

SELEÇÕES DE SUBSTITUIÇÃO EM UM ALGORITMO GENÉTICO DE CODIFICAÇÃO REAL APLICADO À OTIMIZAÇÃO DE UM CICLO RANKINE ORGÂNICO DE DUPLO ESTÁGIO

Guilherme de Paula Prado, guilhermep Prado@alunos.utfpr.edu.br¹
Renan Manozzo Galante, rmgalante@utfpr.edu.br²

Universidade Tecnológica Federal do Paraná (UTFPR), Câmpus Guarapuava, Avenida Professora Laura Pacheco de Bastos, 800. CEP 85053-525. Guarapuava- PR - Brasil,

Resumo. Problemas de otimização são comuns aos projetos de sistemas de engenharia. Nos sistemas térmicos, mais especificamente, pode-se desejar otimizar a potência, a transferência de calor, a eficiência térmica, custos totais, etc. Estes sistemas possuem muitos parâmetros de otimização, cujo os quais, a influência no funcionamento do sistema é imprevisível e de comportamento não linear. Devido à complexidade na otimização de sistemas térmicos, a utilização de algoritmos avançados de otimização torna-se atrativa. O presente trabalho se propõe a comparar a eficiência de um algoritmo genético de codificação real na otimização de um ciclo Rankine orgânico de duplo estágio, considerando-se dois tipos de seleção de substituição. Resultados apontaram para a convergência mais rápida e para valores mais próximos entre si quando utilizada a seleção $(\mu + \lambda)$. A seleção (μ, λ) apresentou uma maior dispersão entre os valores ótimos encontrados, o que pode estar relacionado à maior diversidade da população.

Palavras chave: Algoritmo Genético. Otimização. Seleção...

Abstract. Optimization problems are common in designing engineering systems. More specifically, in thermal systems, it is desirable to optimize the net power, the heat transfer, thermal efficiency, total costs, etc. These systems have various optimization parameters which influence into system operation is unpredictable and with nonlinear behavior. Due the complexity in optimizing thermal system, the use of advanced algorithms of optimization become attractive. The present work propose to compare the efficiency of a real-coded genetic algorithm in optimizing a double stage organic Rankine cycle, considering two types of replacement selection. Results point to a fastest convergence and for values closer each other when using the $(\mu + \lambda)$ selection. The (μ, λ) selection show a greater dispersion between the optimal values found, which may be related to the greater diversity of the population.

Keywords: Genetic Algorithm, Optimization, Selection...

1. INTRODUÇÃO

1.1 Otimização

Otimização é a tarefa de encontrar a melhor solução para um determinado problema. Geralmente procura-se por uma solução ótima global, cuja a qual é a melhor solução para todo o domínio do problema. Problemas de otimização são comuns em diferentes áreas, desde a vida cotidiana às áreas da matemática, engenharia, ciências naturais e sociais, ciência da computação, entre outras. De forma resumida, toda situação em que procura-se maximizar algo, como lucro, ou minimizar algo, como gasto energético, trata-se de um problema de otimização (Kramer, 2017).

Matematicamente, um problema de otimização pode ser descrito de forma genérica como

$$\min_{x \in \mathbb{R}} f_i(x), \quad (i = 1, 2, \dots, M), \quad (1)$$

$$\text{sujeita a } \phi_j(x) = 0, \quad (j = 1, 2, \dots, J), \quad (2)$$

$$\psi_k(x) \leq 0, \quad (k = 1, 2, \dots, K), \quad (3)$$

em que $f_i(x)$, $\phi_j(x)$ e $\psi_k(x)$ são funções do vetor

$$X = \langle x_1, x_2, \dots, x_n \rangle^T. \quad (4)$$

Os componentes x_i do vetor x são chamados parâmetros ou variáveis de decisão ou otimização. As funções f_i são chamadas de funções objetivo e, para os casos em que $M = 1$, há apenas uma função objetivo. O domínio que contém as variáveis de decisão é chamado de espaço de busca \mathbb{R}^n . As igualdades para $\phi_j(x)$ e as desigualdades para $\psi_k(x)$ são restrições deste espaço de busca. As desigualdade também podem ser do tipo ≥ 0 e, a otimização, formulada como um problema de maximização (Yang, 2010).

Nos diversos campos da engenharia, problemas de otimização são abundantes. Otimização é recorrente em aplicações de projeto, modelagem, caracterização, controle, produção, etc. Muitos problemas da vida real possuem mais de uma solução e, assim, a otimização pode ser realizada encontrando-se o conjunto de parâmetros/variáveis que soluciona o problema da melhor forma, em termos de algum critério de desempenho. À melhor solução possível, dá-se o nome de solução ótima. Problemas que admitem somente uma solução não são otimizáveis, pois apenas um único conjunto de parâmetros é aceitável (Andreas Antoniou, 2007).

Lopes e Takahashi (2011) afirmam que o projeto de sistemas de engenharia requer o uso de técnicas capazes de tratar do grande número de soluções que estes sistemas possuem, abordando um número reduzidos de soluções consideradas ótimas que, implicam em sistemas econômicos, confiáveis e de bom desempenho. Estas técnicas são chamadas de técnicas ou métodos de otimização que, com o avanço da tecnologia, tornam-se cada vez mais frequentes e importantes.

