São José dos Campos - SP
LV SBPO
LV Simpósio Brasileiro de Pesquisa Operacional
São José dos Campos, SP - 6 a 9 de novembro de 2023

IMPLEMENTAÇÃO DE UM ALGORITMO GENÉTICO PARA A ESTIMAÇÃO DE PARÂMETROS DE UM PROCESSO DE RENOVAÇÃO GENERALIZADO

Clarissa Maria Rodrigues de Oliveira[a], Matheus Lopes Bittencourt[a], Maria Rita Rocha Mota[a], Bruno Linhares de Souza[a], Rodrigo José Santana de Macedo[a], Hellen Rayssa Oliveira de Souza[a], Márcio das Chagas Moura[a,b]
Universidade Federal de Pernambuco
[a]Departamento de Engenharia de Produção
[b]Centro de Estudos e Ensaios em Risco e Modelagem Ambiental
Av. da Arquitetura, s/n – Cidade Universitária, Recife - PE, Brasil, CEP: 50740-550
{clarissa.maria, matheus.bittencourt, maria.mrrm, bruno.lsouza, santana.macedo, hellen.souza, marcio.cmoura}@ufpe.br

RESUMO

O estudo da modelagem dos processos de falhas e os tipos de reparos realizados são de grande utilidade para manter e mensurar a confiabilidade de equipamentos. Para análise de confiabilidade em sistemas reparáveis pode-se adotar o Processo de Renovação Generalizada (GRP), modelo probabilístico capaz de englobar cinco estados de um sistema após o reparo. Este artigo implementou uma abordagem *Random Walk* a partir de Simulação Monte Carlo (SMC) para estimar o número de falhas de um processo GRP definido em [Yañez et al., 2002] e um Algoritmo Genético (AG) a fim de realizar comparações. Os resultados mostram que a utilização do AG demonstrou melhor desempenho em relação ao esforço computacional e ao ajuste da curva do número falhas esperado vs. tempo. O trabalho desenvolvido promove uma discussão acadêmica sobre a possibilidade de implementação de modelos GRP combinado com métodos de otimização combinatória em indústrias, visando a melhoria do desempenho.

PALAVRAS CHAVE. Sistemas reparáveis, Processo de renovação generalizada, Algoritmo genético.

ABSTRACT

The study of modeling failure processes and the types of repairs performed are very useful for maintaining and measuring the reliability of equipment. For reliability analysis in repairable systems, the Generalized Renewal Process (GRP) can be adopted, a probabilistic model capable of encompassing five states of a system after repair. This article implemented a *Random Walk* approach from Monte Carlo Simulation (SMC) to estimate the number of failures of a GRP process defined in [Yañez et al., 2002] and a Genetic Algorithm (GA) in order to perform comparisons. The results show that the use of GA showed better performance in relation to the computational effort and the adjustment of the curve of the expected number of failures vs. time. The developed work promotes an academic discussion about the possibility of implementing GRP models combined with combinatorial optimization methods in industries, aiming to improve performance.

KEYWORDS. Repairable systems, Generalized renewal process, Genetic algorithm.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

1. Introdução

Para Syamsundar et al. [2020], a maioria dos sistemas reais pode ser considerado como reparável, no qual a vida útil está baseada no tempo total de uso e/ou rendimento dos componentes. Um sistema é dito reparável se, após uma falha, sua função puder ser satisfatoriamente retomada através de reparo, sem a necessidade de substituir todos os componentes do sistema [Syamsundar et al., 2020]. Os equipamentos reparáveis são passíveis de manutenções, seja por meio de manutenção corretiva ou preventiva, ou seja, uma vez detectada uma falha, sua condição operacional pode ser restaurada através de algum tipo de intervenção, diferente de simplesmente substituição ou descarte [Elsayed, 2020].

Para modelar processos de falhas de um sistema reparável é necessário compreender as diferenças dos modelos e o comportamento do tipo de reparo que foi realizado no equipamento ou componente em estudo, pois sua vida virtual é proporcional às condições referentes à qualidade do reparo no equipamento [Wang e Yang, 2012]. Em detrimento disso, Yañez et al. [2002] apresentam o processo de renovação generalizada (GRP), que é um modelo probabilístico capaz de englobar cinco estados de um sistema após o reparo, sendo eles o Reparo Mínimo (RM), no qual o sistema retorna à condição em que se encontrava no momento imediatamente anterior à falha, Reparo Perfeito (RP) retorna o sistema à condição de novo e às condições de "tão bom quanto novo", Reparo Imperfeito (RF) restaurando o sistema a um estado de "tão ruim quanto velho", Reparo Melhor que o Novo (RMN) e o Reparo Pior que Velho (RPV).

Os *Random Walk* (RW) correspodem a modelos de processos estocásticos que descrevem o caminho de passos aleatórios n, cujo a escolha do passo seguinte não está vinculada a escolha do passo anterior [Xia et al., 2019]. Essa ideia foi introduzida por [Pearson, 1905] e o modelo é amplamente utilizado em diversas áreas [Xia et al., 2019]. Com isso, frequentemente, a análise de confiabilidade para sistemas reparáveis por meio do GRP utiliza Simulação Monte Carlo (SMC) ao invés dos tradicionais métodos numéricos para estimar os parâmetros para um modelo de tempos de falha, visto a complexidade e dificuldade de desenvolver esses modelos probabilístico matematicamente [Wang e Yang, 2012].

