

Desenvolvimento de melhorias no Sistema Integrado de Modelagem de Poluição Pontual e Difusa (SIMPPOD)

Identificação:

Grande área do CNPq: Ciências Exatas e da Terra

Área do CNPq: Ciência da Computação

Título do Projeto: Planejamento integrado de controle e redução de cargas pontuais e difusas de poluição em bacias hidrográficas

Professor Orientador: Lucia Catabriga

Estudante PIBIC/PIVIC: Juliane Azeredo Ferreira

Resumo: *Os rios brasileiros são categorizados em classes de qualidade de água, de acordo com o Enquadramento de cursos d'água, em conformidade. Nos últimos anos, algoritmos evolutivos vem sendo utilizados para encontrar a eficiência mínima de redução de poluentes garantindo o enquadramento de rios na legislação vigente. Este trabalho tem por objetivo comparar três algoritmos genéticos na busca de soluções ótimas: algoritmo genético clássico, algoritmo genético das chaves aleatórias viciadas (BRKGA) e um algoritmo genético adaptado ao problema, denominado, AG+. Os algoritmos são comparados no estudo de qualidade de água na bacia hidrográfica do rio Pardo, localizada no estado do Espírito Santo.*

Palavras chaves: Otimização, Algoritmo Genético, BRKGA, Qualidade de Água.

1 Introdução

Os padrões brasileiros de qualidade de água foram fixados pela Resolução do Conselho Nacional de Meio Ambiente (Conama) nº 357 de 2005, a qual foi posteriormente complementada pela resolução Conama nº 430 de 2011, estabelecendo limites de concentração para vários parâmetros de qualidade de água. Todavia, a qualidade de água nos rios brasileiros frequentemente está em desacordo com a legislação vigente, violando o referido padrão de qualidade, e essa violação ocorre principalmente devido à aportes de cargas poluentes aos cursos d'água. Tanto os processos naturais como atividades humanas influenciam na qualidade de água de corpos d'água. Entretanto, a intensificação destas atividades sobre a bacia hidrográfica geralmente resulta na deterioração da qualidade da água dos seus recursos hídricos. Estas atividades intervêm com o equilíbrio do ecossistema resultando, por vezes, em perda de diversidade e alterações nos constituintes químicos e parâmetros físicos dos cursos d'água presentes na bacia.

A poluição em recursos hídricos, ou poluição hídrica, pode ser classificada quanto a origem das fontes das cargas poluidoras. A poluição hídrica é dita pontual quando o lançamento de carga poluente ocorre de modo concentrado e numa localidade específica, como por exemplo, o esgoto de uma cidade. Quando essa carga poluente não possui uma localidade de ocorrência específica, como os poluentes que se encontram presentes nas enxurradas, a poluição é dita difusa, Novotny (2003); Prodanoff (2005); Carvalho (2019). Para o controle e redução destas cargas de poluição existem inúmeros métodos e tecnologias, entretanto o ponto de partida para a otimização destes processos envolve identificar os valores mínimos de redução destas cargas, de modo que os limites vinculados ao padrão de qualidade de água não sejam violados. Na identificação destes percentuais de redução comumente são empregados métodos de busca, especialmente meta-heurísticas, com o objetivo de obter soluções próximas as soluções ótimas. Dada a complexidade presente em sistemas hídricos, principalmente pelas múltiplas variáveis e processos envolvidos a serem representados, o uso de meta-heurísticas como alternativa de otimização para estes sistemas vêm crescendo nos últimos anos, Maringanti (2011); Hernandez (2013).

Os algoritmos evolucionários são métodos de otimização baseados na teoria da evolução das espécies de Darwin, assumindo que apenas indivíduos mais adaptados conseguem sobreviver. Dentre os algoritmos evolucionários, os algoritmos genéticos destacam-se, tendo como característica principal a simulação da evolução das espécies como método de busca pelo indivíduo ou indivíduos ótimos. Sendo, portanto, um método populacional estocástico de busca direta com operadores probabilísticos, Goldberg (1989).

O presente trabalho teve como meta refinar o SIMPPOD, aperfeiçoando os resultados obtidos nos projetos de Iniciação Científica anteriores (2017-2018 e 2018-2019), Ribeiro (2018, 2019). O projeto contemplou a implementação de um conjunto de modelos para redução de cargas poluidoras em bacias hidrográficas comparando três algoritmos genéticos na busca de soluções ótimas: algoritmo genético clássico (AG), Linden (2012), algoritmo genético das chaves aleatórias viciadas ou *Biased Random-Key Genetic Algorithm* (BRKGA), Gonçalves & Resende (2011); Moura (2018), e um algoritmo genético adaptado ao problema, denominado, AG+. Ademais, foram realizados *upgrades* no código de modo a melhorar seu desempenho e inclusão de novas funcionalidades, tais como: elaboração de um conjunto de algoritmos evolutivos, permitindo ao usuário a escolha de diferentes técnicas de otimização, e aprimoramento da relação do usuário com o código, mediante desenvolvimento de uma interface mais amigável.

2 Objetivos

O presente projeto de iniciação científica teve por principais objetivos:

- Estudo da linguagem Python e dos modelos e métodos que compõem o SIMPPOD;
- Implementar novos algoritmos evolutivos;

- Otimizar o funcionamento e, conseqüentemente, o desenvolvimento do código;
- Desenvolver uma interface mais amigável.

Ao decorrer deste será explicitada a execução destes objetivos mencionados.

