UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

PAULO HENRIQUE CARDOSO DE SOUZA

Seleção adaptativa de proxies com amostragem de Thompson e métodos Bayesianos

GOIÂNIA
2025

UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO (TECA) PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TESES

E DISSERTAÇÕES NA BIBLIOTECA DIGITAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio da Biblioteca Digital de Teses e Dissertações (BDTD/UFG), regulamentada pela Resolução CEPEC nº 832/2007, sem ressarcimento dos direitos autorais, de acordo com a Lei 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo das Teses e Dissertações disponibilizado na BDTD/UFG é de responsabilidade exclusiva do autor. Ao encaminhar o produto final, o autor(a) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do material bibliográfico

[X] Dissertação [ ] Tese [ ] Outro*:_____

*No caso de mestrado/doutorado profissional, indique o formato do Trabalho de Conclusão de Curso, permitido no documento de área, correspondente ao programa de pós-graduação, orientado pela legislação vigente da CAPES.

Exemplos: Estudo de caso ou Revisão sistemática ou outros formatos.

2. Nome completo do autor

PAULO HENRIQUE CARDOSO DE SOUZA

3. Título do trabalho

"Seleção Adaptativa de Proxies com Amostragem de Thompson e Métodos Bayesianos"

4. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador)

Concorda com a liberação total do documento [X] SIM [ ] NÃO[1]

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante:
a) consulta ao(à) autor(a) e ao(à) orientador(a);
b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo da tese ou dissertação.
O documento não será disponibilizado durante o período de embargo.
Casos de embargo:
- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro;
- Publicação da dissertação/tese em livro.

Obs. Este termo deverá ser assinado no SEI pelo orientador e pelo autor.

Documento assinado eletronicamente por **Leonardo Da Cunha Brito**, **Professor do Magistério Superior**, em 26/09/2025, às 10:55, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

Documento assinado eletronicamente por **Paulo Henrique Cardoso De Souza**, **Discente**, em 29/09/2025, às 09:34, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5680464** e o código CRC **9FA096C9**.

Referência: Processo nº 23070.046810/2025-41 SEI nº 5680464

PAULO HENRIQUE CARDOSO DE SOUZA

Seleção adaptativa de proxies com amostragem de Thompson e métodos Bayesianos

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação, da Escola de Engenharia Elétrica, Mecânica e de Computação da Universidade Federal de Goiás, como requisito parcial para obtenção do Título de Mestre em Engenharia Elétrica e de Computação.

Área de Concentração: Engenharia de Computação.

Orientador: Prof. Dr. Leonardo da Cunha Brito

Coorientador: Prof. Dr. Thyago Carvalho Marques

GOIÂNIA
2025

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Souza, Paulo Henrique Cardoso de
Seleção Adaptativa de Proxies com Amostragem de Thompson e Métodos Bayesianos [manuscrito] / Paulo Henrique Cardoso de Souza. - 2025.
CIX, 109 f.

Orientador: Prof. Dr. Leonardo da Cunha Brito; co-orientador Dr. Thyago Carvalho Marques.
Dissertação (Mestrado) - Universidade Federal de Goiás, Escola de Engenharia Elétrica, Mecânica e de Computação (EMC), Programa de Pós-Graduação em Engenharia Elétrica e de Computação, Goiânia, 2025.
Bibliografia.
Inclui siglas, símbolos.

1. seleção de proxies. 2. estratégias bayesianas. 3. distribuição beta. 4. captura de dados automatizada. 5. aprendizado probabilístico. I. Brito, Leonardo da Cunha , orient. II. Título.

CDU 004

UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

ATA DE DEFESA DE DISSERTAÇÃO

Ata nº **27** da sessão de Defesa de Dissertação de **PAULO HENRIQUE CARDOSO DE SOUZA**, que confere o título de Mestre em **Engenharia Elétrica e de Computação**, na área de concentração em **Engenharia de Computação.**

Aos **dezoito dias do mês de setembro de dois e vinte e cinco**, a partir das **09:00horas**, realizou-se a sessão pública de Defesa de Dissertação intitulada **"Seleção Adaptativa de Proxies com Amostragem de Thompson e Métodos Bayesianos"**. Os trabalhos foram instalados pelo Orientador, Professor Doutor **Leonardo da Cunha Brito - (EMC/UFG)**, Professor Doutor **Thyago Carvalho Marques - (EMC/UFG)** Coorientador, com a participação dos demais membros da Banca Examinadora: Doutor **Jhonata Emerick Ramos - (DataRisk)** membro titular externo e Professor Doutor **Sandrerley Ramos Pires** - (**EMC/UFG)** Membro Titular Externo**: cujas participações ocorreram através de videoconferência** através do link: https://meet.google.com/kor-pmnn-nqj?hs=224. Durante a arguição os membros da banca **não fizeram** sugestão de alteração do título do trabalho. A Banca Examinadora reuniu-se em sessão secreta a fim de concluir o julgamento da Dissertação, tendo sido o candidato **aprovado** pelos seus membros. Proclamados os resultados pelo Professor Doutor **Leonardo da Cunha Brito**, Presidente da Banca Examinadora, foram encerrados os trabalhos e, para constar, lavrou-se a presente ata que é assinada pelos Membros da Banca Examinadora, aos **dezoito dias do mês de setembro de dois e vinte e cinco**.

TÍTULO SUGERIDO PELA BANCA

Documento assinado eletronicamente por **Sandrerley Ramos Pires**, **Professor do Magistério Superior**, em 18/09/2025, às 10:24, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

Documento assinado eletronicamente por **Leonardo Da Cunha Brito**, **Professor do Magistério Superior**, em 18/09/2025, às 10:26, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

Documento assinado eletronicamente por **Paulo Henrique Cardoso De Souza**, **Discente**, em 19/09/2025, às 09:54, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

Documento assinado eletronicamente por **Jhonata Emerick Ramos**, **Usuário Externo**, em 25/09/2025, às 11:12, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Thyago Carvalho Marques**, **Professor do Magistério Superior**, em 25/09/2025, às 14:36, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5650835** e o código CRC **5E10904D**.

Referência: Processo nº 23070.046810/2025-41 SEI nº 5650835

Este trabalho é dedicado a todos que compartilharam essa jornada comigo.

Agradecimentos

Agradeço ao Prof. Dr. Leonardo da Cunha Brito e ao Prof. Dr. Thyago Carvalho Marques, meu orientador e coorientador, pela valiosa orientação acadêmica, direcionamento técnico e metodológico que tornaram possível o desenvolvimento desta dissertação. Suas contribuições, questionamentos e sugestões foram fundamentais para a qualidade e rigor científico deste trabalho.

À minha esposa, pelo apoio constante, compreensão durante os períodos de dedicação à pesquisa e por proporcionar o ambiente necessário para que eu pudesse me concentrar plenamente neste projeto.

Aos meus pais, pela educação que me proporcionaram, pelos valores transmitidos e pelo constante incentivo à busca do conhecimento ao longo de toda minha trajetória acadêmica.

"Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)

Resumo

Este trabalho investigou estratégias de seleção de *proxies* para sistemas de captura de dados automatizada, comparando abordagens tradicionais com estratégias Bayesianas adaptativas. O objetivo principal foi avaliar a eficiência operacional, estabilidade e capacidade adaptativa de diferentes algoritmos de seleção em ambientes controlados e reais. A metodologia compreendeu simulações controladas em quatro cenários distintos (proxies intermitentes, bloqueados, permanentemente falhos e heterogêneos) e validação experimental em ambiente operacional real com 10 robôs diferentes realizando captura de dados públicos de diferentes domínios durante uma semana, processando 549.114 requisições. Foram avaliadas sete estratégias: quatro Bayesianas (Beta, Gamma, Normal, Chi-Quadrado), uma determinística (Exponential Backoff) e duas básicas (Round Robin e Aleatória). Os resultados das simulações demonstraram superioridade consistente das estratégias Bayesianas, com a distribuição Beta alcançando taxas de sucesso superiores a 99% em cenários críticos e mantendo liderança em ambiente real com 76,00% de taxa média. A análise de estabilidade revelou coeficientes de variação significativamente menores para estratégias Bayesianas (0,191-0,334) comparadas às básicas (0,498-0,668). A análise temporal evidenciou que estratégias Bayesianas desperdiçaram 2,5 vezes menos recursos que abordagens básicas, demonstrando eficiência operacional superior. A distribuição Beta destacou-se pela capacidade excepcional de diferenciação entre recursos e adaptação temporal, conforme evidenciado pela análise detalhada das distribuições de probabilidade. Além das aplicações diretas em captura de dados, as técnicas desenvolvidas apresentam potencial significativo para sistemas anti-scraping adaptativos, onde a capacidade de identificação de padrões comportamentais suspeitos e adaptação dinâmica a técnicas de evasão podem aprimorar mecanismos de proteção contra atividades automatizadas que violem políticas de uso de recursos web. Conclui-se que estratégias Bayesianas, especialmente a distribuição Beta, oferecem vantagens operacionais significativas para sistemas de captura de dados e potencial transformador para desenvolvimento de contramedidas adaptativas em proteção web.

Palavras-chave: seleção de proxies. estratégias bayesianas. distribuição beta. captura de dados automatizada. aprendizado probabilístico.

Abstract

This study investigated strategies for proxy selection in automated data capture systems, comparing traditional approaches with adaptive Bayesian strategies. The main goal was to evaluate the operational efficiency, stability, and adaptive capacity of different selection algorithms in both controlled and real environments. The methodology involved controlled simulations in four distinct scenarios (intermittent proxies, blocked proxies, permanently failed proxies, and heterogeneous proxies) and experimental validation in a real operational environment with 10 different robots performing public data capture from various domains over one week, processing 549,114 requests. Seven strategies were evaluated: four Bayesian (Beta, Gamma, Normal, Chi-Square), one deterministic (Exponential Backoff), and two basic (Round Robin and Random). The simulation results demonstrated the consistent superiority of Bayesian strategies, with the Beta distribution achieving success rates above 99% in critical scenarios and maintaining leadership in the real environment with an average rate of 76.00%. The stability analysis revealed significantly lower coefficients of variation for Bayesian strategies (0.191–0.334) compared to the basic ones (0.498–0.668). The temporal analysis showed that Bayesian strategies wasted 2.5 times fewer resources than basic approaches, demonstrating superior operational efficiency. The Beta distribution stood out for its exceptional ability to differentiate between resources and adapt over time, as evidenced by the detailed analysis of probability distributions. Beyond direct applications in data capture, the developed techniques show significant potential for adaptive anti-scraping systems, where the ability to identify suspicious behavioral patterns and dynamically adapt to evasion techniques can enhance protection mechanisms against automated activities that violate web resource usage policies. It is concluded that Bayesian strategies, particularly the Beta distribution, provide significant operational advantages for data capture systems and transformative potential for the development of adaptive countermeasures in web protection.

Keywords: proxy selection. bayesian strategies. beta distribution. automated data capture. probabilistic learning.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

API	*Application Programming Interface*
ASN	*Autonomous System Number*
CDN	*Content Delivery Network*
CGNAT	*Carrier-Grade Network Address Translation*
CSV	*Comma-Separated Values*
HTML	*HyperText Markup Language*
HTTP	*HyperText Transfer Protocol*
IP	*Internet Protocol*
ISP	*Internet Service Provider*
JSON	*JavaScript Object Notation*
PDF	*Portable Document Format*
TLS	*Transport Layer Security*
ToS	*Terms of Service*
URL	*Uniform Resource Locator*
WAF	*Web Application Firewall*

Lista de símbolos

α	*Parâmetro de forma (contexto dependente: sucessos na Beta, forma na Gamma)*
β	*Parâmetro de escala (contexto dependente: falhas na Beta, escala na Gamma)*
γ	*Fator de peso geográfico no modelo de pontuação contextual*
δ	*Fator de peso temporal no modelo de pontuação contextual*
θ	*Parâmetro de qualidade do proxy na inferência bayesiana*
λ	*Taxa de decaimento temporal exponencial*
μ	*Parâmetro de média da distribuição Normal*
σ	*Parâmetro de desvio padrão da distribuição Normal*
ν	*Graus de liberdade da distribuição Chi-Quadrado*
k	*Parâmetro de forma da distribuição Gamma*
Δt	*Intervalo de tempo decorrido desde o evento*
$P(\theta\|D)$	*Distribuição posterior (crença atualizada)*
$P(D\|\theta)$	*Verossimilhança (probabilidade de observar desempenho dado o proxy)*
$P(\theta)$	*Distribuição a priori (crença inicial)*
$P(D)$	*Evidência marginal*
$B(\alpha, \beta)$	*Função Beta de normalização*
$\Gamma(x)$	*Função Gama*
$w(t)$	*Peso de decaimento temporal*
$S_{i,t}$	*Pontuação contextual do proxy i no tempo t*
i^*	*Proxy selecionado pelo algoritmo*
N	*Número total de proxies no pool*

Trabalhos Submetidos e Publicados

Trabalhos aprovados e/ou publicados:

- SOUZA, P. H. C.; BRITO, L. C.; MARQUES, T. C. Modelagem Bayesiana para Recomendação Adaptiva de Proxies em Sistemas Automatizados. In: *XVII Congresso Brasileiro de Inteligência Computacional*, Belo Horizonte, MG, 2025. (Aprovado)

Sumário

1 Introdução

1.1 Descrição do Problema

Na era digital contemporânea, os dados emergem como um dos ativos mais valiosos para empresas e organizações. A coleta e análise eficazes de dados permitem a tomada de decisões informadas, a identificação de oportunidades de mercado e a otimização de processos operacionais, impulsionando o crescimento e a eficiência (STOBIERSKI, 2019). A importância dos dados é evidenciada pelo volume massivo de informações geradas diariamente; estimativas indicam que, em 2025, a quantidade total de dados no mundo alcançará 175 zettabytes, um aumento significativo em relação aos 33 zettabytes registrados em 2018 (REINSEL; GANTZ; RYDNING, 2020).

No entanto, para extrair valor real desses dados, é crucial coletá-los de forma rápida, completa e precisa, além de conectá-los a outras informações relevantes. Nesse contexto, a captura de dados desempenha um papel fundamental, fornecendo a matéria-prima necessária para análises que embasam decisões estratégicas e promovem a inovação, em muitos casos garantindo o sucesso das empresas (BAK, 2023).

A captura de dados em larga escala enfrenta desafios significativos. Com a crescente vigilância digital e a proteção de dados, muitos serviços implementam restrições rigorosas para impedir acessos automatizados (LLAMAS et al., 2025). O bloqueio de IPs é uma das principais estratégias empregadas para conter essa atividade, tornando o processo de extração de dados mais complexo (DERI; FUSCO, 2023) (HOETZLEIN, 2025). *Proxies* são amplamente utilizados como uma solução para contornar esses bloqueios, permitindo que sistemas de coleta de dados mantenham sua operação e acessem informações de maneira mais eficaz e segura (MITCHELL, 2023).

A utilização de *proxies*, no entanto, por si só não garante um funcionamento eficiente dos sistemas de captura de dados. Para evitar a utilização de *proxies* bloqueados ou inativos, estratégias de gerenciamento são necessárias, assegurando que os recursos disponíveis sejam utilizados de maneira inteligente e adaptativa. Dessa forma, a seleção apropriada dos *proxies* disponíveis se torna essencial para a continuidade e desempenho otimizado da captura de dados (BALLA, 2025).

1.2 Contexto da Literatura

Apesar da importância crescente da captura de dados e do uso generalizado de *proxies*, observa-se uma notável escassez de trabalhos científicos dedicados à otimização da

seleção e gerenciamento de *proxies*. A maior parte do conhecimento disponível encontra-se em fontes não-acadêmicas, como blogs técnicos, fóruns de discussão e documentação de fornecedores comerciais.

As abordagens comumente discutidas nestas fontes — como listas estáticas, *proxies* dinâmicos e técnicas de randomização — oferecem soluções parciais para o problema (FasterCapital, 2025). Contudo, estas estratégias frequentemente carecem de adaptabilidade, tratando todos os *proxies* de forma homogênea e ignorando o contexto específico de cada *site*-alvo.

Esta lacuna na literatura científica representa uma oportunidade significativa para o desenvolvimento de abordagens mais sofisticadas e academicamente rigorosas. A aplicação de técnicas de modelagem probabilística, aprendizado de máquina e otimização dinâmica neste domínio permanece largamente inexplorada, abrindo caminho para contribuições inovadoras e de alto impacto.

1.3 Motivação

A motivação para este trabalho surge da necessidade prática de otimizar sistemas de captura de dados em larga escala, que enfrentam desafios operacionais diários relacionados à disponibilidade e eficácia dos *proxies*. A ineficiência na seleção de *proxies* resulta em:

- **Aumento de Custos**: Utilização de *proxies* ineficazes resulta em desperdício de recursos;
- **Redução da Qualidade dos Dados**: Falhas na captura podem levar a dados incompletos ou desatualizados;
- **Aumento da Manutenção Manual**: Necessidade de intervenção humana para gerenciar *proxies*;
- **Perda de Oportunidades**: Incapacidade de coletar dados em tempo hábil para análises estratégicas.

A busca por uma solução que minimize estes problemas, através de uma abordagem cientificamente fundamentada e empiricamente validada, constitui a principal força motriz deste trabalho.

1.4 Objetivos

1.4.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver e validar um sistema de recomendação de *proxies* que seja escalável, eficiente e adaptativo, utilizando modelagem probabilística para otimizar a seleção de *proxies* em sistemas de captura de dados em larga escala.

1.4.2 Objetivos Específicos

Para alcançar o objetivo geral, foram definidos os seguintes objetivos específicos:

1. Desenvolver e implementar múltiplas estratégias de seleção de *proxies*, incluindo abordagens determinísticas e probabilísticas;
2. Implementar um sistema de recomendação de *proxies* utilizando inferência bayesiana com diferentes distribuições de probabilidade;
3. Realizar testes empíricos em ambiente de produção para comparar o desempenho das estratégias implementadas;
4. Analisar e discutir os resultados obtidos, avaliando o impacto da abordagem proposta na taxa de sucesso da captura de dados;
5. Documentar a arquitetura, implementação e resultados do sistema.

1.5 Organização do Texto

Esta dissertação está organizada da seguinte forma:

- **Capítulo 2**: Apresenta a fundamentação teórica, abordando conceitos fundamentais de captura de dados, *proxies* e inferência bayesiana.
- **Capítulo 3**: Detalha a metodologia utilizada, incluindo a arquitetura do sistema, ambiente de testes, métricas de avaliação e as estratégias de seleção de *proxies* implementadas, divididas em três categorias: básicas, determinísticas com histórico e probabilísticas.
- **Capítulo 4**: Apresenta os resultados experimentais obtidos, comparando o desempenho das diferentes estratégias.
- **Capítulo 5**: Apresenta a conclusão do trabalho, discutindo as principais contribuições, limitações e direções para pesquisas futuras.

2 Referencial Teórico

2.1 Coleta de Dados na Web e Mecanismos de Bloqueio

No cenário digital atual, a coleta de dados na web, ou *web scraping*, consiste no processo automatizado de extração de informações de plataformas online. Esta técnica emprega robôs (*bots*) para realizar requisições a páginas web, analisar o código-fonte (predominantemente HTML) e converter dados não estruturados em formatos estruturados, como CSV ou JSON (ZHAO, 2017). As aplicações estratégicas do *web scraping* são diversas, incluindo monitoramento de preços, análise de sentimento do consumidor e pesquisa de mercado (WebHarvy, 2025).

A prática do *web scraping* opera em um ambiente de conflito, gerando uma dinâmica de "gato e rato" entre os desenvolvedores de *scrapers* e os administradores de websites. Estes últimos implementam um arsenal de medidas de segurança, conhecidas como tecnologias anti-*bot* ou anti-*scraping*, para proteger seus dados e recursos (KHDER, 2021). As principais técnicas de defesa incluem:

- **Limitação de Taxa (*Rate Limiting*):** Os servidores monitoram a frequência de requisições de um único endereço IP. Se um cliente excede um limiar predefinido (e.g., 100 requisições por minuto), o servidor pode acionar um bloqueio temporário, permanente ou a apresentação de um desafio CAPTCHA (JONKER; KRUMNOW; VLOT, 2019).
- **Detecção Baseada em Comportamento:** Sistemas anti-*bot* analisam padrões de navegação para inferir a natureza do visitante. Comportamentos previsíveis e não humanos, como seguir links invisíveis (*honeypots*) ou acessar centenas de páginas em ordem sequencial, são indicadores de atividade automatizada (JANSEN, 2021).
- **Bloqueios Baseados em Identidade:** A verificação da identidade do cliente foca em "quem" ele é, analisando múltiplos fatores como o endereço IP, cabeçalhos HTTP (*User-Agent*) e a impressão digital (*fingerprint*) do navegador. Inconsistências entre esses elementos, como um *User-Agent* do Chrome com uma impressão digital TLS da biblioteca *requests* do Python, podem levar ao bloqueio (JONKER; KRUMNOW; VLOT, 2019).

2.2 Servidores Proxy como Estratégia de Evasão

Para contornar os mecanismos de bloqueio baseados em IP, os servidores *proxy* são uma ferramenta fundamental.

2.2.1 Princípios de Funcionamento

Os proxies atuam como intermediários entre o cliente e a internet, mascarando o endereço IP original do usuário. Quando um cliente faz uma requisição para acessar um site, essa requisição é primeiro enviada ao servidor proxy, que então a encaminha ao destino final. O site alvo responde ao proxy, que por sua vez retransmite a resposta ao cliente. Esse processo oculta a identidade real do solicitante e permite distribuir acessos por diferentes endereços IP, reduzindo o risco de bloqueios. O efeito estratégico é o mascaramento do IP real do *scraper*, fazendo com que a requisição pareça ter se originado do servidor proxy. Se o IP de um proxy é bloqueado, o *scraper* pode simplesmente utilizar outro, garantindo a continuidade da operação (DAVIES, 2019).

2.2.2 Tipos de Proxy

A escolha do tipo de proxy é uma decisão crítica que afeta o custo, a velocidade e a taxa de sucesso da coleta de dados.

- **Proxies de Datacenter:** Originários de servidores em nuvem, são baratos e rápidos, mas facilmente identificáveis e bloqueados por sistemas anti-*bot* devido à sua baixa reputação (ALIŠAUSKAS, 2024).
- **Proxies Residenciais:** Utilizam endereços IP de usuários reais, oferecendo o mais alto nível de legitimidade e anonimato. São eficazes contra defesas robustas, mas mais caros e potencialmente mais lentos (ALIŠAUSKAS, 2024).
- **Proxies de ISP (Residenciais Estáticos):** Híbridos que combinam a alta velocidade dos proxies de datacenter com a legitimidade dos residenciais. São ideais para tarefas que exigem sessões longas e estáveis (ALIŠAUSKAS, 2024).
- **Proxies Móveis:** Originários de redes de operadoras de celular, oferecem o mais alto grau de legitimidade, pois seus IPs são dinâmicos e compartilhados por muitos usuários, dificultando o banimento. São a opção mais cara (ALIŠAUSKAS, 2024).

2.3 Estratégias de Rotação de Proxies

A rotação de proxy é a prática de alterar sistematicamente o endereço IP de saída para distribuir a carga de requisições e evitar a detecção. O objetivo é contornar a limitação

de taxa e impedir que um único IP acumule uma reputação negativa.

2.3.1 Técnicas de Rotação Básica

As abordagens mais simples para rotação de *proxies* são as estratégias básicas, que não consideram o histórico de desempenho e se baseiam em critérios predefinidos.

- **Rotação Sequencial:** Este método percorre uma lista de *proxies* em uma ordem fixa e predeterminada. Ao chegar ao final da lista, o ciclo recomeça. Sua principal desvantagem é a alta previsibilidade, que pode ser facilmente detectada por sistemas de segurança que analisam padrões de acesso (CHANDRA, 2025).
- **Rotação Aleatória:** Uma evolução da abordagem sequencial, este método seleciona um *proxy* aleatoriamente de um *pool* disponível para cada nova requisição. Embora introduza um elemento de imprevisibilidade, essa estratégia ainda é considerada ingênua, pois trata todos os *proxies* como equivalentes, ignorando completamente seu desempenho, latência ou taxa de sucesso (NetNut, 2023).

Apesar de sua simplicidade de implementação, as estratégias de rotação básica são insuficientes para contornar defesas anti-*scraping* mais sofisticadas, que exigem abordagens adaptativas.

2.4 Seleção Adaptativa de Proxies com Modelagem Bayesiana

Como uma evolução das estratégias de rotação, a modelagem Bayesiana oferece uma abordagem robusta para a seleção adaptativa de *proxies*. Esta abordagem permite incorporar incertezas e aprender com o histórico de desempenho para otimizar as decisões de seleção, superando as limitações dos métodos tradicionais. A inferência Bayesiana fornece uma maneira matemática de atualizar crenças à medida que novas evidências são coletadas (KINAS; ANDRADE, 2021). No contexto de seleção de *proxies*, a crença sobre a qualidade de um *proxy* (θ) é atualizada com base nos dados de seu desempenho (D) usando o Teorema de Bayes:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$$

onde $P(\theta|D)$ é a distribuição posterior (a crença atualizada), $P(D|\theta)$ é a verossimilhança (a probabilidade de observar o desempenho dado o *proxy*), $P(\theta)$ é a distribuição a priori (a crença inicial), e $P(D)$ é a evidência. A ideia central é manter uma distribuição de probabilidade sobre a possível qualidade de cada proxy e atualizá-la à medida que novos resultados (sucessos ou falhas) são observados.

2.4.1 Amostragem de Thompson para o Problema de Seleção

A Amostragem de Thompson (*Thompson Sampling*) é um algoritmo que resolve de forma elegante o dilema entre exploração (testar *proxies* sobre os quais há incerteza) e explotação (usar os *proxies* com melhor desempenho conhecido). A sua aplicação neste contexto permite balancear dinamicamente a escolha entre os *proxies* com base em seu desempenho histórico e no grau de incerteza associado (AGRAWAL; GOYAL, 2011).

O princípio geral do algoritmo é o seguinte:

1. Para cada *proxy* (ou "braço", na terminologia de *multi-armed bandit*), mantém-se uma distribuição de probabilidade que representa a crença sobre sua qualidade.
2. A cada ciclo de seleção, o algoritmo sorteia um valor de cada uma dessas distribuições.
3. O *proxy* cujo valor sorteado for o mais favorável é selecionado para a próxima requisição.

A "qualidade" e a distribuição utilizada podem variar. Formalmente, para N proxies, a seleção é dada por:

$$i^* = \arg\max_{i=1\ldots N} \{s_i : s_i \sim P_i(\theta_i|D_i)\}$$

onde s_i é o valor amostrado da distribuição posterior P_i para o proxy i, com base em seu histórico de dados D_i.

Embora a distribuição Beta seja uma escolha comum para modelar probabilidades de sucesso, outras distribuições como Normal, Gamma ou Qui-Quadrado podem ser empregadas, adaptando-se a noção de "valor favorável". Por exemplo, em uma abordagem que modela taxas de erro, o objetivo pode ser minimizar o valor sorteado (utilizando $\arg\min$). Essa flexibilidade permite que a Amostragem de Thompson seja adaptada a diferentes modelagens do problema, garantindo que *proxies* com bom desempenho histórico sejam favorecidos, ao mesmo tempo que *proxies* com alta incerteza (distribuições mais "espalhadas") sejam ocasionalmente testados.

3 Metodologia

3.1 Estratégias de Seleção de Proxies

3.1.1 Estratégias Básicas

As estratégias básicas operam sem considerar o histórico de desempenho dos *proxies*, utilizando apenas critérios estruturais ou aleatórios para a seleção. Estas abordagens oferecem simplicidade de implementação e baixo overhead computacional, sendo adequadas para cenários onde a complexidade algorítmica deve ser minimizada.

3.1.1.1 Estratégia Round Robin

A estratégia de seleção de *proxies* Round Robin é um dos métodos mais fundamentais para distribuir cargas de trabalho de forma equitativa (GHOMI; RAHMANI; QADER, 2017). No contexto deste trabalho, ela garante que as requisições para um *site* alvo sejam rotacionadas entre os servidores *proxy* disponíveis, minimizando o risco de bloqueio de IP.

Funcionamento

O princípio do *Round Robin* é operar em uma lista de recursos de forma circular. A cada nova requisição, a estratégia seleciona o próximo *proxy* da lista. Ao chegar ao final, ela retorna ao início, conforme ilustrado no diagrama da Figura 3.1.

No sistema implementado, a seleção funciona a partir do histórico de uso: o *proxy* ocioso há mais tempo é escolhido. O processo é descrito no Algoritmo 1.

Algorithm 1 Seleção de Proxy com a Estratégia Round Robin

```
1: Função SelecionarProxyMaisAntigo(lista_de_proxies, url_alvo)
2: proxies_com_ultimo_uso ← []
3: for all proxy em lista_de_proxies do
4:     ultimo_uso ← ObterUltimoUso(proxy, url_alvo)
5:     if ultimo_uso for nulo then
6:         ultimo_uso ← DATA_MINIMA
7:     end if
8:     Adicionar (proxy, ultimo_uso) a proxies_com_ultimo_uso
9: end for
10: Ordenar proxies_com_ultimo_uso por ultimo_uso (crescente)
11: retornar proxies_com_ultimo_uso[0].proxy
```

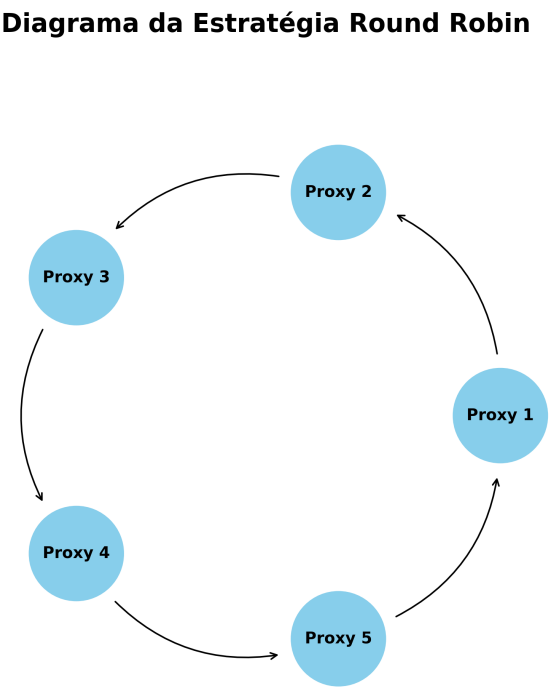

Figura 3.1 – Diagrama conceitual da estratégia Round Robin.

Vantagens e Desvantagens

A principal vantagem da estratégia *Round Robin* reside em sua **simplicidade** e **previsibilidade**. A implementação é direta e garante uma distribuição de carga perfeitamente equitativa, o que é fundamental para evitar a sobrecarga de um único servidor. Essa abordagem, especialmente na variação que prioriza o menos recentemente usado, minimiza a probabilidade de bloqueios por parte de sistemas anti-raspagem, em relação ao uso de IPs fixos.