A otimização é um procedimento comum ao projeto de sistemas térmicos, onde considera-se um requisito específico, como transferência de calor, eficiência térmica, capacidade de resfriamento, etc. Além dos requisitos citados anteriormente, é razoável minimizar o custo total do sistema. Deste modo, o processo de otimização se torna complexo, cuja função objetivo deve considerar um grande número de variáveis e restrições de projeto. A influência das variáveis de otimização geralmente é imprevisível e de comportamento não linear, e a função objetivo pode apresentar vários ótimos locais. Nestes casos, a utilização de algoritmos avançados de otimização torna-se atrativa, cujo os quais encontram soluções suficientemente próximas do ótimo global, com custo computacional e tempo razoável (Patel *et al.*, 2019).

O presente trabalho tem como objetivo estudar o desempenho e a performance de um algoritmo genético de codificação real (RCGA – do inglês *Real-Coded Genetic Algorithm*) em relação às seleções de substituição $(\mu + \lambda)$ e (μ, λ) (lê-se mu plus lambda e mu comma lambda, respectivamente), aplicado à otimização do ciclo Rankine orgânico de duplo estágio (DS ORC – do inglês *Double Stage Organic Rankine Cycle*) estudado por Braimakis e Karellas (2018).

Em seu trabalho, Braimakis e Karellas (2018) otimizaram um DS-ORC conectado em série via força bruta, para 28 diferentes pares de fluidos de trabalho. Prado e Galante (2020) propuseram um RCGA para a otimização do mesmo DS-ORC e os mesmos pares de fluidos, e demonstraram que essa abordagem pode gerar soluções satisfatórias, cuja as quais podem ser melhoradas executando-se o algoritmo diversas vezes. No presente trabalho, para comparar a influência das seleções $(\mu + \lambda)$ e (μ, λ) , apenas um par de fluido será utilizado.

1.2 Algoritmo Genético de Codificação Real

O algoritmo genético (GA – do inglês *Genetic Algorithm*) é descrito por Kramer (2017) como método de otimização bioinspirado, que imita a seleção natural ao traduzir o conceito biológico da evolução em algoritmos. De forma parecida, Yang (2010) descreve um GA como modelo ou abstração da evolução biológica, baseada na teoria da seleção natural de Charles Darwin.

Um GA procura por soluções ótimas ao realizar um processo de evolução artificial. Esse processo começa com uma primeira geração formada por soluções candidatas geradas de forma aleatória. Então, um processo evolucionário começa recombinao duas ou mais soluções por meio de um operador de cruzamento (*crossover*). Essas soluções recombinadas podem ser modificadas por um operador de mutação. As melhores soluções geradas após cruzamento e mutação são selecionados (ou tendem a serem selecionadas) para a próxima geração (Kramer, 2017).

Os GAs foram apresentados como método de otimização pela primeira vez por John Holland em 1975, nos Estados Unidos. O primeiro algoritmo genético tem representação binária, baixa probabilidade de mutação, seleção proporcional à aptidão e ênfase na recombinação para gerar novas soluções candidatas. Este algoritmo genético é comumente referido como "GA canonico" ou GA simples (SGA – do inglês *Simple Genetic Algorithm*) A. E. Eiben (2016).

GAs de representação binária se mostraram ineficientes em aplicações de alta precisão, multidimensionais ou em problemas de domínio contínuo, se comparados ao RCGAs. Nos RCGAs as variáveis, que são representadas por números no formato de ponto flutuante (floating-point), são genes que formam os cromossomos dos indivíduos, cujo os quais são alterados por diferentes operadores genéticos (Chambers, 2001).

Ao comparar os GAs de representação binária e os RCGAs, Michalewicz (1994) concluiu que a representação por ponto flutuante é mais rápida, mais consistente e promove maior precisão. Essa representação por ponto flutuante também é descrita como fácil de implementar já que é mais próxima ao domínio do problema.

Segundo Borne *et al.* (2014), a estratégia/seleção de substituição ou sobrevivência, que escolhe quais indivíduos da geração atual passarão para a próxima geração, têm relação com as propriedades de *exploitation* (capacidade de convergência rápida) e *exploration* (capacidade de exploração do domínio do problema) do GA. Portanto, as seleções $(\mu + \lambda)$ e (μ, λ) serão abordadas e analisadas em relação à convergência e exploração do domínio do problema.

Na seleção $(\mu + \lambda)$, um número μ de indivíduos sobreviventes é escolhido entre o conjunto dos μ indivíduos da população atual e o conjunto de λ indivíduos gerados pelos operadores genéticos (cruzamento e mutação). Na seleção (μ, λ) , toda a população atual é descartada e μ sobreviventes são escolhidos a partir do conjunto dos λ indivíduos gerados (Kramer, 2017). O procedimento de otimização com o RCGA será melhor descrito e detalhado na METODOLOGIA.

2. METODOLOGIA

Braimakis e Karellas (2018) otimizaram um DS ORC via força bruta. Nesse trabalho, a otimização é feita pela maximização da eficiência exergética, variando-se um conjunto de parâmetros de um limite inferior a um limite superior. Esse procedimento de otimização é repetido para diferentes valores de temperatura da fonte quente, indo de 100 a 300°C, em incrementos de 20 K. Além disso, essa otimização é executada para 28 diferentes combinações de fluido de trabalho.

Como o intuito do presente trabalho é comparar duas estratégias de substituição em um RCGA, a mesma planta e a mesma modelagem de Braimakis e Karellas (2018) serão consideradas, mas apenas um par de fluido de trabalho (Ciclopentano e R1234yf) será considerado, com a temperatura da fonte quente fixada em 240°C. A modelagem, algumas considerações e o processo de otimização serão melhor detalhados nas próxima subseções.