3 Metodologia

A seguir são discriminadas as metodologias utilizadas para a realização de cada componente e etapa deste projeto de iniciação científica.

3.1 Documentação de código

Manual de instruções de um programa: uma boa documentação auxilia a ambos, usuários e programadores, no entendimento do código e na detecção de erros nas funções e nos algoritmos desenvolvidos. Ou seja, este método é uma forma de melhorar a relação usuário-código.

3.2 Python

A linguagem de programação Python é uma linguagem de alto nível de multiparadigma, sendo orientada a objetos, funcional, procedural e imperativa. Possui uma codificação próxima à linguagem natural, priorizando, assim, sua legibilidade. Tratando-se de representação visual, Tkinter, uma biblioteca do Python, possibilita o desenvolvimento de interfaces gráficas. O estudo da linguagem foi norteado pelo livro Lutz & Ascher (2007).

3.3 AG

O AG Clássico define um *indivíduo*, chamado de *cromossomo*, como um conjunto de *genótipos*. Este, no modelo de otimização de fontes pontuais, será um vetor composto por n_p números randômicos gerados em uma distribuição uniforme no intervalo [eficiência mínima¹, 0.9]; enquanto que no modelo de otimização de fontes difusas, uma matriz $n_d \times 4$ de valores aleatórios de igual distribuição no intervalo [0, 0.9]. Cada um destes valores randômicos é um *alelo* do genótipo do indivíduo. Assim, o cromossomo é um conjunto de alelos.

A *população* é um grupo de indivíduos. Cromossomos são criados e avaliados com relação às restrições até que se complete o conjunto com o tamanho desejável (n_{sol}). Indivíduos que não atendem as restrições são considerados *inválidos*, sendo então descartados, e novos cromossomos vão sendo gerados para formar a população inicial. Isso ocorre quando o *cenário* resultante da aplicação daquele cromossomo (calculado através do Método de Euler) não atende aos limites de qualidade de água fixados pela resolução Conama 357/2005 - exemplo para a Classe 2,

¹Referente aos limites mínimos de eficiência que as ETEs devem possuir, os quais foram fixados pela CONAMA 430/2011.

representados na Eq. (1).

$$\begin{aligned} DBO5 &\leq 5\text{mg/L}; & OD &\geq 5\text{mg/L}; & NH3 &\leq 3,7\text{mg/L para } \leq pH7; \\ NO &\leq 1\text{mg/L}; & P &\leq 0,1\text{mg/L} \end{aligned} \quad (1)$$

Com a população inicial em mãos, calcula-se a Função Objetivo (FO) de cada um de seus componentes. Neste estudo, consideramos que quanto menor a eficiência da tecnologia de tratamento, menor seria o custo na implementação do tratamento. Assim, minimizar a eficiência representa a solução desejada. Portanto, as FOs dos algoritmos de poluição pontual e difusa são representadas pelas Eqs. (2) e (3).

$$\min_{j=1, n_{sol}} |F(j)| = \sum_{i=1}^{n_p} E^j(i) \quad (2)$$

$$\min_{j=1, n_{sol}} |F(j)| = \sum_{i=1}^{n_d} \left[\left(W_{DBO5} R_{DBO5}^j(i) \right) + \left(W_{NH3} R_{NH3}^j(i) \right) + \left(W_{NO} R_{NO}^j(i) \right) + \left(W_P R_P^j(i) \right) \right] \quad (3)$$

onde n_{sol} é o tamanho da população, $E^j(i)$ é a j -ésima eficiência de tratamento da i -ésima fonte poluidora pontual relativa a DBO5 no contexto da poluição pontual; W_* é o peso da função multi-objetivo e R_*^j é a j -ésima remoção das cargas poluentes na i -ésima sub-bacia.

Após calculadas as funções objetivo de cada indivíduo da população inicial, é realizada uma série de cruzamentos entre dois componentes escolhidos aleatoriamente do conjunto, o pai a e o pai b . Um cruzamento gera o denominado filho através de um crossover dos alelos de seus pais. Define-se um inteiro p_c como o ponto de corte do operador genético. Para realizar-se o cruzamento, um número randômico r é gerado em uma distribuição uniforme no intervalo $[0, 1]$. Se $r < 0.5$, o descendente recebe os p_c primeiros alelos do pai a e os demais, do pai b ; senão, ele os recebe do pai b e o restante, do pai a .

A mutação representa a geração de indivíduos gerados randomicamente. A nova geração será composta por estes mutantes e pelos filhos dos cruzamentos. Este processo é repetido um número fixo de iterações (n_{iter}). Os melhores indivíduos de cada geração, com base em seus alelos, são computados em um conjunto especial, e o cenário otimizado será referente ao melhor indivíduo deste conjunto de iterações.

O Algoritmo 1 ilustra um pseudocódigo para o AG Clássico, onde P é uma população aleatória; P_m é a população mutante; p_c é o ponto de corte do operador genético; n é o número de alelos do cromossomo.

3.4 BRKGA

Outra meta-heurística implementada neste trabalho foi o Biased Random-Key Genetic Algorithm, o BRKGA (Algoritmo Genético de Chaves Aleatórias Viciadas). As principais diferenças deste para o AG clássico são a presença das chaves aleatórias, do decodificador, e de certas modificações no modelo de cruzamento.

