, demonstrando que a arquitetura das conexões humanas é tão determinante para o destino social quanto o próprio código da lei.

Referências

- [1] Katja Drinhausen e Vincent Brussee. *China's Social Credit System in 2021: From fragmentation towards integration*. Mercator Institute for China Studies (MERICS), 2021. URL: <https://merics.org/en/report/chinas-social-credit-system-2021-fragmentation-towards-integration>.
- [2] Louise Matsakis. "How the West Got China's Social Credit System Wrong". Em: *WIRED* (2019). URL: <https://www.wired.com/story/china-social-credit-score-system/>.
- [3] Genia Kostka. "China's social credit systems and public opinion: Explaining high levels of approval". Em: *New Media & Society* 21.7 (2019), pp. 1565–1593. DOI: 10.1177/1461444819826402.
- [4] Gilles Deleuze. "Postscript on the Societies of Control". Em: *October* 59 (1992), pp. 3–7.
- [5] Yongxi Chen e Anne S. Y. Cheung. "The Transparent Self Under Big Data Profiling: Privacy and Chinese Legislation on the Social Credit System". Em: *The Journal of Comparative Law* 12.2 (2019), pp. 356–378.
- [6] *Rongcheng Municipal Personal Credit Appraisal Standards*. China Law Translate. URL: <https://www.chinalawtranslate.com/en/rongcheng-municipal-personal-credit-appraisal-standards/> (acesso em 29/06/2025).
- [7] Rogier Creemers. "China's Social Credit System: An Evolving Practice of Control". Em: *SSRN Electronic Journal* (2018). DOI: 10.2139/ssrn.3175792.
- [8] Severin Engelmann et al. "Clear Sanctions, Vague Rewards: How China's Social Credit System Currently Defines "Good" and "Bad" Behavior". Em: *Proceedings of the Conference on Fairness, Accountability, and Transparency* (2019), pp. 69–78. DOI: 10.1145/3287560.3287585.
- [9] Ye Li e Yeping Liu. "Evolutionary Game Analysis of the Supervision of the Social Credit System". Em: *Frontiers in Psychology* 13 (2022), p. 954132.
- [10] Charles M Macal e Michael J North. "Tutorial on agent-based modelling and simulation". Em: *Journal of simulation* 4.3 (2010), pp. 151–162.
- [11] Jordi Sabater e Carles Sierra. "Review on computational trust and reputation models". Em: *Artificial intelligence review* 24.1 (2005), pp. 33–60.
- [12] *Rongcheng Personal Credit: Reward and Punishment Measures*. China Law Translate. URL: <https://www.chinalawtranslate.com/en/rongcheng-personal-credit-reward-and-punishment-measures/> (acesso em 29/06/2025).
- [13] Xin Dai. "Toward a Reputation State: The Social Credit System Project of China". Em: *SSRN Electronic Journal* (2020). DOI: 10.2139/ssrn.3193577.
- [14] Uri Wilensky. *NetLogo*. Evanston, IL: Center for Connected Learning e Computer-Based Modeling, Northwestern University, 1999. URL: <http://ccl.northwestern.edu/netlogo/>.
- [15] David Masad e Jackie Kazil. "Mesa: An agent-based modeling framework in Python". Em: *Proceedings of the 14th python in science conference*. Vol. 3. Citeseer. 2015, pp. 53–60.
- [16] Steven F Railsback e Volker Grimm. *Agent-based and individual-based modeling: a practical introduction*. 2ª ed. Princeton university press, 2019.
- [17] Uri Wilensky e William Rand. *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. MIT Press, 2015.
- [18] Dirk P. Kroese, Thomas Taimre e Zdravko I. Botev. *Handbook of Monte Carlo Methods*. Wiley, 2013. ISBN: 9781118014967.