2.1 Modelagem

Braimakis e Karellas (2018) modelaram o DS ORC como sendo dois ciclos Rankine orgânicos de único estágio conectados em série, operando em alta e baixa temperatura, respectivamente, como mostrado na Fig .1. Assim como na metodologia de Braimakis e Karellas (2017), a fonte quente será considerada como água quente pressurizada. Os ciclos de alta temperatura (HT – do inglês *High Temperature*) e de baixa temperatura (LT – do inglês *Low Temperature*) têm a mesma configuração. Como o fluido de trabalho do ciclo HT deve possuir temperatura crítica igual ou superior ao ciclo LT, o ciclopentano (temperatura crítica de 238,6°C) é assumido como fluido de trabalho do ciclo HT e o R1234yf (temperatura crítica de 94,70 °C) como fluido de trabalho do ciclo LT.

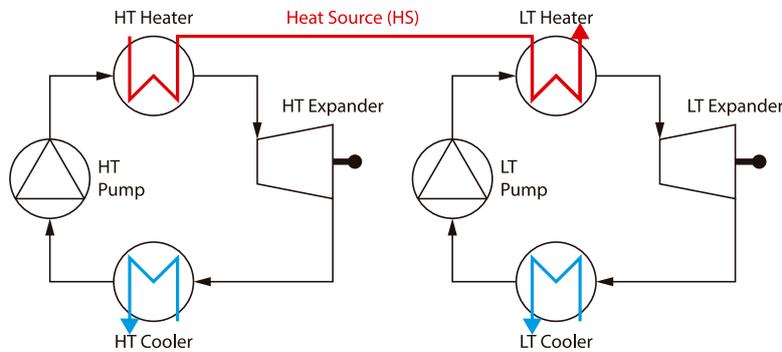


Figura 1: Representação do ciclo Rankine orgânico de duplo estágio, adaptado de Braimakis e Karellas (2018).

Assim como em Braimakis e Karellas (2018), a função objetivo utilizada para a otimização do DS ORC é a eficiência exergética (eficiência da segunda lei), calculada pela Eq. 5.

$$\eta_{ex} = \frac{P_{e,liq}}{\dot{E}_{fq}} \quad (5)$$

Na 5, $P_{e,liq}$ é a soma das potências líquidas geradas nos ciclos HT e LT, e \dot{E}_{fq} é a taxa exergética da corrente da fonte quente na entrada do sistema, dadas pelas Eq. 6 e 7, respectivamente. Assume-se uma capacidade máxima teórica de 500 kW para a fonte quente.

$$P_{e,liq} = P_{e,liq,HT} + P_{e,liq,LT} \quad (6)$$

$$\dot{E}_{fq} = \dot{m}_{fq} [(h_{fq,ent} - h_{fq,ref}) - T_0(s_{fq,ent} - s_{fq,ref})] \quad (7)$$

No presente trabalho, assim como em Braimakis e Karellas (2018), as propriedades termofísicas também são calculadas usando a biblioteca CoolProp, desenvolvida e apresentada por Bell *et al.* (2014).

2.2 Variáveis de Otimização e Considerações

Em Braimakis e Karellas (2018), as seguintes variáveis de otimização são consideradas: pressões de evaporação nos ciclos HT e LT, *pinch point* da caldeira de recuperação (também referida como evaporador) e temperatura de condensação no estágio HT. Essas variáveis de otimização e as principais considerações e suposições para o sistema são apresentadas na Tab. 1.

Tabela 1: Variáveis de otimização e principais considerações (Braimakis e Karellas, 2018).

<u>Eficiências</u>	
Eficiência eletromecânica ($\eta_m \eta_G$)	0,95
Eficiência do motor da bomba (η_M)	0,85
Eficiência isentrópica do expensor ($\eta_{exp, is}$)	0,75
Eficiência isentrópica da bomba ($\eta_{bomba, is}$)	0,70
<u>Trocadores de calor</u>	
Pinch point no evaporador HT (PP_{HT})	a ser otimizada (5 a 40 K)
Pinch point no evaporador LT	5 K
Incremento na temperatura da água de resfriamento	10 K
<u>Limites globais de pressão e temperatura</u>	
Pressão no evaporador HT (p_{HT})	a ser otimizada
Pressão no evaporador LT (p_{LT})	a ser otimizada
Temperatura de condensação no ciclo HT ($T_{cond, HT}$)	a ser otimizada (40 a 140 °C)
Mínimas temperaturas de condensação HT e LT (°C)	$\max(40, T_{sat}(p = 5 \text{ kPa}))$
Pressão de evaporação mínima (HT e LT)	$1,5 p_{cond}$
Pressão de evaporação máxima (HT e LT)	$\min(40E + 2 \text{ kPa}, 1,4 p_{crit})$

2.3 Procedimento de Otimização com o RCGA

O procedimento de otimização com o RCGA se inicia pela criação de indivíduos com cromossomos contendo as variáveis de otimização como seus genes. O cromossomo de um indivíduo é representado por um vetor de números de ponto flutuante. Na Eq. 8, X é um indivíduo e $x_1, x_2, x_3, \dots, x_n$ são genes representados por números de ponto flutuante que formam o cromossomo.

$$X = \langle x_1, x_2, x_3, \dots, x_n \rangle \quad (8)$$

Como visto anteriormente, Braimakis e Karellas (2018) consideram PP_{HT} , p_{HT} , p_{LT} e $T_{cond, HT}$ como variáveis de otimização. Deste modo, para este problema de otimização, cada indivíduo tem 4 genes, e pode ser representado pela Eq. 9.

$$indivíduo = \langle PP_{HT}, p_{HT}, p_{LT}, T_{cond, HT} \rangle \quad (9)$$

Para cada indivíduo da população inicial, os genes ou variáveis de otimização são selecionados aleatoriamente dentro dos limites mostrados na Tab. 2.