Algoritmo 1 AG: Dados P, P_m, p_c, n

```
1: Gere uma população com  $P$  cromossomos definidos por alelos
2: enquanto Critério de parada não for satisfeito faça
3:   Avalie a Função Objetivo de cada solução na população
4:   Inicialize a população da próxima geração
5:   Gere um conjunto com  $P_m$  mutantes aleatórios
6:   Adicione o conjunto de mutantes a população da nova geração
7:   para  $i \leftarrow 1$  até  $(P - P_m)$  faça
8:     Escolhe o pai  $a$  aleatoriamente dentre a população
9:     Escolhe o pai  $b$  aleatoriamente dentre a população
10:    para  $j \leftarrow 1$  até  $n$  faça
11:      Escolhe o alelo do filho, com base em  $p_c$ 
12:    fim para
13:    Adicione o filho a população da nova geração
14:  fim para
15: fim enquanto
16: devolve Melhor solução encontrada
```

No BRKGA, o cromossomo é definido inicialmente como um conjunto de *chaves aleatórias*, números randômicos gerados em uma distribuição uniforme no intervalo $[0, 1]$. Análogo ao Algoritmo Genético Clássico, este indivíduo será um vetor composto por n_p chaves aleatórias no algoritmo de otimização de fontes pontuais, e uma matriz $n_d \times 4$ de chaves aleatórias para as fontes difusas. Ao ser criado, o cromossomo é então *decodificado* para dentro do espaço de busca em questão: [eficiência mínima, 0.9] e $[0, 0.9]$ (referente às poluições pontual e difusa, respectivamente). Para isso, o decodificador faz uso de regra de três simples, encontrando o valor de alelo correspondente à chave. Portanto, o indivíduo é definido por dois conjuntos, um de alelos e um de chaves aleatórias.

Cromossomos são gerados e testados quanto ao atendimento das restrições (descritas na Eq. 1), sendo chamados de *inválidos* aqueles que não estão de acordo com as restrições. Estas soluções inválidas são descartadas e novas soluções são geradas e novamente testadas. Este processo se repete até que se forme o conjunto da população inicial com o tamanho desejado (n_{sol}). Com a população em mãos, esta será particionada através do uso da técnica denominada *elitismo*, que separa os melhores indivíduos dos demais. Assim, as soluções que apresentam menores valores de eficiência de remoção (melhor desempenho) constituem um conjunto denominado *elite*.

Os cruzamentos ocorrem entre um membro elite e um não elite. Define-se p_e como a probabilidade da cria herdar o componente genético do pai elite, com $p_e > 0.5$. Para realizar-se o cruzamento, um número randômico é gerado em uma distribuição uniforme no intervalo $[0, 1]$. Se este for menor que o parâmetro p_e fornecido, o descendente recebe o valor da chave aleatória do pai elite; senão, ele o recebe do pai não elite. Este processo é repetido até que se estabeleça

um novo ser completo, de tamanho n_p (no caso da poluição pontual) ou $n_d \times 4$ (se tratar-se da poluição difusa). Em seguida, este novo cromossomo é traduzido pelo decodificador, tendo então os conjunto de alelos e de chaves aleatórias.

O grupo de mutantes é gerado, e a próxima geração será composta pelos mesmos, pela elite e pelos filhos dos cruzamentos. Por fim, este processo é repetido um número fixo de iterações (n_{iter}). Novamente, os melhores indivíduos de cada geração, com base em seus alelos, são computados em um conjunto especial, e o cenário otimizado será referente ao melhor indivíduo deste conjunto de iterações.

O Algoritmo 2 ilustra um pseudocódigo para o BRKGA, onde P é uma população aleatória; P_e é a população pertencente ao grupo elite; P_m é a população mutante; p_e é a probabilidade de herdar-se do pai elite; n é o número de alelos do cromossomo.

Algoritmo 2 BRKGA: Dados P, P_e, P_m, p_e, n

```
1: Gere uma população com  $P$  cromossomos definidos por chaves aleatórias
2: Decodifique os indivíduos em alelos
3: enquanto Critério de parada não for satisfeito faça
4:   Avalie a Função Objetivo de cada solução na população
5:   Particione a população nos conjuntos Elite e não-Elite
6:   Inicialize a população da próxima geração
7:   Gere um conjunto com  $P_m$  mutantes aleatórios
8:   Adicione o conjunto de mutantes a população da nova geração
9:   Copie o conjunto Elite para a próxima geração
10:  para  $i \leftarrow 1$  até  $(P - P_m - P_e)$  faça
11:    Escolhe o pai  $a$  dentro do conjunto elite
12:    Escolhe o pai  $b$  dentro do conjunto não elite
13:    para  $j \leftarrow 1$  até  $n$  faça
14:      Escolhe a chave aleatória do filho, com probabilidade  $p_e > 0.5$  de ser o do pai Elite
15:    fim para
16:    Decodifique o filho em alelos
17:    Adicione o filho a população da nova geração
18:  fim para
19: fim enquanto
20: devolve Melhor solução encontrada
```

3.5 AG+

A alternativa AG+ desenvolvida consiste em uma combinação dos dois algoritmos apresentados, sendo em suma um AG Clássico com o modelo de cruzamento do BRKGA. Todavia tal procedimento é realizado com os alelos dos cromossomos, visto que não há a definição das chaves aleatórias no algoritmo clássico.