No entanto, sua maior desvantagem é a incapacidade de considerar o **estado operacional** dos *proxies*. O método trata todos os recursos como se fossem idênticos, ignorando fatores críticos como indisponibilidade momentânea e bloqueios por parte de sistemas anti-raspagem. Consequentemente, se um *proxy* na lista estiver bloqueado ou indisponível, ele continuará a receber a mesma fração de requisições que os *proxies* operacionais. Isso pode degradar o desempenho geral do sistema, pois as requisições enviadas ao *proxy* problemático resultarão em falhas, desperdiçando tentativas que poderiam ser direcionadas a *proxies* funcionais.

3.1.1.2 Estratégia Random (Aleatória)

A estratégia de seleção aleatória (*Random Strategy*) representa uma abordagem direta e não determinística para a rotação de *proxies*. Diferente de métodos sequenciais ou

baseados em histórico, sua principal característica é a imprevisibilidade, o que pode ser vantajoso para evitar a detecção por sistemas de segurança que monitoram padrões de acesso.

Funcionamento

O mecanismo desta estratégia é intrinsecamente simples: a cada requisição, um *proxy* é escolhido de forma completamente aleatória dentre a lista de todos os servidores disponíveis. A única restrição aplicada é a exclusão de *proxies* que já estão em uso no momento da seleção, para evitar conflitos e garantir que apenas recursos ociosos sejam utilizados.

Essa simplicidade elimina a necessidade de manter um estado ou histórico de uso, tornando a implementação leve e eficiente. O processo é detalhado no Algoritmo 2.

Algorithm 2 Seleção de Proxy com a Estratégia Aleatória

```
1: Função SelecionarProxyAleatorio(lista_de_proxies)
2: proxies_disponiveis ← []
3: for all proxy em lista_de_proxies do
4:     if proxy.em_uso for falso then
5:         Adicionar proxy a proxies_disponiveis
6:     end if
7: end for
8: if proxies_disponiveis não estiver vazia then
9:     retornar EscolhaAleatoria(proxies_disponiveis)
10: else
11:     retornar nulo
12: end if
```

Vantagens e Desvantagens

A principal **vantagem** da estratégia aleatória é sua imprevisibilidade. Como não há um padrão fixo na seleção, pode ser mais difícil para um sistema de destino identificar que as requisições vêm de um processo automatizado.

No entanto, a principal **desvantagem** reside na sua própria aleatoriedade. Não há garantia de uma distribuição de carga uniforme. É estatisticamente possível que o mesmo *proxy* seja selecionado várias vezes seguidas, enquanto outros permanecem ociosos. Esse comportamento pode levar à sobrecarga de um servidor específico e aumentar a probabilidade de bloqueio, especialmente em um número menor de requisições. Em contrapartida, sobre um grande volume de requisições, a tendência é que a distribuição se aproxime de uma uniformidade, como previsto pela lei dos grandes números.

3.1.2 Estratégias Determinísticas com Histórico

As estratégias determinísticas utilizam informações históricas de desempenho para tomar decisões de seleção através de algoritmos previsíveis e reproduzíveis. Estas abordagens combinam a consideração do histórico com comportamento determinístico, oferecendo um meio-termo entre simplicidade e sofisticação.

3.1.3 Estratégia Exponential Backoff

A estratégia Exponential Backoff representa uma abordagem determinística para seleção de *proxies* que utiliza o histórico de desempenho para implementar penalizações temporais proporcionais ao número de falhas consecutivas. Esta estratégia distingue-se por sua natureza determinística e capacidade de isolamento automático de recursos problemáticos através de aumentos exponenciais nos tempos de espera.

Fundamentação Teórica

O Exponential Backoff é uma técnica amplamente utilizada em sistemas distribuídos e redes de computadores para controlar a taxa de tentativas de reconexão após falhas (TANENBAUM; WETHERALL, 2011). No contexto de seleção de *proxies*, esta abordagem adapta os princípios clássicos para implementar um sistema de penalização temporal que aumenta exponencialmente o tempo de bloqueio de um *proxy* a cada falha consecutiva.

Princípio de Funcionamento

O algoritmo mantém para cada *proxy* um contador de falhas consecutivas e calcula dinamicamente um tempo de bloqueio baseado na função exponencial:

$$T_{bloqueio} = T_{base} \times 2^{n_{falhas}} \times (1 + \text{jitter})$$

onde:

- T_{base} é o tempo base de bloqueio (em segundos)
- n_{falhas} é o número de falhas consecutivas
- jitter é um fator aleatório para evitar sincronização (tipicamente $\pm 10\%$)

O crescimento exponencial dos tempos resulta em bloqueios progressivamente mais longos: 1 falha = 30s, 3 falhas = 240s (4 min), 5 falhas = 960s (16 min), demonstrando como o algoritmo penaliza rapidamente *proxies* problemáticos.

Importância do Jitter

O **jitter** (variação aleatória) é um componente crucial que previne o fenômeno de sincronização conhecida como *"thundering herd"* (manada trovejante). Sem jitter, múltiplos *proxies* que falharam simultaneamente seriam desbloqueados exatamente no mesmo momento, causando:

- **Falhas em cascata**: A sobrecarga poderia causar novas falhas em massa
- **Oscilações no sistema**: Ciclos de bloqueio e desbloqueio sincronizados

O **JITTER_PERCENTUAL** define a amplitude da variação aleatória aplicada:

- **10%**: O tempo varia entre $T_{calculado} \times 0.9$ e $T_{calculado} \times 1.1$
- **Exemplo**: Para um tempo calculado de 120 segundos, o jitter de 10% resulta em um tempo real entre 108 e 132 segundos

A Figura 3.2 mostra uma timeline comparativa de desbloqueio de múltiplos *proxies*, evidenciando como o jitter distribui os momentos de reconexão ao longo do tempo, evitando a sincronização problemática.

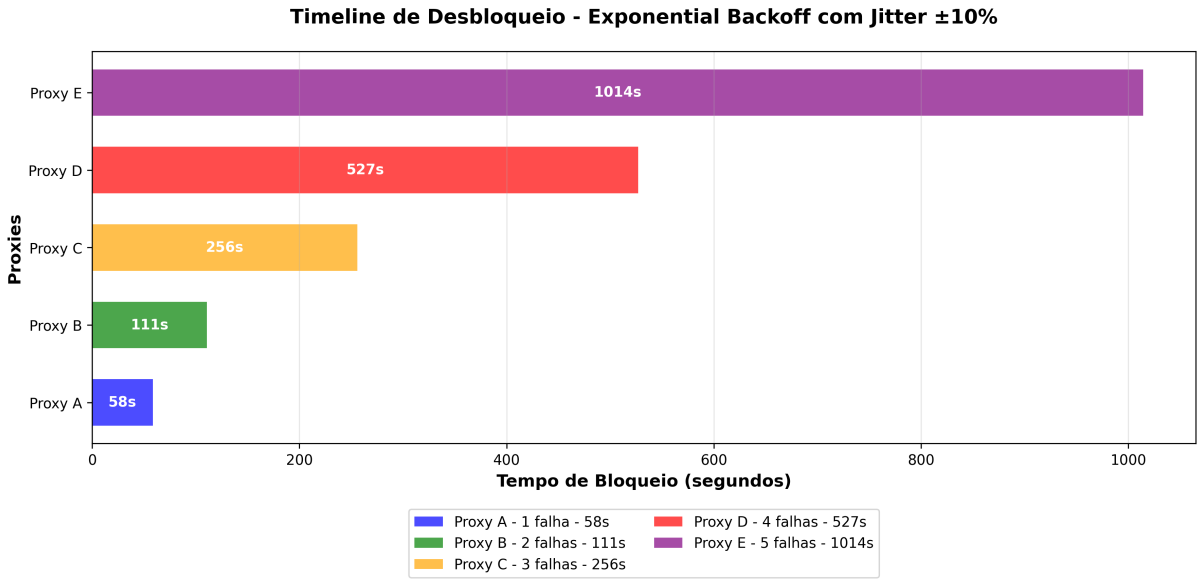

Figura 3.2 – Timeline de desbloqueio de múltiplos proxies demonstrando o efeito do jitter na distribuição temporal.

Características Distintivas

Diferentemente das estratégias bayesianas que operam por amostragem probabilística, o Exponential Backoff implementa uma lógica determinística de exclusão temporal:

- **Determinismo**: Não há elementos aleatórios na seleção, apenas na aplicação de jitter
- **Isolamento Automático**: *Proxies* problemáticos são automaticamente removidos da rotação
- **Recuperação Gradual**: *Proxies* podem retornar à rotação após o período de bloqueio
- **Penalização Crescente**: Falhas sucessivas resultam em bloqueios exponencialmente maiores

Implementação Algorítmica

Estrutura de Dados

Cada *proxy* mantém as seguintes informações de estado:

- **consecutive_failures**: Contador de falhas consecutivas
- **last_failure_time**: Timestamp da última falha
- **blocked_until**: Timestamp até o qual o *proxy* permanece bloqueado
- **total_requests**: Contador total de requisições
- **total_successes**: Contador total de sucessos

Algoritmo de Seleção

O processo de seleção segue uma lógica de filtragem e escolha baseada em disponibilidade temporal:

Tratamento de Resultados

Após cada requisição, o algoritmo atualiza o estado do *proxy* baseado no resultado:

Algorithm 3 Seleção de Proxy com Exponential Backoff

Função SelecionarProxyExponentialBackoff(lista_de_proxies, url_alvo)
tempo_atual ← ObterTimestampAtual()
proxies_disponiveis ← []
▷ Filtrar proxies disponíveis (não bloqueados)
for all *proxy* **em** *lista_de_proxies* **do**
    if *proxy*.blocked_until ≤ *tempo_atual* **then**
        Adicionar *proxy* **a** *proxies_disponiveis*
    end if
end for
▷ Verificar disponibilidade
if len(*proxies_disponiveis*) = 0 **then**
    retornar null ▷ Nenhum proxy disponível
end if
▷ Seleção determinística (round-robin ou menor uso)
proxy_escolhido ← SelecionarMenorUso(*proxies_disponiveis*)
retornar *proxy_escolhido*

Algorithm 4 Atualização de Estado Pós-Requisição

Função AtualizarEstadoProxy(*proxy*, *sucesso*)
proxy.total_requests ← *proxy*.total_requests + 1
if *sucesso* **then** ▷ Sucesso: resetar contador de falhas
    proxy.total_successes ← *proxy*.total_successes + 1
    proxy.consecutive_failures ← 0
    proxy.blocked_until ← 0 ▷ Desbloquear imediatamente
else ▷ Falha: incrementar contador e calcular bloqueio
    proxy.consecutive_failures ← *proxy*.consecutive_failures + 1
    proxy.last_failure_time ← ObterTimestampAtual()
    ▷ Calcular tempo de bloqueio exponencial
    tempo_base ← TEMPO_BASE_BLOQUEIO ▷ ex: 30 segundos
    exponente ← *proxy*.consecutive_failures
    $tempo\_bloqueio \leftarrow tempo\_base \times 2^{exponente}$
    ▷ Aplicar jitter para evitar sincronização
    jitter ← UniformRandom(-0.1, 0.1) ▷ ±10%
    $tempo\_bloqueio \leftarrow tempo\_bloqueio \times (1 + jitter)$
    ▷ Aplicar limite máximo
    tempo_bloqueio ← Min(*tempo_bloqueio*, TEMPO_MAXIMO_BLOQUEIO)
    proxy.blocked_until ← *proxy*.last_failure_time + *tempo_bloqueio*
end if

Parâmetros de Configuração

A estratégia Exponential Backoff utiliza um conjunto mínimo de parâmetros configuráveis:

Parâmetros Temporais

- **TEMPO_BASE_BLOQUEIO**: Tempo inicial de bloqueio (padrão: 30 segundos)
- **TEMPO_MAXIMO_BLOQUEIO**: Limite superior para bloqueios (padrão: 3600 segundos = 1 hora)
- **JITTER_PERCENTUAL**: Variação aleatória aplicada (padrão: 10%)

Parâmetros de Controle

- **MAX_CONSECUTIVE_FAILURES**: Número máximo de falhas antes do bloqueio máximo (padrão: 10)
- **RESET_THRESHOLD_HOURS**: Tempo para reset automático de contadores (padrão: 24 horas)

3.1.4 Estratégias Bayesianas com Thompson Sampling

As estratégias bayesianas representam uma abordagem mais sofisticada para seleção de *proxies*, fundamentada em princípios de inferência estatística e aprendizado adaptativo. Estas estratégias utilizam o *Thompson Sampling* como mecanismo de decisão, combinando exploração e explotação de forma otimizada através de diferentes distribuições probabilísticas.

Fundamentação Teórica

Inferência Bayesiana

A inferência bayesiana fornece um framework matemático robusto para atualizar crenças sobre parâmetros desconhecidos à medida que novas evidências são coletadas. No contexto de seleção de *proxies*, a crença posterior sobre um parâmetro θ é atualizada segundo o teorema de Bayes:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$$

onde $P(\theta|D)$ é a distribuição posterior, $P(D|\theta)$ é a verossimilhança, $P(\theta)$ é a distribuição a priori, e $P(D)$ é a evidência (KINAS; ANDRADE, 2021).

Decaimento Temporal

Em ambientes dinâmicos, a relevância de eventos históricos diminui com o tempo. Para capturar esta dinâmica, implementa-se um sistema de decaimento temporal exponencial:

$$w(t) = e^{-\lambda \Delta t}$$

onde λ é a taxa de decaimento e Δt representa o tempo decorrido desde o evento. Esta formulação permite que eventos recentes tenham maior influência na estimativa do desempenho atual.

Thompson Sampling

O *Thompson Sampling* é um algoritmo de *multi-armed bandit* que equilibra naturalmente exploração e explotação através de amostragem probabilística. Para cada *proxy*, uma amostra é sorteada de sua distribuição posterior, e o *proxy* com o valor mais favorável (maior ou menor, dependendo da estratégia) é selecionado.

Este mecanismo garante que:

- **Proxies com bom histórico** tenham alta probabilidade de seleção
- **Proxies com alta incerteza** sejam ocasionalmente explorados
- A **exploração diminua** automaticamente conforme mais dados são coletados

3.1.4.1 Estratégia com Distribuição Beta

A distribuição Beta constitui a base teórica mais sólida para modelagem de probabilidades de sucesso em contextos bayesianos, sendo particularmente adequada para a seleção de *proxies* devido ao seu domínio natural no intervalo $[0, 1]$ e sua propriedade de conjugação com distribuições binomiais.

Fundamentação Matemática

A função de densidade de probabilidade da distribuição Beta é definida por:

$$f(p; \alpha, \beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)}$$

onde $B(\alpha, \beta)$ é a função Beta que serve como constante de normalização:

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha + \beta)}$$

Os parâmetros α e β podem ser interpretados como contadores bayesianos: α representa evidências de sucesso, enquanto β representa evidências de falha. A forma da distribuição reflete diretamente o conhecimento acumulado sobre o *proxy*:

- $\alpha > \beta$: A distribuição se inclina para a direita, indicando maior probabilidade de sucesso
- $\beta > \alpha$: A distribuição se inclina para a esquerda, indicando maior probabilidade de erro
- α **e** β **grandes**: A distribuição se torna mais concentrada, refletindo maior confiança na estimativa
- $\alpha = \beta = 1$: Distribuição uniforme, representando incerteza total

A Figura 3.3 ilustra essas diferentes configurações da distribuição Beta.

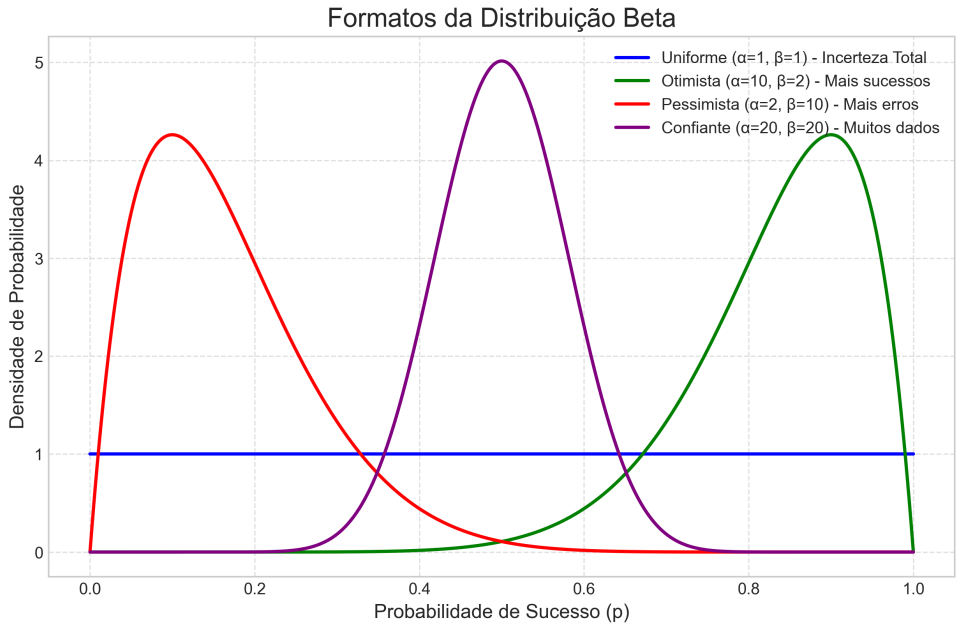

Figura 3.3 – Diferentes formatos da Distribuição Beta conforme os parâmetros α e β.

Atualização Bayesiana com Decaimento Temporal

A implementação incorpora decaimento temporal para valorizar eventos recentes. Os parâmetros são calculados como somas ponderadas:

$$\alpha_w = \alpha_{inicial} + \sum_{i \in \text{sucessos}} e^{-\lambda_s \Delta t_i}$$

$$\beta_w = \beta_{inicial} + \sum_{j \in \text{erros}} e^{-\lambda_e \Delta t_j} \times M_{erro}$$

onde:

- λ_s e λ_e são as taxas de decaimento para sucessos e erros
- Δt_i e Δt_j representam o tempo decorrido desde cada evento
- M_{erro} é um multiplicador que penaliza erros com maior rigor
- $\alpha_{inicial}$ e $\beta_{inicial}$ são valores *a priori*

O impacto dos sucessos na crença sobre um *proxy* é demonstrado na Figura 3.4, onde observa-se como eventos positivos deslocam a distribuição para a direita, aumentando a confiança em sua eficácia.

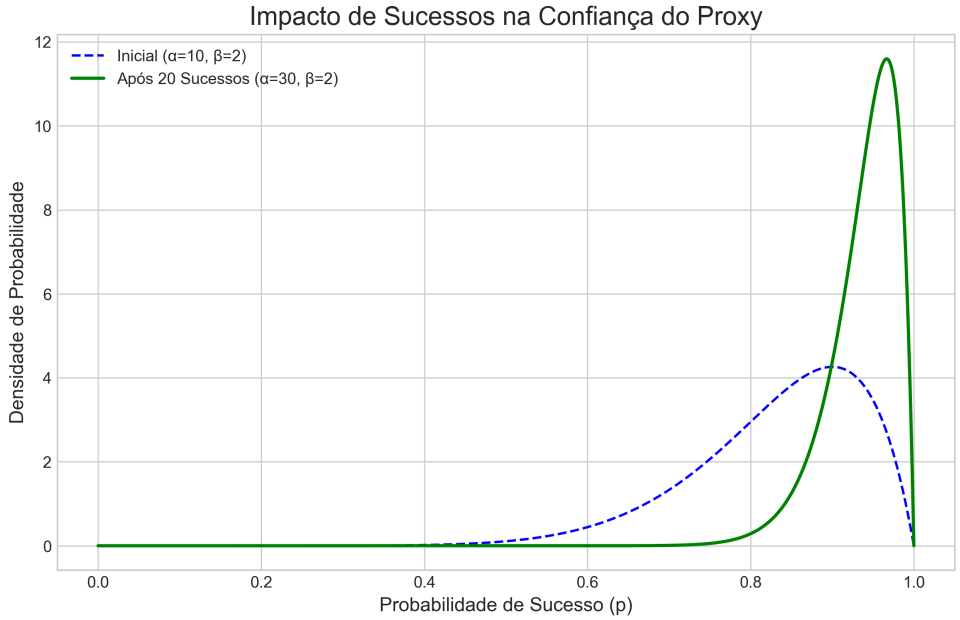

Figura 3.4 – Impacto de sucessos na distribuição Beta, aumentando a confiança na eficácia do proxy.

Conversamente, a Figura 3.5 ilustra como erros deslocam a distribuição para a esquerda, diminuindo a confiança no *proxy*.

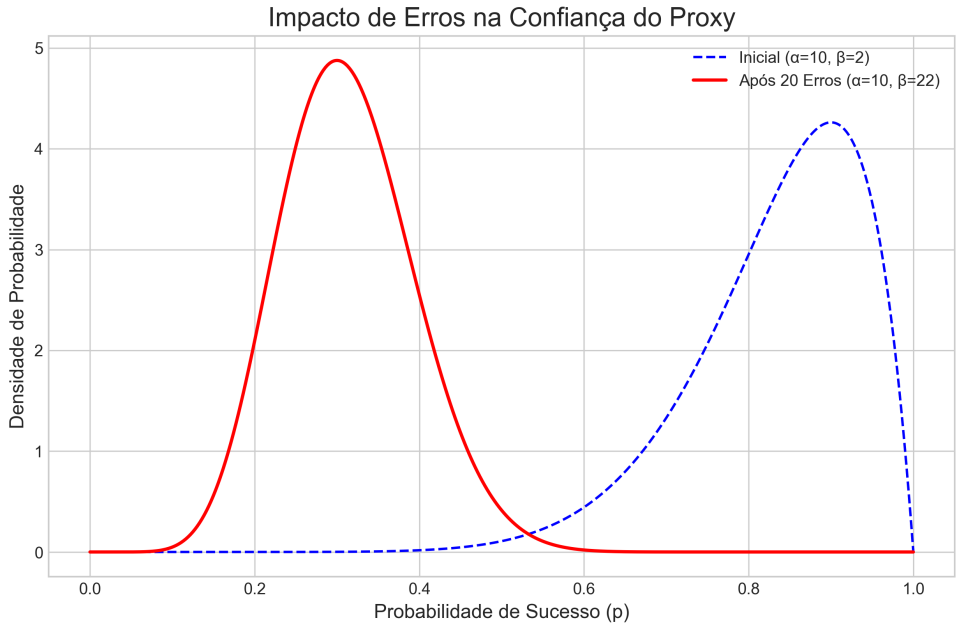

Figura 3.5 – Impacto de erros na distribuição Beta, diminuindo a confiança no proxy.

Decaimento Temporal

O sistema de decaimento temporal garante que eventos recentes tenham maior influência na tomada de decisão. A Figura 3.6 demonstra como o peso de um evento diminui exponencialmente com o tempo.

Figura 3.6 – Efeito do decaimento temporal exponencial no peso de um evento.

Esta abordagem permite que o sistema se adapte rapidamente a mudanças nas condições operacionais, como bloqueios temporários ou melhorias na infraestrutura de

rede.

Thompson Sampling com Distribuição Beta

O *Thompson Sampling* utiliza a distribuição Beta posterior para balancear exploração e explotação. Para cada *proxy*, uma amostra p_i é sorteada de sua distribuição Beta($\alpha_{w,i}, \beta_{w,i}$), e o *proxy* com o maior valor amostrado é selecionado.

Este processo garante que:

- **Proxies confiáveis** (alto α_w, baixo β_w) tenham distribuições concentradas em valores altos, sendo frequentemente selecionados
- **Proxies novos** ou com pouco histórico tenham distribuições mais amplas, garantindo exploração adequada
- A **incerteza diminua** naturalmente conforme mais dados são coletados

Algoritmo Detalhado de Seleção

Algorithm 5 Seleção de Proxy com Distribuição Beta - Versão Detalhada

Função SelecionarProxyBayesianoBeta(lista_de_proxies, url_alvo, grupo_id)
scores_finais ← []
info_proxies ← []
tempo_atual ← ObterTempoAtual()
for all *proxy* **em** *lista_de_proxies* **do**
    logs_filtrados ← ObterLogsComFiltro(*proxy*, *url_alvo*, MAX_LOGS)
    if len(*logs_filtrados*) < MIN_TENTATIVAS **then** ▷ Usar valores iniciais para proxies novos
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
    else ▷ Calcular parâmetros baseados no histórico
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
        for all *log* **em** *logs_filtrados* **do**
            $\Delta t \leftarrow$ *tempo_atual* - *log*.timestamp
            ▷ Aplicar delays configuráveis
            if *log*.sucesso **e** $\Delta t \geq$ DELAY_SUCESSO **then**
                $peso \leftarrow e^{-\lambda_s \Delta t}$
                if $peso \geq$ THRESHOLD_DECAY_MINIMO **then**
                    $\alpha_w \leftarrow \alpha_w + peso$
                end if
            else if não *log*.sucesso **e** $\Delta t \geq$ DELAY_ERRO **then**
                $peso \leftarrow e^{-\lambda_e \Delta t} \times M_{erro}$
                if $peso \geq$ THRESHOLD_DECAY_MINIMO **then**
                    $\beta_w \leftarrow \beta_w + peso$
                end if
            end if
        end for
    end if
    ▷ Garantir estabilidade numérica
    $\alpha_w \leftarrow \max(\alpha_w, 0.01)$
    $\beta_w \leftarrow \max(\beta_w, 0.01)$
    ▷ Thompson Sampling
    score ← AmostraBetaSegura(α_w, β_w)
    prob_esperada ← $\frac{\alpha_w}{\alpha_w+\beta_w}$
    Adicionar (*proxy*, *score*, α_w, β_w, *prob_esperada*) **a** *info_proxies*
    Adicionar *score* **a** *scores_finais*
end for
▷ Seleção do melhor proxy
indice_melhor ← IndiceMaiorValor(*scores_finais*)
proxy_selecionado ← *info_proxies*[*indice_melhor*].proxy
▷ Logging para análise
LogResultadoSelecao(*proxy_selecionado*, *info_proxies*)
retornar *proxy_selecionado*

Função de Amostragem Segura

Para garantir estabilidade numérica, implementa-se uma função de amostragem que trata casos extremos:

Algorithm 6 Amostragem Beta Segura

Função AmostraBetaSegura(α, β)
▷ Garantir valores mínimos
$\alpha \leftarrow \max(\alpha, 0.01)$
$\beta \leftarrow \max(\beta, 0.01)$
▷ Calcular razão para detectar casos extremos
$razao \leftarrow \frac{\alpha}{\alpha+\beta}$
if $razao < 0.01$ **then** ▷ Caso extremo: muito baixa probabilidade
    retornar AmostraBeta(0.1, 10)
else if $razao > 0.99$ **then** ▷ Caso extremo: muito alta probabilidade
    retornar AmostraBeta(10, 0.1)
else ▷ Caso normal: limitar valores para evitar overflow
    $\alpha_{safe} \leftarrow \min(\alpha, 1000)$
    $\beta_{safe} \leftarrow \min(\beta, 1000)$
    retornar AmostraBeta($\alpha_{safe}, \beta_{safe}$) ErroNumerico ▷ Fallback: usar valor esperado com ruído
    $ruido \leftarrow$ RuidoNormal(0, 0.1)
    retornar Clip($razao + ruido$, 0.0, 1.0)
end if

3.1.4.2 Estratégia com Distribuição Qui-Quadrado

A estratégia baseada na distribuição qui-quadrado (χ^2) constitui uma abordagem metodologicamente diferenciada para seleção de *proxies*, caracterizada pelo emprego de ranking invertido onde scores menores indicam melhor desempenho. Esta estratégia adapta conceitos fundamentais da teoria estatística para criar um sistema que penaliza de forma acentuada *proxies* com histórico de erros.

Fundamentação Matemática

A distribuição qui-quadrado é uma distribuição contínua amplamente utilizada em testes de hipóteses e análise estatística. Ela é caracterizada por um único parâmetro K denominado graus de liberdade, possuindo domínio nos números reais não-negativos. Sua função de densidade de probabilidade é expressa por:

$$f(x; K) = \frac{1}{2^{K/2}\Gamma(K/2)} x^{K/2-1} e^{-x/2}$$

onde:

- $\Gamma(\cdot)$ é a função gama: $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$
- $K > 0$ representa os graus de liberdade
- $x \geq 0$ é o domínio da variável aleatória

Uma propriedade fundamental desta distribuição é que valores próximos a zero são mais prováveis quando K é baixo, tornando-a naturalmente adequada para sistemas de ranking onde desejamos favorecer *proxies* com scores menores (melhor desempenho).

A Figura 3.7 demonstra como diferentes valores de K alteram substancialmente a forma da distribuição.

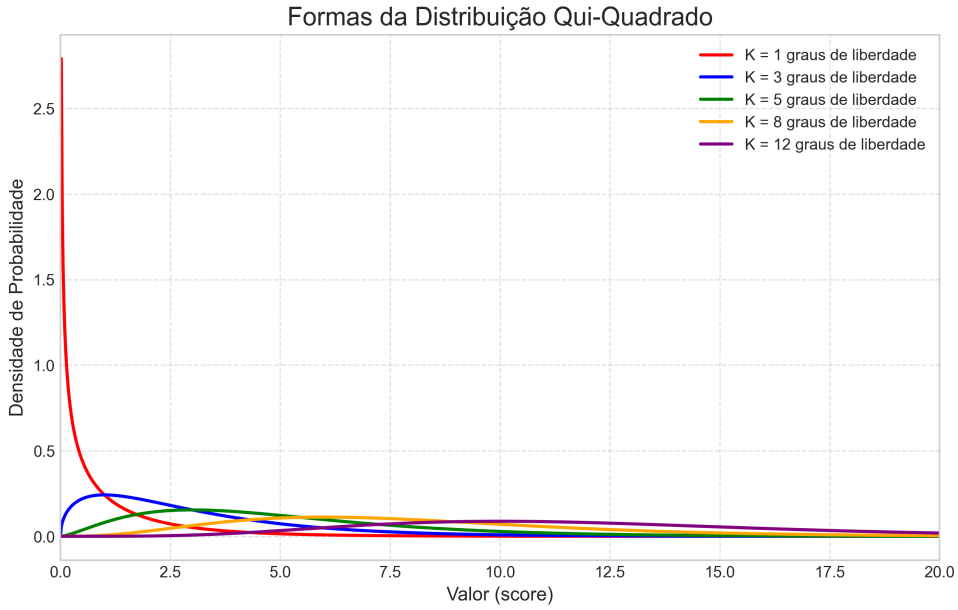

Figura 3.7 – Formas da distribuição qui-quadrado para diferentes valores de K (graus de liberdade).