Tabela 2: Limites das variáveis de otimização.

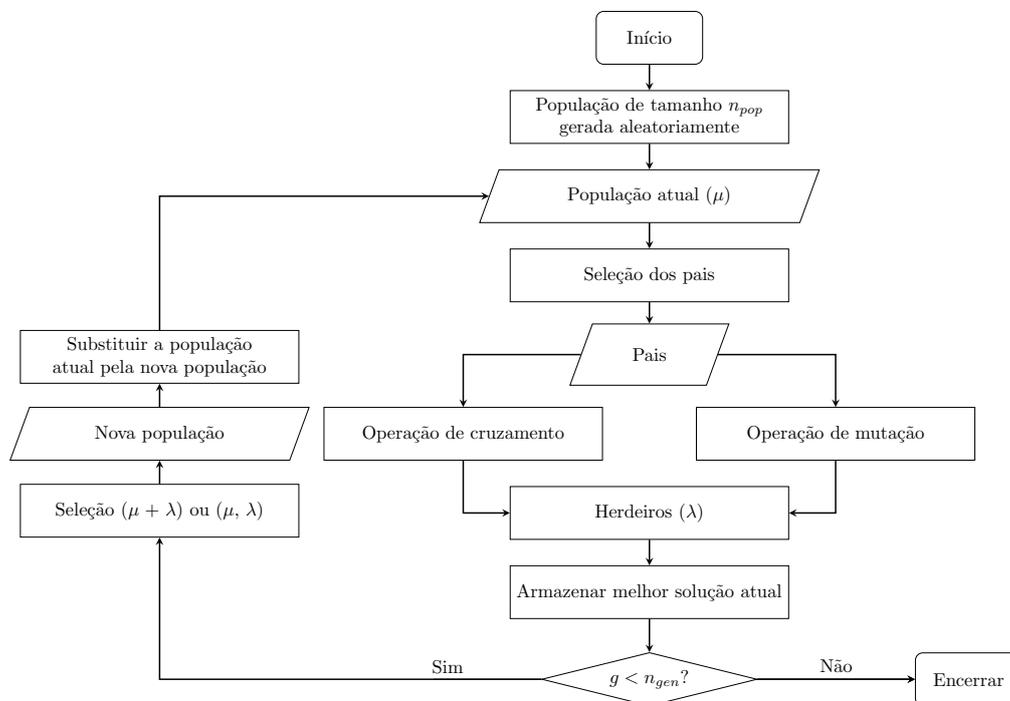
5	\leq	PP_{HT}	\leq	40 K
$1,5 p_{cond}$	\leq	p_{HT}	\leq	$\min(40E + 2 \text{ kPa}, 1,4 p_{crit})$
$1,5 p_{cond}$	\leq	p_{LT}	\leq	$\min(40E + 2 \text{ kPa}, 1,4 p_{crit})$
40	\leq	$T_{cond, HT}$	\leq	140°C

Inicialmente, o tamanho da população (n_{pop}), geralmente fixo, é definido. Então, n_{pop} indivíduos são criados aleatoriamente como mencionado anteriormente, dentro dos limites apresentados na Tab. 2. Um fluxograma que representa o procedimento de otimização com o RCGA a partir da criação da população inicial é mostrado na Fig. 2.

Após a população inicial ser criada, cada indivíduo é avaliado baseado em sua aptidão. No problema do presente trabalho, a aptidão é calculada utilizando os genes dos indivíduos como parâmetros na modelagem do DS ORC, e calculando a eficiência exergetica pela Eq. 5. Como o objetivo dessa otimização é a maximização da eficiência exergetica, a aptidão dos indivíduos pode ser considerada como sendo o próprio valor calculado pela Eq. 5 e, deste modo, os indivíduos mais aptos são aqueles com maiores valores de eficiência exergetica.

Com a avaliação de todos os indivíduos da população, é possível selecionar alguns deles como pais, baseados em suas aptidões. Os pais são indivíduos selecionados (par a par) para gerarem novas soluções através de cruzamento ou mutação. As novas soluções candidatas são chamadas de filhos ou herdeiros. Existem diversas formas de seleção parental, e a utilizada no presente trabalho é a seleção pelo método da roleta (*roulette wheel selection*), também conhecida como método de seleção proporcional à aptidão (FPS – do inglês *Fitness Proportionate Selection*). Segundo Brabazon *et al.* (2015), o método FPS escolhe os pais de forma aleatória, mas sendo tendenciosa àqueles com maiores aptidões.

Figura 2: Fluxograma do processo evolutivo do RCGA.