O AG+, assim como o AG Clássico, define o *cromossomo* como um conjunto de *alelos*. O

indivíduo é então representado por um vetor composto por n_p números randômicos gerados em uma distribuição uniforme no intervalo [eficiência mínima, 0.9], e por uma matriz $n_d \times 4$ de valores aleatórios de igual distribuição no intervalo [0, 0.9], nos modelos de otimização de fontes pontuais e difusas respectivamente. As soluções são geradas e testadas quanto ao atendimento das restrições (descritas na Eq. 1). Serão *inválidas* aquelas que não satisfazem as restrições. Estas são descartadas e novos cromossomos são gerados e testados até que se forme o conjunto da população inicial com o tamanho desejado (n_{sol}). Com o então conjunto em mãos, este será particionado em dois grupos através do uso da técnica denominada *elitismo*, separando as melhores soluções das demais. Portanto, os indivíduos com os menores valores de eficiência de remoção (que apresentam o melhor desempenho) constituem o conjunto *elite*.

Análogo ao BRKGA, os cruzamentos ocorrem entre um membro elite e um não elite. Tome p_e como a probabilidade do filho herdar o alelo do pai elite, com $p_e > 0.5$, e r um número gerado aleatoriamente em uma distribuição uniforme no intervalo [0, 1]. Se $r > p_e$, a cria recebe o valor do alelo do pai elite; senão, ela o recebe do pai não elite. Este processo é repetido até que se estabeleça um novo ser completo, de tamanho n_p (no caso da poluição pontual) ou $n_d \times 4$ (se tratar-se da poluição difusa).

Gera-se o grupo de mutantes, e a próxima geração será composta por este, pela elite e pelos filhos dos cruzamentos. Por fim, este processo é repetido um número fixo de iterações (n_{iter}). Finalmente, os melhores cromossomos de cada geração, com base em seus alelos, são computados em um conjunto especial, e o cenário otimizado será referente ao melhor indivíduo deste conjunto de iterações.

O Algoritmo 3 ilustra um pseudocódigo para o AG+, onde P é uma população aleatória; P_e é a população pertencente ao grupo elite; P_m é a população mutante; p_e é a probabilidade de herdar-se do pai elite; n é o número de alelos do cromossomo.

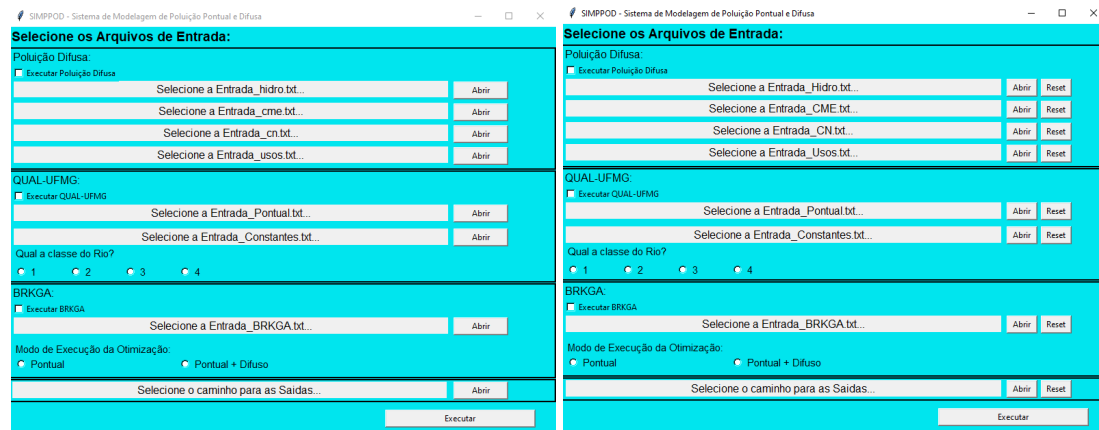
Algoritmo 3 AG+: Dados P, P_e, P_m, p_e, n

```
1: Gere uma população com  $P$  cromossomos definidos por alelos
2: enquanto Critério de parada não for satisfeito faça
3:   Avalie a Função Objetivo de cada solução na população
4:   Particione a população nos conjuntos Elite e não-Elite
5:   Inicialize a população da próxima geração
6:   Gere um conjunto com  $P_m$  mutantes aleatórios
7:   Adicione o conjunto de mutantes a população da nova geração
8:   Copie o conjunto Elite para a próxima geração
9:   para  $i \leftarrow 1$  até  $(P - P_m - P_e)$  faça
10:     Escolhe o pai  $a$  dentro do conjunto elite
11:     Escolhe o pai  $b$  dentro do conjunto não elite
12:     para  $j \leftarrow 1$  até  $n$  faça
13:       Escolhe o alelo do filho, com probabilidade  $p_e > 0.5$  de ser o do pai Elite
14:     fim para
15:     Adicione o filho a população da nova geração
16:   fim para
17: fim enquanto
18: devolve Melhor solução encontrada
```