As abordagens clássicas usadas para estimar parâmetros são realizadas por meio maximização da função log-verossimilhança (LLF) ou por meio da minimização do erro quadrático médio (EQM) [Kim et al., 2015]. Contudo, a diversificação da aplicação de métodos de otimização combinatória em diversas campos de pesquisa, fez com que a utilização de métodos de busca evolucionária também tenha sido aplicada para a estimação de parâmetos, como o algoritmo genético (AG) [Kim et al., 2015; Hsu e Huang, 2010], evolução diferencial (ED) [Wu et al., 2020] e o algoritmo de otimização de enxame de partícula (PSO) [Jin e Jin, 2016]. Esses métodos tem sido amplamente implementados devido às vantagens presentes em suas estruturas, como a capacidade de resolução de problemas complexos, a qualidade das soluções encontradas e a simplicidade de implementação [Yaghoobi, 2020].

O AG é uma meta-heurística de busca populacional baseada na evolução natural, a qual aplica princípios que guiam na procura de respostas com qualidade, sendo um método que possui implementação em instâncias que dispõem de um amplo espaço de soluções e diversas aplicações práticas nas mais diferentes áreas, se destacando por sua robustez e eficiência [Mirjalili et al., 2020]. Diante disso, esse trabalho apresenta como objetivo principal a implementação do AG aplicado para a estimação de parâmetros de um (GRP) para sistemas reparáveis. A abordagem proposta é comparada com a metodologia desenvolvida por Yañez et al. [2002], que utiliza um modelo baseado na Simulação Monte Carlo para aferir os parâmetros do modelo.

O restante do artigo está organizado da seguinte maneira: a Seção 2 apresenta a descrição

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

do problema GRP abordado na pesquisa; a Seção 3 contém conceitos relacionados a AG; a Seção 4 descreve a abordagem utilizada para o problema GRP implementado nesse trabalho; a Seção 5 demonstra os resultados e experimentos computacionais realizados; por fim, a Seção 6 apresenta as conclusões e perspectivas para trabalhos futuros.

2. Processo de Renovação Generalizado - GRP

Em GRP é apresentado o conceito de era virtual (A_n), que representa a idade calculada do sistema imediatamente após a ocorrência do enésimo reparo, dessa maneira, se $A_n = y$, então o sistema tem um tempo para a $(n+1)$ésima falha acontecer $(X_n + 1)$, que é distribuído de acordo com uma função acumulativa de probabilidades (CDF) [Yañez et al., 2002]. Como demonstrado na Equação (1)

$$F(X|A_n = y) = \frac{F(X+y) - F(y)}{1 - F(y)} \tag{1}$$

Onde $F(x)$ é a CDF da distribuição do tempo até a primeira falha de um novo sistema ou componente. Assim, o somatório, considerando $S_0 = 0$, é conhecido como idade real do sistema. Sabendo que:

$$S_n = \sum_{i=1}^{n} X_i \tag{2}$$

Com isso, supõe-se que o enésimo reparo compensaria apenas o dano acumulado entre o tempo da $(n-1)$ésima e a (n)ésima falha. Diante disso, a idade virtual do sistema, após o (n)ésimo reparo é:

$$A_n = A_{n-1} + qX_n = qS_n, n = 1, 2, ... \tag{3}$$

No qual, q representa a efetividade do reparo (ou parâmetro de rejuvelhecimento), portanto, considerando o tempo entre todas as n falhas consecutivas, temos que:

$$A_n = q\sum_{i=1}^{n} X_i \tag{4}$$

De acordo com este modelo, o parâmetro q pode assumir um valor de $q = 0$, que representa o reparo perfeito (tão bom quanto novo), enquanto o pressusposto de $q = 1$ representa um reparo mínimo (tão ruim quanto velho), e os valores contidos no intervalo $0 < q < 1$ representam o estado pós-reparo no qual a condição do sistema é melhor do que o antigo, porém pior do que o novo. Para casos em $q > 1$, o sistema está em uma condição pior que antes do reparo, e o $q < 0$ sugere que o sistema foi restaurado para uma condição melhor do que novo [González-Prida et al., 2014]. Dessa forma, o q pode ser entendido como um índice que representa a eficácia e a qualidade do reparo [Yañez et al., 2002].

A distribuição Weibull é a mais comumente usada na análise de confiabilidade para sistemas reparáveis devido a sua flexibilidade e aplicabilidade a vários processos de falha[González-Prida et al., 2014]. Os trabalhos de Kijima e Sumita [1986] e Krivtsov [2000] utilizam essa distribuição para modelar as falhas. Em Yañez et al. [2002], o primeiro tempo entre falhas é modelado por uma distribuição Weibul, com CDF:

$$F(t_i) = 1 - exp\left(-\left(\frac{t_i}{\alpha}\right)^{\beta}\right) \tag{5}$$

Logo:

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

$$t_i = \alpha \times \{-ln[1 - F(t_i)]^{1/\beta}\} \text{ para } i = 1 \tag{6}$$

E os tempos subsequentes seguem uma distribuição Weibul condicionada, com CDF:

$$F(t_i) = 1 - exp\left[\left(\frac{q}{\alpha}\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(\frac{t_i + q\sum_{j=1}^{i-1} t_j}{\alpha}\right)^{\beta}\right] \tag{7}$$