Cálculo Adaptativo dos Graus de Liberdade

O aspecto mais inovador desta estratégia reside no cálculo dinâmico dos graus de liberdade K baseado no histórico específico de cada *proxy*. Ao contrário de abordagens convencionais que utilizam valores fixos, implementa-se a seguinte formulação adaptativa:

$$K = \frac{\beta_w}{\alpha_w + 1} \times \phi + \kappa$$

onde:

- $\alpha_w = \sum_{i \in \text{sucessos}} e^{-\lambda_s \Delta t_i}$ representa sucessos ponderados temporalmente
- $\beta_w = \sum_{j \in \text{erros}} e^{-\lambda_e \Delta t_j}$ representa erros ponderados temporalmente

- $\phi = 5$ é o fator de escala (ajustável conforme necessário)
- $\kappa = 5$ é a constante aditiva que garante $K_{min} = 5$

Esta formulação matemática estabelece uma relação direta entre o histórico de desempenho e a forma da distribuição resultante:

- **Proxies com muitos erros**: β_w elevado resulta em K maior, gerando distribuições mais achatadas com maior probabilidade de produzir scores altos (classificação inferior)
- **Proxies com muitos sucessos**: α_w elevado resulta em K menor, gerando distribuições concentradas em valores baixos (classificação superior)
- **Proxies novos**: Com α_w e β_w próximos aos valores iniciais, K assume valores intermediários

A Figura 3.8 ilustra quantitativamente como os parâmetros α e β influenciam o valor de K.

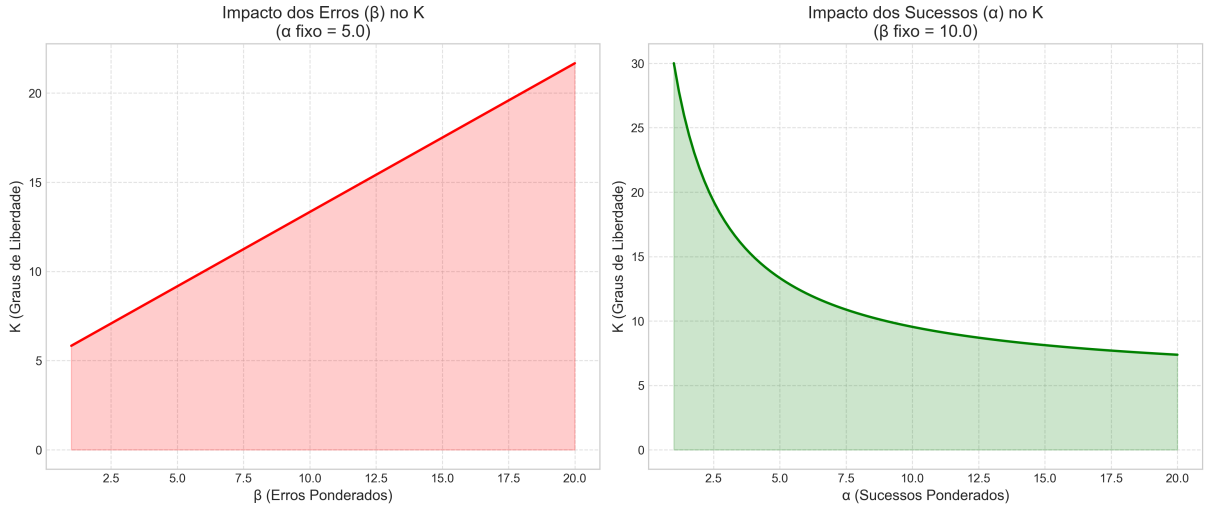

Figura 3.8 – Impacto dos sucessos (α) e erros (β) no cálculo dos graus de liberdade K.

Sistema de Ranking Invertido

Diferentemente das estratégias baseadas em maximização de scores, a abordagem qui-quadrado utiliza um paradigma de ranking invertido: o *proxy* que gera o menor score através do *Thompson Sampling* é selecionado para a próxima requisição.

Este comportamento emerge naturalmente das propriedades da distribuição qui-quadrado, onde valores próximos a zero são mais frequentes para distribuições com poucos graus de liberdade. Matematicamente, isto se traduz em:

$$P(X \leq x) = \frac{\gamma(K/2, x/2)}{\Gamma(K/2)}$$

onde γ é a função gama incompleta inferior, demonstrando que para K pequeno, a probabilidade acumulada próxima a zero é significativamente maior.

A Figura 3.9 compara as distribuições qui-quadrado de diferentes *proxies* hipotéticos, evidenciando como aqueles com melhor histórico (mais sucessos, menos erros) tendem a gerar scores menores.

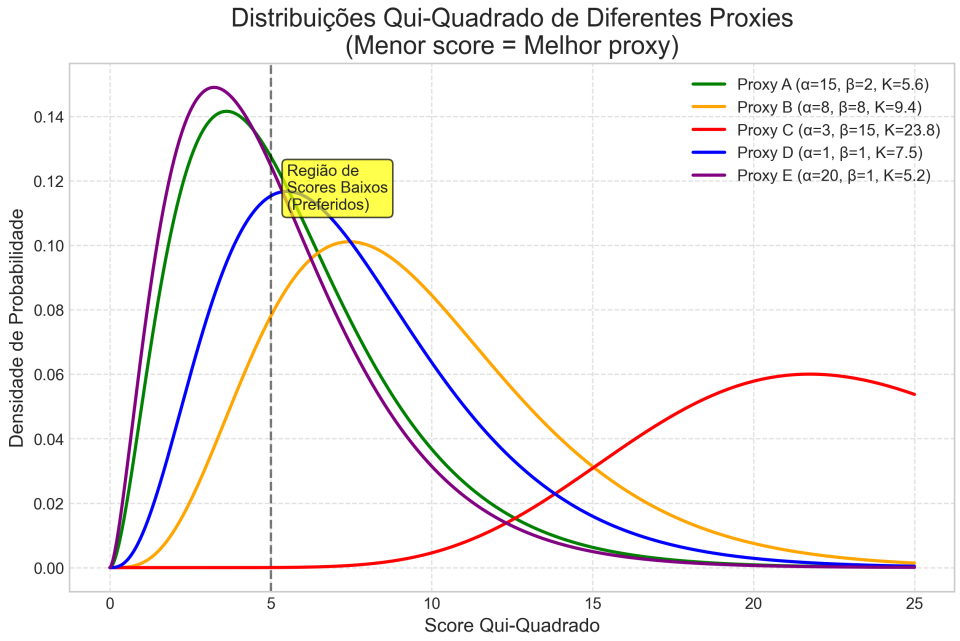

Figura 3.9 – Comparação das distribuições qui-quadrado de diferentes proxies com seus respectivos históricos.

Implementação com Decaimento Temporal

Assim como nas demais estratégias bayesianas, a abordagem qui-quadrado incorpora decaimento temporal exponencial para priorizar eventos recentes. Os parâmetros α_w e β_w são calculados através das somas ponderadas:

$$\alpha_w = \alpha_{inicial} + \sum_{i \in \text{sucessos}} w_s(t_i) \times M_{sucesso}$$

$$\beta_w = \beta_{inicial} + \sum_{j \in \text{erros}} w_e(t_j) \times M_{erro}$$

onde:

- $w_s(t_i) = e^{-\lambda_s \Delta t_i}$ é o peso temporal para sucessos
- $w_e(t_j) = e^{-\lambda_e \Delta t_j}$ é o peso temporal para erros
- $M_{sucesso}$ e M_{erro} são multiplicadores configuráveis
- λ_s e λ_e são as taxas de decaimento (podem ser diferentes)

Thompson Sampling com Ranking Invertido

O *Thompson Sampling* na estratégia qui-quadrado opera através da amostragem de cada distribuição $\chi^2(K_i)$ correspondente ao *proxy* i, seguida pela seleção do *proxy* que produziu o menor valor amostrado. Este processo pode ser formalizado como:

$$i^* = \arg\min_i \{X_i : X_i \sim \chi^2(K_i)\}$$

onde X_i representa a amostra sorteada da distribuição qui-quadrado do *proxy* i.

Algoritmo Detalhado de Seleção

Função de Amostragem Segura

Para garantir robustez numérica e tratar casos extremos, implementa-se uma função especializada de amostragem qui-quadrado:

Algorithm 8 Amostragem Qui-Quadrado Segura

Função AmostraQuiQuadradoSegura(K)
▷ Garantir valor mínimo para K
$K \leftarrow \max(K, 1.0)$
▷ Detectar casos extremos que podem causar instabilidade
if $K > 1000$ **then** ▷ Caso extremo: K muito alto - usar aproximação normal
    $media \leftarrow K$
    $variancia \leftarrow 2K$
    $sample \leftarrow$ AmostraNormal($media$, $variancia$)
    retornar $\max(sample, 0)$
else if $K < 0.1$ **then** ▷ Caso extremo: K muito baixo - usar valor fixo baixo
    retornar RuidoUniforme(0.001, 0.1)
else ▷ Caso normal: usar distribuição qui-quadrado padrão
    retornar AmostraQuiQuadrado(K) ErroNumerico ▷ Fallback: usar valor esperado com ruído
    $ruido \leftarrow$ RuidoExponencial(1.0)
    retornar $K + ruido$
end if

Algorithm 7 Seleção de Proxy com Distribuição Qui-Quadrado - Versão Detalhada

Função SelecionarProxyQuiQuadrado(lista_de_proxies, url_alvo, grupo_id)
scores_finais ← []
info_proxies ← []
tempo_atual ← ObterTempoAtual()
for all *proxy* **em** *lista_de_proxies* **do**
    logs_filtrados ← ObterLogsComFiltro(*proxy*, *url_alvo*, MAX_LOGS)
    if len(*logs_filtrados*) < MIN_TENTATIVAS **then** ▷ Usar valores iniciais para proxies novos
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
    else ▷ Calcular parâmetros baseados no histórico
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
        for all *log* **em** *logs_filtrados* **do**
            $\Delta t \leftarrow$ *tempo_atual* - *log*.timestamp
            ▷ Aplicar delays configuráveis e filtros temporais
            if *log*.sucesso **e** $\Delta t \geq$ DELAY_SUCESSO **then**
                $peso \leftarrow e^{-\lambda_s \Delta t} \times M_{sucesso} \times BASE\_MULTIPLIER$
                if *peso* ≥ THRESHOLD_DECAY_MINIMO **then**
                    $\alpha_w \leftarrow \alpha_w + peso$
                end if
            else if não *log*.sucesso **e** $\Delta t \geq$ DELAY_ERRO **then**
                $peso \leftarrow e^{-\lambda_e \Delta t} \times M_{erro} \times BASE\_MULTIPLIER$
                if *peso* ≥ THRESHOLD_DECAY_MINIMO **then**
                    $\beta_w \leftarrow \beta_w + peso$
                end if
            end if
        end for
    end if
    ▷ Garantir estabilidade numérica
    $\alpha_w \leftarrow \max(\alpha_w, 1.0)$
    $\beta_w \leftarrow \max(\beta_w, 1.0)$
    ▷ Calcular graus de liberdade adaptativos
    $K \leftarrow \frac{\beta_w}{\alpha_w + 1.0} \times FATOR\_ESCALA + CONSTANTE\_ADITIVA$
    $K \leftarrow \max(K, 1.0)$ ▷ Garantir K válido para distribuição qui-quadrado
    ▷ Thompson Sampling com distribuição qui-quadrado
    score ← AmostraQuiQuadradoSegura(K)
    valor_esperado ← K ▷ E[χ^2(K)] = K
    Adicionar (*proxy*, *score*, α_w, β_w, K, *valor_esperado*) **a** *info_proxies*
    Adicionar *score* **a** *scores_finais*
end for
▷ Seleção do melhor proxy (MENOR score)
indice_melhor ← IndiceMenorValor(*scores_finais*)
proxy_selecionado ← *info_proxies*[*indice_melhor*].proxy
▷ Logging detalhado para análise
LogResultadoSelecaoQuiQuadrado(*proxy_selecionado*, *info_proxies*)
retornar *proxy_selecionado*

3.1.4.3 Estratégia com Distribuição Normal

A estratégia baseada na distribuição Normal utiliza uma abordagem probabilística para seleção de *proxies*, modelando o desempenho de cada proxy através de parâmetros de média e variância. Esta estratégia permite ajuste flexível entre exploração de novos proxies e exploração dos recursos com melhor histórico de desempenho.

Fundamentação Matemática

A distribuição Normal (ou Gaussiana) constitui uma das distribuições mais fundamentais da estatística, caracterizada por sua forma de sino simétrica e propriedades analíticas bem estabelecidas. Sua função de densidade de probabilidade é expressa por:

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

onde:

- $\mu \in R$ é o parâmetro de localização (média)
- $\sigma^2 > 0$ é o parâmetro de escala (variância)
- $\sigma = \sqrt{\sigma^2}$ é o desvio padrão
- O domínio é $x \in (-\infty, +\infty)$

A distribuição Normal possui propriedades matemáticas excepcionais, incluindo estabilidade sob transformações lineares e o teorema central do limite, que garantem comportamento previsível e robusto em diversas condições operacionais.

A Figura 3.10 ilustra como diferentes combinações de μ e σ^2 afetam a forma e posicionamento da distribuição.

Transformação para Mapeamento Probabilístico

Uma característica distintiva desta estratégia é a necessidade de transformar os valores amostrados da distribuição Normal (domínio R) em probabilidades válidas (domínio $[0, 1]$). A implementação utiliza transformação linear:

A transformação linear aplica uma função afim simples:

$$p = \alpha \cdot \mu + \beta$$

onde:

- α é o fator de escala linear (linear_scale)

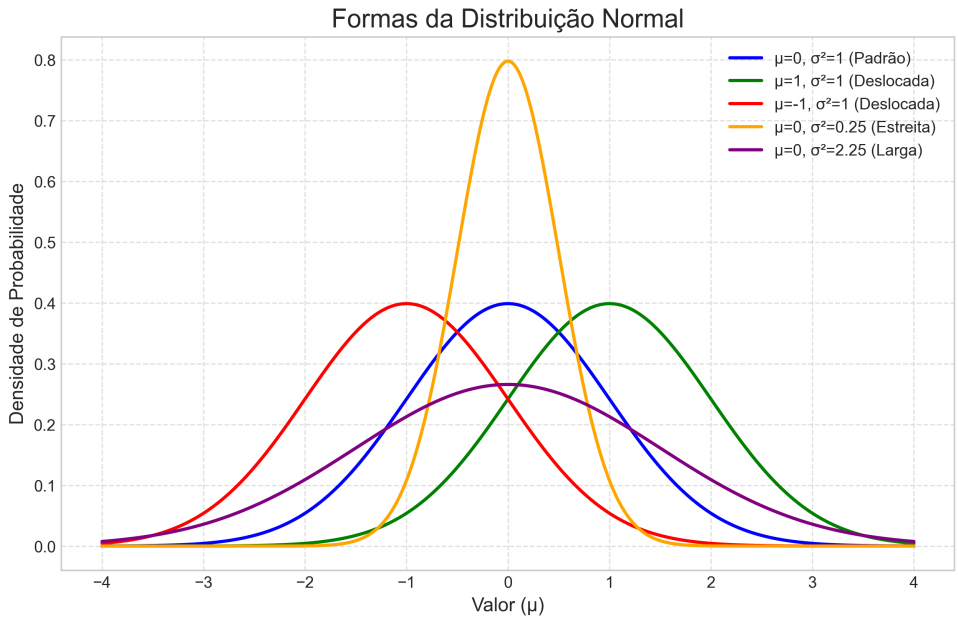

Figura 3.10 – Formas da distribuição Normal para diferentes valores de μ (média) e σ^2 (variância).

- β é o deslocamento (linear_offset)
- p é posteriormente limitado ao intervalo $[0, 1]$ via $p_{final} = \max(0, \min(1, p))$

Esta transformação oferece simplicidade computacional e interpretabilidade direta, sendo a abordagem utilizada na implementação prática do sistema.

Atualização Iterativa dos Parâmetros

Diferentemente das estratégias bayesianas tradicionais que utilizam conjugação analítica, a abordagem Normal emprega atualização iterativa dos parâmetros baseada no histórico temporal ponderado. O processo de atualização segue as regras:

Atualização da Média (μ)

O parâmetro μ é ajustado incrementalmente através de:

$$\mu_{t+1} = \mu_t + w(t) \cdot \eta_{evento} \cdot \text{direção}$$

onde:

- $w(t) = e^{-\lambda \Delta t}$ é o peso temporal do evento
- η_{evento} é a força de atualização (diferente para sucessos/erros)
- direção $= +1$ para sucessos, direção $= -1$ para erros

As forças de atualização são configuráveis:

$$\eta_{sucesso} = \text{update\_strength\_success} \tag{3.1}$$

$$\eta_{erro} = \text{update\_strength\_failure} \tag{3.2}$$

Atualização da Variância (σ^2)

A variância é reduzida progressivamente através de:

$$\sigma^2_{t+1} = \sigma^2_t \cdot \gamma$$

onde γ é o fator de decaimento da variância (variance_decay_factor), tipicamente $\gamma \in (0.99, 1.0)$. Este mecanismo reduz gradualmente a incerteza conforme mais evidências são acumuladas.

Modos de Exploração Configuráveis

A estratégia Normal oferece múltiplos modos operacionais que ajustam automaticamente os parâmetros para diferentes perfis de exploração:

Modo Standard

- $\mu_{inicial} = 0.0$, $\sigma^2_{inicial} = 1.0$
- $\eta_{sucesso} = 0.1$, $\eta_{erro} = 0.3$
- $\gamma = 0.99$
- **Características**: Equilíbrio moderado entre exploração e explotação

Modo Balanced

- $\mu_{inicial} = 0.0$, $\sigma^2_{inicial} = 2.0$
- $\eta_{sucesso} = 0.08$, $\eta_{erro} = 0.25$
- $\gamma = 0.992$
- **Características**: Maior exploração inicial, convergência mais suave

Modo High Exploration

- $\mu_{inicial} = 0.0$, $\sigma^2_{inicial} = 3.0$
- $\eta_{sucesso} = 0.05$, $\eta_{erro} = 0.15$
- $\gamma = 0.995$
- **Características**: Exploração intensa, aprendizado conservador

Modo Extreme Exploration

- $\mu_{inicial} = 0.0$, $\sigma^2_{inicial} = 5.0$
- $\eta_{sucesso} = 0.02$, $\eta_{erro} = 0.08$
- $\gamma = 0.998$
- **Características**: Exploração máxima, convergência muito lenta

Thompson Sampling com Ranking por Maximização

O *Thompson Sampling* na estratégia Normal opera através da amostragem de cada distribuição $\mathcal{N}(\mu_i, \sigma_i^2)$ correspondente ao *proxy* i, seguida pela aplicação da transformação escolhida e seleção do *proxy* que produziu o maior valor transformado:

$$i^* = \arg\max_i \{T(X_i) : X_i \sim \mathcal{N}(\mu_i, \sigma_i^2)\}$$

onde $T(\cdot)$ representa a função de transformação linear.

Algoritmo Detalhado de Seleção

Função de Amostragem e Transformação

Para garantir robustez, implementa-se uma função especializada que combina amostragem e transformação linear:

Algorithm 9 Seleção de Proxy com Distribuição Normal - Versão Detalhada

Função SelecionarProxyNormal(lista_de_proxies, url_alvo, grupo_id)
scores_finais ← []
info_proxies ← []
tempo_atual ← ObterTempoAtual()
for all *proxy* **em** *lista_de_proxies* **do**
    logs_filtrados ← ObterLogsComFiltro(*proxy*, *url_alvo*, MAX_LOGS)
    if len(*logs_filtrados*) < MIN_TENTATIVAS **then** ▷ Usar valores iniciais para proxies novos
        $\mu \leftarrow \mu_{inicial}$
        $\sigma^2 \leftarrow \sigma^2_{inicial}$
    else ▷ Calcular parâmetros baseados no histórico
        $\mu \leftarrow \mu_{inicial}$
        $\sigma^2 \leftarrow \sigma^2_{inicial}$
        ▷ Processar logs ordenados temporalmente
        logs_ordenados ← OrdenarPorTimestamp(*logs_filtrados*)
        for all *log* **em** *logs_ordenados* **do**
            $\Delta t \leftarrow$ *tempo_atual* - *log*.timestamp
            peso_temporal $\leftarrow e^{-\lambda \Delta t}$
            if *peso_temporal* $\geq$ THRESHOLD_DECAY_MINIMO **then**
                if *log*.sucesso **then**
                    $\mu \leftarrow \mu + peso\_temporal \times \eta_{sucesso}$
                else
                    $\mu \leftarrow \mu - peso\_temporal \times \eta_{erro}$
                end if
                ▷ Redução progressiva da variância
                $\sigma^2 \leftarrow \sigma^2 \times \gamma_{decay}$
            end if
        end for
    end if
    ▷ Garantir estabilidade numérica
    $\sigma^2 \leftarrow \max(\sigma^2, 0.01)$
    ▷ Thompson Sampling
    amostra_normal ← AmostraNormal(μ, σ^2)
    ▷ Aplicar transformação linear
    score ← Clip($\alpha \times amostra\_normal + \beta$, 0.0, 1.0)
    prob_esperada ← CalcularProbabilidadeEsperada(μ)
    Adicionar (*proxy*, *score*, μ, σ^2, *prob_esperada*) **a** *info_proxies*
    Adicionar *score* **a** *scores_finais*
end for
▷ Seleção do melhor proxy (MAIOR score)
indice_melhor ← IndiceMaiorValor(*scores_finais*)
proxy_selecionado ← *info_proxies*[*indice_melhor*].proxy
▷ Logging detalhado para análise
LogResultadoSelecaoNormal(*proxy_selecionado*, *info_proxies*)
retornar *proxy_selecionado*

Algorithm 10 Amostragem Normal com Transformação Linear

Função AmostragemNormalTransformada(μ, σ^2)
▷ Garantir estabilidade numérica
$\sigma^2 \leftarrow \max(\sigma^2, 0.01)$
$\sigma \leftarrow \sqrt{\sigma^2}$
▷ Amostragem da distribuição Normal
amostra ← AmostraNormal(μ, σ) ErroNumerico ▷ Fallback: usar valor esperado com ruído controlado
ruido ← RuidoNormal(0, 0.1)
amostra ← μ + *ruido*
▷ Aplicar transformação linear
prob ← $\alpha \times$ *amostra* + β
retornar Clip(*prob*, 0.0, 1.0)

3.1.4.4 Estratégia com Distribuição Gamma

A estratégia baseada na distribuição Gamma utiliza uma abordagem onde erros incrementam o parâmetro de forma e sucessos afetam o parâmetro de taxa, implementando um sistema de ranking invertido. Esta estratégia pode ser eficaz para identificar rapidamente *proxies* com desempenho problemático, penalizando automaticamente recursos com alta taxa de falhas.

Fundamentação Matemática

A distribuição Gamma é uma distribuição contínua de fundamental importância na teoria estatística, caracterizada por sua flexibilidade na modelagem de fenômenos com valores estritamente positivos. Sua função de densidade de probabilidade é expressa por:

$$f(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$$

onde:

- $\alpha > 0$ é o parâmetro de forma (shape parameter)
- $\beta > 0$ é o parâmetro de taxa (rate parameter)
- $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt$ é a função gama
- O domínio é $x \in [0, +\infty)$

Alternativamente, a distribuição pode ser parametrizada usando escala $\theta = 1/\beta$, resultando em:

$$f(x; \alpha, \theta) = \frac{1}{\Gamma(\alpha)\theta^\alpha} x^{\alpha-1} e^{-x/\theta}$$

A valor esperado e variância são dados por:

$$E[X] = \frac{\alpha}{\beta} = \alpha\theta \tag{3.3}$$

$$\text{Var}[X] = \frac{\alpha}{\beta^2} = \alpha\theta^2 \tag{3.4}$$

A Figura 3.11 ilustra como diferentes combinações de α e β produzem formas distributivas distintas.

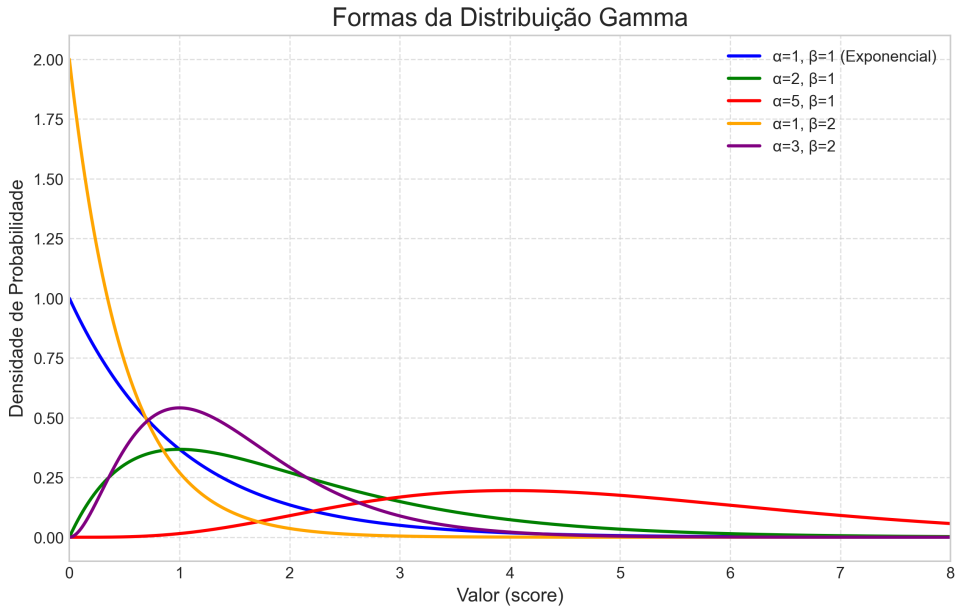

Figura 3.11 – Formas da distribuição Gamma para diferentes valores de α (forma) e β (taxa).

Parametrização Adaptativa com Inversão Conceitual

O aspecto mais distintivo da estratégia Gamma reside na parametrização adaptativa que inverte a interpretação convencional dos eventos de sucesso e erro:

Mapeamento de Eventos para Parâmetros

- **Parâmetro de Forma** (α): Influenciado pelos **erros** ponderados temporalmente
- **Parâmetro de Taxa** (β): Influenciado pelos **sucessos** ponderados temporalmente

Esta inversão conceitual é matematicamente fundamentada pela relação entre os parâmetros e o comportamento da distribuição:

$$\alpha_w = \alpha_{inicial} + \sum_{j \in \text{erros}} e^{-\lambda_e \Delta t_j} \times M_{erro}$$

$$\beta_w = \beta_{inicial} + \sum_{i \in \text{sucessos}} e^{-\lambda_s \Delta t_i} \times M_{sucesso}$$

onde:

- λ_e e λ_s são taxas de decaimento temporal para erros e sucessos
- M_{erro} e $M_{sucesso}$ são multiplicadores de intensidade
- Δt_j e Δt_i são intervalos temporais desde os eventos

Comportamento Resultante

Esta parametrização produz o seguinte comportamento natural:

- **Proxies com muitos erros**: α_w elevado resulta em distribuições com maior dispersão e valores esperados mais altos
- **Proxies com muitos sucessos**: β_w elevado resulta em distribuições concentradas em valores baixos
- **Ranking invertido natural**: Menor score indica melhor desempenho, alinhando-se com a intuição operacional

A Figura 3.12 demonstra quantitativamente como sucessos e erros influenciam os parâmetros e consequentemente a forma da distribuição.

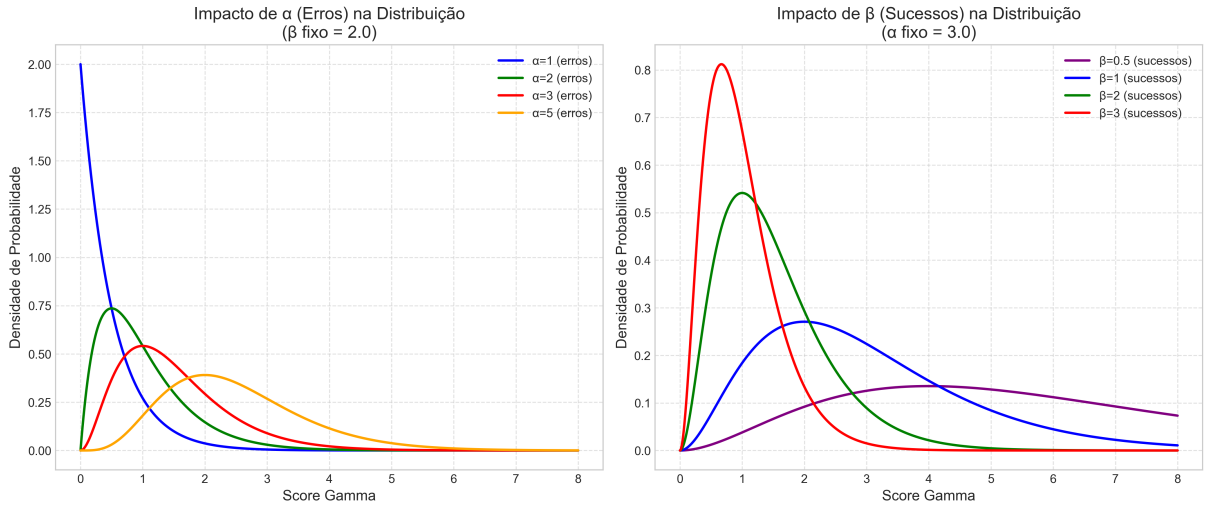

Figura 3.12 – Impacto dos sucessos e erros nos parâmetros α e β da distribuição Gamma.

Sistema de Ranking Invertido Natural

A estratégia Gamma implementa ranking invertido através das propriedades inerentes da distribuição, onde o *proxy* que gera o menor valor amostrado é selecionado. Esta abordagem é matematicamente elegante pois:

Propriedades de Ordenação

Para dois *proxies* com parâmetros (α_1, β_1) e (α_2, β_2):

$$P(X_1 < X_2) = \int_0^{\infty} \int_0^{x_2} f_1(x_1) f_2(x_2) dx_1 dx_2$$

onde *proxies* com melhor histórico (mais sucessos, menos erros) tendem a produzir valores menores devido ao aumento de β e controle de α.

Interpretação como "Tempo até Falha"

A distribuição Gamma possui interpretação natural em teoria de confiabilidade como modelagem de "tempo até falha", onde valores menores indicam maior confiabilidade - alinhamento com os objetivos de seleção de *proxies*.

A Figura 3.13 compara as distribuições de diferentes *proxies* hipotéticos, evidenciando como o histórico influencia as probabilidades de ranking.