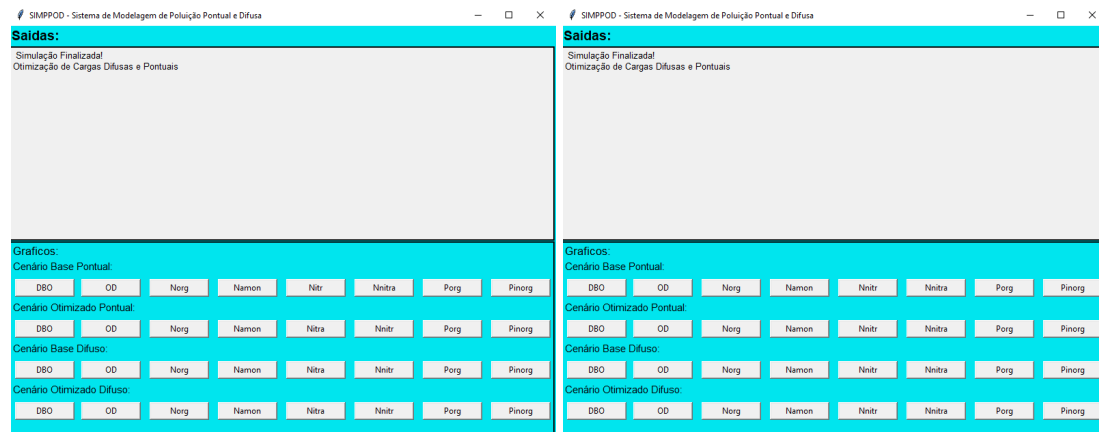
4 Resultados

Realizou-se revisão de literatura dos modelos ambientais desenvolvidos no SIMPPOD, tendo sido estudados conteúdos dos temas relativos a poluição hídrica (cargas pontuais e difusas) e equações de perfil dos componentes biológicos (estes descritos em VON SPERLING (1996)). Ademais, fez-se o estudo e revisão da linguagem Python e do SIMPPOD, juntamente com os algoritmos evolutivos envolvidos (tomando como base o exposto em Linden (2012)). Foram executadas algumas melhorias e otimizações no desempenho e em funcionalidades do SIMPPOD, como limpeza, reparo de erros e documentação de suas funções. Definições de variáveis, parâmetros de entrada/saída, métodos, módulos e importação de bibliotecas que não eram mais utilizados foram removidos, assim como a inclusão de novos destes para melhor funcionamento e entendimento do programa e de sua interface. Consequentemente, aperfeiçoou-se a relação usuário-código a partir da criação e manipulação de métodos ligados à representação visual, com melhorias na interface previamente desenhada, tais como: criação de um botão de reset para cada parâmetro de entrada na página de simulação, correção na ordem e chamada dos botões na tela de resultados, colocação de título nos gráficos gerados de acordo com cada componente biológico e labels nos eixos dos gráficos. A Figura 1 mostra algumas telas do SIMPPOD, enfatizando as mudanças efetuadas na interface.

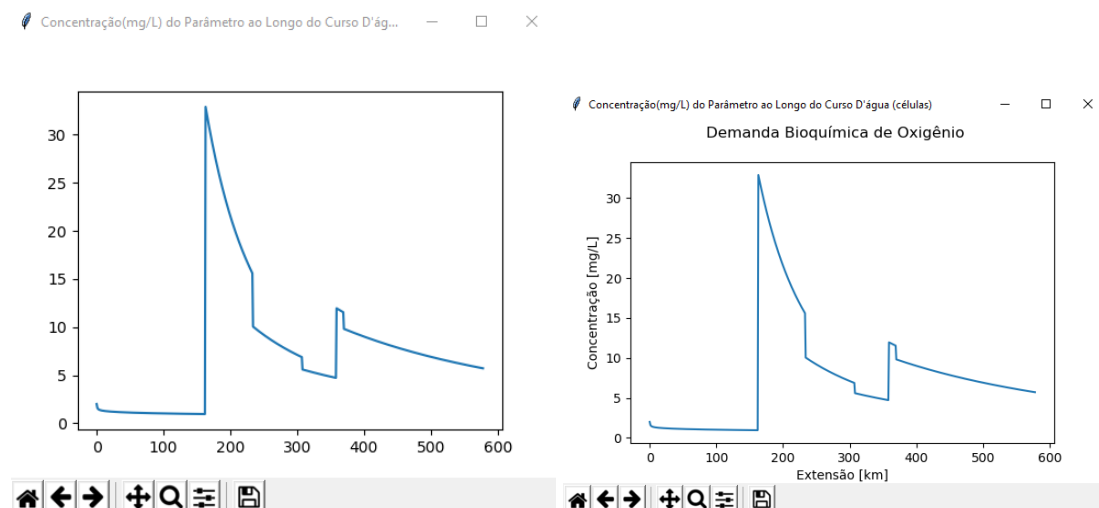
Também foram desenvolvidos novos modos de execução, totalizando três algoritmos genéticos, descritos anteriormente: o AG Clássico, o BRKGA e o AG+. Acerca destes foi realizado um



(a) Nova tela de simulação, com botões de reset para cada parâmetro de entrada do SIMPPOD.



(b) Nova tela de resultados, com correção na ordem e na chamada dos botões de geração dos gráficos.



(c) Gráficos possuem título indicando o componente sendo tratado; acréscimo dos labels nos eixos.

Figura 1: Telas do programa SIMPPOD.

estudo com base nos resultados obtidos. Os experimentos foram realizados a partir dos dados da Bacia do rio Pardo (em sua modelagem básica), utilizando valores reais de uso e ocupação

referentes ao ano de 2015, disponibilizados pelo GEOBASES².

Em todos os nossos experimentos nos algoritmos evolutivos AG Clássico, AG+ e BRKGA é considerado um número fixo de iterações ($n_{iter} = 100$), sendo apenas uma população por iteração de tamanho fixo, sendo $n_{sol} = 50$ indivíduos. Com o objetivo de minimizar os efeitos aleatórios naturais de algoritmos evolutivos realizamos um total de 30 repetições dos testes para todos os algoritmos considerados. Assim, todos os resultados apresentados a seguir mostram as médias obtidas em cada execução dos algoritmos.