Então:

$$t_i = \left\{\left[\frac{q}{a}\sum_{j=1}^{i-1} t_j\right]^{\beta} - ln[1 - F(t_i)]\right\}^{1/\beta} - \frac{q}{\alpha}\sum_{j=1}^{i-1} t_j, \text{para } i = 2, 3, ..., n \tag{8}$$

Com isso, o modelo GRP possui a seguinte PDF derivada:

$$f(t_i) = \left(\frac{\beta}{\alpha^{\beta}}\right)\left[t_i + q\sum_{j=1}^{i-1} t_j\right]^{\beta-1} \times exp\left[\left(\frac{q}{\alpha}\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(\frac{ti + q\sum_{j=1}^{i-1} t_j}{\alpha}\right)^{\beta}\right], \text{para } i = 2, 3, ..., n \tag{9}$$

Portanto, a função de log-verossimilhança dos tempos entre falhas para um sistema reparável sem censura de tempo é:

$$LLF = ln\left(\frac{\beta}{\alpha}\right) + ln\left[\left(\frac{t_1}{\alpha}\right)^{\beta-1}\right] - \left(\frac{t_1}{\alpha}\right)^{\beta} + (n-1)ln\left(\frac{\beta}{\alpha^{\beta}}\right) + \sum_{i=2}^{n} ln\left[\left(t_i + q\sum_{j=1}^{i-1} t_j\right)^{\beta-1}\right] + \sum_{i=2}^{n}\left[\left(\frac{q}{\alpha}\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(\frac{t_i + q\sum_{j=1}^{i-1} t_j}{\alpha}\right)^{\beta}\right] \tag{10}$$

E com censura de tempo, no qual T representa o truncamento (a diferença entre o tempo total observado e o último tempo resgistrado de falha), é apresentada:

$$LLF = ln\left(\frac{\beta}{\alpha}\right) + ln\left[\left(\frac{t_1}{\alpha}\right)^{\beta-1}\right] - \left(\frac{t_1}{\alpha}\right)^{\beta} + (n-1)ln\left(\frac{\beta}{\alpha^{\beta}}\right) + \sum_{i=2}^{n} ln\left[\left(t_i + q\sum_{j=1}^{i-1} t_j\right)^{\beta-1}\right] + \sum_{i=2}^{n}\left[\left(\frac{q}{\alpha}\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(\frac{t_i + q\sum_{j=1}^{i-1} t_j}{\alpha}\right)^{\beta}\right] + \left(\frac{q}{\alpha}\sum_{j=1}^{n} t_j\right)^{\beta} - \left(\frac{T + q\sum_{j=1}^{n} t_j}{\alpha}\right)^{\beta} \tag{11}$$

Dessa forma, realizando a diferenciação das duas funções LLF apresentadas (Equação (10) e (11)) em relação aos parâmetros α, β e q e igualando cada derivada a zero, encontra-se um sistema de três equações com três variáveis desconhecidas. Essas são extremamente complexas e requerem métodos númericos e inovadores para a sua solução. Assim, apenas para os α's obtem-se

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

uma forma fechada, como observado nas Equações (12) e (13), as outras variáveis ficam inviávies de serem isoladas como pode ser observado em [Yañez et al., 2002], sendo necessário a utilização de outros métodos para a obtenção e estimação desses valores.

$$\alpha = \sqrt[\beta]{\frac{\sum_{i=2}^{n}\left[\left(t_i + q\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(q\sum_{j=1}^{i-1} t_j\right)^{\beta}\right] + t_1^{\beta}}{n}} \tag{12}$$

$$\alpha = \sqrt[\beta]{\frac{(t_1)^{\beta} + \sum_{i=2}^{n}\left[\left(t_i + q\sum_{j=1}^{i-1} t_j\right)^{\beta} - \left(q\sum_{j=1}^{i-1} t_j\right)^{\beta}\right] - \left(q\sum_{j=1}^{n} t_j\right)^{\beta} + \left(T + q\sum_{j=1}^{n} t_j\right)^{\beta}}{n}} \tag{13}$$

O problema abordado neste artigo se refere a estimação dos parâmetros α, β e q para maximizar as Equações (10) e (11), onde β e q são gerados aleatoriamente e α é calculado em função desses dois parâmetros.

3. Algoritmo Genético

O algoritmo genético é uma meta-heurística de busca populacional e funciona de modo análogo à evolução natural, a partir da qual são analisados os indivíduos, de uma determinada população, cujas características genéticas o façam ter uma maior probabilidade de sobrevivência e de produzirem filhos com essas características, enquanto os indivíduos menos aptos tenderiam a desaparecer [Katoch et al., 2021]. Com isso, alguns termos da teoria da evolução assumem um novo significado no âmbito computacional, a exemplo disso, tem-se os cromossomos, estrutura composta por genes, que, nesse contexto, representam os indivíduos que compõem a solução do problema a ser otimizado [Mirjalili et al., 2020].

Visando utilizar esses algoritmos, deve-se, primeiro, gerar a população inicial, a partir da qual ocorrerão os cruzamentos para criar a próxima geração. Além disso, é necessário avaliar a função *fitness* desses indivíduos gerados, com a finalidade de analisar o quão bom é o indivíduo do ponto de vista da função que se deseja otimizar. Com esses valores já definidos, selecionam-se os indivíduos mais aptos para a operação de *crossover*, a partir da qual esses cromossomos são combinados, a fim de gerar filhos para a próxima geração [Reis, 2018].