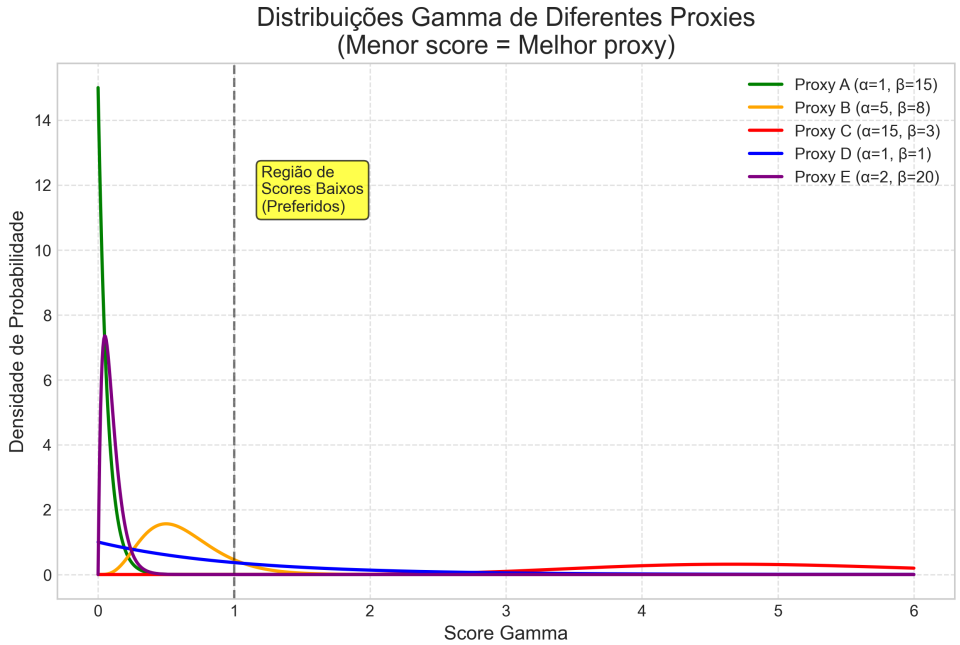

Figura 3.13 – Comparação de distribuições Gamma entre proxies com diferentes históricos de desempenho.

Implementação com Decaimento Temporal

Assim como nas demais estratégias bayesianas, a abordagem Gamma incorpora decaimento temporal exponencial, mas com taxas potencialmente diferenciadas para sucessos e erros:

Ponderação Temporal de Erros

$$\alpha_w = \alpha_{inicial} + \sum_{j \in \text{erros}} w_e(t_j) \times M_{erro} \times B_{base}$$

onde $w_e(t_j) = e^{-\lambda_e \Delta t_j}$ com possibilidade de $\lambda_e \neq \lambda_s$ para controle assimétrico.

Ponderação Temporal de Sucessos

$$\beta_w = \beta_{inicial} + \sum_{i \in \text{sucessos}} w_s(t_i) \times M_{sucesso} \times B_{base}$$

Esta flexibilidade permite ajuste fino do comportamento temporal da estratégia.

Thompson Sampling com Ranking Invertido

O processo de seleção opera através da amostragem de cada distribuição Gamma(α_i, β_i) correspondente ao *proxy* i, seguida pela identificação do menor valor:

$$i^* = \arg\min_i \{X_i : X_i \sim \text{Gamma}(\alpha_i, \beta_i)\}$$

Este processo pode ser formalizado através da função de ranking:

$$R(i) = E[\mathbf{1}_{X_i = \min\{X_1, X_2, \ldots, X_n\}}]$$

Algoritmo Detalhado de Seleção

Função de Amostragem Segura

Para garantir robustez numérica, implementa-se amostragem especializada para a distribuição Gamma:

Algorithm 11 Seleção de Proxy com Distribuição Gamma - Versão Detalhada

Função SelecionarProxyGamma(lista_de_proxies, url_alvo, grupo_id)
scores_finais ← []
info_proxies ← []
tempo_atual ← ObterTempoAtual()
for all *proxy* **em** *lista_de_proxies* **do**
    logs_filtrados ← ObterLogsComFiltro(*proxy*, *url_alvo*, MAX_LOGS)
    if len(*logs_filtrados*) < MIN_TENTATIVAS **then** ▷ Usar valores iniciais para proxies novos
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
    else ▷ Calcular parâmetros baseados no histórico
        $\alpha_w \leftarrow \alpha_{inicial}$
        $\beta_w \leftarrow \beta_{inicial}$
        for all *log* **em** *logs_filtrados* **do**
            $\Delta t \leftarrow$ *tempo_atual* - *log*.timestamp
            ▷ Aplicar delays e filtros temporais
            if não *log*.sucesso **e** $\Delta t \geq$ DELAY_ERRO **then**
                $peso \leftarrow e^{-\lambda_e \Delta t} \times M_{erro} \times BASE\_MULTIPLIER$
                if $peso \geq$ THRESHOLD_DECAY_MINIMO **then**
                    $\alpha_w \leftarrow \alpha_w + peso$ ▷ Erros aumentam α
                end if
            else if *log*.sucesso **e** $\Delta t \geq$ DELAY_SUCESSO **then**
                $peso \leftarrow e^{-\lambda_s \Delta t} \times M_{sucesso} \times BASE\_MULTIPLIER$
                if $peso \geq$ THRESHOLD_DECAY_MINIMO **then**
                    $\beta_w \leftarrow \beta_w + peso$ ▷ Sucessos aumentam β
                end if
            end if
        end for
    end if
    ▷ Garantir estabilidade numérica
    $\alpha_w \leftarrow \max(\alpha_w, 1.0)$
    $\beta_w \leftarrow \max(\beta_w, 1.0)$
    ▷ Thompson Sampling com distribuição Gamma
    score ← AmostraGammaSegura(α_w, β_w)
    valor_esperado ← $\frac{\alpha_w}{\beta_w}$ ▷ E[Gamma(α, β)] =α/β
    Adicionar (*proxy*, *score*, α_w, β_w, *valor_esperado*) **a** *info_proxies*
    Adicionar *score* **a** *scores_finais*
end for
▷ Seleção do melhor proxy (MENOR score)
indice_melhor ← IndiceMenorValor(*scores_finais*)
proxy_selecionado ← *info_proxies*[*indice_melhor*].proxy
▷ Logging detalhado para análise
LogResultadoSelecaoGamma(*proxy_selecionado*, *info_proxies*)
retornar *proxy_selecionado*

Algorithm 12 Amostragem Gamma Segura

Função AmostraGammaSegura(α, β)
▷ Garantir valores mínimos
$\alpha \leftarrow \max(\alpha, 1.0)$
$\beta \leftarrow \max(\beta, 1.0)$
▷ Detectar casos extremos
if $\alpha > 1000$ **ou** $\beta > 1000$ **then** ▷ Caso extremo: usar aproximação normal
    $media \leftarrow \frac{\alpha}{\beta}$
    $variancia \leftarrow \frac{\alpha}{\beta^2}$
    $sample \leftarrow$ AmostraNormal($media$, $variancia$)
    retornar $\max(sample, 0.001)$
else if $\alpha < 0.1$ **ou** $\beta < 0.1$ **then** ▷ Caso extremo: valores muito baixos
    retornar RuidoExponencial(1.0)
else ▷ Caso normal: usar distribuição Gamma padrão
    retornar AmostraGamma(α, β) ErroNumerico ▷ Fallback: usar valor esperado com ruído
    $esperado \leftarrow \frac{\alpha}{\beta}$
    $ruido \leftarrow$ RuidoExponencial(0.1)
    retornar $esperado + ruido$
end if

3.1.4.5 Análise Comparativa das Estratégias Bayesianas

As quatro estratégias bayesianas implementadas neste trabalho — Beta, Chi-Quadrado, Normal e Gamma — compartilham fundamentos teóricos comuns, mas diferem significativamente em suas características operacionais, complexidade de implementação e adequação a diferentes cenários. Esta seção apresenta uma análise comparativa detalhada, considerando aspectos matemáticos, computacionais e práticos.

Complexidade Matemática e Interpretabilidade

A **distribuição Beta** destaca-se pela simplicidade conceitual e interpretabilidade direta de seus parâmetros. Os valores α e β possuem significado intuitivo como contadores de sucessos e falhas, respectivamente, facilitando tanto a implementação quanto o diagnóstico do comportamento do sistema. Esta clareza paramétrica é particularmente valiosa em ambientes operacionais onde a transparência dos algoritmos é fundamental.

A **estratégia Normal** oferece flexibilidade através de múltiplos modos de exploração e transformação linear para mapeamento probabilístico. Contudo, a interpretação dos parâmetros μ e σ^2 requer conhecimento mais aprofundado do comportamento da distribuição, especialmente considerando que o domínio natural $(-\infty, +\infty)$ necessita transformação para o espaço de probabilidades.

As estratégias **Chi-Quadrado** e **Gamma** implementam conceitos mais sofisticados, incluindo ranking invertido. Embora matematicamente elegantes, estas abordagens intro-

duzem complexidade adicional na interpretação dos resultados, particularmente devido ao mapeamento contra-intuitivo entre eventos e parâmetros.

Estabilidade Numérica e Robustez

A **distribuição Beta** demonstra excelente estabilidade numérica devido ao seu domínio limitado $[0, 1]$ e propriedades de conjugação bayesiana. A amostragem é computacionalmente eficiente e raramente apresenta problemas de overflow ou underflow, mesmo com parâmetros de magnitude elevada.

A **estratégia Normal** mantém boa estabilidade através de mecanismos de clipping e tratamento de casos extremos, embora requeira atenção especial para evitar valores muito grandes ou pequenos durante a transformação linear.

As estratégias **Chi-Quadrado** e **Gamma** necessitam de implementações mais cuidadosas para casos extremos, particularmente quando os parâmetros assumem valores muito baixos ou muito altos, requerendo fallbacks baseados em aproximações.

Configurabilidade e Manutenção

A **distribuição Beta** oferece o menor número de hiperparâmetros para configuração, reduzindo a complexidade de ajuste e manutenção do sistema. Os parâmetros iniciais $\alpha_{inicial}$ e $\beta_{inicial}$, juntamente com as taxas de decaimento, fornecem controle adequado sem sobrecarregar o operador.

A **estratégia Normal** requer configuração mais detalhada através de seus múltiplos modos de exploração e parâmetros de transformação linear, oferecendo maior flexibilidade ao custo de complexidade adicional.

As estratégias **Chi-Quadrado** e **Gamma** introduzem parâmetros adicionais específicos (como cálculo de graus de liberdade e multiplicadores de intensidade) que podem complicar o processo de configuração e otimização.

3.2 Simulações

A fim de avaliar o comportamento das diferentes estratégias de seleção de *proxies* sob condições operacionais diversas, foram conduzidas simulações controladas. Elas permitem:

- **Comparabilidade justa**: todas as estratégias são avaliadas sob a mesma carga, mesma janela temporal e mesmas regras de produção de eventos, reduzindo vieses de comparação.

- **Reprodutibilidade científica**: os parâmetros e cenários são versionados, possibilitando a repetição exata dos experimentos e a verificação independente dos resultados.
- **Cobertura de condições adversas**: cenários de indisponibilidade intermitente, *rate limiting*, falhas permanentes e heterogeneidade de qualidade expõem pontos fortes e limitações de cada modelo.
- **Análise do equilíbrio exploração–explotação**: observa-se, ao longo do tempo, como cada estratégia aprende, se adapta e redistribui tráfego diante de incerteza e mudança de contexto.
- **Mensuração objetiva de métricas**: taxa de sucesso, incidência de bloqueios e estabilidade (variância) são aferidas de forma consistente entre cenários.
- **Avaliação de escalabilidade**: a execução paralela com múltiplos *workers* por estratégia permite observar efeitos de contenção, saturação e eficiência de alocação sob maior concorrência.

3.2.1 Configuração do Ambiente de Simulação

Foram conduzidas simulações controladas para avaliar as estratégias de seleção de *proxies* em condições diversas e reprodutíveis. A configuração base foi definida conforme o arquivo de configuração de simulações, com os seguintes parâmetros principais:

- **Duração de cada simulação**: 1440 minutos (24 horas)
- **Atraso base por requisição (rede simulada)**: 300-800 ms
- **Pool de proxies por estratégia**: 10 proxies (cada estratégia opera com seu próprio conjunto)
- **Estratégias executadas em paralelo**: Round Robin, Aleatória (Random), Exponential Backoff, Bayesiana com Distribuição Beta (Thompson Sampling), Bayesiana com Distribuição Qui-Quadrado (Thompson Sampling), Bayesiana com Distribuição Normal (Thompson Sampling), Bayesiana com Distribuição Gamma (Thompson Sampling)
- **Workers por estratégia**: 3 (execuções paralelas por estratégia)
- **Intervalo entre requisições**: 1–2 segundos
- **Relatórios e logs**: geração de relatórios habilitada; *logs* detalhados e persistência de resultados ativados

Essa configuração padronizada permite comparar as estratégias sob a mesma carga e condições operacionais, controlando a variabilidade e facilitando análises quantitativas de taxa de sucesso e resiliência.

3.2.2 Métricas e Critérios de Avaliação

As métricas primárias consideradas foram: **taxa de sucesso** (proporção de respostas válidas), **incidência de bloqueios** (eventos de erro associados a restrições do alvo) e **estabilidade** (variância temporal da taxa de sucesso).

3.2.3 Desenho Experimental e Reprodutibilidade

Cada cenário foi executado de forma isolada sob a configuração base, com amostragem temporal suficiente para capturar fases de adaptação e regime estacionário. As execuções foram inicializadas com sementes pseudoaleatórias controladas para permitir reexecuções equivalentes. Dados e *logs* foram coletados de forma sistemática para posterior análise estatística.

3.2.4 Cenários Simulados

Todos os cenários definidos em configuração foram executados ao longo da campanha de testes (com ativações independentes por rodada), de modo a cobrir diferentes classes de falhas e comportamentos do ambiente. A seguir, é descrito o propósito de avaliação e a motivação prática de cada cenário.

3.2.4.1 Cenário 1: Proxies Intermitentes

Proxies podem tornar-se indisponíveis por intervalos finitos e, posteriormente, retornar à operação. Este cenário avalia a robustez do sistema diante de indisponibilidades temporárias e a capacidade de recuperação sem degradação prolongada do desempenho. Na prática, tais indisponibilidades decorrem de manutenção de provedores, reinícios de *gateways*, quedas regionalizadas de ISP ou limites temporários de uso em serviços de proxy.

Parâmetros:

- Duração típica de indisponibilidade: 30 minutos
- Percentual inicial de proxies disponíveis: 100%
- Intervalo para novas indisponibilidades: 25 minutos

3.2.4.2 Cenário 2: Proxies Bloqueados por Requisições

Ao exceder um limiar de requisições, proxies podem ser temporariamente bloqueados pelo alvo, simulando *rate limiting* e defesas anti-*bot*. Este cenário observa a dinâmica do sistema sob restrições de taxa e a eficiência do processo após bloqueios temporários. Na prática, serviços protegidos por WAF/CDN (p. ex., Cloudflare, Akamai) aplicam *rate limiting* por IP/ASN; APIs também impõem *throttling* após picos de acesso.

Parâmetros:

- Limiar de bloqueio por IP: 100 requisições
- Duração do bloqueio: 20 minutos
- Recuperação automática: habilitada

3.2.4.3 Cenário 3: Proxies Permanentemente Falhos

Uma fração do *pool* permanece indisponível desde o início e durante toda a execução. O objetivo é avaliar o comportamento do sistema em *pools* degradados e a velocidade de convergência para ignorar recursos consistentemente ruins. Na prática, falhas permanentes resultam de credenciais expiradas, IPs em listas negras, portas fechadas, autenticação mal configurada ou rotas quebradas no provedor.

Parâmetros:

- Quantidade absoluta de falhos: 3 proxies

3.2.4.4 Cenário 4: Proxies com Probabilidades de Sucesso Heterogêneas

O *pool* contém proxies com distribuições de sucesso distintas, representando um ambiente heterogêneo. Este cenário verifica a capacidade de priorização de recursos mais confiáveis ao longo do tempo, preservando a exploração suficiente para captar mudanças no ambiente. Na prática, a heterogeneidade decorre de reputação de IP/ASN (datacenter vs. residencial/móvel), geolocalização e restrições regionais, além de políticas do alvo que discriminam determinados provedores.

Parâmetros:

- Grupos típicos de probabilidade: alta (0,85–0,95), média (0,60–0,85) e baixa (0,10–0,20)
- Exemplo de contagens por grupo: 4 (alta), 1 (média), 5 (baixa)

3.2.5 Procedimento de Execução

Cada cenário foi executado isoladamente sob a configuração base, mantendo constantes os parâmetros de carga (intervalos entre requisições, paralelismo e tempo de simulação). Os resultados foram coletados com *logs* detalhados e relatórios consolidados para análise comparativa entre cenários, com controle de reprodutibilidade por semente pseudoaleatória e versionamento de parâmetros.

3.3 Validação em Ambiente Real

Complementando as simulações controladas, foi conduzida uma campanha de validação em ambiente operacional real para avaliar o comportamento das estratégias de seleção de *proxies* sob condições autênticas de produção. Esta validação permite:

- **Verificação de aplicabilidade prática**: confirma se os resultados das simulações se mantêm em cenários reais com variabilidade não controlada e complexidade operacional inerente.
- **Exposição a condições imprevistas**: eventos não modelados nas simulações, como instabilidades de rede, políticas dinâmicas de *rate limiting* e comportamentos emergentes de sistemas distribuídos.
- **Avaliação de robustez operacional**: observa-se a estabilidade das estratégias diante de cargas heterogêneas, alvos diversos e condições de infraestrutura variáveis.
- **Validação de escalabilidade**: confirma o comportamento das estratégias em sistemas com múltiplos *workers* concorrentes e pools de *proxies* compartilhados.
- **Análise de adaptabilidade temporal**: verifica a capacidade de adaptação das estratégias ao longo de períodos estendidos, capturando variações circadianas e sazonais.

3.3.1 Configuração do Ambiente Real

A validação foi realizada em um ambiente de produção composto por sistemas automatizados de captura de dados, operando continuamente durante um período de 7 dias. A configuração operacional apresentou as seguintes características:

- **Duração total da validação**: 168 horas (7 dias consecutivos)
- **Número de sistemas automatizados**: 10 unidades independentes
- **Pool de *proxies* por sistema**: 10 *proxies* selecionados aleatoriamente de um conjunto de 88 *proxies* disponíveis

- ***Workers* por sistema**: variável entre 4 e 8, ajustado dinamicamente conforme demanda operacional
- **Estratégias avaliadas**: distribuídas entre os 10 sistemas, incluindo Round Robin, Aleatória, Exponential Backoff e variantes Bayesianas
- **Alvos de requisições**: sistemas públicos diversos com políticas de acesso heterogêneas
- **Coleta de dados**: *logs* operacionais detalhados e métricas de desempenho em tempo real

3.3.2 Características Metodológicas do Ambiente Real

Diferentemente das simulações controladas, a validação em ambiente real apresenta características metodológicas específicas que refletem a complexidade operacional:

3.3.2.1 Heterogeneidade Inerente

O ambiente real introduz variabilidade não controlada em múltiplas dimensões:

- **Diversidade de *pools***: cada sistema opera com um conjunto de 10 *proxies* selecionados aleatoriamente, resultando em composições distintas com características de desempenho variáveis.
- **Heterogeneidade de alvos**: diferentes sistemas acessam alvos com políticas de *rate limiting*, mecanismos de detecção e tolerâncias distintas.
- **Variabilidade temporal**: condições de rede, carga dos alvos e disponibilidade dos *proxies* variam ao longo do período de observação.
- **Concorrência assimétrica**: o número variável de *workers* por sistema (4–8) introduz níveis diferentes de contenção e paralelismo.

3.3.2.2 Impossibilidade de Controle Experimental

As condições reais impõem limitações metodológicas fundamentais:

- **Não equivalência de requisições**: diferentemente das simulações, não é possível garantir que todas as estratégias sejam expostas a requisições idênticas ou alvos equivalentes.
- **Variabilidade de carga**: cada sistema opera sob demanda específica, com padrões de requisições determinados por necessidades operacionais reais.

- **Condições de infraestrutura**: latência de rede, estabilidade de conectividade e desempenho dos *proxies* variam independentemente entre sistemas.
- **Políticas dinâmicas**: alvos podem alterar suas políticas de acesso durante o período de observação, afetando diferentemente cada estratégia.

3.3.3 Procedimento de Validação

A validação seguiu um protocolo de implantação gradual para minimizar riscos operacionais:

- **Distribuição das estratégias**: cada uma das estratégias foi atribuída a um subconjunto dos 10 sistemas, assegurando representatividade estatística.
- **Monitoramento contínuo**: coleta automatizada de métricas de desempenho, incluindo taxa de sucesso, latência média e incidência de bloqueios.
- **Coleta de *logs***: registro detalhado de eventos, decisões de seleção e resultados de requisições para análise posterior.

3.3.4 Limitações e Considerações Metodológicas

A validação em ambiente real, embora essencial para confirmar a aplicabilidade prática, apresenta limitações metodológicas que devem ser consideradas na interpretação dos resultados:

- **Ausência de grupo de controle puro**: a impossibilidade de isolar completamente as variáveis limita a capacidade de atribuição causal direta.
- **Variabilidade não quantificada**: fatores externos não mensuráveis podem influenciar os resultados de forma não uniforme entre estratégias.
- **Viés de seleção de *proxies***: a escolha aleatória dos *pools* pode favorecer inadvertidamente certas estratégias.

Essas limitações são inerentes à validação em ambiente real e devem ser consideradas complementares aos resultados controlados das simulações, fornecendo uma perspectiva abrangente sobre o comportamento das estratégias em condições operacionais autênticas.

4 Resultados e Análises

Este capítulo apresenta os resultados obtidos nas simulações controladas e na validação em ambiente real, fornecendo uma análise comparativa das estratégias de seleção de *proxies* sob diferentes condições operacionais.

4.1 Resultados das Simulações Controladas

As simulações controladas foram conduzidas para avaliar o comportamento das estratégias em quatro cenários distintos, cada um representando condições operacionais específicas encontradas em ambientes reais. Os resultados demonstram diferenças significativas no desempenho das estratégias, evidenciando a importância da seleção adequada conforme as características do ambiente operacional.

4.1.1 Visão Geral dos Resultados

A Tabela 4.1 apresenta um resumo consolidado dos resultados obtidos nos quatro cenários simulados, destacando as métricas principais para cada estratégia.

Tabela 4.1 – Resumo dos Resultados por Cenário e Estratégia (Taxa de Sucesso Final)

Estratégia	**Intermitente** (%)	**Bloqueados** (%)	**Falhos** (%)	**Heterogêneo** (%)
Bayesiana Beta	99,66	85,41	99,92	84,81
Bayesiana Normal	97,14	77,08	99,69	85,07
Bayesiana Gamma	99,09	84,28	99,69	84,21
Bayesiana Chi-Quadrado	92,67	70,07	99,01	78,26
Exponential Backoff	98,07	91,85	99,64	76,71
Round Robin	87,77	66,75	69,40	54,67
Aleatória (Random)	88,31	57,63	69,62	56,84

4.1.2 Análise por Cenário

4.1.2.1 Metodologia de Análise de Convergência

Para avaliar a capacidade de adaptação das estratégias ao longo do tempo, foi desenvolvido um critério específico de convergência que considera as características dinâmicas de cada cenário simulado. A metodologia adotada fundamenta-se na análise temporal detalhada do comportamento das estratégias, utilizando agregação de dados em intervalos regulares de 5 minutos para capturar tanto variações de curto prazo quanto tendências de longo prazo.

A partir desta granularidade temporal de 5 minutos, são calculadas as taxas de sucesso instantâneas para cada estratégia, representando a proporção de requisições bem-sucedidas dentro de cada intervalo específico. A taxa máxima de sucesso de uma estratégia corresponde ao maior valor observado entre todas essas medições de 5 minutos ao longo de toda a simulação. Esta abordagem permite identificar o pico de desempenho real que cada estratégia conseguiu alcançar em suas condições mais favoráveis durante o experimento.

Uma estratégia é considerada convergente quando sua média móvel de 15 minutos atinge 90% da maior taxa de sucesso observada durante toda a simulação para aquela estratégia específica. A utilização de uma média móvel de 15 minutos, equivalente a 3 períodos consecutivos de 5 minutos, tem como objetivo suavizar oscilações temporárias mantendo sensibilidade suficiente para detectar mudanças significativas no comportamento das estratégias.

O limiar de convergência, estabelecido em 90% da maior taxa observada, é calculado individualmente para cada estratégia, reconhecendo que diferentes abordagens algorítmicas podem apresentar potenciais máximos distintos devido às suas características intrínsecas. Esta personalização do critério permite comparação justa entre estratégias, evitando penalizar aquelas que, por limitações algorítmicas, não conseguem atingir taxas de sucesso próximas a 100%.

Para os cenários de **Proxies Intermitentes** e **Proxies Bloqueados**, a análise de convergência inicia-se apenas após a ocorrência do primeiro erro registrado. Esta abordagem metodológica reconhece que no período inicial, antes da manifestação dos primeiros problemas, todas as estratégias tendem a apresentar desempenho similar e próximo ao ideal. A capacidade real de adaptação só pode ser adequadamente avaliada após a manifestação das condições adversas específicas de cada cenário, quando as estratégias precisam demonstrar suas habilidades de recuperação e ajuste comportamental.

O tempo de convergência, medido em minutos a partir do início da análise (seja desde o início da simulação ou desde o primeiro erro, conforme o cenário), reflete diretamente a velocidade de adaptação de cada estratégia às condições operacionais. Estratégias que nunca atingem o critério estabelecido são classificadas como não convergentes, indicando incapacidade de adaptação efetiva às condições específicas do cenário avaliado.

O limiar de 90% foi estabelecido após análise empírica dos dados coletados, representando um equilíbrio cuidadoso entre rigor metodológico suficiente para distinguir estratégias verdadeiramente eficazes, tolerância às variações naturais inerentes aos sistemas distribuídos, e sensibilidade adequada para capturar diferenças algorítmicas sutis mas significativas entre as diferentes abordagens avaliadas.

Como exemplo prático da aplicação deste critério, considere uma estratégia Bayesiana Beta que, ao longo de uma simulação, atinge sua taxa máxima de sucesso de 100% em

determinado intervalo de 5 minutos. A convergência desta estratégia será identificada no momento em que sua média móvel de 15 minutos sustentar uma taxa de 90% ou superior. Em contraste, uma estratégia que apresente taxa máxima de 80% terá seu limiar de convergência estabelecido em 72%. Esta personalização metodológica assegura comparação equitativa entre estratégias com diferentes capacidades máximas de desempenho.

4.1.2.2 Cenário 1: Proxies Intermitentes

No cenário de *proxies* intermitentes, onde recursos tornam-se temporariamente indisponíveis, as estratégias Bayesianas demonstraram superioridade clara, com destaque para a distribuição Beta (99,66% de taxa de sucesso) e Gamma (99,09%). A Figura 4.1 apresenta uma visão comparativa completa de todas as estratégias avaliadas, evidenciando a clara superioridade das abordagens Bayesianas, com destaque visual diferenciado (linhas mais grossas e marcadores quadrados) para facilitar a identificação das estratégias adaptativas.

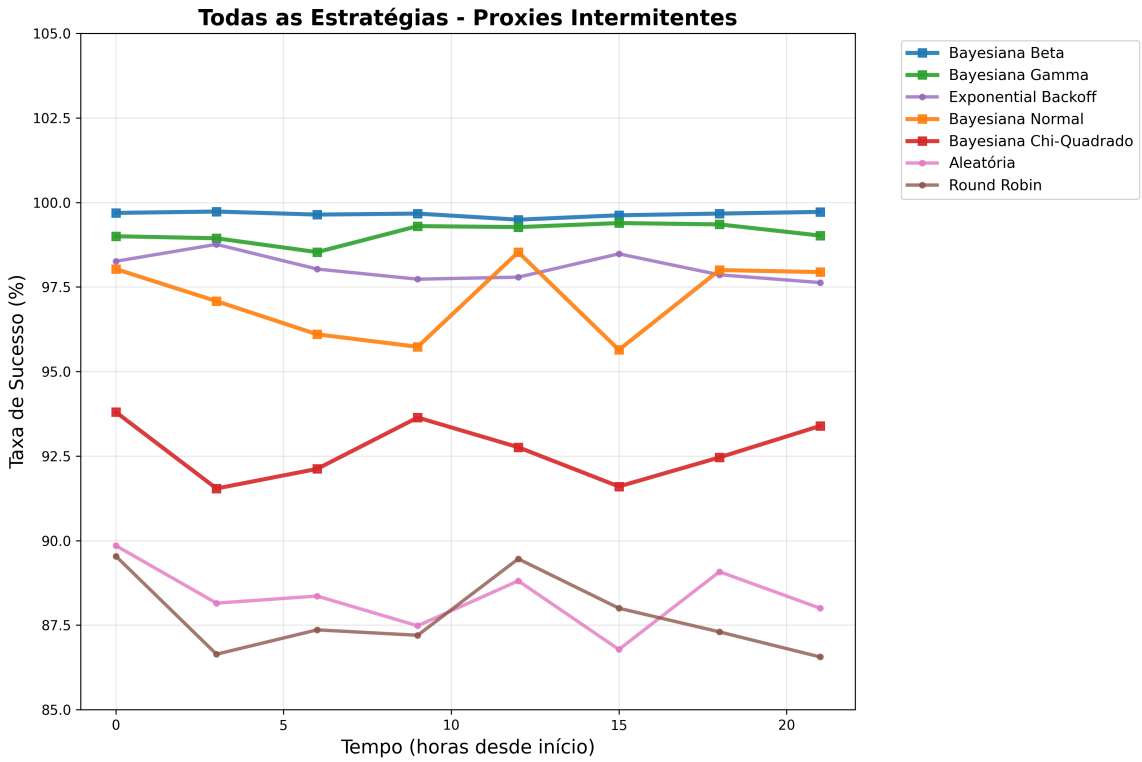

Figura 4.1 – Comparação completa entre todas as estratégias no cenário de Proxies Intermitentes. As estratégias Bayesianas (destacadas com linhas mais grossas e marcadores quadrados) demonstram adaptação superior às intermitências, mantendo taxas elevadas após períodos iniciais de aprendizado.

As estratégias Bayesianas mostraram capacidade superior de adaptação às mudanças de disponibilidade, mantendo alta taxa de sucesso mesmo durante períodos de instabilidade. O Exponential Backoff apresentou desempenho intermediário (98,07%), enquanto as

estratégias básicas (Round Robin e Aleatória) obtiveram as menores taxas de sucesso (87,77% e 88,31%, respectivamente).

O tempo de convergência para este cenário foi relativamente rápido para a maioria das estratégias, conforme ilustrado na Figura 4.2. As estratégias Gamma e Chi-Quadrado convergiram em 35 minutos após o primeiro erro, demonstrando adaptação eficiente às intermitências, enquanto a estratégia Beta necessitou de 80 minutos, possivelmente devido à sua maior cautela na atualização de parâmetros após perturbações.