- [19] Aric A. Hagberg, Daniel A. Schult e Pieter J. Swart. “Exploring Network Structure, Dynamics, and Function using NetworkX”. Em: *Proceedings of the 7th Python in Science Conference (SciPy 2008)* (2008), pp. 11–15. URL: <https://networkx.org/> (acesso em 29/06/2025).
- [20] The Qt Company. *PySide6 Documentation*. <https://doc.qt.io/qtforpython/>. Acesso em: 2025-06-29. 2023. (Acesso em 29/06/2025).
- [21] J. D. Hunter. *Matplotlib: A 2D Graphics Environment*. 2007. DOI: 10.1109/MCSE.2007.55.
- [22] National Bureau of Statistics of China. *Main Data of the Seventh National Population Census*. Acessado em: 2025-11-21. National Bureau of Statistics of China, 2021. URL: http://www.stats.gov.cn/english/PressRelease/202105/t20210510_1816980.html.
- [23] World Bank. *China Economic Update, December 2022: Navigating Uncertainty*. Washington, DC: The World Bank, 2022. URL: <https://www.worldbank.org/en/country/china/publication/china-economic-update-december-2022>.
- [24] Duncan J. Watts e Steven H. Strogatz. “Collective dynamics of ‘small-world’ networks”. Em: *Nature* 393.6684 (1998), pp. 440–442. DOI: 10.1038/30918.
- [25] Albert-László Barabási e Réka Albert. “Emergence of Scaling in Random Networks”. Em: *Science* 286.5439 (1999), pp. 509–512. DOI: 10.1126/science.286.5439.509.
- [26] Paul Erdős e Alfréd Rényi. “On Random Graphs”. Em: *Publicationes Mathematicae Debrecen* 6 (1959), pp. 290–297.
- [27] M. E. J. Newman. “The Structure and Function of Complex Networks”. Em: *SIAM Review* 45.2 (2003), pp. 167–256. DOI: 10.1137/S003614450342480.
- [28] Henry B. Mann e Donald R. Whitney. “On a Test of Whether One of Two Random Variables is Stochastically Larger than the Other”. Em: *Annals of Mathematical Statistics* 18.1 (1947), pp. 50–60. DOI: 10.1214/aoms/1177730491.
- [29] BBC. “Life Inside China’s Social Credit Laboratory”. Em: *Foreign Policy* (2018). URL: <https://www.foreignpolicy.com/2018/04/03/life-inside-chinas-social-credit-laboratory/> (acesso em 29/06/2025).
- [30] Reuters. “China bank protest stopped by health codes turning red, depositors say”. Em: *Reuters World News* (2022). Reportagem Investigativa. URL: <https://www.reuters.com/world/china/china-bank-protest-stopped-by-health-codes-turning-red-depositors-say-2022-06-14/>.
- [31] Fulong Wu. “The re-emergence of grid management in China”. Em: *Urban Studies* 57.12 (2020), pp. 2403–2422.
- [32] Cyberspace Administration of China. *Notice on Further Strengthening the Management of Chaos in Fandoms (‘Qinglang’ Campaign)*. Documento Oficial. 2021.
- [33] Shoshana Zuboff. *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. Public Affairs, 2019.
- [34] *Getting Rongcheng Right*. China Law Translate. URL: <https://www.chinalawtranslate.com/en/getting-rongcheng-right/> (acesso em 29/06/2025).

A Sistema de Pontuação de Rongcheng

Nível	Intervalo	Consequências/Recompensas
AAA – Modelo de Credibilidade	≥ 1.050	<ul style="list-style-type: none"> • 200 RMB para seguro de aposentadoria; • 100 RMB para seguro médico; • Subsídio de 5 % para cuidados hospitalares residenciais; • Subsídio de 100 RMB para transporte (se < 64) ou isenção da taxa (se ≥ 65); • Check-up gratuito (se < 60); • Empréstimo sem juros de até 100.000 RMB para projetos de baixo lucro; • Benefício de 3 % na taxa de empréstimos elegíveis; • Outras.
AA – Crédito Excepcional	1.030 a 1.049	Mesmas consequências do nível AAA.
A – Crédito (A ⁻ , A, A ⁺)	960 a 1.029	<ul style="list-style-type: none"> • Inclusão em lista-modelo para publicidade; • Prioridade em matrícula escolar e assistência social; • Preferência em emprego, com subsídios e pontuação adicional na qualificação; • Recomendações para filiação partidária, premiações, avaliações, contratações e titulações; • Apoio prioritário em políticas governamentais; • Redução ou dispensa de inspeções de fiscalização; • Pontos extras em licitações e inscrições em projetos científicos; • Procedimentos simplificados em serviços municipais; • Treinamentos gratuitos e preferência em promoções; • Outras medidas.
B – Relativamente Crédito	850 a 959	<ul style="list-style-type: none"> • Orientações e materiais para aprimorar o crédito; • Possibilidade de acessar benefícios de nível A com melhora.
C – Alerta de Crédito	600 a 849	<ul style="list-style-type: none"> • Fiscalização intensificada; • Divulgação restrita; • Suspensão ou redução temporária de benefícios, subsídios e políticas; • Cancelamento de qualificações para premiações; • Inabilitação para licitações ou concursos; • Ação disciplinar conjunta; • Outras medidas.
D – Não Crédito	≤ 599	<ul style="list-style-type: none"> • Inspeções rigorosas de monitoramento; • Cancelamento de apoios e subsídios; • Inclusão em lista negra (divulgação pública); • Revogação de títulos honoríficos e proibição em premiações; • Inabilitação para concursos públicos; • Suspensão ou cancelamento de qualificações profissionais; • Restrição de empréstimos; • Ação disciplinar conjunta severa; • Outras medidas.