Contudo, mesmo que os filhos gerados sejam uma combinação de boas soluções, aplica-se a operação de mutação objetivando criar uma diversidade através de uma mudança aleatória em um ou mais genes dos indivíduos gerados após o *crossover*. Essa ação também é utilizada para evitar que o algoritmo fique preso em mínimos ou máximos locais [Reis, 2018]. Por fim, após a adição da nova geração de filhos, é realizada uma nova avaliação dos indivíduos da população e, aqueles que sobrevivem, são os responsáveis por definir a geração dos novos elementos na próxima iteração [Lambora et al., 2019]. Assim, todos esses processos do algoritmo AG pode ser observado na Figura 1.

Como demonstrado na Figura 1, e complementado por Lambora et al. [2019], a exploração do espaço de busca ocorre por meio de um conjunto de soluções, que serão submetidas aos processos de recombinação genética, de mutação, *crossover* e seleção, permitindo que ocorra a evolução da população, dando-se uma maior probabilidade de reprodução para as soluções com maior qualidade, semelhante ao que ocorre na seleção natural. E, dentro dessas etapas, existem diversos parâmetros, como a taxa de mutação e de *crossover*, número de iterações e tamanho da população, tais parâmetros influenciam diretamente nas respostas obtidas pelo método, e os mesmos não devem ser escolhidos de forma genérica [Chebbi e Chaouachi, 2015].

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

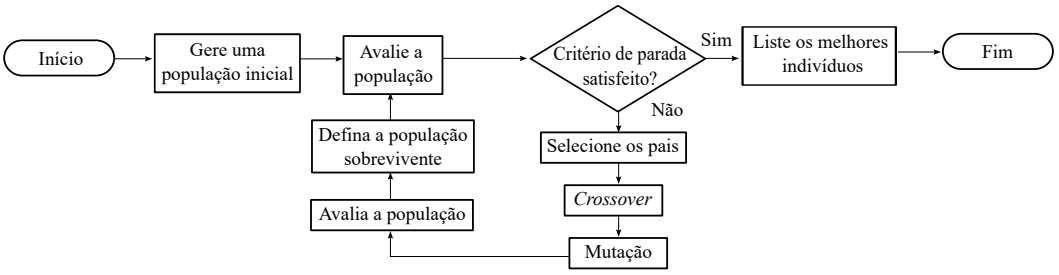


Figura 1: Estrutura do algoritmo genético

Nesse contexto, um algoritmo genético possibilita, por meio de várias repetições lógicas, o refinamento da população inicial de indivíduos, de maneira que, com o passar das interações, as gerações que são concebidas carregarão as melhores características dos seus pais. Portanto, ao final do algoritmo, é possível identificar qual cromossomo (solução) possui o melhor *fitness* (i.e. função objetivo) daquela população [Colherinhas, 2016]. Entretanto, é importante destacar que nem sempre um algoritmo genético encontra a solução ótima para um problema, porém, frequentemente ela é capaz de se aproximar dela [Linden, 2012].

4. Abordagem proposta

O artigo de Yañez et al. [2002] propõe uma abordagem para análise de confiabiliadade de sistemas reparáveis para o caso do GRP aplicado a dois exemplos, um considerando falhas sem censura e um com censura, no qual estima-se os parâmetros por meio de um algoritmo RW, utilizando Simulação Monte Carlo para prever o número esperado de falhas para uma distribuição Weibull.

Na Figura 2 observa-se a Simulação Monte Carlo utilizada para estimar os parâmetros GRP, em que valores aleatórios para β são gerados dentro do intervalo contínuo de 0 a 10 e q, no intervalo contínuo de 0 a 1, de forma a selecionar os parâmetros no qual as suas derivadas do logaritmo da verossimilhança (ver em [Yañez et al., 2002]) respeitem uma determinada tolerância, implementadas nesse artigo como sendo 10^{-3}. Dessa forma, ao escolher β e q, α é definido conforme uma equação fechada apresentada nas Equações (12) e (13).

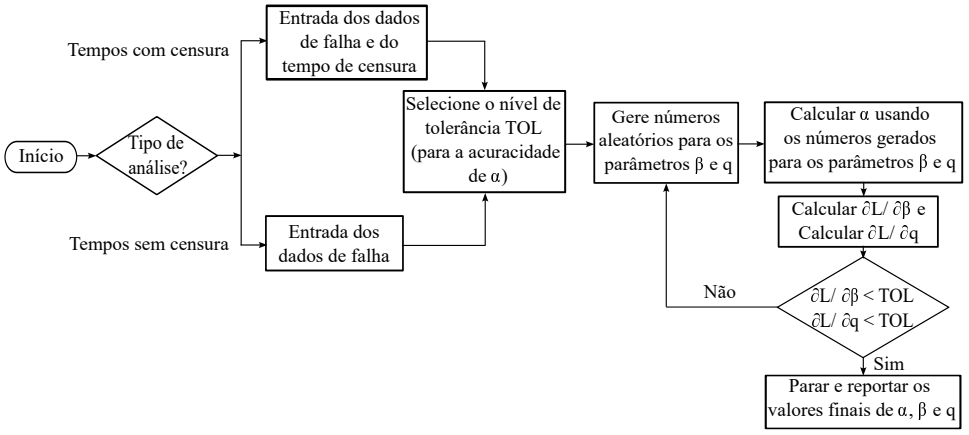


Figura 2: Simulação Monte Carlo para estimar os parâmetros do GRP. Fonte: Yañez et al. [2002]