Com o objetivo de observar o comportamento dos algoritmos implementados, a Tab. 1 apresenta a média e o desvio padrão do número de inválidos da população inicial, dos cruzamentos e dos mutantes para os algoritmos AG Clássico, AG+ e BRKGA nas duas versões, AGOFP (fontes pontuais) e AGOFD (fontes difusas). Claramente o número de inválidos para as versões relativas a poluição difusa são mais elevados. Essa característica é esperada uma vez que a poluição difusa contém um número maior de poluentes, um número maior de fontes de poluição (número de sub-bacias) e também pela magnitude das entradas correspondentes a esta fonte. Os números de cruzamentos para as versões pontuais são sempre nulos, provavelmente devido a simplicidade da aplicação que contém um número muito pequeno de fontes pontuais. O desvio padrão do número de inválidos para a população inicial é grande, o que também era esperado, dado a aleatoriedade natural das soluções iniciais geradas, como descrito na Seção 3. Em contra partida, observa-se que o desvio padrão dos demais parâmetros são bem menores, demonstrando a regularidade dos algoritmos evolutivos na geração de novas soluções.

A média e o desvio padrão das FOs, dadas pelas Eqs. (2) e (3), respectivamente, para a poluição pontual e difusa são apresentadas na Tab. 2. Podemos notar que a média das eficiências mínimas (FO) pouco varia para os 3 algoritmos evolutivos considerados para a poluição pontual. Entretanto, para a poluição difusa as médias das eficiências mínimas para os algoritmos AG+ e BRKGA foram praticamente a metade daquela obtida para o algoritmo AG Clássico. O desvio padrão das eficiências mínimas para o algoritmo BRKGA foi o menor para a fonte de poluição pontual, enquanto que para a fonte de poluição difusa o menor desvio padrão foi o do algoritmo AG Clássico.

A Tab. 3 apresenta as eficiências da Demanda Bioquímica de Oxigênio (DBO5) e do Oxigênio Dissolvido (OD) para a poluição pontual. Mais uma vez podemos observar que os 3 algoritmos geram eficiências praticamente iguais com desvios padrão muito pequenos para os dois parâmetros de qualidade de água avaliados. O mesmo não ocorre para a poluição difusa. Na Tab. 4 apresentamos o somatório das eficiências referente a cada poluente que é otimizado e de todo o conjunto (FO) para a poluição difusa. Em geral, podemos dizer que os resultados do AG clássico são inferiores aos dos algoritmos AG+ e BRKGA, que apresentam soluções semelhantes, indicando que a estratégia de cruzamento presente em ambos gera bons resultados para problemas de otimização de cargas poluentes em rios.

²Sistema Integrado de Bases Geoespaciais do Estado do Espírito Santo (<https://geobases.es.gov.br>)

Tabela 1: Média e Desvio Padrão Percentual dos conjuntos de inválidos gerados na criação da população inicial, nos cruzamentos e na criação do grupo de mutantes nos modos de execução AG Clássico, AG+ e BRKGA, para ambas poluição pontual e difusa.

Modo	Poluição	Conjunto	Média	DP (%)
AG Clássico	Pontual	População Inicial	169,13	7,52
		Cruzamentos	0,00	0,00
		Mutantes	1845,50	4,34
	Difusa	População Inicial	1144,13	11,26
		Cruzamentos	4933,13	0,74
		Mutantes	11483,38	2,99
AG+	Pontual	População Inicial	171,50	10,82
		Cruzamentos	0,00	0,00
		Mutantes	1779,25	3,44
	Difusa	População Inicial	1073,38	8,46
		Cruzamentos	3521,00	0,89
		Mutantes	11495,25	2,70
BRKGA	Pontual	População Inicial	185,63	8,38
		Cruzamentos	0,00	0,00
		Mutantes	1793,00	4,34
	Difusa	População Inicial	1103,75	9,49
		Cruzamentos	3479,00	0,81
		Mutantes	11630,13	4,65

Tabela 2: Média e Desvio Padrão Percentual dos valores da FO nos modos de execução AG Clássico, AG+ e BRKGA, para ambas poluição pontual e difusa.

Modo	Poluição	Média	DP (%)
AG Clássico	Pontual	1,61	0,43
	Difusa	2,87	3,14
AG+	Pontual	1,59	0,04
	Difusa	1,36	4,18
BRKGA	Pontual	1,59	0,03
	Difusa	1,33	3,86

Para alguns poluentes o BRKGA encontrou valores menores de eficiência, entretanto a similaridade nos resultados do AG+ e do BRKGA sugerem que a estratégia de chaves aleatórias, presente no BRKGA, não apresenta desempenho significativo para este tipo de problema. De modo geral, o BRKGA apresentou resultados mais satisfatórios, dado que este encontrou o menor valor de eficiência (FO = 5,33) para todo o conjunto, sendo este o objetivo caracterizado pelas FOs implementadas. Quanto aos desvios padrão, podemos observar que estes são relativamente altos para todos os parâmetros, excetuando para a DBO5 e, no caso do AG+ e do

Tabela 3: Média e Desvio Padrão Percentual das eficiências da Demanda Bioquímica de Oxigênio (DBO5) e do Oxigênio Dissolvido (OD) nos modos de execução AG Clássico, AG+ e BRKGA para a poluição pontual.