Os operadores de cruzamento e mutação são aplicados paralelamente até que um conjunto de filhos de tamanho λ (número inteiro escolhido pelo usuário) seja criado. Deste modo, deve-se definir as probabilidades de cruzamento e mutação ρ_c e ρ_m , respectivamente. Os valores de ρ_c e ρ_m definem com que frequência os operadores de cruzamento e mutação ocorrerão, e ambos devem estar dentro do intervalo $[0, 1]$, obedecendo à condição $(\rho_c + \rho_m) \leq 1, 0$. A criação do conjunto de filhos ocorre de forma iterativa até que o conjunto de herdeiros tenha λ indivíduos. Em cada iteração um número r no intervalo $[0, 1]$ é escolhido aleatoriamente; se $r \leq \rho_c$, então dois pais são escolhidos e aplica-se o operador de cruzamento, gerando-se dois filhos, mas apenas o primeiro filho gerado é adicionado à população de herdeiros; se $\rho_c < r \leq (\rho_c + \rho_m)$, um pai é escolhido, e aplica-se o operador de mutação, em que cada gene do cromossomo pode ser modificado individualmente, de acordo com um número ind_{pb} definido pelo usuário, e o indivíduo resultante dessa operação é adicionado ao conjunto de herdeiros; se $(\rho_c + \rho_m) < 1$ e $r > (\rho_c + \rho_m)$, então um indivíduo é escolhido pela seleção parental e adicionado ao conjunto de herdeiros, sem sofrer modificações pelos operadores anteriores.

Existem diversos operadores de cruzamento para os RCGAs, alguns deles são revisados por Herrera *et al.* (1998) e Sorsa *et al.* (2008). O operador de cruzamento escolhido para o presente RCGA é o SBX (do inglês – *Simulated Binary Crossover*), apresentado por Deb e Agrawal (1995). No SBX, a probabilidade de cruzamento (ρ_c) e um índice de distribuição (η_c), geralmente fixos, são definidos. η_c define o quão próximos (em relações aos valores das variáveis de otimização) os filhos serão de seus respectivos pais.

O operador de mutação utilizado neste RCGA é o Polynomial Mutation Operator, que é estudado por Deb e Ayan Deb (2014). Nesse operador, o usuário deve definir uma probabilidade de mutação (ρ_m) e um índice de distribuição

(η_m). η_m define o quão próximos os filhos serão de seus respectivos pais. Os operadores de cruzamento e mutação descritos anteriormente são limitados de forma a garantir que os valores das variáveis de otimização não excedam os valores mostrados na Tab. 2.

Após o conjunto de herdeiros é completado, a nova população é selecionada pela estratégia de substituição. No presente trabalho serão consideradas duas diferentes abordagens para o RCGA, as seleções $(\mu + \lambda)$ e (μ, λ) . Na seleção $(\mu + \lambda)$, a nova geração é selecionada entre o conjunto de herdeiros (λ) e o conjunto da população atual (μ), preservando-se as melhores soluções encontradas em cada geração. Na seleção (μ, λ) , o conjunto (μ) é descartado, e a nova população é escolhida somente entre os indivíduos do conjunto de herdeiros (λ), e desta forma, nem sempre as melhores soluções encontradas são preservadas. Vale destacar que no presente RCGA o tamanho da população se mantém constante ao longo das gerações, e desta forma o número μ , que define quantos indivíduos serão selecionados para a próxima geração, deve ser considerado igual a n_{pop} .

A substituição da população atual pela nova população determina uma nova geração. Todos os processo de seleção de pais e criação de filhos por cruzamento e mutação, são repetidos até que um número máximo de gerações (definido pelo usuário) seja alcançado, e a solução ótima é considerada a melhor solução encontrada em todas as gerações.

3. RESULTADOS

O RCGA foi executado trinta vezes para cada estratégia de substituição. A cada execução do algoritmo, o valor da melhor solução encontrada em cada geração é armazenado. Considerou-se os seguintes parâmetros: $n_{pop} = \mu = 25$, $\lambda = 50$, $\rho_c = 0,7$, $\eta_c = 0,5$, $\rho_m = 0,3$, $ind_{pb} = 0,3$, $\eta_m = 2$, $n_{gen} = 50$. A Fig. 3 apresenta as melhores soluções encontradas pelas várias execuções do RCGA ao longo das gerações.

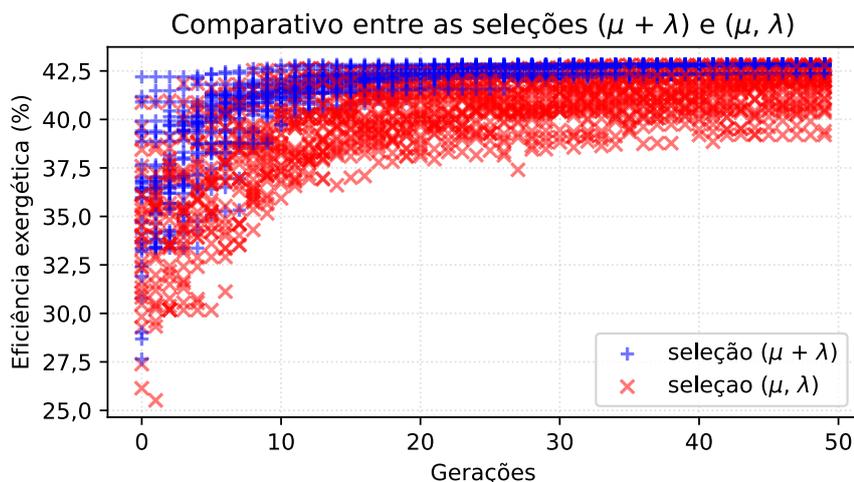


Figura 3: Melhores soluções encontradas a cada geração, em trinta execuções do RCGA, para cada seleção de substituição.