Em prosseguimento, conforme a Figura 2 uma Simulação Monte Carlo é utilizada para gerar números aleatórios com probabilidades uniformes no intervalo de 0 a 1 para a função não-

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

confiabilidade, a fim de obter valores aleatórios para o tempo t_i, conforme a Equação (7) para o primeiro tempo de falha e Equação (8) para os tempos subsequentes (no qual utilizam-se os parâmetros α, β e q estimados anteriormente). Assim, para cada t_i gerado é contabilizado uma falha até que os $t_i's$ acumulados ultrapassem um tempo T observado. Dessa forma, após uma quantidade de n componentes analisados no período T, o número esperado de falhas para esse intervalo de tempo é calculado conforme o número total de falhas observadas dividido pelo total de componentes testados n, verificado na Equação (14):

$$E[N(T)] = \sum_{j=1}^{m} \frac{n_j}{m} \tag{14}$$

m é a quantidade de tempos aleatórios gerados para cada componente simulado e tempo de interesse (Ver em [Yañez et al., 2002]). É permitido averiguar o ajuste da curva do número de falhas esperada vs. tempo por meio do calculo de EQM, como demonstrado na Equação (15).

$$EQM = \frac{1}{n} \sum_{i=1}^{n} (X_i - Y_i)^2 \tag{15}$$

X_i é o número de falhas observado na amostra, Y_i é o número de falhas esperado encontrado pelo modelo. Dessa forma, esse método pode ser utilizado como uma medida de qualidade dos modelos apresentados, almejando-se que o valor do EQM seja o menor possível, descrevendo uma distância média menor entre o estimador e o parâmetro.

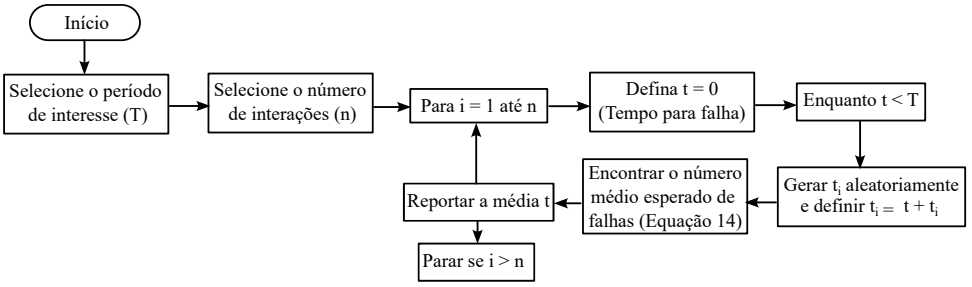


Figura 3: Simulação Monte Carlo para estimação do número de falhas. Fonte: Yañez et al. [2002]

É possível observar que o método proposto pelos autores para estimar os parâmetros de uma função de confiabilidade para uma determinada amostra analisada não apresenta grande eficiência, dado que não emprega conhecimentos obtidos anteriormente pelos testes de combinação de parâmetros para estimar, com maior precisão, melhores combinações. Portanto, métodos de otimização combinatória podem ser uma estratégia de busca mais eficiente no espaço de soluções do problema.

Verifica-se que o critério de EQM obtem um melhor ajuste para a curva do número de falhas esperada vs. tempo, no entanto, utilizá-lo como função de otimização ocasiona em um custo computacional muito alto, por conta disso, um meio alternativo comumente utilizado é a aplicação de métodos de otimização para estimar os parâmetros de forma a maximizar a LLF. Portanto, no caso do algoritmo desse trabalho implementa-se o AG para definir os parâmetros α, β e q, objetivando maximizar a LLF. Contudo, ao final (n-ésima geração), ao invés de selecinar o indivíduo, ou melhor, a solução de maior valor *fitness* (LLF), é escolhida aquela com o melhor ajuste em relação ao número de falhas estimada e a observada na amostra, isto é, o menor EQM.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

O AG implementado inicia com a geração de uma população inicial p aleatoriamente, no qual cada indivíduo é composto por 3 genes, sendo o primeiro referente ao valor de β (gerado aleatoriamente seguindo uma distribuição uniforme no intervalo de 0 a 10), o segundo representa o valor de q (gerado aleatoriamente seguindo uma distribuição uniforme no intervalo de 0 a 1) e o último corresponde ao α obtido por equação fechada (Equações (12) e (13)).

Em seguida, dois pais (P_1 e P_2) dessa população são selecionados confome a aptidão, no qual soluções com maior valor *fitness* apresentam maiores probabilidades de serem escolhidos para a realização da combinação (*crossover*), gerando dois filhos. O *crossover* realizado é o aritmético, no qual define-se um β e q para cada filho (F_1 e F_2) conforme as Equações (16) e (17) que insere características dos pais nestes. Por outro lado, o α é obtido para cada filho por equação fechada de acordo com os outros dois parâmetros calculados, no qual λ é um peso aleatório responável por definir uma média ponderada dos pais, pertencente ao intervalo [0, 0,5].

$$F_1 = \lambda P_1 + (1 - \lambda)P_2 \tag{16}$$

$$F_2 = \lambda P_2 + (1 - \lambda)P_1 \tag{17}$$

Após a combinação que gera os filhos, um dos dois, escolhido aleatoriamente, é submetido a um processo de mutação conforme uma probabilidade p_m pertencente ao intervalo [0, 1], no qual altera-se um dos genes β ou q, selecionados aleatoriamente, obtendo um α correspondente. Seguindo para o processo de substituição da população, no qual os filhos formados a cada geração são comparados com os piores indivíduos da população atual, isto é, aqueles com menores valores de *fitness* (LLF), portanto, se apresentarem valor de LLF melhores que os deles, os substituem para a próxima geração. Assim, esse processo se repete por η gerações, retornando, ao final, o indivíduo da η-ésima população que apresenta o menor EQM.