Modo	Parâmetro	Média	DP (%)
AG Clássico	DBO5	0,88	0,53
	OD	0,73	0,63
AG+	DBO5	0,87	0,02
	OD	0,72	0,08
BRKGA	DBO5	0,87	0,02
	OD	0,72	0,05

Tabela 4: Média e Desvio Padrão Percentual da soma das eficiências da Demanda Bioquímica de Oxigênio (DBO5), Nitrogênio Amoniacal (NH3), Nitrito (NO) e Fósforo (P) nos modos de execução AG Clássico, AG+ e BRKGA para a poluição difusa.

Modo	Parâmetro	Média	DP (%)
AG Clássico	DBO5	4,15	5,47
	NH3	2,44	13,57
	NO	2,71	15,26
	P	2,24	20,15
	FO	11,47	3,14
AG+	DBO5	3,03	4,52
	NH3	0,41	18,00
	NO	1,54	5,28
	P	0,40	31,66
	FO	5,42	4,18
BRKGA	DBO5	3,04	4,82
	NH3	0,49	11,60
	NO	1,44	6,28
	P	0,36	26,29
	FO	5,33	3,87

BRKGA, também para o NO. Este fato ocorre, em partes, devido a complexidade existente nos processos ambientais referentes aos parâmetros NH3, NO e P e em função das características randômicas presentes nestes algoritmos, especialmente na geração da população inicial e na mutação. Entretanto, quando se trata dos desvios padrão das FOs observa-se que estes não são tão altos, demonstrando que de modo geral as soluções obtidas nos testes tem características similares quando se trata do objeto de otimização, ou seja, o conjunto de eficiências de todos os parâmetros.

Os comportamentos das eficiências da DBO5 e do OD ao longo da extensão do rio Pardo

para a poluição pontual não apresentam muita variação para os 3 algoritmos implementados, conforme pode ser observado na Fig. 2.

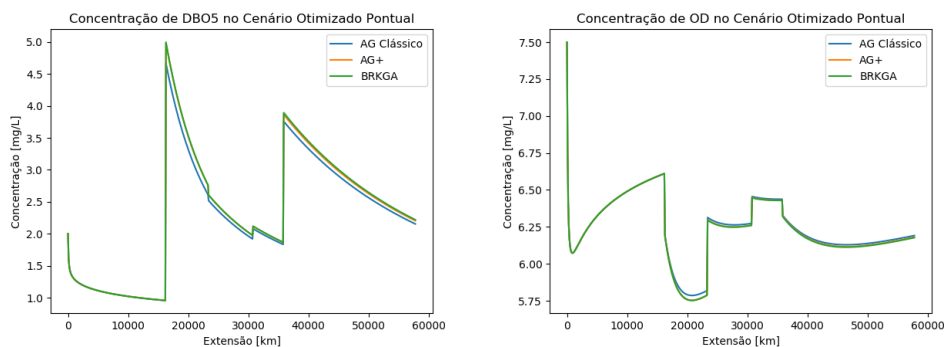


Figura 2: Comportamento das eficiências da Demanda Bioquímica de Oxigênio (DBO5) e do Oxigênio Dissolvido (OD) no cenário otimizado de poluição pontual nos modos de execução AG Clássico, AG+ e BRKGA.

Já para a poluição difusa o comportamento das eficiências de DBO5, NH3, NO e P apresentam comportamentos diferentes, conforme mostra a Fig. 3.

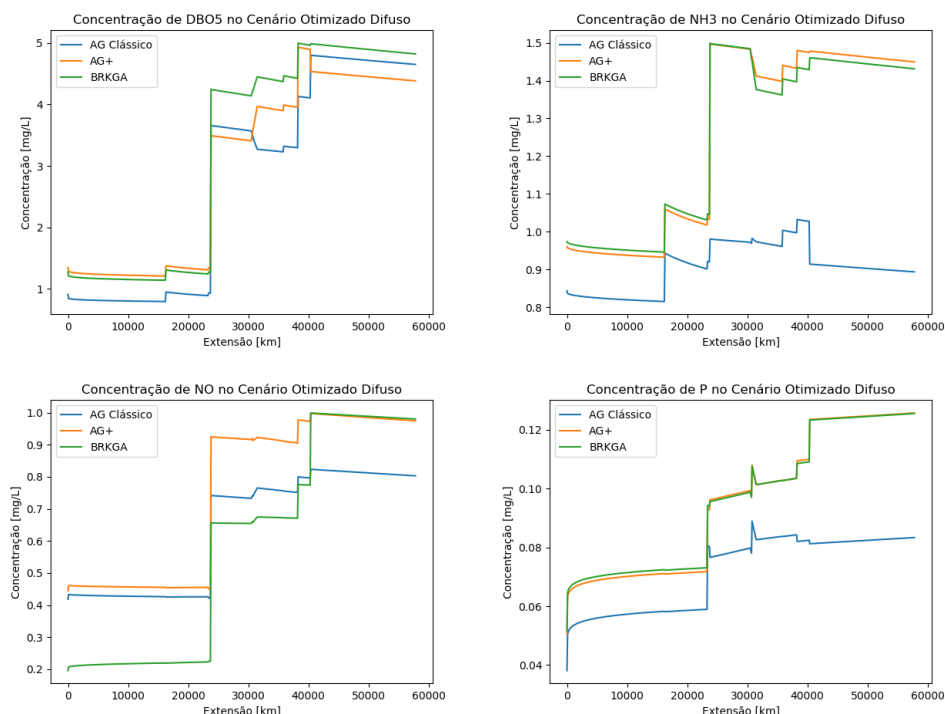


Figura 3: Comportamento das eficiências da Demanda Bioquímica de Oxigênio (DBO5), do Nitrogênio Amoniacal (NH3), do Nitrito (NO) e do Fósforo (P) no cenário otimizado de poluição difusa nos modos de execução AG Clássico, AG+ e BRKGA.