Como pode-se observar no gráfico da Fig. 3, os pontos referentes aos melhores valores encontrados pela seleção (μ, λ) apresentam uma maior dispersão se comparados aos valores encontrados pela seleção $(\mu + \lambda)$, em todo o processo de busca pela solução ótima. Essa dispersão pode ser melhor observada na Fig. 4, que apresenta o desvio padrão entre os melhores valores encontrados em cada geração, para cada seleção de substituição.

A diferença na dispersão dos valores encontrados pelos dois tipos de seleção de substituição, apresentadas nas Figuras 3 e 4, pode estar relacionada às propriedades de *exploration* e *exploitation*. A propriedade de *exploration*, segundo Borne *et al.* (2014) está associada à diversidade da população, o que permite uma maior exploração do domínio do problema e evita a estagnação do algoritmo em um mínimo ou máximo local. Utilizando o desvio padrão como medida de diversidade, é possível observar que a seleção (μ, λ) se sobressai em relação à $(\mu + \lambda)$. Kenny (2003) além de estudar o desvio padrão da aptidão como medida de diversidade, afirma que a queda do mesmo está relacionada à convergência da população. A habilidade de evitar ótimos locais não pôde ser observada na otimização do problema abordado no presente trabalho.

A propriedade de *exploitation*, isto é, de convergência rápida segundo Borne *et al.* (2014), pode ser observada pela seleção $(\mu + \lambda)$ em relação à (μ, λ) em análise da Fig. 3. Como Kenny (2003) afirma, esta convergência mais rápida também pode ser observada pelo gráfico da Fig. 4, em que o desvio padrão das aptidões da seleção $(\mu + \lambda)$ se mostra menor em praticamente toda a busca, com exceção das primeiras gerações. Além da convergência mais rápida, a seleção $(\mu + \lambda)$ encontrou melhores soluções em todas as execuções, resultado que é apresentado na Fig. 5, em que o valor das melhores soluções encontrada pelas duas formas de seleção de substituição são apresentados de forma crescente. Outra propriedade apresentada pela seleção $(\mu + \lambda)$ diz respeito à encontrar soluções ótimas com valores mais próximos entre

si: o desvio padrão das soluções ótimas foi calculado, sendo igual a 0,19% para a seleção $(\mu + \lambda)$ e 0,48% para a seleção (μ, λ) .

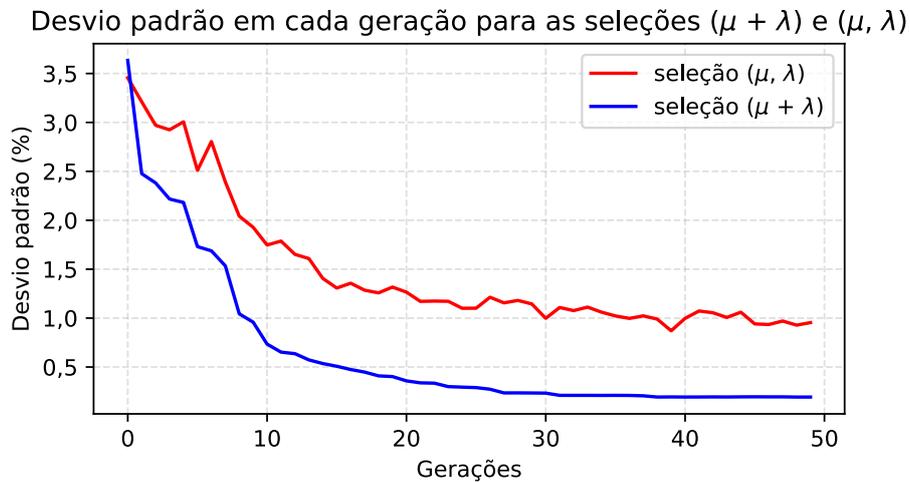


Figura 4: Desvio padrão das aptidões ao longo das gerações, em 30 execuções do RCGA, para cada seleção de substituição.

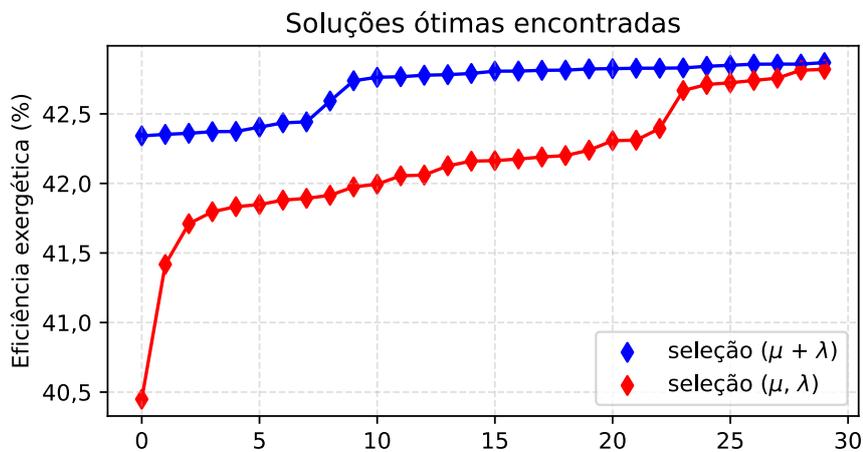


Figura 5: Melhores soluções encontradas por cada seleção de substituição, de forma crescente.