5. Experimentos Computacionais

O GRP em Yañez et al. [2002] e o AG descritos na Seção 4 foram implementados em linguagem de programação Julia, realizando os experimentos computacionais em um computador com processador Intel(R) Core (TM) i5-7200 com CPU de @2.50GHz e 8 GB de memória RAM, com sistema operacional Windows 10. Além disso, utilizou-se o pacote IRace de López-Ibáñez et al. [2016] para calibração automática dos parâmetros dos algoritmos, tomando como base 4 instâncias de treinamento, as quais foram resolvidas 1500 vezes (*runs of budget*).

As quatro instâncias de treinamento, referente aos tempos de falha, foram geradas aleatoriamente seguindo diferentes configurações de características para os parâmetros de escala (α), forma (β) e rejuvelhecimento q, os quais foram determinados de maneira aleatória e uniforme no intervalo [0, 10000], [0, 10] e [0, 1], respectivamente.

5.1. Calibração dos parâmetros

Visando obter um melhor desempenho dos algoritmos, os parâmetros p (número da população inicial), t_m (Taxa de mutação), λ (operador do cruzamento) e η (Número de gerações) foram calibrados utilizando um intervalo de valores categóricos para delimitar os testes do IRace, apresentados na Tabela 1 na coluna "Variação", o qual exibe os intervalos de valores considerados e os passos. Assim, o referido software definiu os valores exibidos na coluna "Resposta" como o melhor conjunto dos parâmetros do AG, que minimizam o EQM, para os dois tipos de amostras em estudo (tempos de falha com e sem censura). No qual, para o segundo considerou-se um tempo de censura T de 300.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

Tabela 1: Parâmetros

Parâmetros	Descrição	Variação	Resposta	
			Sem censura	Com censura
p	População inicial	(50 : 25 : 150)	150	150
t_m	Taxa de mutação	(0.05 : 0.0 : 0.10)	0.10	0.10
λ	Operador do cruzamento	(0.30 : 0.0 : 0.40)	0.40	0.40
η	Número de gerações	(200 : 50 : 400)	200	200

5.2. Resultados computacionais

Para avaliar os algoritmos propostos, foram resolvidas as mesmas instâncias de experimentação consideradas por Yañez et al. [2002] o qual abordam tempos entre 24 falhas. Implementou-se o algoritmo evolucionário AG e a busca por RW conforme proposto pelos autores, rodando os algoritmos um total de 5 vezes e selecionando a solução de menor EQM. Os resultados obtidos foram comparados com aqueles encontrados pelos mesmos, o qual fizeram uso de uma abordagem de RW para estimar os parâmetros do modelo GRP explanado anteriormente.

A Tabela 2 exibe os resultados para os parâmetros α, β e q, além do LLF, EQM, o tempo computacional em segundos para cada abordagem explanada e o tipo de falha (sem e com censura, representado por i_1 e i_2). Na primeira e na segunda coluna, observam-se os dados obtidos com o AG, as duas colunas seguintes correspondem as respostas encontradas para a implementação do RW, e as duas próximas são referentes aos resultados vistos em Yañez et al. [2002].

Tabela 2: Resultados para os parâmetros do GRP, LLF, EQM e tempos computacionais em segundos

Resultado	Algoritmo Genético		*Random Walk*		Yañez et al. [2002]	
	i_1	i_2	i_1	i_2	i_1	i_2
α	1956.76	1572.32	2933.40	1827.39	3072	1828
β	1.399	2.131	1.547	2.026	1.62	2.026
q	0.287	0.080	0.723	0.146	0.70	0.146
LLF	-189.63	-182.27	-189.30	-181.79	-189.34	-181.79
EQM	0.662	0.896	1.707	1.243	2.388	1.355
$Tempo$	16.00	19.92	53.12	3937.03	–	–

A Tabela 3 exibe na coluna "AG vs. Yañez et al. [2002]" uma relação entre o EQM obtido por meio do AG e o definido em Yañez et al. [2002] para os dados sem e com sensura (representados pelas instâncias i_1 e i_2) e igualmente a coluna "AG vs. *Random Walk*" exibe uma relação o entre os EQM's das abordagens AG e RW.