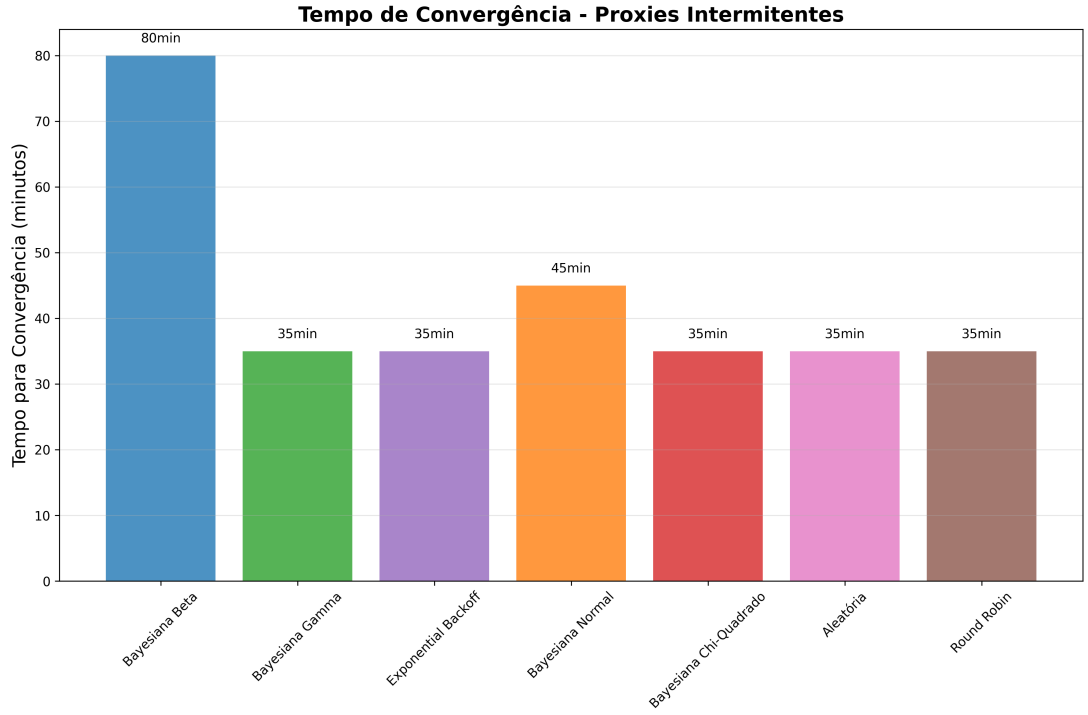

Figura 4.2 – Tempo de convergência no cenário de Proxies Intermitentes. Critério: média móvel de 15 minutos atinge 90% da maior taxa observada, iniciando análise após primeiro erro.

A análise de erros acumulados, apresentada na Figura 4.3, demonstra a eficiência superior das estratégias Bayesianas em minimizar tentativas desnecessárias em recursos problemáticos. A Bayesiana Beta acumulou apenas 60 erros em 24 horas, contrastando dramaticamente com os 3.234 erros da estratégia Aleatória no mesmo período, evidenciando a importância dos mecanismos adaptativos para eficiência operacional.

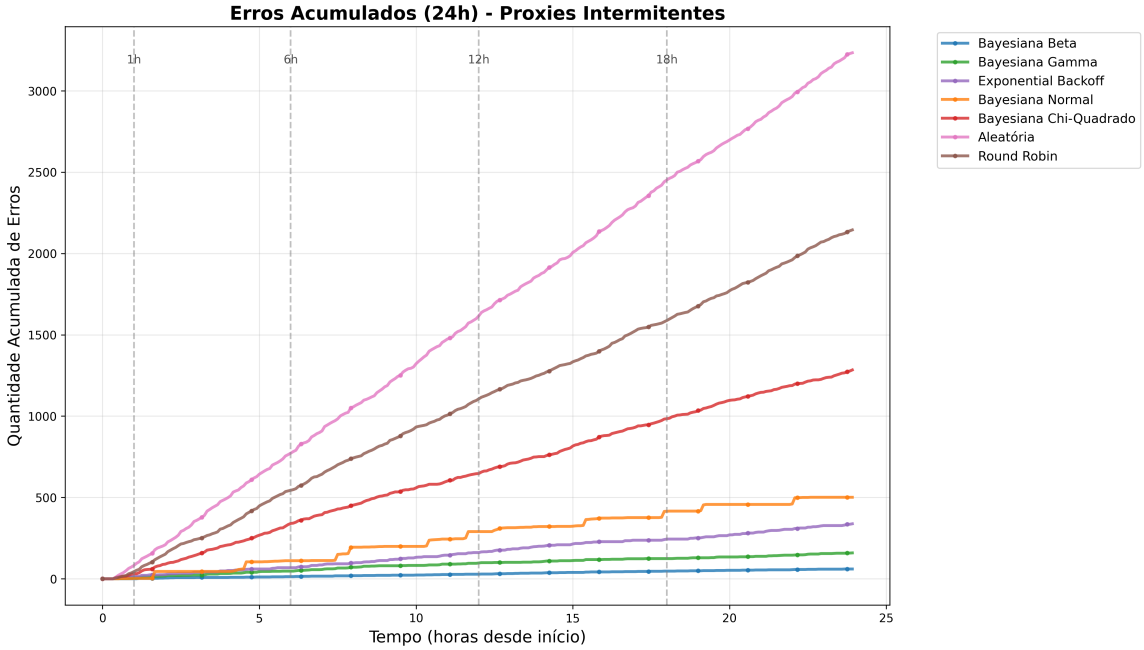

Figura 4.3 – Erros acumulados ao longo de 24 horas no cenário de Proxies Intermitentes. As estratégias Bayesianas (Beta, Gamma) demonstram eficiência superior na minimização de erros.

4.1.2.3 Cenário 2: Proxies Bloqueados por Requisições

Este cenário, que simula *rate limiting* e bloqueios temporários, revelou o desempenho excepcional do Exponential Backoff (91,85%), superando todas as estratégias Bayesianas. A Bayesiana Beta obteve o segundo melhor resultado (85,41%), seguida pela Gamma (84,28%). A Figura 4.4 oferece uma visão comparativa completa, destacando o desempenho excepcional do Exponential Backoff neste cenário específico.

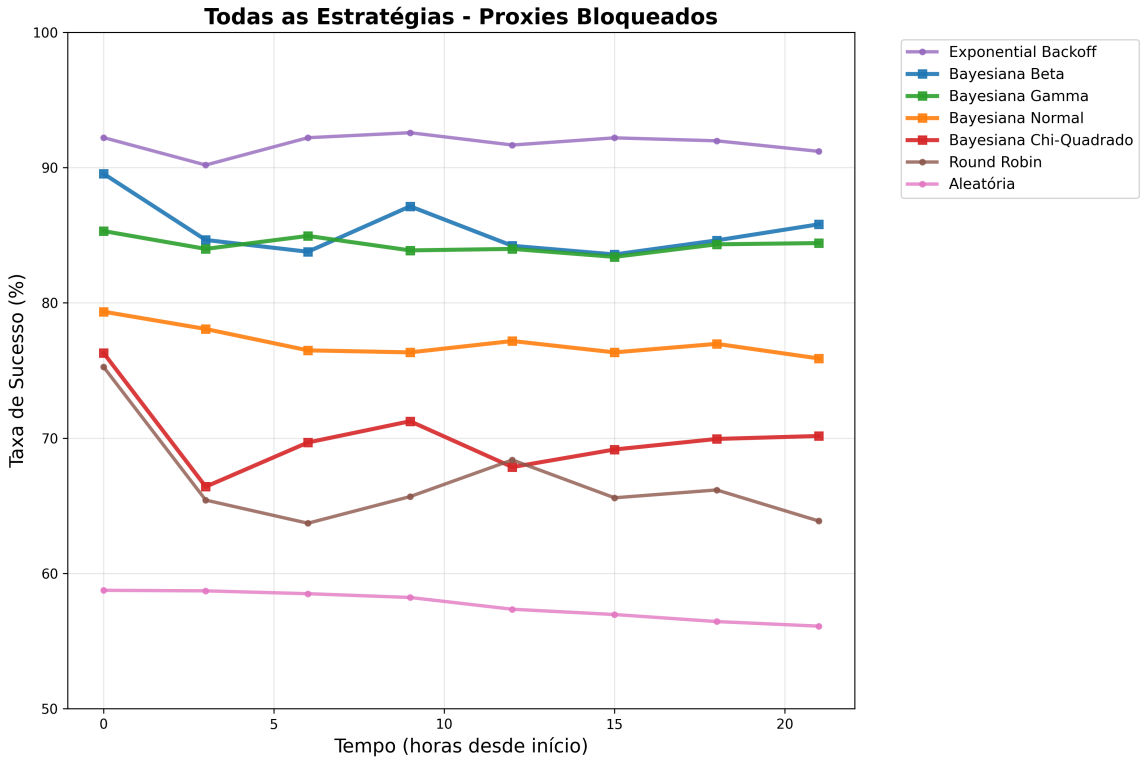

Figura 4.4 – Comparação completa entre todas as estratégias no cenário de Proxies Bloqueados. O Exponential Backoff (destacado) supera significativamente as demais estratégias, demonstrando sua eficácia especializada na gestão de rate limiting.

A análise de convergência, ilustrada na Figura 4.5, revela tempos de convergência variados que refletem diferentes capacidades de gestão de bloqueios entre as estratégias. A Bayesiana Gamma demonstrou adaptação mais rápida (30 minutos), seguida pela Beta (35 minutos), enquanto o Exponential Backoff e Round Robin apresentaram convergência mais lenta (135 minutos), sugerindo que a complexidade dos bloqueios temporários demanda períodos mais extensos para otimização de parâmetros determinísticos.

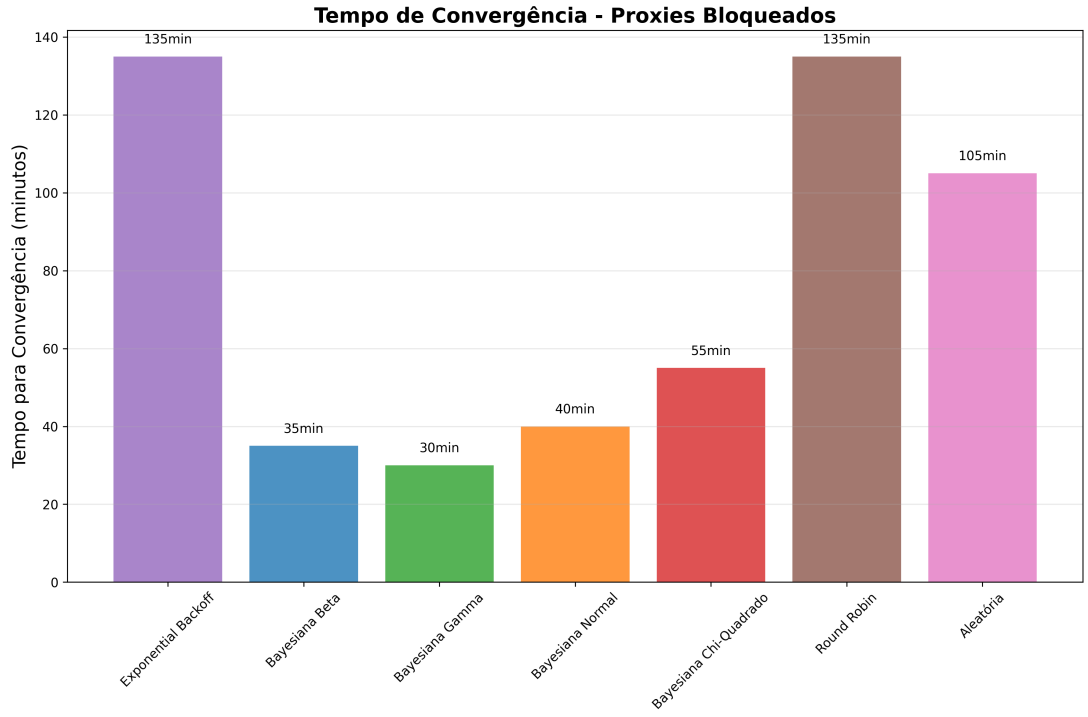

Figura 4.5 – Tempo de convergência no cenário de Proxies Bloqueados. Observa-se grande variação nos tempos de adaptação entre estratégias, refletindo diferentes capacidades de gestão de bloqueios.

O impacto em erros acumulados, evidenciado na Figura 4.6, confirma a eficiência excepcional do Exponential Backoff neste cenário específico. A estratégia manteve apenas 691 erros acumulados, validando sua superioridade em ambientes com rate limiting ativo, enquanto a estratégia Aleatória acumulou 11.733 erros, demonstrando o custo operacional da ausência de mecanismos especializados para gestão de bloqueios.

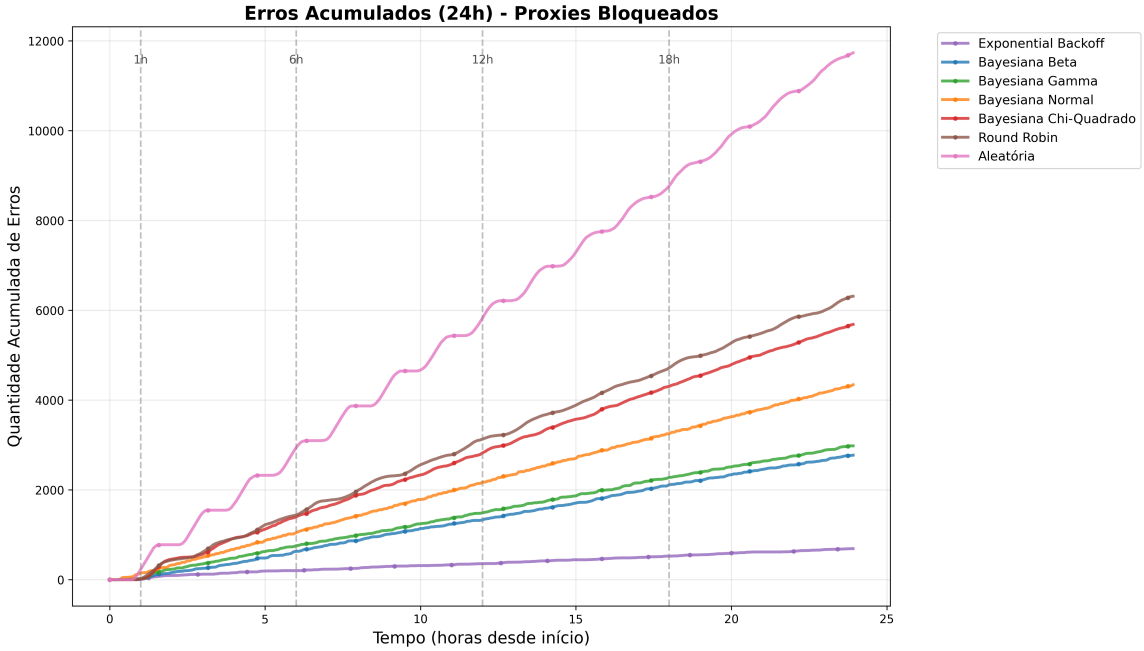

Figura 4.6 – Erros acumulados no cenário de Proxies Bloqueados. O Exponential Backoff demonstra eficiência excepcional na minimização de erros, validando sua superioridade neste cenário específico.

4.1.2.4 Cenário 3: Proxies Permanentemente Falhos

No cenário com *proxies* permanentemente indisponíveis, as estratégias Bayesianas dominaram completamente, com a distribuição Beta alcançando resultados próximos de 100,0% de taxa de sucesso. A Figura 4.7 apresenta uma comparação completa que evidencia dramaticamente a diferença entre estratégias adaptativas e básicas.

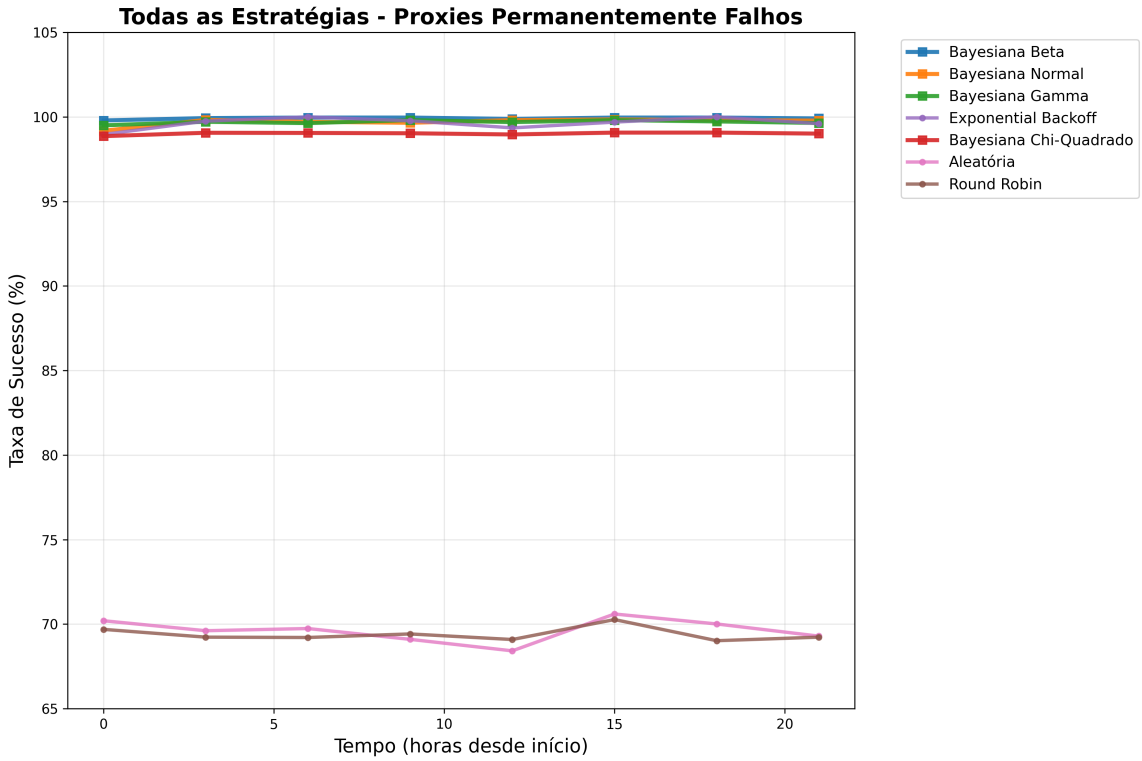

Figura 4.7 – Comparação completa no cenário de Proxies Permanentemente Falhos. As estratégias Bayesianas e Exponential Backoff convergem rapidamente para taxas próximas a 100%, enquanto as estratégias básicas permanecem estagnadas em aproximadamente 70%.

Este cenário evidenciou a capacidade de aprendizado das estratégias Bayesianas, que rapidamente identificaram e evitaram *proxies* consistentemente falhos. O contraste com as estratégias básicas foi dramático: Round Robin (69,3%) e Aleatória (69,1%) continuaram desperdiçando recursos em *proxies* inoperantes.

A análise temporal de convergência, apresentada na Figura 4.8, revela convergência extremamente rápida e uniforme para as estratégias adaptativas (10 minutos), demonstrando eficiência equivalente na identificação de recursos consistentemente falhos. A estratégia Aleatória convergiu em 40 minutos, evidenciando não uma limitação de aprendizado (que não possui), mas sim o tempo estatisticamente necessário para que a seleção probabilística aleatória concentre-se predominantemente nos recursos funcionais disponíveis.

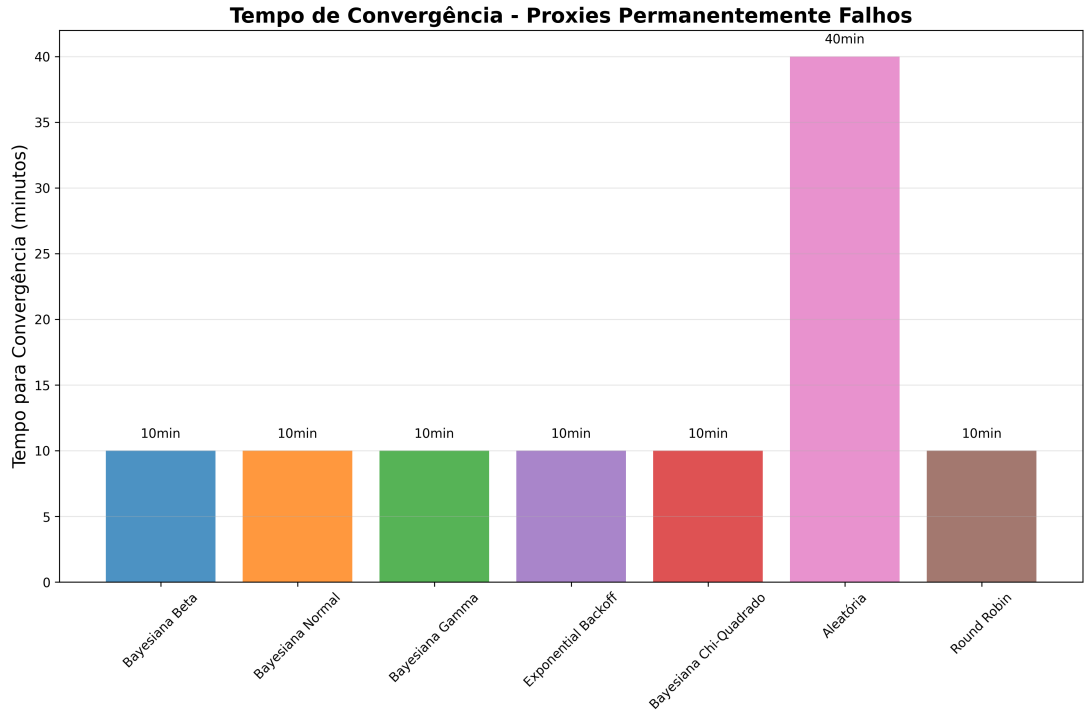

Figura 4.8 – Tempo de convergência no cenário de Proxies Permanentemente Falhos. Estratégias Bayesianas e Exponential Backoff demonstram aprendizado rápido e uniforme.

Uma análise granular dos primeiros 60 minutos, período crítico para identificação de recursos falhos, é apresentada na Figura 4.9. Este intervalo temporal revela como as diferentes estratégias se comportam durante a fase inicial de descoberta de padrões, onde a velocidade de identificação de recursos problemáticos impacta diretamente a eficiência subsequente do sistema.

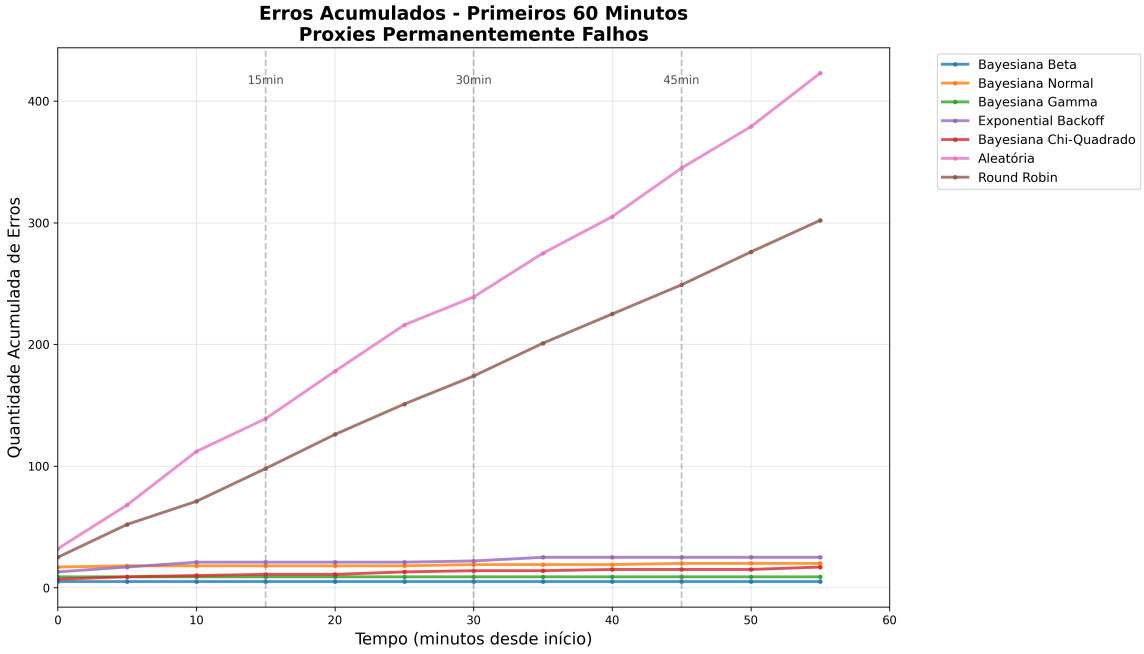

Figura 4.9 – Erros acumulados nos primeiros 60 minutos - Proxies Permanentemente Falhos. Estratégias Bayesianas minimizam rapidamente tentativas em recursos falhos.

O impacto cumulativo das diferentes abordagens algorítmicas ao longo de 24 horas, demonstrado na Figura 4.10, evidencia diferenças dramáticas na eficiência operacional. A Bayesiana Beta acumulou apenas 17 erros durante todo o período, enquanto a estratégia Aleatória atingiu 10.663 erros, uma diferença de mais de 600 vezes que ilustra claramente o valor dos mecanismos adaptativos em cenários com recursos consistentemente problemáticos.

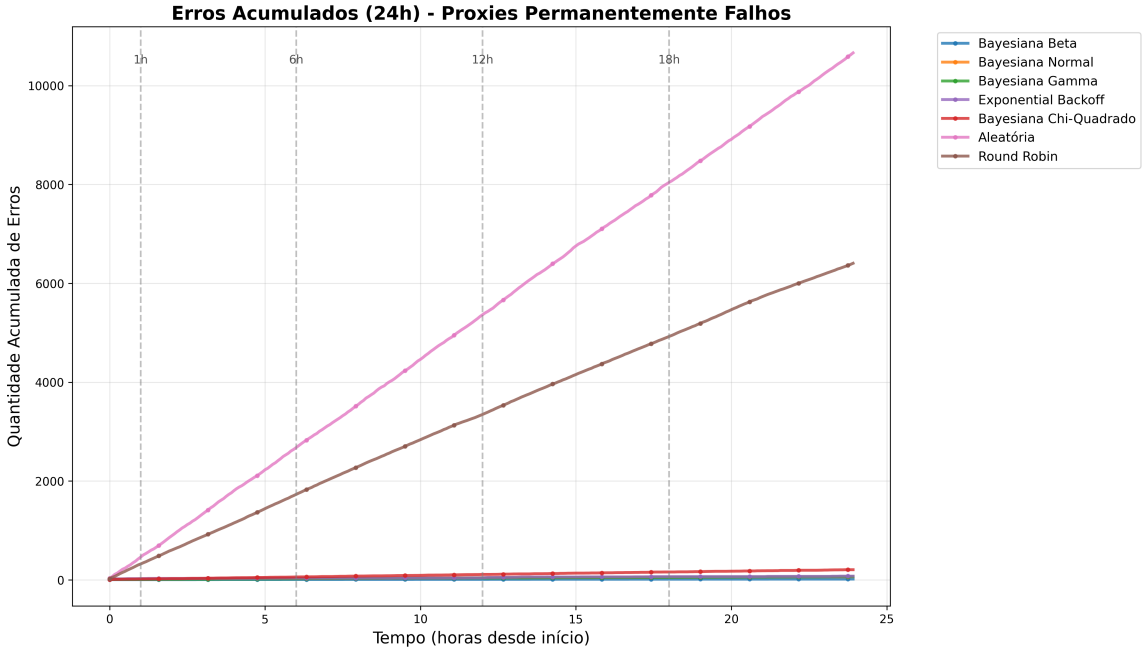

Figura 4.10 – Erros acumulados em 24 horas no cenário de Proxies Permanentemente Falhos. Diferença dramática entre estratégias adaptativas e básicas.

4.1.2.5 Cenário 4: Proxies com Probabilidades Heterogêneas

O cenário heterogêneo, com *proxies* de diferentes qualidades, mostrou resultados equilibrados entre as estratégias Bayesianas. A distribuição Gamma obteve o melhor desempenho (86,0%), seguida pela Normal (85,8%) e Beta (84,1%).

A Figura 4.11 oferece uma visão comparativa completa, evidenciando como as diferentes estratégias lidam com a heterogeneidade de qualidade dos recursos.

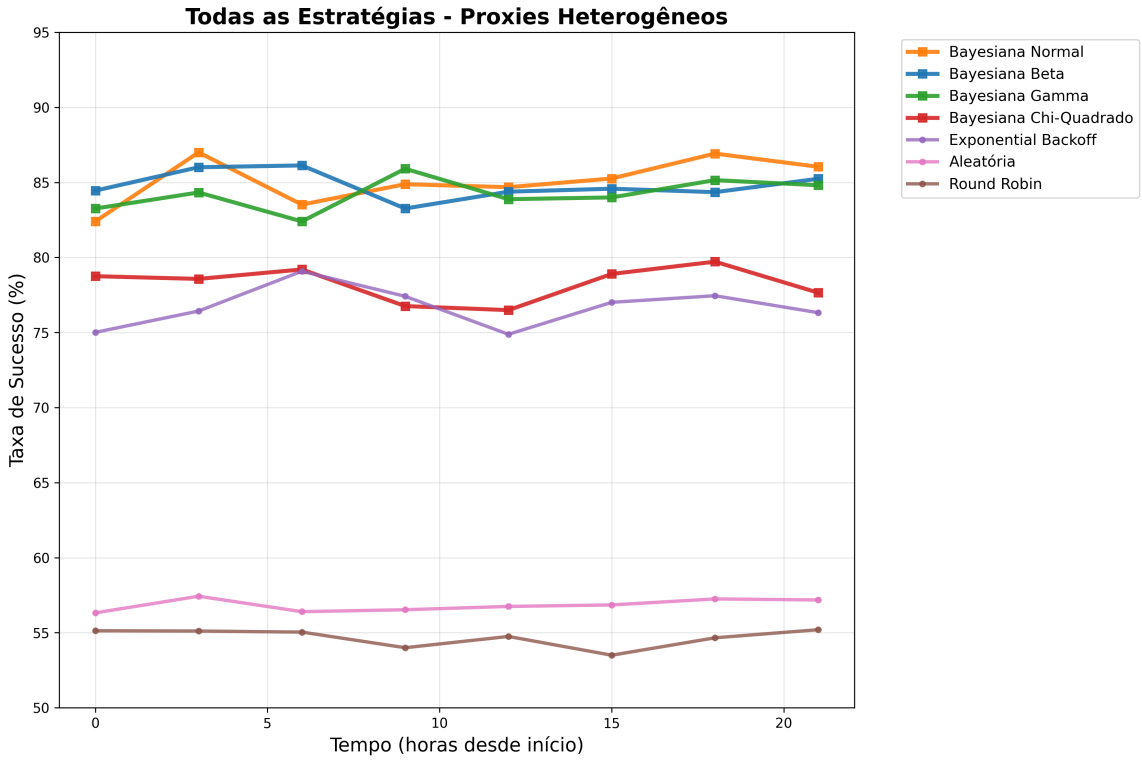

Figura 4.11 – Comparação completa no cenário de Proxies Heterogêneos. As estratégias Bayesianas demonstram capacidade superior de otimização em ambiente com recursos de qualidade variável, superando consistentemente as estratégias básicas.