A melhor solução encontrada em todas as execuções foi a eficiência exergética de 42,87%, com os valores para as variáveis de otimização apresentados na Tab. 3. Para estes valores, um diagrama Q-T do DS ORC é mostrado na Fig. 6. Este diagrama apresenta a quantidade de calor da fonte quente que é absorvida pelos estágios HT e LT; a linha vermelha representa a temperatura da fonte quente, que diminui conforme fornece calor para o sistema; a linha azul representa a temperatura dos fluidos de trabalho dos estágios HT e LT. Neste diagrama também é possível observar como é feita a análise de *pinch point*: a partir de uma discretização da caldeira de recuperação, em que o processo de fornecimento de calor para o sistema na caldeira de recuperação é dividido em diversas partes, e o *pinch point* é basicamente a menor diferença de temperatura entre os fluidos quente e frio encontrada. Graficamente, o *pinch point* é a menor distância entre as linhas vermelha e azul, em cada estágio do DS ORC.

Tabela 3: Valores ótimos para as variáveis de otimização.

PP_{HT}	5 K
p_{HT}	1473,4 kPa
p_{LT}	1874,4 kPa
$T_{cond, HT}$	40 K

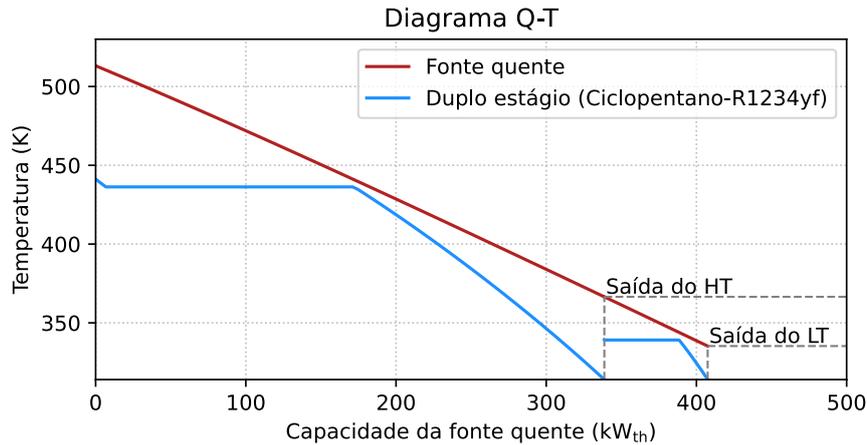


Figura 6: Melhores soluções encontradas por cada seleção de substituição, de forma crescente.

4. CONCLUSÃO

A utilização de um RCGA para a otimização do DS ORC abordado no presente trabalho já havia sido estudada por Prado e Galante (2020), que obtiveram resultados similares à otimização por força bruta realizada por Braimakis e Karellas (2018). Prado e Galante (2020), assim como Braimakis e Karellas (2018), também observaram que a otimização do DS ORC pode recomendar a opção por um SS ORC (ciclo rankine de único estágio – do inglês *Single Stage Organic Rankine Cycle*), a depender principalmente da temperatura da fonte quente.

O presente trabalho apresentou a utilização de duas formas de seleção de substituição no RCGA, na otimização do mesmo DS ORC apresentado por Braimakis e Karellas (2018). Em um processo iterativo do RCGA, a seleção $(\mu + \lambda)$ escolhe para a próxima geração uma quantidade μ de indivíduos com melhor aptidão entre os conjuntos da população atual e dos herdeiros gerados nesta iteração. A seleção (μ, λ) por sua vez, descarta o conjunto da população atual e escolhe uma quantidade μ de indivíduos, mas somente entre o conjunto dos herdeiros gerados nesta iteração, também com base na aptidão.

Segundo Borne *et al.* (2014), os GAs devem apresentar um bom *trade-off* entre *exploration* e *exploitation*, duas propriedades que se referem à diversidade da população e a convergência rápida, respectivamente. A diversidade da população está ligada a uma maior exploração do domínio do problema e em evitar máximos ou mínimos locais. Ambas as formas de seleção, $(\mu + \lambda)$ e (μ, λ) foram analisadas conforme as propriedades de *exploration* e *exploitation*. Para a análise, o RCGA foi executado trinta vezes para cada tipo de seleção, totalizando em sessenta execuções do algoritmo.

As várias execuções do algoritmo possibilitaram analisar uma maior dispersão dos valores das soluções ótimas encontrados pela otimização com a seleção (μ, λ) , o que pode ser observado pelo desvio padrão em cada geração, menor em praticamente todas as gerações se comparada à seleção $(\mu + \lambda)$, como mostrado na Fig. 4. O desvio padrão, segundo Kenny (2003), é uma das formas de medir a diversidade da população. A maior dispersão na aptidão da população da seleção (μ, λ) pode estar ligada à maior diversidade da população e exploração do domínio do problema, no entanto, no problema de otimização proposto, não foi possível verificar a capacidade de evitar ótimos locais, sendo recomendada em trabalhos futuros o estudo da seleção (μ, λ) na otimização de funções de comportamento conhecido, com grande quantidade de ótimos locais.

Os resultados referentes à seleção $(\mu + \lambda)$ apontaram para uma menor dispersão dos valores das soluções encontradas, ou seja, menor valor do desvio padrão em praticamente todas as gerações, como mostrado na Fig. 4. Segundo Kenny (2003), a diminuição do desvio padrão das aptidões da população é um precursor da convergência do algoritmo. A propriedade de convergência mais rápida foi observada pela seleção $(\mu + \lambda)$, que ao mesmo tempo encontrou soluções ótimas melhores que a seleção (μ, λ) , em todas as execuções. Além disso, esta seleção encontrou soluções ótimas muito mais próximas entre si, apresentando um desvio padrão de 0,19% contra 0,48% da solução (μ, λ) .