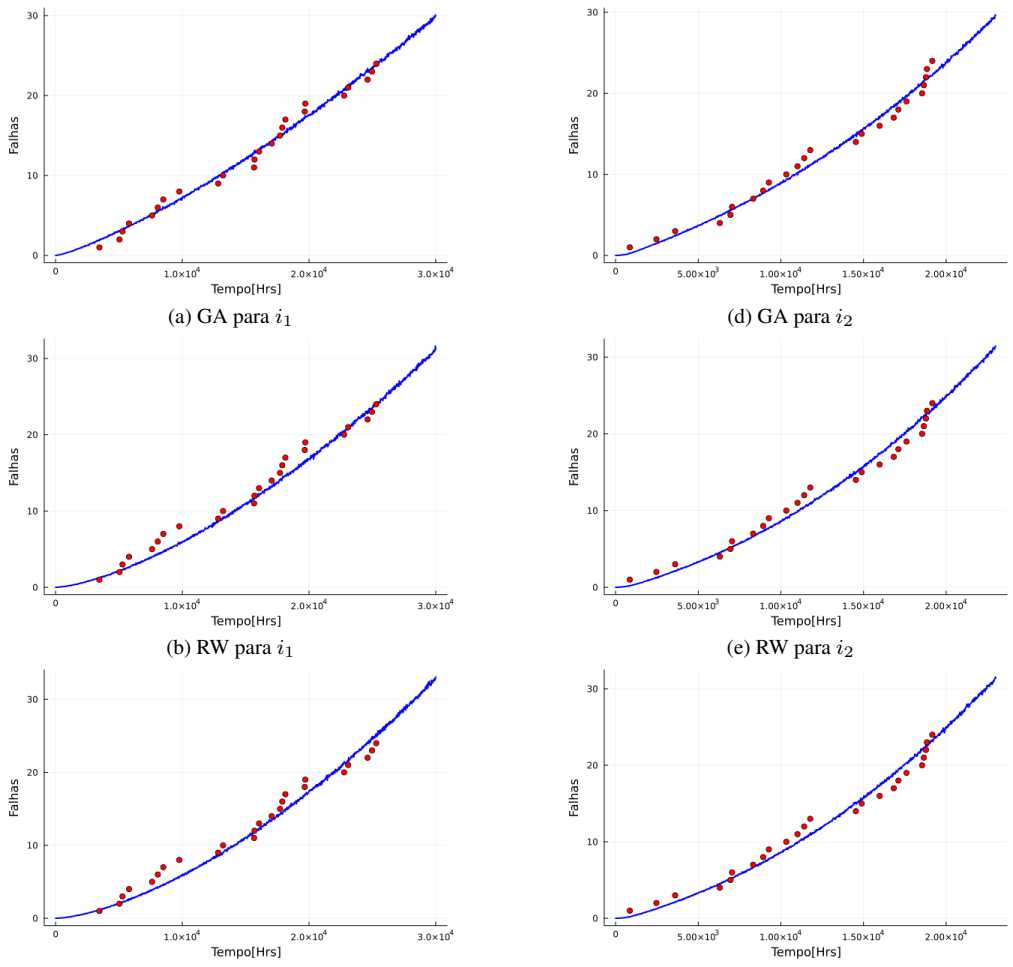
Tabela 3: GAP's

Instâncias	AG vs. Yañez et al. [2002]	AG vs. *Random Walk*
i_1	0.723	0.612
i_2	0.339	0.279

Verifica-se na Tabela 2 que, para os dois casos, o conjunto de parâmetros de GRP obtém dados de LLF significamente próximos, no entanto, por meio da análise do EQM, observam-se ganhos superiores nas respostas do AG, onde, para a instância i_1, o ajuste do número de falhas esperado pelo tempo foi 72.3% melhor que o do artigo de Yañez et al. [2002] (com o número esperado de falhas no tempo alcançado conforme α, β e q do referido trabalho). O primeiro apresenta EQM de 0.662 e o segundo de 2.388 (i.e., cerca de 3.6 vezes maior), já para i_2 essa melhora é de 33.8% (Tabela 3). Comparando a implementação do RW com o AG também é permitido averiguar um

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

EQM menor em 61.20% para i_1 e de 27.9% para i_2 (Tabela 3). É possível observar, na Figura 4, o ajuste da curva de número de falhas pelo tempo para os três conjuntos de parâmetros mostrados como resultado, no qual os pontos em vermelho são referentes aos dados de tempos de falhas das instâncias i_1 e i_2 e em azul apresenta-se a predição do modelo.



(a) GA para i_1 (d) GA para i_2

(b) RW para i_1 (e) RW para i_2

(c) Yañez et al. [2002] para i_1 (f) [Yañez et al., 2002] para i_2

Figura 4: Ajuste dos parâmetros α, β e q obtidos pelo GA, RW e de Yañez et al. [2002] para os dados i_1 e i_2

Comparando os esforços computacionais, observa-se que o AG encontrou respostas melhores em tempos significativamente menores (16 e 19.92 segundos para i_1 e i_2). Em contrapartida, a abordagem RW implementada com base nos algoritmos propostos de Yañez et al. [2002] obteve respostas em 53.12 e 3937.03 segundos.

6. Conclusões

O presente estudo implementou a abordagem RW e um algoritmo genético para estimar os parâmetros de um modelo GRP para dados de tempos de falhas com e sem censura. As abordagens foram comparadas utilizando as instâncias experimentais propostas em Yañez et al. [2002]. Os resultados experimentais mostraram que apesar do método de Simulação de Monte Carlo consistir em uma maneira alternativa diante da complexidade e dificuldade de resolução matemática de modelos probabilísticos, a definição de valores aleatórios para os parâmetros de forma a maximizar a LLF torna a abordagem extremamente demorada e de pouca eficiência.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

Assim, permitiu-se ratificar que a utilização de métodos de otimização combinatória para a obtenção de soluções aproximadas promove uma busca eficiente, culminando em soluções promissoras com menores esforços computacionais. Isso foi perceptível na obtenção dos parâmetros para o modelo GRP com valores satisfatórios para estimação do número de falhas de componentes não reparáveis em um determinado tempo, evidenciado pela análise do EQM.