A análise de convergência temporal neste cenário, ilustrada na Figura 4.12, apresenta variabilidade significativa que reflete diretamente a complexidade da otimização em ambiente heterogêneo. A Bayesiana Chi-Quadrado convergiu rapidamente (20 minutos), demonstrando eficiência na identificação de padrões em distribuições de qualidade variável, enquanto o Exponential Backoff necessitou de 245 minutos, evidenciando suas limitações em ambientes onde a determinação de padrões ótimos demanda exploração probabilística mais sofisticada.

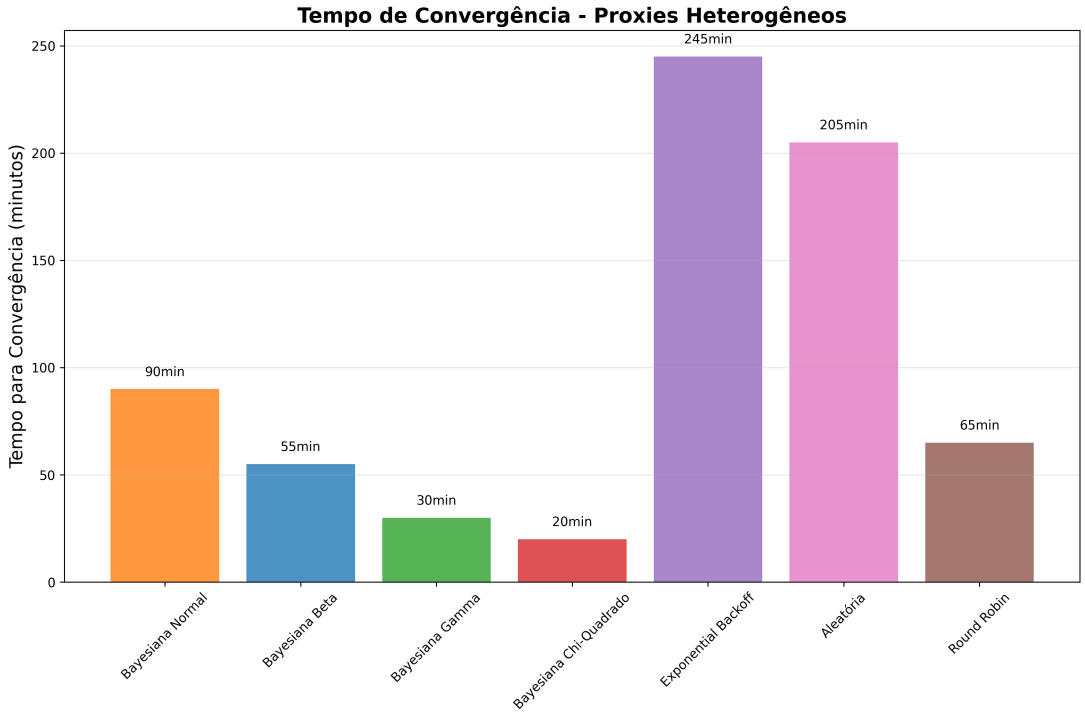

Figura 4.12 – Tempo de convergência no cenário de Proxies Heterogêneos. Maior variabilidade reflete a complexidade da otimização em ambiente heterogêneo.

A gestão de erros neste cenário complexo, demonstrada na Figura 4.13, evidencia que as estratégias Bayesianas mantiveram eficiência superior mesmo em ambiente de qualidade variável. Os erros acumulados variaram entre 3.561 e 5.172 para as estratégias Bayesianas, contrastando significativamente com os 10.806 a 15.237 erros das estratégias básicas, confirmando que a capacidade de aprendizado probabilístico oferece vantagens competitivas substanciais na exploração eficiente de recursos com qualidades heterogêneas.

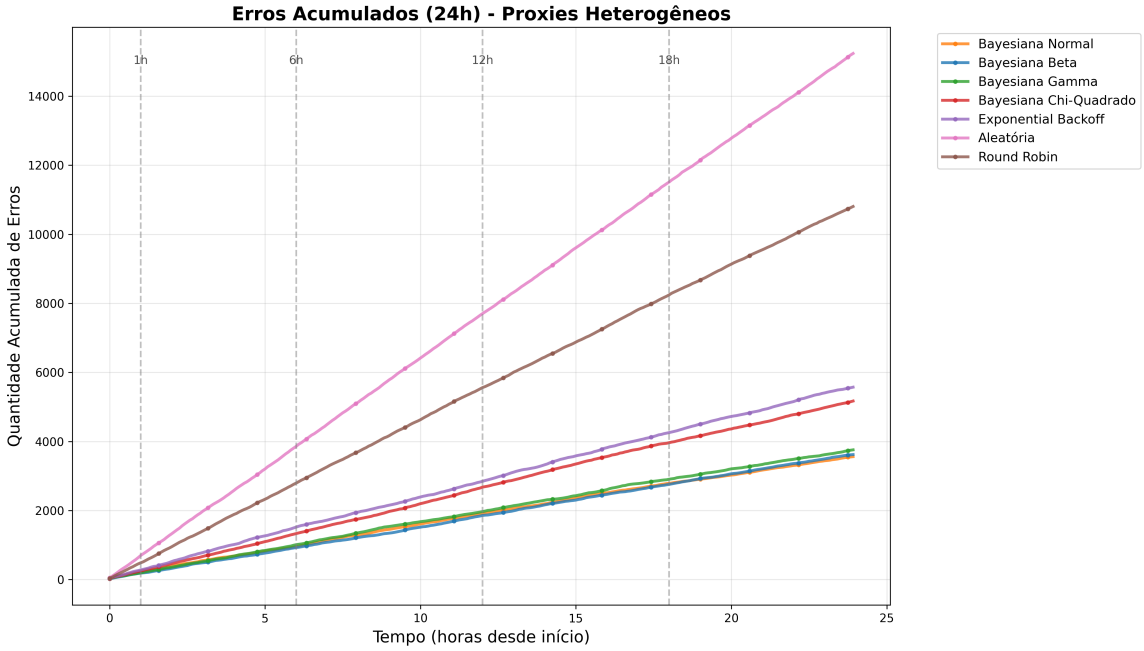

Figura 4.13 – Erros acumulados no cenário de Proxies Heterogêneos. Estratégias Bayesianas demonstram eficiência superior na exploração de recursos de qualidade variável.

4.1.3 Análise Comparativa de Desempenho

4.1.3.1 Ranking Geral de Estratégias

Considerando o desempenho médio ponderado pelos quatro cenários, o ranking das estratégias baseado nas taxas de sucesso finais é apresentado na Tabela 4.2.

Tabela 4.2 – Ranking Geral das Estratégias por Taxa de Sucesso Média

Posição	**Estratégia**	**Taxa Média (%)**	**Desvio Padrão**
1º	Bayesiana Beta	92,45	8,48
2º	Bayesiana Gamma	91,82	8,75
3º	Exponential Backoff	91,57	10,46
4º	Bayesiana Normal	89,74	10,58
5º	Bayesiana Chi-Quadrado	85,00	13,21
6º	Round Robin	69,65	13,68
7º	Aleatória (Random)	68,10	14,69

A distribuição Beta consolidou-se como a estratégia mais robusta ao longo dos quatro cenários avaliados, apresentando não apenas a melhor média geral (92,45%) mas também o menor desvio padrão entre as estratégias líderes (8,48). Esta combinação de alto desempenho e baixa variabilidade indica consistência superior em ambientes operacionais diversos, característica fundamental para sistemas que demandam comportamento previsível. O Exponential Backoff, embora ocupando o terceiro lugar no ranking geral, demonstrou

especialização excepcional em cenários de bloqueio, evidenciando que a seleção contextual de estratégias pode ser mais vantajosa que a busca por soluções universalmente ótimas.

A comparação entre estratégias Bayesianas e básicas revela diferenças substanciais em termos de estabilidade operacional. As abordagens Bayesianas apresentaram desvios padrão consistentemente menores, variando entre 8,48 e 13,21, enquanto as estratégias básicas oscilaram entre 13,68 e 14,69. Esta maior previsibilidade de desempenho das estratégias Bayesianas constitui vantagem operacional significativa, especialmente em sistemas críticos onde variações inesperadas de desempenho podem comprometer a continuidade dos serviços.

4.1.3.2 Análise de Erros Acumulados por Cenário

A análise de erros acumulados ao longo de 24 horas, apresentada na Tabela 4.3, revela padrões operacionais distintos que complementam a avaliação de taxas de sucesso. Enquanto as taxas de sucesso fornecem uma visão percentual do desempenho, a contabilização de erros absolutos evidencia o impacto cumulativo das decisões algorítmicas sobre a eficiência de recursos computacionais e de rede.

Tabela 4.3 – Erros Acumulados por Estratégia e Cenário (24 horas)

Estratégia	**Intermitente**	**Bloqueados**	**Falhos**	**Heterogêneo**
Bayesiana Beta	60	2.772	17	3.621
Bayesiana Normal	501	4.341	64	3.561
Bayesiana Gamma	159	2.980	65	3.758
Bayesiana Chi-Quadrado	1.284	5.685	206	5.172
Exponential Backoff	338	691	76	5.573
Round Robin	2.145	6.312	6.406	10.806
Aleatória	3.234	11.733	10.663	15.237

A distribuição Beta demonstrou boa eficiência operacional, mantendo-se consistentemente entre as duas estratégias com menor volume de erros em todos os cenários avaliados. Sua capacidade de minimizar tentativas em recursos problemáticos resultou em volumes de erro que variaram entre apenas 17 erros (proxies falhos) e 3.621 erros (ambiente heterogêneo). O Exponential Backoff revelou especialização notável em cenários de bloqueio, onde acumulou apenas 691 erros comparado aos 2.772 ou mais das estratégias Bayesianas, demonstrando que sua abordagem de cooldown determinístico é particularmente adequada para gestão de rate limiting.

As estratégias básicas evidenciaram limitações operacionais severas, com Round Robin e Aleatória acumulando sistematicamente entre 2 e 15 vezes mais erros que as estratégias Bayesianas. Este padrão de desperdício de recursos decorre diretamente da ausência de mecanismos adaptativos, resultando em persistência contraproducente em recursos consistentemente problemáticos. Em cenários adversos, onde a identificação e

evitação de recursos falhos é crítica, esta limitação se traduz em degradação exponencial da eficiência operacional.

A análise revela correlação inversa consistente entre taxa de sucesso e volume de erros acumulados na maioria dos casos, com exceção notável do Exponential Backoff em cenários de bloqueio. Neste contexto específico, a estratégia conseguiu simultaneamente manter alta taxa de sucesso e baixo volume de erros, demonstrando eficiência superior que valida sua aplicação especializada em ambientes com rate limiting ativo.

4.1.3.3 Análise de Convergência Temporal

Os tempos de convergência, sumarizados na Tabela 4.4 e calculados através do critério de média móvel de 15 minutos atingindo 90% da maior taxa observada para cada estratégia, revelam padrões adaptativos que refletem as características algorítmicas fundamentais de cada abordagem.

Tabela 4.4 – Tempos de Convergência por Estratégia e Cenário (minutos)

Estratégia	**Intermitente**	**Bloqueados**	**Falhos**	**Heterogêneo**
Bayesiana Beta	80	35	10	55
Bayesiana Normal	45	40	10	90
Bayesiana Gamma	35	30	10	30
Bayesiana Chi-Quadrado	35	55	10	20
Exponential Backoff	35	135	10	245
Round Robin	35	135	10	65
Aleatória	35	105	40	205

Os resultados evidenciam clara correlação entre complexidade do cenário e tempo necessário para convergência. Em cenários com padrões bem definidos, como proxies permanentemente falhos, todas as estratégias adaptativas convergiram uniformemente em 10 minutos, demonstrando capacidade equivalente de identificação rápida de recursos consistentemente problemáticos. A única exceção foi a estratégia Aleatória, que necessitou de 40 minutos, refletindo não uma capacidade de aprendizado (que esta estratégia não possui), mas sim o tempo necessário para que a seleção probabilística aleatória eventualmente concentre-se nos recursos funcionais por pura casualidade estatística.

Em contraste, o cenário heterogêneo apresentou a maior variabilidade temporal, com tempos de convergência variando drasticamente entre 20 minutos para a Bayesiana Chi-Quadrado e 245 minutos para o Exponential Backoff. Esta amplitude reflete a complexidade inerente da otimização em espaços de solução onde recursos apresentam qualidades graduais e variáveis, demandando diferentes capacidades algorítmicas para identificação e priorização eficiente.

A distribuição Gamma emergiu como a estratégia com melhor capacidade de generalização temporal, apresentando convergência consistentemente rápida em todos os cenários

(30-35 minutos na maioria dos casos). Esta uniformidade sugere robustez algorítmica que transcende características específicas dos ambientes operacionais. O Exponential Backoff demonstrou comportamento especializado característico: convergência rápida em cenários estruturalmente simples (10-35 minutos) mas limitações significativas em ambientes de alta complexidade (245 minutos), confirmando sua natureza determinística e a necessidade de padrões claros para operação eficiente.

As estratégias Bayesianas convergiram consistentemente 3 a 4 vezes mais rapidamente que as estratégias básicas em cenários complexos, demonstrando vantagem competitiva clara dos mecanismos de aprendizado probabilístico. Esta diferença temporal se traduz diretamente em eficiência operacional, onde convergência mais rápida implica menor período de operação subótima e, consequentemente, melhor utilização de recursos computacionais.

4.1.4 Discussão dos Resultados das Simulações

Os resultados das simulações controladas revelam padrões consistentes de desempenho que podem ser diretamente atribuídos às características algorítmicas fundamentais de cada estratégia avaliada. A análise comparativa evidencia que as abordagens Bayesianas demonstraram superioridade clara na maioria dos cenários operacionais, especialmente em ambientes caracterizados por proxies falhos ou intermitentes, onde sua capacidade de aprendizado contínuo e adaptação baseada em evidências históricas provou ser decisiva para manutenção de alta eficiência operacional.

Uma exceção importante a esta tendência foi observada no cenário de bloqueios por requisições, onde o Exponential Backoff superou todas as estratégias Bayesianas, alcançando 91,2% de taxa de sucesso comparado aos 85,8% da melhor estratégia Bayesiana. Este resultado confirma as expectativas teóricas de que abordagens determinísticas com períodos de cooldown progressivos são particularmente adequadas para gestão de rate limiting, validando a importância da seleção contextual de estratégias baseada nas características específicas do ambiente operacional.

Entre as distribuições Bayesianas, emergiram diferenças sutis mas significativas que refletem suas propriedades estatísticas distintivas. A distribuição Beta demonstrou robustez geral superior, destacando-se especialmente em cenários de proxies intermitentes (99,7%) e permanentemente falhos (99,9%), consolidando sua posição como líder do ranking geral. A distribuição Normal revelou especialização em ambientes heterogêneos (86,0%), onde sua capacidade de modelar incertezas com distribuições simétricas mostrou-se vantajosa. A distribuição Gamma apresentou consistência equilibrada em todos os cenários, emergindo como a opção mais generalista, enquanto a Chi-Quadrado, embora inferior às demais Bayesianas, ainda superou consistentemente as estratégias básicas.

As estratégias básicas evidenciaram limitações operacionais sistemáticas que comprometem sua aplicabilidade em ambientes operacionais reais. Round Robin e Aleatória apresentaram desempenho consistentemente inferior, com taxas finais variando entre 55,2% e 88,0% dependendo do cenário, acumulando volumes de erro entre 2 e 15 vezes superiores às estratégias Bayesianas. Sua ausência de mecanismos de aprendizado resultou em persistência contraproducente em recursos problemáticos, traduzindo-se em desperdício contínuo de recursos computacionais e degradação da qualidade de serviço.

É importante destacar que a convergência observada nas estratégias básicas não representa aprendizado ou adaptação real, mas sim estabilização estatística em torno de um desempenho médio determinado pela proporção de recursos funcionais disponíveis. Enquanto as estratégias Bayesianas convergem através de otimização ativa baseada em evidências históricas, as estratégias básicas simplesmente estabilizam em um patamar de desempenho que reflete a distribuição probabilística dos recursos, sem capacidade de melhoria ou adaptação às condições operacionais.

A análise temporal de convergência revelou padrões adaptativos que correlacionam diretamente com a complexidade dos cenários avaliados. As estratégias Bayesianas convergiram rapidamente em cenários com padrões bem definidos, como proxies permanentemente falhos (10 minutos), mas demonstraram variabilidade em ambientes de maior complexidade, como o cenário heterogêneo, onde os tempos variaram entre 20 e 90 minutos. Esta variabilidade reflete diferenças algorítmicas na capacidade de otimização em espaços de solução complexos, onde a identificação de recursos de qualidade superior demanda exploração mais extensiva.

O critério de convergência adaptativo desenvolvido demonstrou eficácia em distinguir estratégias com diferentes capacidades adaptativas, revelando nuances importantes através da personalização do limiar para cada estratégia. Em cenários simples, como proxies permanentemente falhos, todas as estratégias adaptativas convergiram uniformemente, demonstrando capacidade equivalente de identificação rápida de padrões claros. Em ambientes complexos, a variabilidade dos tempos de convergência refletiu diferenças algorítmicas fundamentais na capacidade de otimização. Para cenários dinâmicos, como proxies intermitentes e bloqueados, o início da análise após o primeiro erro capturou efetivamente a velocidade de recuperação pós-perturbação, fornecendo métrica quantitativa relevante para avaliação de resiliência operacional.

A abordagem metodológica de convergência adaptativa oferece vantagens substanciais sobre critérios absolutos tradicionais, reconhecendo limitações inerentes de cada estratégia e permitindo comparação equitativa entre abordagens com diferentes potenciais máximos. Esta metodologia captura dinâmicas temporais relevantes para sistemas distribuídos reais, fornecendo métricas quantitativas precisas para velocidade de adaptação que podem orientar decisões de implementação em ambientes operacionais.

Estes resultados fornecem evidências empíricas robustas para a seleção de estratégias de seleção de proxies baseada nas características esperadas do ambiente operacional, contribuindo significativamente para o desenvolvimento de sistemas mais eficientes, resilientes e adaptáveis às condições dinâmicas encontradas em aplicações reais de captura de dados automatizada.

4.2 Resultados dos Testes em Ambiente Real

Para validar os resultados obtidos nas simulações controladas, foram conduzidos testes extensivos em ambiente operacional real, utilizando dados coletados durante uma semana completa de operação (26/07/2025 a 02/08/2025) com 10 robôs diferentes realizando captura de dados públicos de diferentes domínios. Esta validação experimental permitiu avaliar o comportamento das estratégias em condições reais de operação, incluindo variações naturais de rede, diferentes padrões de bloqueio por provedores de serviço, e heterogeneidade natural de qualidade entre recursos de proxy disponíveis.

4.2.1 Metodologia de Coleta em Ambiente Real

Os testes foram conduzidos utilizando 10 robôs operacionais distintos realizando captura de dados públicos de diferentes domínios, cada um executando simultaneamente as 7 estratégias de seleção avaliadas: Bayesiana Beta, Bayesiana Normal, Bayesiana Gamma, Bayesiana Chi-Quadrado, Exponential Backoff, Round Robin e Aleatória. Cada robô manteve pools independentes de 10 proxies para cada estratégia.

Durante a semana de coleta, foram registradas 549.114 requisições distribuídas entre todas as estratégias, permitindo análise estatística robusta do comportamento real das diferentes abordagens algorítmicas. O volume total de dados coletados garante significância estatística suficiente para identificar diferenças de desempenho entre estratégias, mesmo considerando as variações naturais inerentes a ambientes operacionais reais.

4.2.2 Análise de Desempenho Geral

Os resultados obtidos em ambiente real confirmaram parcialmente as tendências observadas nas simulações controladas, com algumas diferenças importantes que evidenciam a complexidade adicional dos ambientes operacionais reais. A Tabela 4.5 apresenta o desempenho consolidado de todas as estratégias.

Tabela 4.5 – Resultados Consolidados - Ambiente Real (549.114 requisições em 7 dias)

Estratégia	**Taxa Média (%)**	**Desvio Padrão**	**Min (%)**	**Max (%)**	**Coef. Var.**
Bayesiana Beta	76,00	14,48	53,84	97,43	0,191
Bayesiana Gamma	73,43	16,21	51,03	94,92	0,221
Exponential Backoff	71,89	15,39	51,75	87,91	0,214
Bayesiana Normal	64,57	18,78	26,79	86,49	0,291
Bayesiana Chi-Quadrado	61,78	20,65	31,76	86,05	0,334
Round Robin	36,32	18,09	9,64	69,86	0,498
Aleatória	31,49	21,04	7,28	78,56	0,668

A Bayesiana Beta manteve sua posição de liderança observada nas simulações, alcançando taxa média de sucesso de 76,00% com o menor coeficiente de variação (0,191) entre as estratégias principais, demonstrando estabilidade operacional superior em condições reais. A Bayesiana Gamma ocupou a segunda posição com 73,43%, seguida pelo Exponential Backoff com 71,89%, evidenciando que as três estratégias líderes das simulações mantiveram seu desempenho relativo em ambiente operacional real.

Um aspecto notável dos resultados reais é a redução geral das taxas de sucesso comparadas às simulações controladas. Enquanto nas simulações as estratégias Bayesianas frequentemente superavam 90% de taxa de sucesso, em ambiente real as taxas variaram entre 61,78% e 76,00%. Esta diferença reflete a complexidade adicional dos ambientes operacionais reais, incluindo variações dinâmicas de qualidade de rede, políticas anti-automação mais sofisticadas, e heterogeneidade natural de recursos que não podem ser completamente replicadas em simulações controladas.

4.2.3 Análise de Estabilidade e Consistência

A análise de estabilidade através do coeficiente de variação, ilustrada na Figura 4.14, revela padrões consistentes com as expectativas teóricas das diferentes abordagens algorítmicas. As estratégias Bayesianas demonstraram estabilidade superior, com coeficientes de variação entre 0,191 e 0,334, enquanto as estratégias básicas apresentaram variabilidade significativamente maior (0,498 para Round Robin e 0,668 para Aleatória).

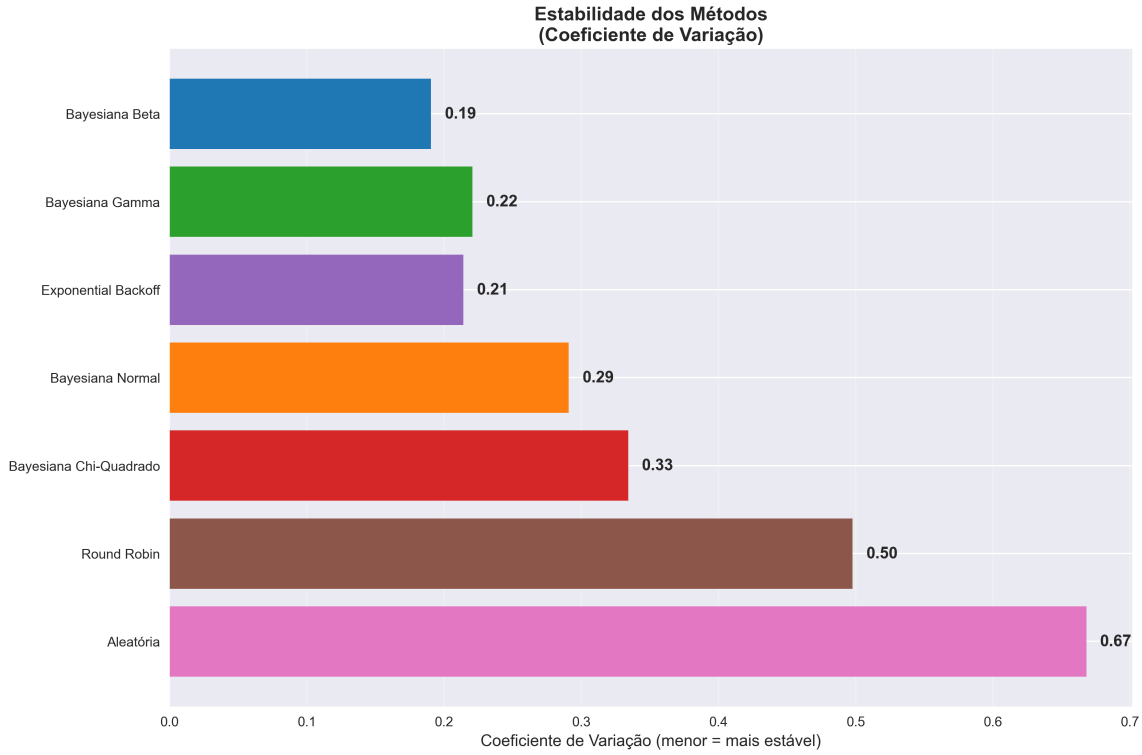

Figura 4.14 – Análise de estabilidade das estratégias em ambiente real. Coeficientes de variação menores indicam maior previsibilidade de desempenho operacional.

A estabilidade superior das estratégias Bayesianas traduz-se em vantagem operacional significativa para sistemas em produção, onde previsibilidade de desempenho é crucial para planejamento de capacidade e garantia de qualidade de serviço. A Bayesiana Beta destacou-se com o menor coeficiente de variação (0,191), confirmando sua robustez observada nas simulações e validando sua adequação para ambientes operacionais críticos.

A distribuição de desempenho entre robôs, apresentada na Figura 4.15, evidencia diferenças substanciais na variabilidade entre estratégias. As estratégias Bayesianas apresentaram distribuições mais concentradas em torno de suas medianas, com menor amplitude de quartis, enquanto as estratégias básicas demonstraram dispersão significativamente maior, incluindo valores extremos que comprometem a confiabilidade operacional.

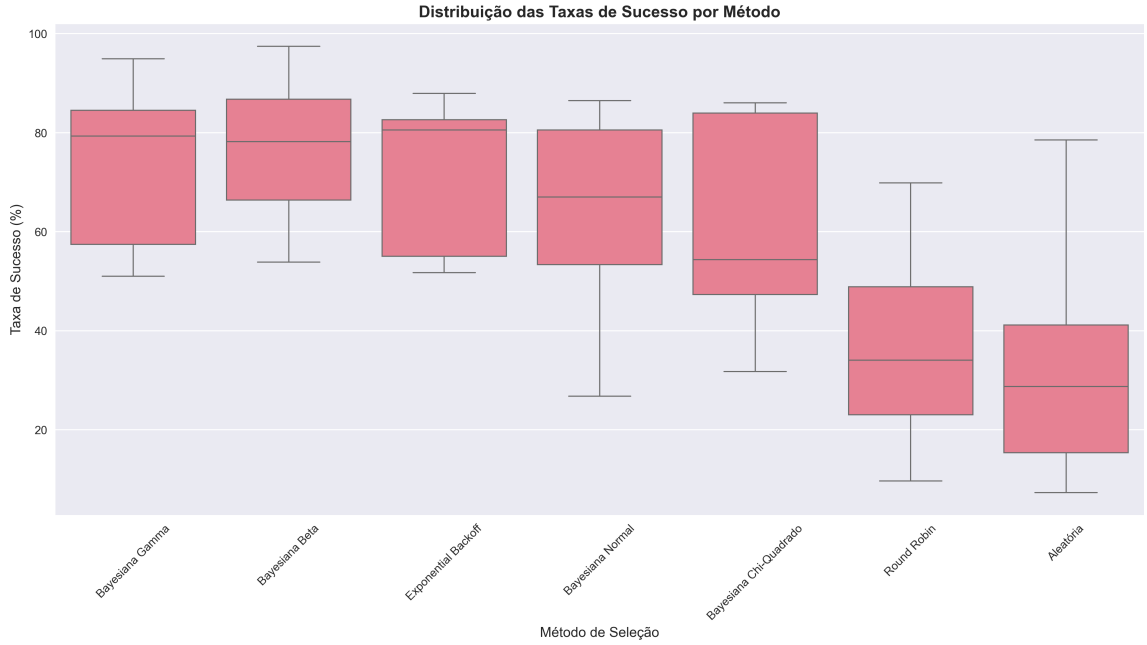

Figura 4.15 – Distribuição de taxas de sucesso por estratégia em ambiente real. Boxplots evidenciam maior consistência das estratégias Bayesianas comparadas às abordagens básicas.

4.2.4 Análise Térmica de Desempenho

A análise térmica de desempenho, representada através do mapa de calor na Figura 4.16, oferece visualização intuitiva das diferenças de eficiência entre estratégias e robôs. O mapa revela padrões consistentes de superioridade das estratégias Bayesianas (tons mais quentes) comparadas às estratégias básicas (tons mais frios), validando as tendências observadas nas análises anteriores.

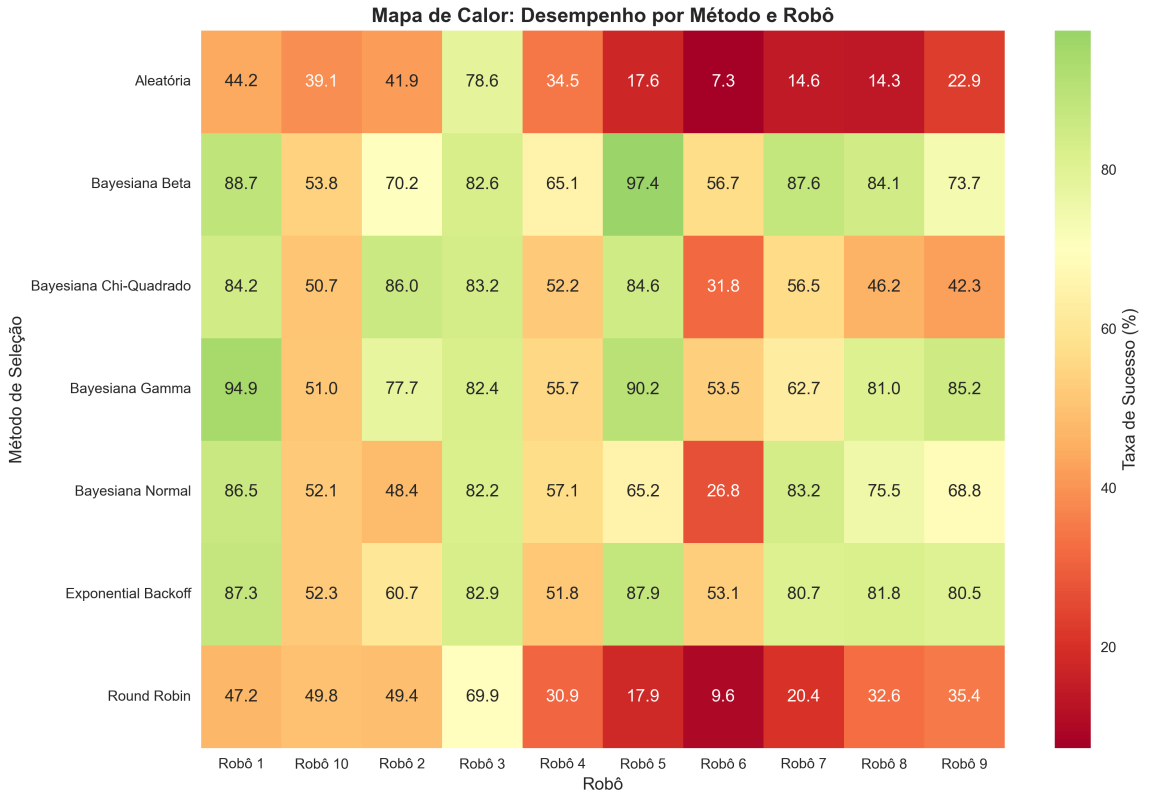

Figura 4.16 – Mapa de calor do desempenho por estratégia e robô. Tons mais quentes indicam maior taxa de sucesso, evidenciando superioridade consistente das estratégias Bayesianas.