Com os resultados apresentados fica claro que, para o problema abordado, a seleção $(\mu + \lambda)$ se mostrou mais vantajosa em relação à (μ, λ) . Essa vantagem, no entanto, não deve ser estendida à todos os problemas de otimização, pois cada problema tem suas particularidades e, como afirmado anteriormente, o problema da otimização do DS ORC aparentou não possuir problemas com ótimos locais. Para determinados problemas de otimização, a maior dispersão apresentada pela seleção (μ, λ) , relacionada à maior diversidade da população, pode auxiliar o algoritmo a evitar a convergência para ótimos locais e torna-se interessante o estudo de ambas as seleções em trabalhos futuros, na otimização de funções conhecidas pela grande quantidade de máximos e mínimos locais, como o caso da função *eggholder*, cuja otimização é apresentada no estudo de Czerniak e Zarzycki (2017) e em muitos outros trabalhos.

5. REFERÊNCIAS

- A. E. Eiben, J.E.S., 2016. *Introduction to Evolutionary Computing*. Springer, Berlin. ISBN 3662499851.
- Andreas Antoniou, W.S.L., 2007. *Practical Optimization*. Springer-Verlag New York Inc. ISBN 0387711066.
- Bell, I.H., Wronski, J., Quoilin, S. e Lemort, V., 2014. “Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library coolprop”. *Industrial & Engineering Chemistry Research*, Vol. 53, No. 6, pp. 2498–2508. doi:10.1021/ie4033999. URL <http://pubs.acs.org/doi/abs/10.1021/ie4033999>.
- Borne, P., Popescu, D. e Philip, F.G., 2014. *Optimization in Engineering Sciences*. John Wiley & Sons, Ltd. ISBN 1848214987.
- Brabazon, A., O’Neill, M. e McGarraghy, S., 2015. *Natural Computing Algorithms*. Springer Berlin Heidelberg. ISBN 3662436302.
- Braimakis, K. e Karellas, S., 2017. “Integrated thermoeconomic optimization of standard and regenerative ORC for different heat source types and capacities”. *Energy*, Vol. 121, pp. 570–598. doi:10.1016/j.energy.2017.01.042.
- Braimakis, K. e Karellas, S., 2018. “Exergetic optimization of double stage organic rankine cycle (ORC)”. *Energy*, Vol. 149, pp. 296–313. doi:10.1016/j.energy.2018.02.044.
- Chambers, L., 2001. *The practical handbook of genetic algorithms : applications*. Chapman & Hall/CRC, Boca Raton, Fla. ISBN 9781584882404.
- Czerniak, J.M. e Zarzycki, H., 2017. “Artificial acari optimization as a new strategy for global optimization of multimodal functions”. *Journal of Computational Science*, Vol. 22, pp. 209–227. doi:10.1016/j.jocs.2017.05.028.
- Deb, K. e Agrawal, R.B., 1995. “Simulated Binary Crossover for Continuous Search Space”. URL <https://www.semanticscholar.org/paper/Simulated-Binary-Crossover-for-Continuous-Search-Deb-Agrawal/b8ee6b68520ae0291075cb1408046a7dff9dd9ad>.
- Deb, K. e ayan Deb, 2014. “Analysing mutation schemes for real-parameter genetic algorithms”. *International Journal of Artificial Intelligence and Soft Computing*, Vol. 4, No. 1, p. 1. doi:10.1504/ijaisc.2014.059280.
- Herrera, F., Lozano, M. e Verdegay, J., 1998. *Artificial Intelligence Review*, Vol. 12, No. 4, pp. 265–319. doi:10.1023/a:1006504901164.
- Kenny, Q., 2003. “Population Diversity in Genetic Algorithm for Vehicle Routing Problem with Time Windows”.
- Kramer, O., 2017. *Genetic Algorithm Essentials*. Springer-Verlag GmbH. ISBN 3319521551.
- Lopes, H.S. e Takahashi, R.H.C., eds., 2011. *Computação Evolucionária em Problemas de Engenharia*. Omnipax, Curitiba (PR), 1st edition. ISBN 978-85-64619-00-5. 385 p.
- Michalewicz, Z., 1994. *Genetic algorithms + data structures = evolution programs*. Springer-Verlag, Berlin New York. ISBN 9783540580904.
- Patel, V.K., Savsani, V.J. e Tawhid, M.A., 2019. *Thermal System Optimization*. Springer-Verlag GmbH. ISBN 3030104761.
- Prado, G.P. e Galante, R.M., 2020. “Real-Coded Genetic Algorithm for Double Stage Organic Rankine Cycle Exergy Optimization”. *ABCM*. doi:doi://10.26678/ABCM.ENCIT2020.CIT20-0487.
- Sorsa, A., Peltokangas, R. e Leiviska, K., 2008. “Real-coded genetic algorithms and nonlinear parameter identification”. In *2008 4th International IEEE Conference Intelligent Systems*. IEEE. doi:10.1109/is.2008.4670495.
- Yang, 2010. *Engineering Optimization*. John Wiley & Sons. ISBN 0470582464.

6. RESPONSABILIDADE PELAS INFORMAÇÕES

Os autores são os únicos responsáveis pelas informações incluídas neste trabalho.