Diante disso, os resultados encontrados nesse artigo são relevantes tendo em vista que promovem discussões acadêmicas acerca do tema e instigam a possibilidade de implementação de modelos GRP combinado a métodos de otimização combinatória (i.e., como o AG desenvolvido nesse artigo) em indústrias com o intuito de melhorar desempenhos. Para tanto, como trabalhos futuros, sugere-se a implementação de outras meta-heurísticas de otimização de forma a realizar comparações em busca de métodos que promovam melhores resultados, além da aplicação em casos reais.

Referências

Chebbi, O. e Chaouachi, J. (2015). Effective parameter tuning for genetic algorithm to solve a real world transportation problem. In *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR)*, p. 370–375.

Colherinhas, G. B. (2016). Ferramenta de otimização via algoritmos genético com aplicações em engenharia. Mestrado em engenharia mecânica, Faculdade de Tecnologia - Universidade de Brasília, Brasília.

Elsayed, A. (2020). *Reliability Engineering*. Wiley, 3 edition.

González-Prida, V., Barberá, L., Márquez, A. C., e Fernández, J. G. (2014). Modelling the repair warranty of an industrial asset using a non-homogeneous Poisson process and a general renewal process. *IMA Journal of Management Mathematics*, 26(2):171–183. ISSN 1471-678X. URL https://doi.org/10.1093/imaman/dpu002.

Hsu, C.-J. e Huang, C.-Y. (2010). A study on the applicability of modified genetic algorithms for the parameter estimation of software reliability modeling. In *2010 IEEE 34th Annual Computer Software and Applications Conference*, p. 531–540.

Jin, C. e Jin, S.-W. (2016). Parameter optimization of software reliability growth model with s-shaped testing-effort function using improved swarm intelligent optimization. *Applied Soft Computing*, 40:283–291. ISSN 1568-4946. URL https://www.sciencedirect.com/science/article/pii/S1568494615007723.

Katoch, S., Chauhan, S. S., e Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80:8091–8126.

Kijima, M. e Sumita, U. (1986). A useful generalization of renewal theory: counting processes governed by non-negative markovian increments. *Journal of Applied Probability*, 23(1):71–88.

Kim, T., Lee, K., e Baik, J. (2015). An effective approach to estimating the parameters of software reliability growth models using a real-valued genetic algorithm. *Journal of Systems and Software*, 102:134–144. ISSN 0164-1212. URL https://www.sciencedirect.com/science/article/pii/S0164121215000023.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995

Krivtsov, V. (2000). *A Monte Carlo Approach to Modeling and Estimation of the Generalized Renewal Process in Repairable System Reliability Analysis*. University of Maryland, College Park. URL https://books.google.com.br/books?id=ANppNwAACAAJ.

Lambora, A., Gupta, K., e Chopra, K. (2019). Genetic algorithm- a literature review. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, p. 380–384.

Linden, R. (2012). *Algoritmos genéticos*. Ciência Moderna, 3 edition.

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Birattari, M., e Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58.

Mirjalili, S., Song Dong, J., Sadiq, A. S., e Faris, H. (2020). *Genetic Algorithm: Theory, Literature Review, and Application in Image Reconstruction*, p. 69–85. Springer International Publishing, Cham. ISBN 978-3-030-12127-3.

Pearson, K. (1905). The problem of the random walk. *Nature*, 72(1865):294–294.

Reis, L. R. D. (2018). Determinação de parâmetros de módulo fótovoltaicos utilizando o método de newton raphson e algoritmos genéticos. Master's thesis, Universidade Federal de Uberlândia, Uberlância.

Syamsundar, A., Naikan, V., e Wu, S. (2020). Alternative scales in reliability models for a repairable system. *Reliability Engineering & System Safety*, 193:106599. ISSN 0951-8320. URL https://www.sciencedirect.com/science/article/pii/S0951832018313310.

Wang, Z.-M. e Yang, J.-G. (2012). Numerical method for weibull generalized renewal process and its applications in reliability analysis of nc machine tools. *Computers & Industrial Engineering*, 63(4):1128–1134. ISSN 0360-8352. URL https://www.sciencedirect.com/science/article/pii/S0360835212001787.

Wu, Y., Chen, R., Li, C., Zhang, L., e Cui, Z. (2020). Hybrid symbiotic differential evolution moth-flame optimization algorithm for estimating parameters of photovoltaic models. *IEEE Access*, 8: 156328–156346.

Xia, F., Liu, J., Nie, H., Fu, Y., Wan, L., e Kong, X. (2019). Random walks: A review of algorithms and applications. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(2): 95–107.

Yaghoobi, T. (2020). Parameter optimization of software reliability models using improved differential evolution algorithm. *Mathematics and Computers in Simulation*, 177:46–62. ISSN 0378-4754. URL https://www.sciencedirect.com/science/article/pii/S0378475420301142.

Yañez, M., Joglar, F., e Modarres, M. (2002). Generalized renewal process for analysis of repairable systems with limited failure experience. *Reliability Engineering & System Safety*, 77 (2):167–180. ISSN 0951-8320. URL https://www.sciencedirect.com/science/article/pii/S0951832002000443.

DOI: 10.59254/sbpo-2023-174995
http://live-proceedings.pantheonsite.io/p/174995