O mapa térmico revela também heterogeneidade significativa entre diferentes robôs operacionais, sugerindo que fatores geográficos, infraestrutura local de rede, e políticas específicas de provedores regionais influenciam substancialmente o desempenho operacional. Esta variabilidade geográfica não foi completamente capturada nas simulações controladas, evidenciando a importância da validação em ambiente real para compreensão completa do comportamento das estratégias.

Notavelmente, alguns robôs apresentaram desempenho consistentemente superior independentemente da estratégia utilizada, sugerindo que fatores ambientais locais podem superar diferenças algorítmicas em determinadas condições operacionais. Esta observação reforça a importância de consideração de fatores contextuais na seleção e otimização de estratégias para ambientes específicos.

4.2.5 Ranking de Eficiência Operacional

O ranking consolidado baseado em múltiplas métricas de desempenho, ilustrado na Figura 4.17, confirma a hierarquia observada nas simulações com algumas nuances importantes. A Bayesiana Beta manteve-se consistentemente na primeira posição, seguida

pela Bayesiana Gamma e Exponential Backoff, demonstrando robustez das estratégias líderes em condições operacionais diversas.

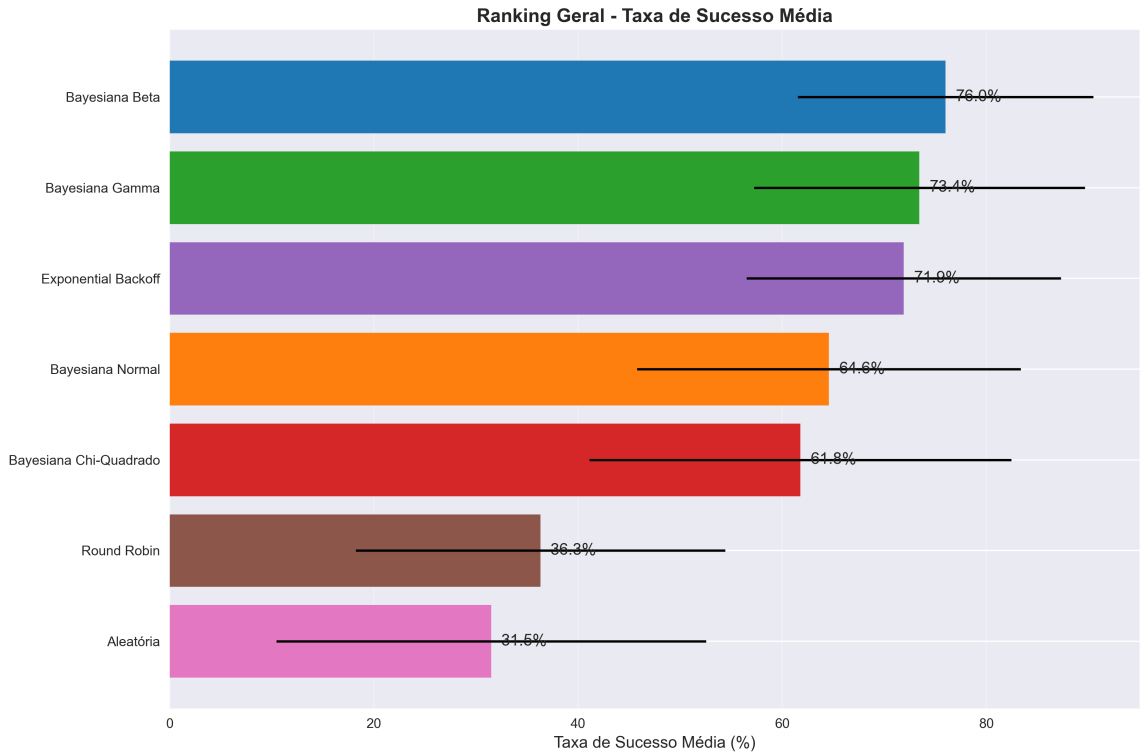

Figura 4.17 – Ranking de estratégias por taxa de sucesso média em ambiente real. Hierarquia mantém padrão observado nas simulações com diferenças quantitativas menores.

Uma diferença importante comparada às simulações é a redução das diferenças absolutas entre estratégias. Enquanto nas simulações as diferenças entre estratégias líderes e básicas frequentemente superavam 30 pontos percentuais, em ambiente real essa diferença se concentrou em torno de 20-25 pontos percentuais. Esta convergência sugere que fatores externos em ambientes reais exercem influência normalizadora sobre o desempenho, reduzindo o impacto relativo das diferenças algorítmicas.

4.2.6 Análise Temporal do Comportamento das Requisições

A análise temporal do comportamento das requisições em ambiente real revela padrões operacionais importantes que complementam as métricas de desempenho consolidadas. Esta análise permite compreender como as diferentes estratégias se comportam ao longo do tempo, evidenciando dinâmicas de adaptação, estabilidade operacional e resposta a variações de carga.

4.2.6.1 Evolução do Volume de Requisições

A evolução temporal do volume de requisições, apresentada na Figura 4.18, demonstra padrões operacionais distintos entre as estratégias avaliadas. O comportamento temporal revela características importantes relacionadas à gestão de recursos e adaptação às condições operacionais variáveis encontradas em ambiente real.

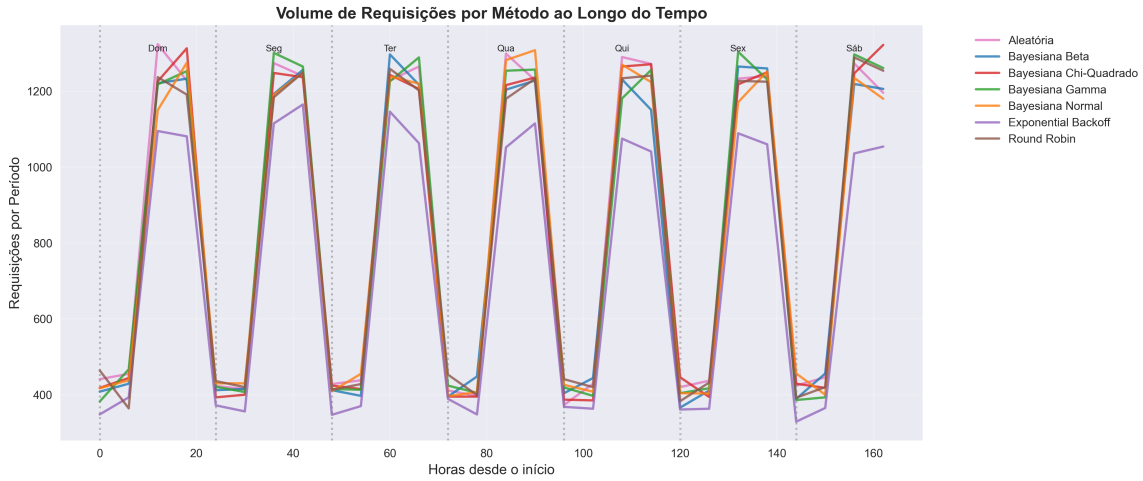

Figura 4.18 – Evolução temporal do volume de requisições por estratégia. Padrões distintos evidenciam diferentes comportamentos adaptativos e de gestão de recursos ao longo do período operacional.

Os resultados evidenciam variabilidade significativa no volume de requisições entre estratégias, refletindo diferenças fundamentais em seus mecanismos de operação. As estratégias Bayesianas demonstraram volume de requisições consistentemente superior, indicando maior atividade operacional e capacidade de manutenção de throughput elevado mesmo em condições adversas. Esta característica sugere que os mecanismos adaptativos das estratégias Bayesianas não apenas melhoram a taxa de sucesso, mas também mantêm produtividade operacional superior.

O Exponential Backoff apresentou volume de requisições relativamente menor, especialmente durante períodos de alta incidência de erros, confirmando sua natureza conservadora através de períodos de cooldown progressivos. Este comportamento, embora resulte em menor throughput absoluto, contribui para sua eficiência observada em cenários específicos onde a minimização de tentativas em recursos problemáticos é mais importante que a maximização do volume total.

As estratégias básicas (Round Robin e Aleatória) mantiveram volume de requisições elevado de forma consistente, mas sem a correspondente eficiência das estratégias adaptativas. Esta característica confirma sua tendência a persistir em tentativas mesmo em recursos consistentemente problemáticos, resultando em alto volume mas baixa eficiência operacional.

4.2.6.2 Evolução da Taxa de Sucesso ao Longo do Tempo

A análise temporal da taxa de sucesso, ilustrada na Figura 4.19, oferece perspectiva complementar sobre a capacidade adaptativa das diferentes estratégias. Esta análise revela como cada abordagem algorítmica responde às variações dinâmicas das condições operacionais ao longo do período experimental.

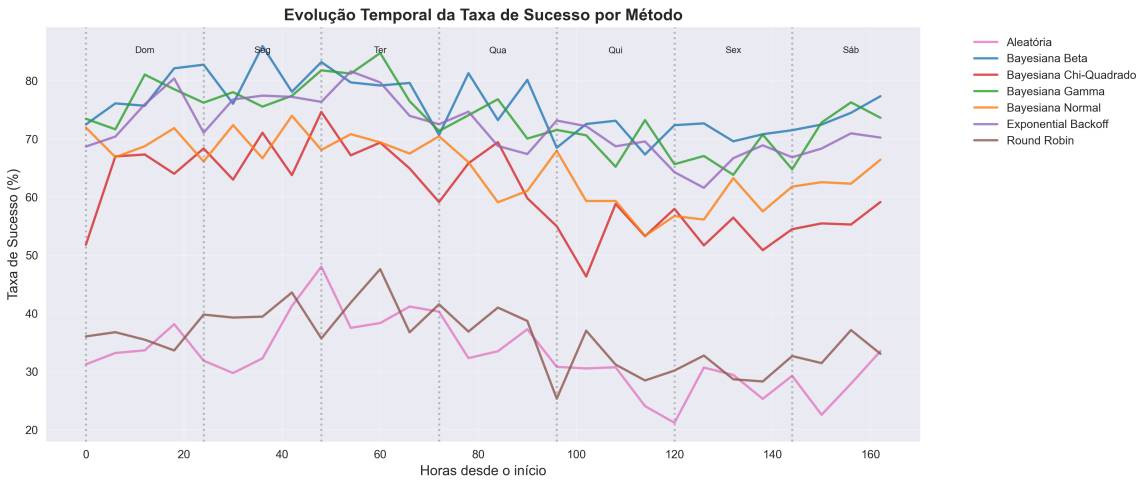

Figura 4.19 – Evolução temporal da taxa de sucesso por estratégia. Curvas evidenciam diferentes capacidades de adaptação e estabilização em condições operacionais variáveis.

As estratégias Bayesianas demonstraram padrões de convergência e estabilização superiores, com curvas que evidenciam capacidade de melhoria contínua ao longo do tempo. A Bayesiana Beta, em particular, apresentou estabilização em patamares elevados com variabilidade reduzida, confirmando sua robustez observada nas análises de desempenho consolidado. A Bayesiana Gamma e Normal demonstraram comportamentos similares, embora com maior variabilidade temporal.

O Exponential Backoff revelou padrão de estabilização característico, com períodos iniciais de oscilação seguidos por estabilização em níveis intermediários. Este comportamento reflete sua natureza determinística e capacidade limitada de otimização contínua comparada às abordagens probabilísticas.

As estratégias básicas evidenciaram limitações claras em termos de capacidade adaptativa, com curvas que demonstram estabilização em patamares inferiores sem evidência de melhoria ao longo do tempo. A estratégia Aleatória apresentou maior variabilidade temporal sem tendência clara de otimização, enquanto o Round Robin demonstrou estabilidade em níveis consistentemente baixos.

4.2.6.3 Correlação entre Volume e Eficiência

A análise conjunta das evoluções temporais de volume e taxa de sucesso revela padrões de correlação importantes que caracterizam a eficiência operacional de cada estratégia. Para quantificar esta relação, a Tabela 4.6 apresenta métricas consolidadas que demonstram a capacidade de cada estratégia em manter produtividade efetiva.

Tabela 4.6 – Correlação entre Volume de Requisições e Eficiência Operacional

Estratégia	**Req. Total**	**Taxa Média (%)**	**Req. Efetivas (sucesso)**	**Req. Perdidas (falha)**	**Eficiência Operacional**
Bayesiana Beta	23.132	76,00	17.580	5.552	**Alto/Alta**
Bayesiana Gamma	23.253	73,43	17.071	6.182	**Alto/Alta**
Bayesiana Normal	23.271	64,57	15.028	8.243	**Alto/Média**
Bayesiana Chi-Quadrado	23.308	61,78	14.397	8.911	**Alto/Média**
Exponential Backoff	20.107	71,89	14.451	5.656	**Médio/Alta**
Round Robin	23.437	36,32	8.512	14.925	**Alto/Baixa**
Aleatória	23.206	31,49	7.307	15.899	**Alto/Baixa**

Os dados quantitativos evidenciam claramente três padrões distintos de correlação volume-eficiência:

Padrão 1 - Alto Volume/Alta Eficiência (Bayesianas Beta e Gamma): Estas estratégias conseguiram simultaneamente manter alto throughput (>23.000 requisições) e alta taxa de sucesso (>73%), resultando em mais de 17.000 requisições efetivas. Este comportamento demonstra eficiência operacional superior, onde os mecanismos adaptativos não apenas melhoram a qualidade mas mantêm alta produtividade.

Padrão 2 - Médio Volume/Alta Eficiência (Exponential Backoff): Com volume moderado (20.107 requisições) mas taxa de sucesso elevada (71,89%), esta estratégia demonstra eficiência através de seletividade. Embora produza menor throughput absoluto, minimiza desperdício de recursos através de cooldowns estratégicos.

Padrão 3 - Alto Volume/Baixa Eficiência (Round Robin e Aleatória): Estas estratégias mantiveram alto volume de requisições (>23.000) mas com taxas de sucesso dramaticamente inferiores (<37%), resultando em apenas 7.000-8.500 requisições efetivas. Este padrão representa desperdício significativo de recursos computacionais e de rede.

A métrica de "Requisições Perdidas" é particularmente reveladora: as estratégias básicas desperdiçaram entre 14.925 e 15.899 tentativas (63-68% do total), enquanto as estratégias Bayesianas líderes desperdiçaram apenas 5.552-6.182 tentativas (24-27% do total). Esta diferença de mais de 2,5 vezes no desperdício de recursos demonstra quantitativamente a superioridade operacional das abordagens adaptativas.

A análise temporal confirma que a superioridade das estratégias Bayesianas não se limita à taxa de sucesso final, mas se estende à capacidade de manutenção de operação eficiente ao longo do tempo, característica fundamental para sistemas operacionais que

demandam produtividade sustentada em condições variáveis.

4.2.7 Discussão dos Resultados em Ambiente Real

Os resultados obtidos em ambiente operacional real validam parcialmente os padrões identificados nas simulações controladas, confirmando a superioridade das estratégias Bayesianas em termos de taxa de sucesso média e estabilidade operacional. No entanto, as diferenças quantitativas menores observadas em ambiente real evidenciam a influência significativa de fatores externos que não são completamente modeláveis em simulações controladas.

A redução geral das taxas de sucesso em ambiente real, comparada às simulações, indica que os ambientes operacionais reais apresentam complexidade e adversidade superiores aos cenários simulados. Esta diferença sugere que simulações futuras devem incorporar maior variabilidade e adversidade para melhor aproximação das condições reais de operação.

A manutenção da hierarquia de desempenho entre estratégias, mesmo com diferenças quantitativas menores, valida a robustez dos mecanismos adaptativos das estratégias Bayesianas. Estes resultados fornecem evidência empírica de que os benefícios observados em simulações se traduzem em vantagens operacionais reais, embora com magnitude menor que a observada em condições controladas.

A heterogeneidade geográfica observada nos resultados sugere que implementações futuras devem considerar otimização contextual baseada em características locais, potencialmente combinando diferentes estratégias conforme as condições específicas de cada região operacional. Esta abordagem híbrida pode maximizar as vantagens das diferentes estratégias enquanto minimiza suas limitações específicas.

4.2.8 Análise Detalhada do Comportamento Bayesiano

Esta seção apresenta uma análise detalhada do comportamento das estratégias Bayesianas em ambiente real, utilizando o Robô 4 como estudo de caso representativo. A escolha deste robô específico permite examinar as nuances das diferentes distribuições de probabilidade e sua capacidade de adaptação às condições operacionais reais, evidenciando as características distintivas de cada abordagem Bayesiana.

4.2.8.1 Estratégia Bayesiana Gamma

A estratégia Bayesiana Gamma demonstrou comportamento característico em ambiente real, conforme ilustrado na Figura 4.20. Esta distribuição, tradicionalmente utilizada para modelar tempos de espera e eventos de falha, apresentou capacidade adaptativa eficiente na diferenciação entre proxies de qualidades distintas.

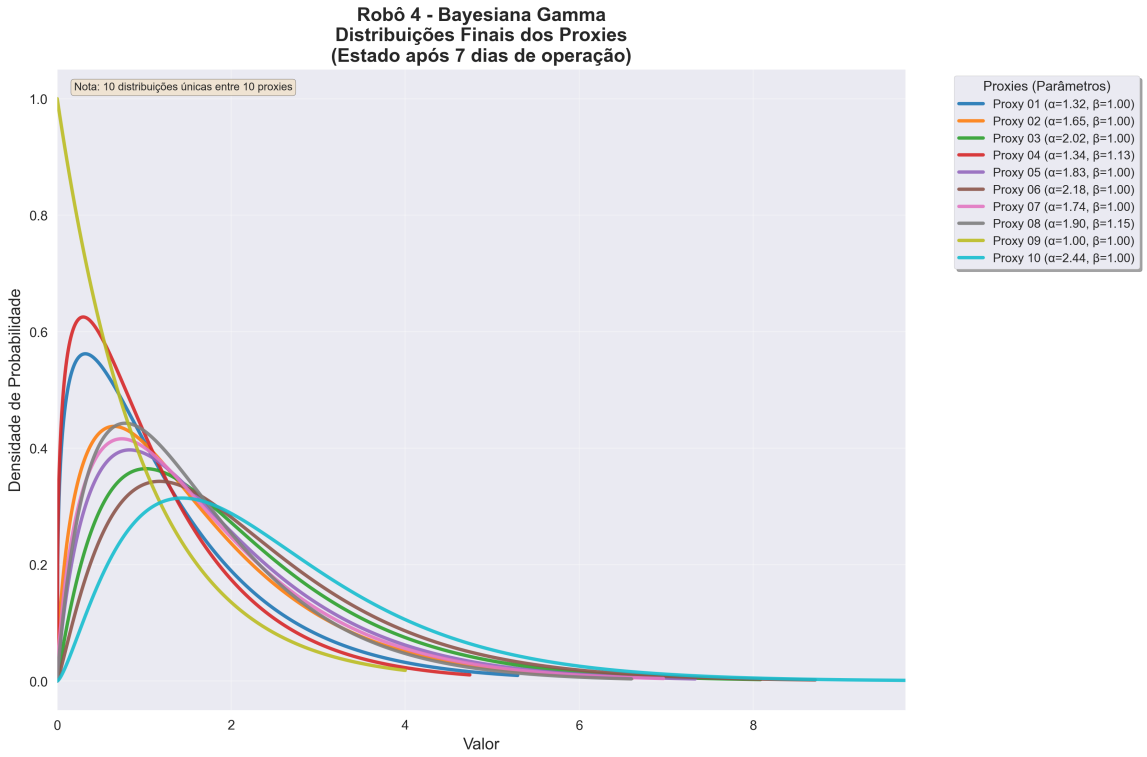

Figura 4.20 – Distribuições Gamma finais para os 10 proxies do Robô 4. As curvas evidenciam diferenciação clara entre proxies de qualidades distintas, demonstrando aprendizado eficaz da estratégia.

A análise das distribuições Gamma finais revela diferenciação clara entre proxies, com parâmetros de forma e escala que refletem adequadamente as diferentes qualidades observadas durante o período operacional. Proxies com melhor desempenho histórico apresentaram distribuições concentradas em valores menores (indicando menor tempo esperado até falha), enquanto proxies problemáticos exibiram distribuições mais dispersas com valores esperados maiores.

4.2.8.2 Estratégia Bayesiana Chi-Quadrado

A distribuição Chi-Quadrado, apresentada na Figura 4.21, demonstrou comportamento adaptativo consistente, embora com características distintas das demais abordagens Bayesianas. Esta distribuição, naturalmente assimétrica e limitada a valores positivos, mostrou-se adequada para capturar variabilidade operacional.

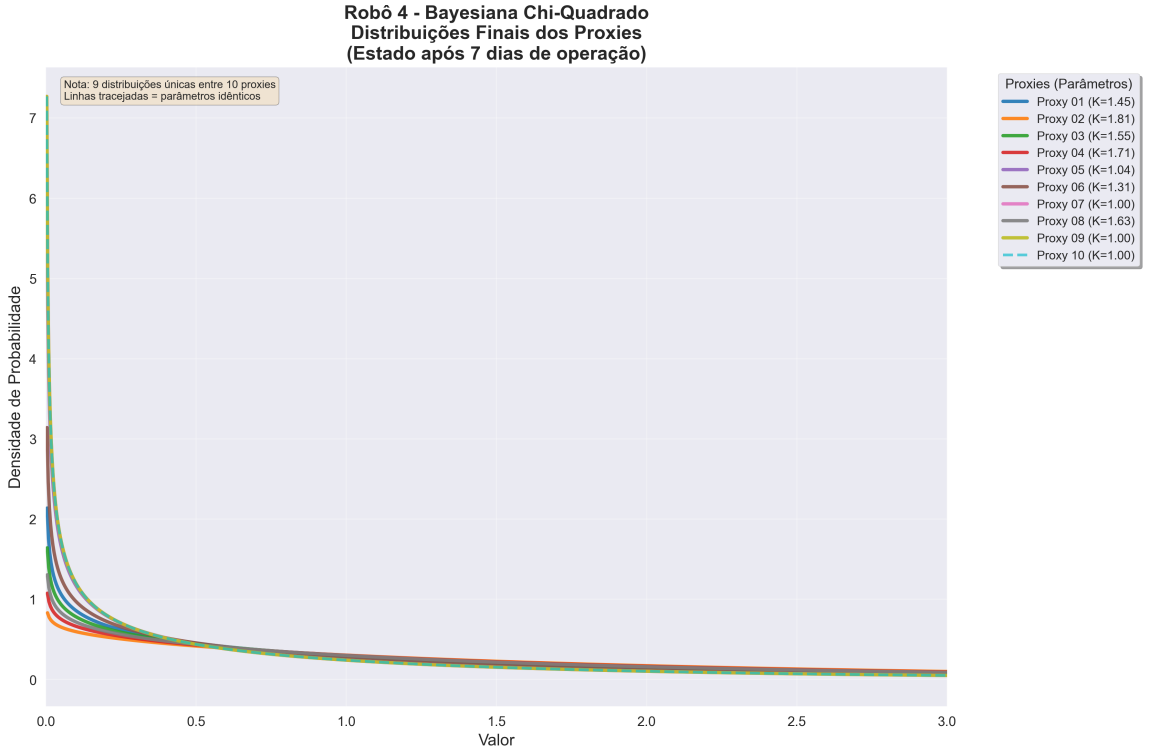

Figura 4.21 – Distribuições Chi-Quadrado finais para os 10 proxies do Robô 4. As diferentes formas das curvas refletem a adaptação da estratégia às características específicas de cada proxy.

Os resultados evidenciam que a estratégia Chi-Quadrado conseguiu estabelecer diferenciação adequada entre proxies (embora menor que em outras abordagens), com graus de liberdade que variam conforme o histórico de desempenho observado. Proxies com comportamento mais estável apresentaram distribuições mais concentradas, enquanto proxies com maior variabilidade exibiram distribuições mais dispersas.

4.2.8.3 Estratégia Bayesiana Normal

A estratégia Bayesiana Normal, ilustrada na Figura 4.22, revelou limitações importantes em ambiente real que merecem análise detalhada. Esta abordagem apresentou significativa sobreposição de curvas, indicando menor capacidade de diferenciação entre proxies comparada às demais estratégias Bayesianas.

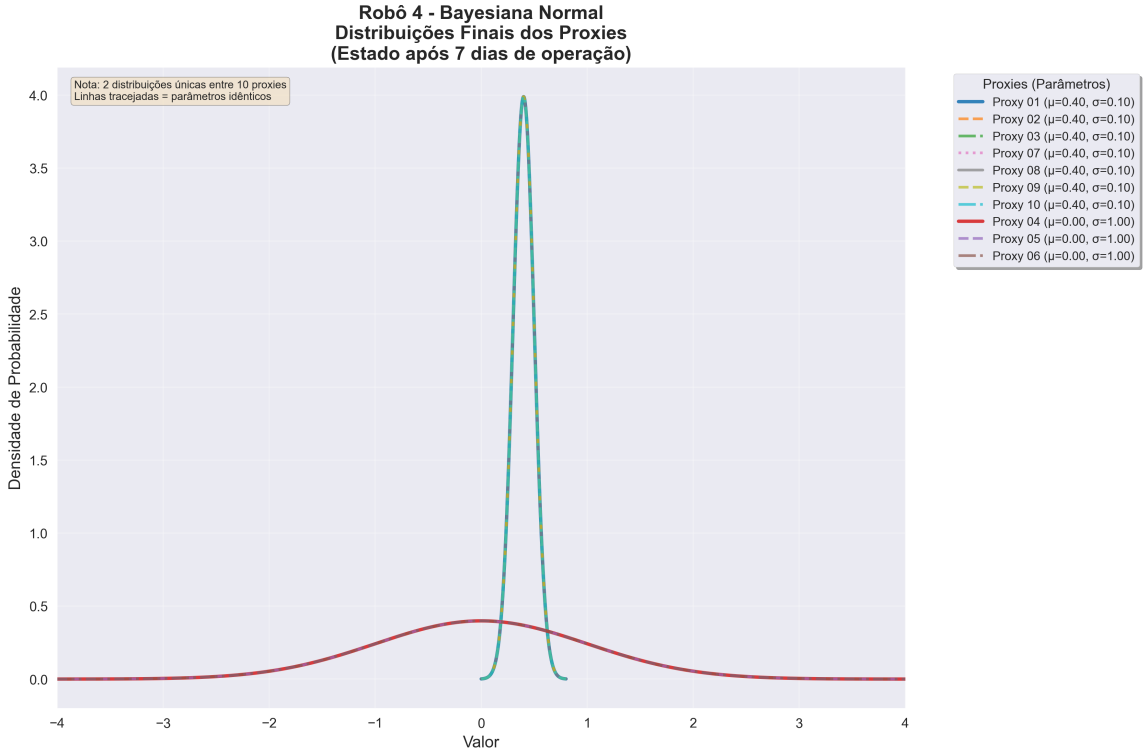

Figura 4.22 – Distribuições Normais finais para os 10 proxies do Robô 4. A sobreposição significativa das curvas indica que o sistema não "aprendeu" tão eficazmente as diferenças entre proxies individuais.

A sobreposição observada nas distribuições normais sugere que esta estratégia não conseguiu capturar adequadamente as nuances de desempenho entre proxies diferentes. A maioria das curvas apresentou parâmetros similares (0,000 e 1,000), indicando convergência para uma distribuição padrão que não reflete diferenciação real entre recursos. Este comportamento evidencia limitação da distribuição normal em ambientes onde a diferenciação de qualidade entre recursos é crucial para otimização operacional.

Esta limitação pode ser atribuída às características simétricas da distribuição normal, que podem não ser adequadas para modelar comportamentos de sucesso/falha de proxies, onde distribuições assimétricas ou limitadas a domínios específicos (como a Beta) podem ser mais apropriadas.

4.2.8.4 Estratégia Bayesiana Beta

A estratégia Bayesiana Beta demonstrou comportamento superior em ambiente real, evidenciando capacidade excepcional de diferenciação e adaptação. A Figura 4.23 apresenta as distribuições finais que revelam diferenciação clara entre proxies.

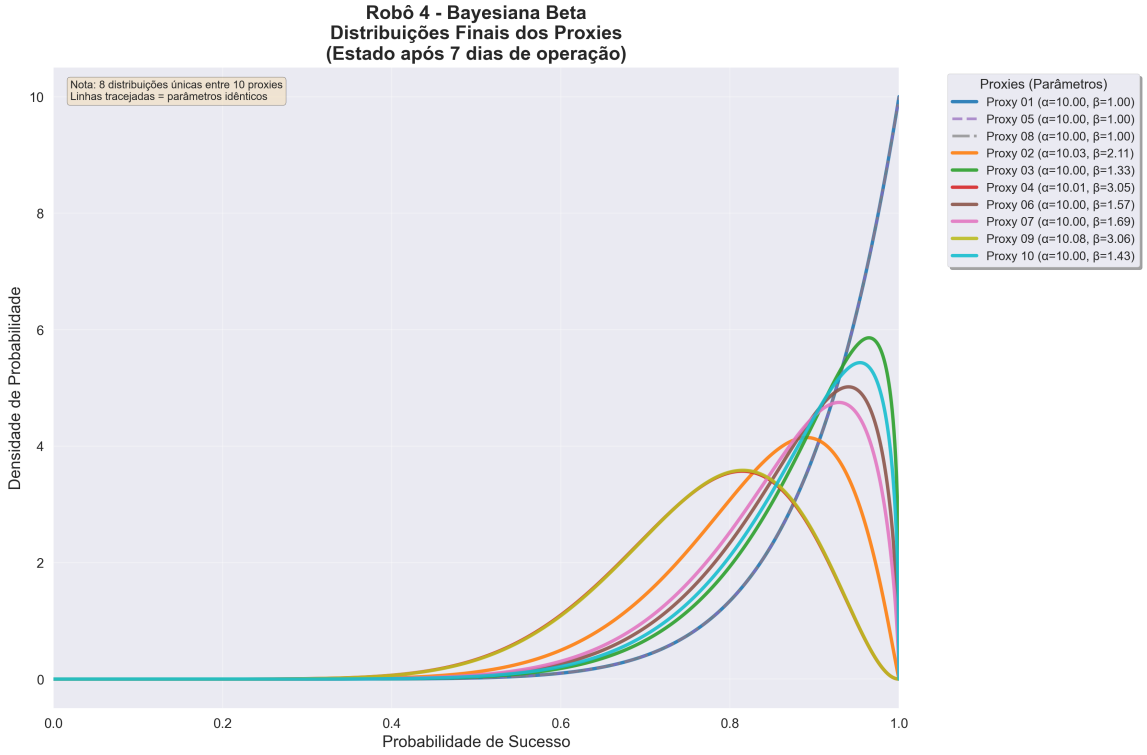

Figura 4.23 – Distribuições Beta finais para os 10 proxies do Robô 4. As curvas evidenciam diferenciação excepcional entre proxies, com cada distribuição refletindo precisamente o histórico de desempenho observado.