Tabela 3: Resumo do sistema de pontuação de crédito social de Rongcheng por nível de classificação. Dados complementares disponíveis em [34].

B Exemplos de Estrutura JSON de Eventos

Para ilustrar a estrutura de dados utilizada na modelagem de eventos, apresentam-se dois exemplos representativos extraídos do arquivo de configuração do simulador: um evento de recompensa por comportamento cívico e um evento de punição moderada.

Exemplo 1: Evento Positivo — Denúncia de Segurança Alimentar

```
{
  "id": "addition_information_social_mores_reported_food_
        or_production_safety_issues",
  "description": "Reportou problemas de segurança alimentar...",
  "points_change": 5.0,
  "base_prob_daily": 0.00109006,
  "cooldown_days": 30,
  "influence_factor_sensitivity": 1.0,
  "min_age": 14,
  "max_age": 90,
  "age_sensitivity": 0.2,
  "income_bias": 0.0,
  "income_sensitivity": 0.0,
  "category_tags": ["addition", "addition_information", "minor", "social_mores"]
}
```

Este evento de recompensa leve (+5 pontos) incentiva vigilância comunitária sobre segurança pública: influência social neutra (1.0), indicando ausência de amplificação ou resistência à pressão dos pares; ausência de viés de renda (0.0), permitindo que qualquer cidadão possa reportar irregularidades independente de classe socioeconômica; probabilidade base ajustada por calibração automática; idade mínima de 14 anos, permitindo participação de adolescentes em atos de cidadania; e cooldown de 30 dias para evitar acumulação artificial de pontos.

Exemplo 2: Evento Negativo — Atraso em Pagamento de Impostos

```
{
  "id": "social_management_sector_tax_related_matters_
        having_a_record_of_tax_arrears",
  "description": "Ter registro de atraso no pagamento de impostos",
  "points_change": -20.0,
  "base_prob_daily": 0.00066167,
  "cooldown_days": 30,
  "influence_factor_sensitivity": 0.85,
  "min_age": 18,
  "max_age": 90,
  "age_sensitivity": 1.0,
  "income_bias": -0.4,
  "income_sensitivity": 0.8,
  "category_tags": ["minor", "deduction", "social_management_sector", "tax_related_matters"]
}
```

Este evento de punição moderada (−20 pontos) reflete inadimplência fiscal: sensibilidade à influência social abaixo da neutralidade (0.85), indicando resistência parcial à pressão social; viés de renda negativo (−0.4) com alta sensibilidade (0.8), modelando maior vulnerabilidade de indivíduos de menor renda a atrasos fiscais; probabilidade base ajustada por calibração automática; idade mínima de 18 anos (maioridade legal para responsabilidade tributária); e cooldown de 30 dias.

C Sistema de Gravação de Dados em Tempo Real

Para viabilizar visualizações em tempo real e análise pós-processamento de experimentos batch, implementou-se um sistema de gravação modular que captura estado da simulação em intervalos configuráveis. Este sistema (*SimulationRecorder*) é desacoplado da lógica de simulação, permitindo diferentes backends de armazenamento.