De modo geral, o algoritmo AG básico obtém concentrações menores para todos os parâmetros. Isto ocorre porque o AG clássico encontrou um conjunto de eficiências de maior valor que

aquele encontrado pelos demais algoritmos. Ou seja, os resultados do AG clássico possuem eficiências de remoção de maior valor, entretanto o objetivo dos algoritmos implementados é de minimizar o conjunto de eficiências. Uma vez que o AG clássico encontra soluções com maior percentual de remoção, é esperado que as concentrações poluentes sejam menores. Entretanto, o custo destas soluções será maior, uma vez que maiores eficiência tendem a ser mais custosas.

Os resultados encontrados pelos AG+ e BRKGA são semelhantes para os parâmetros de DBO5, NH3 e P, uma vez que as eficiências de remoção/tratamento encontradas por estes algoritmos são similares. Entretanto, mesmo para parâmetros que obtiveram valores similares de eficiência de remoção, o comportamento dos perfis resultantes da aplicação apresentam diferenças relevantes ao longo do rio. Por exemplo, para a DBO5, o perfil resultante do BRKGA apresentou maiores concentrações deste parâmetro entre os quilômetros 23 e 39 (aproximadamente), indicando que para estes trechos este algoritmo encontrou menores eficiências de remoção. Todavia, o menor somatório de eficiências referente ao parâmetro DBO5 foi encontrado pelo AG+. Analogamente, acerca do NO, o perfil resultante do AG+ apresentou maiores concentrações deste parâmetro no mesmo intervalo, indicando que para estes trechos este algoritmo encontrou menores eficiências de remoção. Todavia, o menor somatório de eficiências referente ao parâmetro NO foi encontrado pelo BRKGA. Para o parâmetro NH3, os valores do somatório de eficiências foram de 0,41 para o AG+ e 0,49 para o BRKGA, explicando a similaridade entre os perfis resultantes. O mesmo pode ser verificado quanto ao P, sendo de 0,40 para o AG+ e 0,36 para o BRKGA.

5 Discussão e Conclusões

Este trabalho apresenta uma aplicação de algoritmos genéticos para encontrar a eficiência mínima de redução de poluentes em rios garantindo o enquadramento na legislação vigente para fontes pontuais e difusas. Foi realizado um estudo comparativo do algoritmo genético clássico, algoritmo genético das chaves aleatórias viciadas (BRKGA) e um algoritmo genético adaptado ao problema, denominado, AG+. Os experimentos apresentam o estudo de qualidade de água na bacia hidrográfica do rio Pardo, localizada no estado do Espírito Santo. Para o conjunto de testes realizados concluímos que para poluição pontual os três algoritmos encontraram soluções similares, enquanto que para a poluição difusa os algoritmos BRKGA e AG+ encontraram soluções superiores e bastante próximas.

Referências

- Carvalho, S. L. (2019). *Sistema De Suporte A Decisão Para Planejamento, Controle E Redução De Cargas Difusas E Pontuais Em Bacias Hidrográficas*. PhD thesis, Programa de Pós-Graduação em Engenharia Ambiental - UFES. Tese de Doutorado em Engenharia Ambiental.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Boston: Addison-Wesley.
- Gonçalves, J. F. & Resende, M. G. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17.
- Hernandez, E. A. e Uddameri, V. (2013). An assessment of optimal waste load allocation and assimilation characteristics in the arroyo colorado river watershed, tx along the us–mexico border. *Clean Technologies and Environmental Policy*, 15, 617–630.
- Linden, R. (2012). *Algoritmos Genéticos*. Porto Alegre: Ciência Moderna, 3. ed. edition.
- Lutz, M. & Ascher, D. (2007). *Aprendendo Python*. Porto Alegre: Bookman, 2. ed. edition.
- Maringanti, C. e. a. (2011). Application of a multi-objective optimization method to provide least cost alternatives for nps pollution control. *Environmental Management*, 48, 448–461.
- Moura, M. A. (2018). *Algoritmo Genético De Chaves Aleatórias Segundo Distribuição De Levy Para Otimização Global*. Master's thesis, Universidade Federal de Pernambuco, Recife.
- Novotny, V. (2003). *Water quality: diffuse pollution and watershed management*. New Jersey: John Wiley.
- Prodanoff, J. H. A. (2005). *Avaliação da Poluição Difusa Gerada por Enxurradas em Meio Urbano*. PhD thesis, Coordenação dos Programas de Pós-Graduação de Engenharia da Universidade Federal do Rio de Janeiro - COPPE/UFRJ. Tese de Doutorado em Engenharia Civil.
- Ribeiro, L. F. O. (2018). *Implementação de rotinas de Otimização de Ferramentas de Controle e Redução de Poluição para Rios e Córregos*. Technical report, Universidade Federal do Espírito Santo (UFES).
- Ribeiro, L. F. O. (2019). *Implementação de rotinas de Otimização de Ferramentas de Controle e Redução de Poluição para Rios e Córregos - Aperfeiçoamento e Complementação do SSD*. Technical report, Universidade Federal do Espírito Santo (UFES).
- VON SPERLING, M. (1996). *Introdução À Qualidade Das Águas E Ao Tratamento De Esgotos*. Belo Horizonte: Departamento de Engenharia Sanitária e Ambiental, UFMG, 2. ed. edition.