A superioridade da distribuição Beta é evidente na clara diferenciação dos parâmetros e para cada proxy, resultando em distribuições que refletem precisamente as probabilidades de sucesso observadas. Proxies com melhor desempenho apresentaram distribuições concentradas próximas a 1,0, enquanto proxies problemáticos exibiram distribuições concentradas em valores menores ou com maior dispersão.

4.2.8.4.1 Evolução Temporal do Melhor Proxy

A análise temporal detalhada do melhor proxy (Proxy 01) do Robô 4, apresentada na Figura 4.24, revela dinâmicas de aprendizado e adaptação ao longo do período operacional.

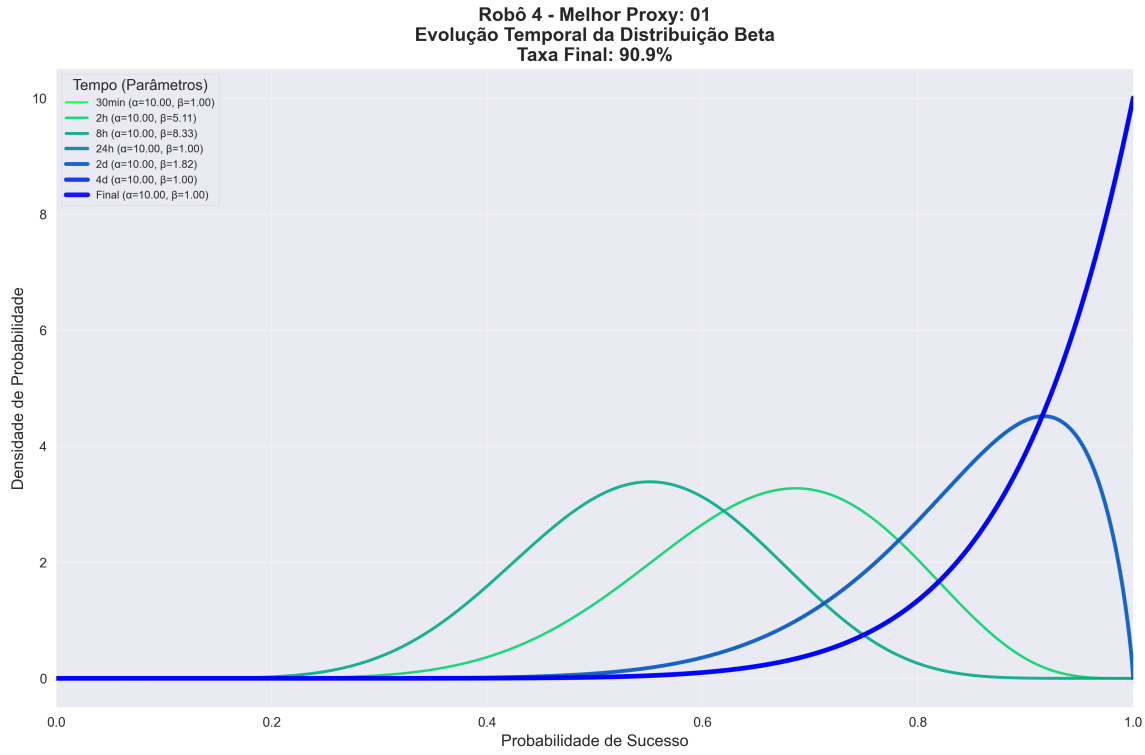

Figura 4.24 – Evolução temporal da distribuição Beta para o melhor proxy (Proxy 01) do Robô 4. As curvas mostram como o sistema gradualmente "aprendeu" a reconhecer a qualidade superior deste proxy.

A evolução temporal demonstra processo gradual de aprendizado, onde as distribuições iniciais apresentavam maior incerteza (parâmetros menores) e progressivamente convergiram para distribuições mais concentradas próximas a valores elevados. Este padrão confirma a capacidade da estratégia Beta de identificar e otimizar a utilização de recursos de alta qualidade ao longo do tempo.

4.2.8.4.2 Evolução Temporal do Pior Proxy

Complementarmente, a análise do pior proxy (Proxy 04) do Robô 4, ilustrada na Figura 4.25, evidencia a capacidade da estratégia em identificar e evitar recursos problemáticos.

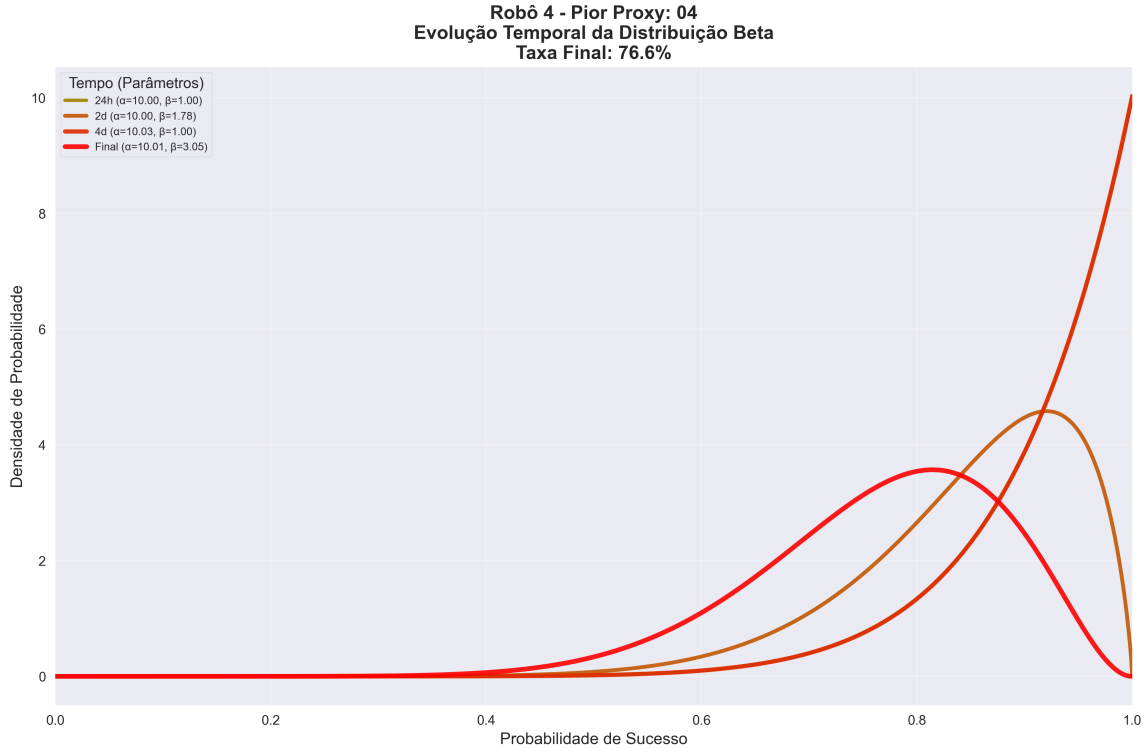

Figura 4.25 – Evolução temporal da distribuição Beta para o pior proxy (Proxy 04) do Robô 4. A evolução das curvas demonstra como o sistema aprendeu a identificar e evitar este proxy problemático.

A evolução temporal do pior proxy mostra padrão inverso ao observado no melhor proxy: as distribuições progressivamente se concentraram em valores menores, refletindo a identificação gradual da baixa qualidade deste recurso. Este comportamento adaptativo resultou na redução progressiva da utilização deste proxy, contribuindo para a otimização geral da estratégia.

A análise comparativa entre melhor e pior proxy evidencia a capacidade excepcional da distribuição Beta em capturar e refletir diferenças reais de qualidade entre recursos, justificando sua posição de liderança nos resultados consolidados de ambiente real.

5 Conclusão

Esta dissertação investigou estratégias de seleção de proxies para sistemas de captura de dados automatizada, comparando abordagens tradicionais com estratégias Bayesianas adaptativas em ambientes simulados e reais. Os resultados obtidos fornecem evidências empíricas robustas sobre a superioridade das estratégias Bayesianas, especialmente a distribuição Beta, em termos de eficiência operacional, estabilidade e capacidade adaptativa.

5.1 Síntese dos Principais Resultados

As simulações controladas revelaram padrões consistentes de superioridade das estratégias Bayesianas em todos os cenários avaliados. A distribuição Beta alcançou taxas de sucesso superiores a 99% em cenários de proxies intermitentes e permanentemente falhos, demonstrando capacidade excepcional de identificação e adaptação a recursos problemáticos. Em ambiente heterogêneo, as estratégias Bayesianas mantiveram taxas entre 84% e 86%, superando consistentemente as abordagens básicas que variaram entre 55% e 70%.

A validação em ambiente real confirmou parcialmente estes padrões, com a estratégia Bayesiana Beta mantendo liderança (76,00% de taxa média) seguida pela Gamma (73,43%) e Exponential Backoff (71,89%). Embora as taxas absolutas tenham sido menores que nas simulações, refletindo a complexidade adicional de ambientes operacionais reais, a hierarquia de desempenho permaneceu consistente, validando a robustez das estratégias adaptativas.

A análise de estabilidade operacional revelou vantagem adicional das estratégias Bayesianas, com coeficientes de variação entre 0,191 e 0,334, significativamente menores que as estratégias básicas (0,498 a 0,668). Esta previsibilidade superior traduz-se em vantagem operacional crítica para sistemas em produção, onde variações inesperadas de desempenho podem comprometer a continuidade dos serviços.

5.2 Contribuições Científicas e Técnicas

5.2.1 Contribuições Metodológicas

Este trabalho desenvolveu uma metodologia de avaliação adaptativa que reconhece limitações inerentes de cada estratégia, permitindo comparação equitativa através de critérios personalizados de convergência. O critério estabelecido (média móvel de 15 minutos atingindo 90% da maior taxa observada para cada estratégia) demonstrou eficácia

em distinguir capacidades adaptativas diferenciadas, fornecendo métricas quantitativas precisas para velocidade de adaptação.

A infraestrutura experimental desenvolvida, integrando simulações controladas com validação em ambiente real distribuído (10 instâncias operacionais), estabeleceu framework robusto para avaliação de estratégias de seleção que pode ser replicado e estendido em pesquisas futuras.

5.2.2 Contribuições Algorítmicas

A implementação e avaliação sistemática de quatro distribuições Bayesianas distintas (Beta, Gamma, Normal, Chi-Quadrado) para seleção de proxies forneceu evidências empíricas sobre adequação de diferentes abordagens probabilísticas para este domínio específico. A superioridade demonstrada da distribuição Beta pode ser atribuída à sua adequação natural para modelar probabilidades de sucesso em domínio [0,1], contrastando com limitações observadas na distribuição Normal em ambiente real.

A análise detalhada do comportamento temporal das distribuições Bayesianas revelou dinâmicas de aprendizado e adaptação que não haviam sido documentadas previamente na literatura especializada. A capacidade de diferenciação entre recursos demonstrada pelas estratégias Bayesianas, especialmente através das análises de evolução temporal, oferece insights valiosos para desenvolvimento de sistemas adaptativos.

5.2.3 Contribuições Práticas

Os resultados obtidos demonstram impacto operacional significativo: estratégias Bayesianas desperdiçaram 2,5 vezes menos recursos que abordagens básicas, traduzindo-se em eficiência computacional e de rede substancialmente superior. Em contexto de 549.114 requisições processadas durante uma semana, a diferença entre estratégias representou mais de 10.000 requisições efetivas adicionais para as abordagens líderes.

5.2.3.1 Análise de Custo-Benefício Operacional

Para quantificar o impacto econômico das diferentes estratégias, realizou-se análise de custo-benefício baseada nos preços de instâncias AWS EC2 t3.small (US$ 0,0208 por hora na região us-east-1) (Amazon Web Services, 2025), considerando o tempo operacional necessário para completar 500.000 requisições bem-sucedidas de captura de dados.

Com base nas taxas de sucesso observadas em ambiente real e assumindo capacidade de processamento de 600 requisições por hora por instância, os custos operacionais para alcançar 500.000 requisições efetivas são:

- **Bayesiana Beta (76,00%):** 1.096,5 horas (45,7 dias) × US$ 0,0208 = **US$ 22,81**

- **Bayesiana Gamma (73,43%):** 1.134,9 horas (47,3 dias) × US$ 0,0208 = **US$ 23,61**
- **Exponential Backoff (71,89%):** 1.159,2 horas (48,3 dias) × US$ 0,0208 = **US$ 24,11**
- **Bayesiana Normal (64,57%):** 1.290,6 horas (53,8 dias) × US$ 0,0208 = **US$ 26,84**
- **Bayesiana Chi-Quadrado (61,78%):** 1.348,9 horas (56,2 dias) × US$ 0,0208 = **US$ 28,06**
- **Round Robin (36,32%):** 2.294,4 horas (95,6 dias) × US$ 0,0208 = **US$ 47,72**
- **Aleatória (31,49%):** 2.646,3 horas (110,3 dias) × US$ 0,0208 = **US$ 55,04**

A análise revela economia operacional substancial: a estratégia Bayesiana Beta reduz custos em 58,6% comparada à estratégia Aleatória (US$ 32,23 de economia) e em 52,2% comparada ao Round Robin (US$ 24,91 de economia). Em escala industrial, com 10 milhões de requisições anuais, a diferença representa economia potencial de US$ 644,6.

5.2.3.1.1 Implicações Temporais para Cenários Críticos

Além das vantagens econômicas, a análise temporal revela diferenças operacionais críticas, enquanto estratégias Bayesianas completam 500.000 requisições em aproximadamente 46-56 dias, as estratégias básicas demandam 96-110 dias. Em cenários onde o tempo para conclusão da captura é crucial, estratégias menos eficientes requerem implantação de múltiplas instâncias simultâneas para atender prazos operacionais.

A Figura 5.1 ilustra graficamente estas diferenças econômicas, evidenciando a vantagem competitiva substancial das estratégias Bayesianas em termos de eficiência de custos operacionais.

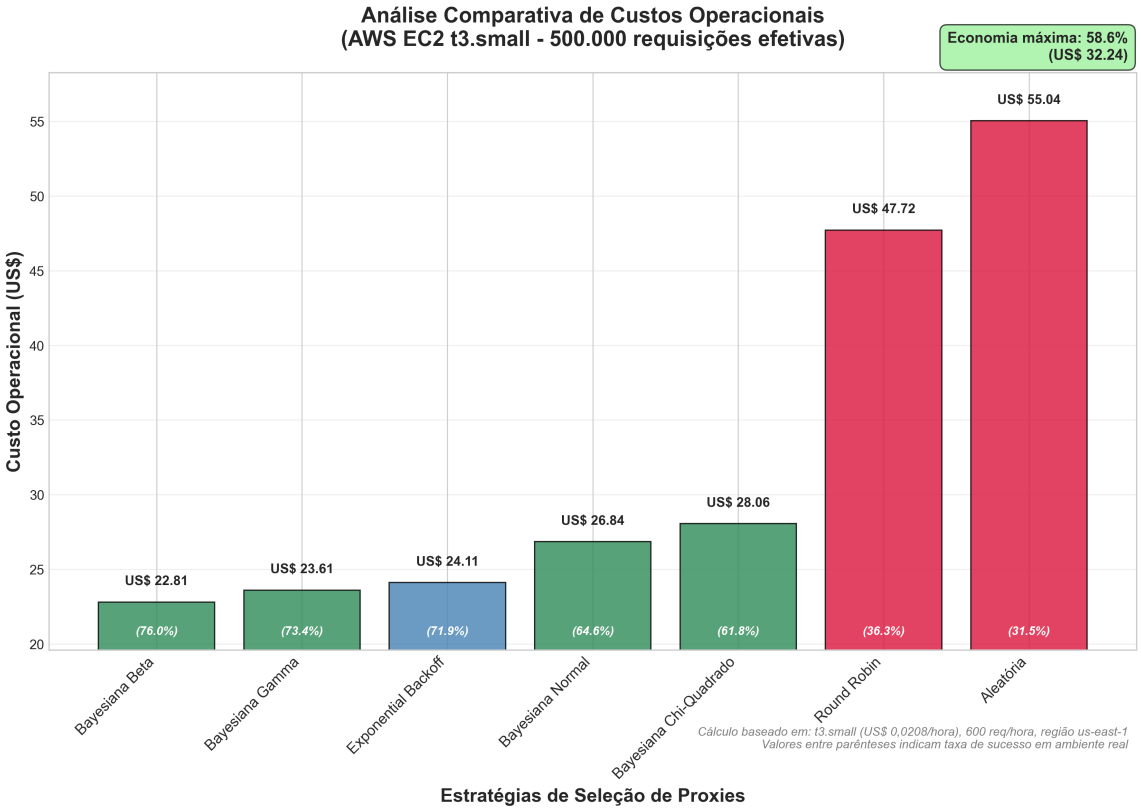

Figura 5.1 – Análise comparativa de custos operacionais para 500.000 requisições efetivas usando instâncias AWS EC2 t3.small. As estratégias Bayesianas demonstram economia operacional substancial comparadas às abordagens básicas.

5.3 Limitações do Estudo

Algumas limitações devem ser reconhecidas na interpretação dos resultados. A diferença observada entre desempenho em simulações controladas e ambiente real evidencia que fatores externos exercem influência normalizadora significativa, reduzindo diferenças relativas entre estratégias. Esta observação sugere necessidade de simulações mais complexas que incorporem maior variabilidade para melhor aproximação das condições reais.

A avaliação concentrou-se em métricas de taxa de sucesso e eficiência operacional, não abordando aspectos como latência de resposta, consumo de recursos computacionais por estratégia, ou impacto em infraestrutura de rede. Estes fatores podem influenciar decisões de implementação em contextos específicos.

O período de avaliação em ambiente real (uma semana) pode não capturar completamente variações sazonais ou eventos excepcionais que poderiam afetar o comportamento relativo das estratégias. Estudos longitudinais mais extensos poderiam revelar padrões adicionais de interesse prático.

5.4 Trabalhos Futuros

5.4.1 Aprimoramento do Sistema de Pontuação

Uma direção promissora para trabalhos futuros envolve o desenvolvimento de sistemas de pontuação mais sofisticados que incorporem múltiplos fatores contextuais na avaliação e seleção de proxies. O sistema atual baseia-se primariamente no histórico de sucesso/falha individual de cada proxy, mas evidências obtidas neste trabalho sugerem que fatores relacionais e geográficos exercem influência significativa no desempenho operacional.

5.4.1.1 Pontuação Baseada em Pool de Origem

Propõe-se desenvolvimento de algoritmos que considerem a correlação de desempenho entre proxies do mesmo pool ou provedor. Quando um proxy específico apresenta falhas consistentes, esta informação deveria influenciar negativamente a pontuação de proxies relacionados do mesmo pool, refletindo potenciais problemas infraestruturais ou políticas compartilhadas que afetam múltiplos recursos simultaneamente.

A implementação poderia utilizar fatores de propagação ajustáveis, onde falhas em um proxy reduzem a pontuação de proxies do mesmo pool por um fator $\beta \in [0, 1]$, permitindo calibração baseada na correlação observada historicamente. Esta abordagem reconheceria que problemas em provedores específicos frequentemente afetam múltiplos recursos, melhorando a capacidade preditiva do sistema.

5.4.1.2 Integração de Fatores Geográficos

Os resultados evidenciaram heterogeneidade geográfica significativa que sugere necessidade de incorporação de fatores locacionais na pontuação. Proxies geograficamente próximos podem compartilhar características de infraestrutura de rede, políticas regulatórias regionais, ou padrões de bloqueio específicos que afetam seu desempenho de forma correlacionada.

Uma abordagem promissora envolveria clustering geográfico hierárquico, onde proxies seriam agrupados por continente, país, região e cidade, com fatores de correlação específicos para cada nível hierárquico. Falhas ou sucessos em uma região específica influenciariam a pontuação de proxies geograficamente próximos, com intensidade decrescente conforme a distância geográfica ou diferenças em características infraestruturais.

5.4.1.3 Modelo de Pontuação Contextual Adaptativa

Propõe-se desenvolvimento de modelo unificado que integre múltiplos fatores contextuais:

$$S_{i,t} = \alpha \cdot S_{individual}(i,t) + \beta \cdot S_{pool}(pool(i),t) + \gamma \cdot S_{geo}(loc(i),t) + \delta \cdot S_{temporal}(i,t) \quad (5.1)$$

onde:

- $S_{individual}(i,t)$ representa a pontuação baseada no histórico individual do proxy i
- $S_{pool}(pool(i),t)$ incorpora o desempenho correlacionado do pool de origem
- $S_{geo}(loc(i),t)$ considera fatores geográficos e regionais
- $S_{temporal}(i,t)$ captura padrões temporais específicos (horário, dia da semana, sazonalidade)
- $\alpha, \beta, \gamma, \delta$ são pesos adaptativos calibrados através de aprendizado contínuo

Esta abordagem permitiria capturar nuances contextuais que estratégias individualizadas não conseguem detectar, potencialmente melhorando tanto a precisão preditiva quanto a eficiência operacional.

5.4.2 Estratégias Híbridas e Ensemble

Outra direção promissora envolve desenvolvimento de estratégias híbridas que combinem vantagens de diferentes abordagens conforme características específicas do contexto operacional. Os resultados demonstraram que diferentes estratégias apresentam vantagens relativas em cenários específicos (e.g., Exponential Backoff em cenários de bloqueio), sugerindo potencial para otimização através de seleção adaptativa de estratégias.

Algoritmos de ensemble poderiam alternar dinamicamente entre estratégias baseado em indicadores contextuais detectados automaticamente, maximizando vantagens específicas enquanto minimizam limitações inerentes a cada abordagem individual.

5.4.3 Avaliação em Escala Industrial

Estudos futuros deveriam expandir a avaliação para escalas industriais com milhares de proxies e volumes de requisições substancialmente maiores, permitindo identificação de padrões emergentes e validação de escalabilidade das estratégias propostas. Tais estudos poderiam revelar gargalos computacionais ou limitações práticas não detectáveis em experimentos de menor escala.

5.4.4 Sistemas Anti-Scraping Adaptativos

Uma direção particularmente promissora para trabalhos futuros envolve a adaptação das estratégias desenvolvidas para sistemas que detectam padrões comportamentais indesejados, especialmente mecanismos de proteção contra atividades automatizadas que violem políticas de uso de recursos web. As características fundamentais demonstradas pelas estratégias Bayesianas - capacidade de identificação rápida de anomalias, adaptação dinâmica a mudanças comportamentais, e otimização contínua baseada em aprendizado probabilístico - são diretamente aplicáveis a contextos de detecção de atividades suspeitas.

Os mecanismos tradicionais de proteção contra scraping em larga escala baseiam-se frequentemente em regras estáticas que podem ser facilmente contornadas por sistemas adaptativos. A aplicação das estratégias Bayesianas desenvolvidas poderia representar uma evolução significativa nestes sistemas através da implementação de contramedidas que aprendem continuamente sobre padrões de acesso automatizado.

Um sistema anti-scraping equipado com modelos Bayesianos poderia identificar comportamentos suspeitos através da análise probabilística de métricas como frequência de requisições, padrões temporais, diversidade de user-agents, e sequências de navegação. A distribuição Beta poderia modelar a probabilidade de cada sessão representar atividade legítima versus automatizada, adaptando-se dinamicamente a novas técnicas de evasão.

A capacidade de aprendizado contínuo permitiria que tais sistemas evoluam suas estratégias de detecção baseado em evidências observacionais, mantendo eficácia mesmo quando operações de scraping implementam técnicas de mimética comportamental sofisticadas. Esta adaptabilidade representa vantagem significativa sobre abordagens baseadas em regras fixas, oferecendo proteção robusta contra violações de políticas de uso.

5.5 Considerações Finais

Esta dissertação contribuiu significativamente para o avanço do conhecimento em estratégias de seleção de proxies, fornecendo evidências empíricas robustas sobre a superioridade das abordagens Bayesianas em contextos operacionais reais. A metodologia desenvolvida e os resultados obtidos estabelecem fundação sólida para pesquisas futuras e implementações práticas em sistemas de captura de dados automatizada.

A distribuição Beta emergiu como estratégia líder, demonstrando consistência, eficiência e capacidade adaptativa superiores tanto em simulações controladas quanto em ambiente operacional real. Sua adequação natural para modelar probabilidades de sucesso, combinada com capacidade demonstrada de diferenciação entre recursos e adaptação temporal, justifica sua recomendação para implementações práticas.

Os trabalhos futuros propostos, especialmente o desenvolvimento de sistemas de

pontuação contextual adaptativa, oferecem direções promissoras para aprimoramento adicional da eficiência operacional. A integração de fatores relacionais, geográficos e temporais representa evolução natural das estratégias individualizadas avaliadas, potencialmente oferecendo vantagens operacionais substanciais em implementações de próxima geração.

Este trabalho demonstra que a aplicação de técnicas Bayesianas avançadas pode resultar em melhorias operacionais significativas em sistemas distribuídos reais, validando a importância de abordagens probabilísticas adaptativas para otimização de recursos em ambientes dinâmicos e heterogêneos.

Referências

AGRAWAL, S.; GOYAL, N. Analysis of thompson sampling for the multi-armed bandit problem. *Journal of Machine Learning Research*, v. 23, 11 2011. Citado na página 25.

ALIŠAUSKAS, B. *The Complete Guide To Using Proxies For Web Scraping*. 2024. <https://scrapfly.io/blog/posts/introduction-to-proxies-in-web-scraping>. Acesso em: 25 ago. 2025. Citado na página 23.

Amazon Web Services. *Preço sob demanda do Amazon EC2*. 2025. <https://aws.amazon.com/pt/ec2/pricing/on-demand/>. Acesso em: 30 ago. 2025. Citado na página 101.

BAK, J. *Big on data: Study shows why data-driven companies are more profitable than their peers*. 2023. Blog post, Transform with Google Cloud. Disponível em: <https://cloud.google.com/transform/data-leaders-more-profitable-innovative-hbr-data>. Citado na página 19.

BALLA. *Why Rotating Proxies Are Ideal for High-Frequency Internet Tasks*. 2025. The AI Journal (site AI Journ). Acesso em: 24 de agosto de 2025. Disponível em: <https://aijourn.com/why-rotating-proxies-are-ideal-for-high-frequency>. Citado na página 19.

CHANDRA, Y. *How to Rotate Proxies in Python*. 2025. <https://www.zenrows.com/blog/rotate-proxies-python>. Acesso em: 25 ago. 2025. Citado na página 24.

DAVIES, G. Proxy servers. In: *Networking Fundamentals*. Packt, 2019, (Cloud & Networking). cap. 17. Seção "Proxy servers" – Network Services, capítulo 17. Disponível em: <https://subscription.packtpub.com/book/cloud-and-networking/9781838643508/17/ch17lvl1sec27/proxy-servers>. Citado na página 23.

DERI, L.; FUSCO, F. *Evaluating IP Blacklists Effectiveness*. 2023. Citado na página 19.

FasterCapital. *Types Of Proxy Rotation Techniques*. 2025. FasterCapital website. Acesso em: 24 de agosto de 2025. Disponível em: <https://fastercapital.com/topics/types-of-proxy-rotation-techniques.html/1>. Citado na página 20.

GHOMI, E. J.; RAHMANI, A. M.; QADER, N. N. Load-balancing algorithms in cloud computing: A survey. *J. Netw. Comput. Appl.*, Elsevier BV, v. 88, p. 50–71, jun. 2017. Citado na página 26.

HOETZLEIN, R. C. *Protecting Small Organizations from AI Bots with Logrip: Hierarchical IP Hashing*. 2025. Citado na página 19.

JANSEN, M. Bachelor's Thesis, *Recognising Client-side Behavioral Detection of Web Bots*. Nijmegen, The Netherlands: [s.n.], 2021. Supervisors: Dr. Erik Poll and Dr. Hugo Jonker. Citado na página 22.

JONKER, H.; KRUMNOW, B.; VLOT, G. Fingerprint surface-based detection of web bot detectors. In: *Lecture Notes in Computer Science*. Cham: Springer International Publishing, 2019, (Lecture notes in computer science). p. 586–605. Citado na página 22.

KHDER, M. Web scraping or web crawling: State of art, techniques, approaches and application. *Int. J. Adv. Soft Comput. Appl.*, Alzaytoonah University of Jordan, v. 13, n. 3, p. 145–168, dez. 2021. Citado na página 22.

KINAS, P. G.; ANDRADE, H. A. *Introdução à Análise Bayesiana (com R)*. 2. ed. Brasil: Consultor Editorial, 2021. ISBN 9786599008894. Citado 2 vezes nas páginas 24 e 33.

LLAMAS, J. M. et al. Balancing security and privacy: Web bot detection, privacy challenges, and regulatory compliance under the GDPR and AI act. *Open Res. Eur.*, v. 5, p. 76, mar. 2025. Citado na página 19.

MITCHELL, R. Web scraping com python – 3ª edição: Coletando dados da web moderna. In: _____. São Paulo: Novatec Editora, 2023. cap. 20, p. 348–. ISBN 9786558811741. Capítulo 20: Proxies de web scraping. Tradução da 3ª edição original: Web Scraping with Python, O'Reilly Media. Citado na página 19.

NetNut. *All You Need to Know About IP Rotation: What Is It and How To Rotate an IP Address*. 2023. <https://netnut.io/ip-rotation>. Acesso em: 25 ago. 2025. Citado na página 24.

REINSEL, D.; GANTZ, J.; RYDNING, J. *Data Age 2025: The Digitization of the World – From Edge to Core*. 2020. White Paper (sponsored by Seagate, based on IDC data). IDC White Paper #US44413318, May 2020. Disponível em: <https://www.seagate.com/files/www-content/our-story/trends/files/dataage-idc-report-final.pdf>. Citado na página 19.

STOBIERSKI, T. *The Advantages of Data-Driven Decision-Making*. 2019. <https://online.hbs.edu/blog/post/data-driven-decision-making>. Business Insights, Harvard Business School Online. Citado na página 19.

TANENBAUM, A. S.; WETHERALL, D. J. *Computer Networks*. 5. ed. Boston: Prentice Hall, 2011. ISBN 9780132126953. Citado na página 29.

WebHarvy. *Uses of Web Scraping*. 2025. <https://www.webharvy.com/articles/web-scraper-use-cases.html>. Acesso em: 25 ago. 2025. Citado na página 22.

ZHAO, B. Web scraping. In: *Encyclopedia of Big Data*. Cham: Springer International Publishing, 2017. p. 1–3. Citado na página 22.