Dados Capturados por Amostra Temporal

- **Distribuição de Pontuações:** Vetor completo de scores de todos os agentes em cada amostra, permitindo cálculo em tempo real de estatísticas (média, mediana, desvio padrão, índice de Gini, percentis P90-P10).
- **Eventos Registrados:** Para cada evento aplicado, registra-se: timestamp (dia da simulação), ID do agente afetado, ID do evento, probabilidade calculada (com modificadores aplicados), pontos de recompensa/penalidade aplicados. Este log permite rastreamento posterior de cascatas de eventos e análise de causalidade.
- **Distribuição de Níveis de Crédito:** Contagem de agentes em cada nível (D, C, B, A, AA, AAA) em cada amostra temporal. Essencial para análise de mobilidade social vertical e mudança de classe ao longo da simulação.
- **Métricas de Rede:** Cálculo de índices estruturais por amostra: Moran's I (autocorrelação espacial de pontuações), grau médio, clustering coefficient local. Permite avaliar como mudanças topológicas impactam dinâmica.
- **Fatores de Influência Social:** Médias de capacidade (c_i) e resistência (r_i) da população, permitindo análise de como heterogeneidade individual afeta dinâmica coletiva.

D Otimizações Computacionais

O simulador passou por múltiplas rodadas de otimização para viabilizar experimentos em larga escala. As principais técnicas empregadas foram:

Compilação JIT com Numba

A função crítica, implementada via compilação JIT (Numba) e responsável por calcular probabilidades e aplicar eventos para todos os agentes, foi anotada com `@jit(cache=True, fastmath=True)`. Esta otimização resultou na eliminação de overhead do interpretador Python nos loops mais internos.

Cache de Estruturas de Rede

Três níveis de cache foram implementados:

1. **Neighbor Cache:** Pré-computa listas de vizinhos para evitar queries repetidas ao grafo *NetworkX*.
2. **Capacity/Resistance Cache:** Armazena valores de c_i e r_i calculados na inicialização.
3. **Event Arrays Cache:** Mantém arrays NumPy com propriedades de eventos (pontos, probabilidades, sensibilidades) e invalida apenas quando eventos mudam.

Vetorização com NumPy

Substituição de loops Python por operações vetorizadas:

- Cálculo de scores em batch: uso de arrays NumPy pré-alocados para processamento em lote.

- Matriz de cooldowns: matriz pré-alocada para controle de cooldowns entre agentes e eventos.
- Números aleatórios pré-alocados: geração de aleatórios em arrays NumPy para reduzir overhead de alocação.

Análise de Complexidade

A eficiência do simulador é governada pelo laço temporal central gerenciado pelo motor de simulação. Define-se N como o número de agentes, T o número de passos temporais (dias), E o número de tipos de eventos ($E = 30$) e $\langle k \rangle_{\text{ext}}$ o grau médio da vizinhança estendida (1° e 2° graus). A complexidade temporal do algoritmo é dada por:

$$\mathcal{O}(T \cdot N \cdot (E + \langle k \rangle_{\text{ext}})) \quad (15)$$

A decomposição dos termos reflete as principais operações por agente a cada dia:

- **Processamento de Eventos ($\mathcal{O}(E)$):** A função de probabilidade, otimizada com Numba, avalia os 30 eventos possíveis em tempo constante amortizado através de operações aritméticas vetorizadas, sem alocação dinâmica de memória.
- **Cálculo de Influência ($\mathcal{O}(\langle k \rangle_{\text{ext}})$):** O gargalo potencial de percorrer o grafo para buscar vizinhos de 2° grau foi eliminado através de uma estratégia de *caching*. A lista de vizinhos é pré-computada na inicialização. Assim, o custo computacional diário reduz-se à iteração sobre essa lista para o cálculo da média ponderada, que em redes de mundo pequeno (Watts-Strogatz) permanece pequena e controlada.

A complexidade espacial é dominada pelo armazenamento da topologia e do cache de vizinhança:

$$\mathcal{O}(N + M + N \cdot \langle k \rangle_{\text{ext}}) \quad (16)$$

Onde M é o número de arestas. A estrutura `neighbor_cache` armazena explicitamente os vizinhos de até 2° grau para cada nó. Este *trade-off* consome memória linear adicional em relação ao tamanho da rede estendida, mas é fundamental para viabilizar a execução de 5.475 passos temporais em tempo hábil, eliminando a necessidade de travessia recursiva do grafo ($\mathcal{O}(\langle k \rangle^2)$) a cada iteração do loop principal.