

Geração de Sequências Curriculares Adaptativas baseada no Perfil dos Alunos e Materiais Didáticos utilizando o Algoritmo NSGA-III

Bernadete Aquino¹, Jairo Francisco de Souza^{1,2},
Luciana Brugiolo Gonçalves², Stênio São Rorário Furtado Soares²

¹LApIC Research Group
Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Juiz de Fora (UFJF)

²Departamento de Ciência da Computação
Universidade Federal de Juiz de Fora (UFJF)
Juiz de Fora, MG – Brasil

{baquino, jairo.souza, lbrugiolo, ssoares}@ice.ufjf.br

Abstract. *Adaptive Curriculum Sequencing (ACS) problem has been treated in the literature with metaheuristic approaches and with techniques that reduce it to a mono-objective optimization problem. The objectives used are mostly conflicting, that is, an improvement in one of these objectives does not necessarily result in the improvement of the others. Thus, the objective of this article is to propose a new approach to ACS based on many objectives optimization using the NSGA-III algorithm. Although it is still an unexplored solution, it proved to be adequate for the problem, according to experiments carried out in the laboratory.*

Resumo. *O problema do Sequenciamento Curricular Adaptativo (SCA) vem sendo tratado na literatura com técnicas de Inteligência Computacional e com abordagens que o reduzem a um problema de otimização mono-objetivo. Em geral, os objetivos tratados são, em sua maioria, conflitantes, ou seja, uma melhora em um desses objetivos não resulta, necessariamente, na melhora dos demais. Este trabalho propõe uma nova abordagem para o SCA baseada em otimização de problema multiobjetivo utilizando o algoritmo NSGA-III. Embora seja uma solução ainda não explorada, os experimentos realizados mostraram que a abordagem é adequada ao problema.*

1. Introdução

Com o rápido crescimento da tecnologia da informação, observa-se uma quantidade crescente de sistemas *e-Learning*. Desse modo, alunos e professores são expostos a uma ampla variedade de recursos e objetos de aprendizagem. Para obter o conhecimento desejado, alunos precisam escolher cursos, lições e sua sequência nessa ampla variedade de conteúdos [Shmelev et al. 2015]. Estudos mostram que considerar particularidades em relação aos estilos de aprendizagem, interesses, motivação e habilidades cognitivas tem um efeito positivo nos resultados de aprendizagem e no progresso dos estudantes [Gonçalves et al. 2016]. Essa personalização e flexibilização de recursos e atividades é tema de intensas pesquisas na área de Sistemas Tutores Inteligentes [Anschau et al. 2017].

O objetivo dos Sistemas Tutores Inteligentes (STIs) é auxiliar o aluno a alcançar o ganho máximo de aprendizagem e melhorar seu engajamento, capturando seus interesses através do comportamento adaptativo do sistema [Sani and Aris 2014]. Nos STIs, o Sequenciamento Curricular Adaptativo (SCA) é utilizado para proporcionar um caminho de aprendizagem personalizado, ajustando o aprendizado de acordo com conhecimentos prévios, preferências e metas do estudante [Dwivedi et al. 2018].

O problema do SCA apresenta, tipicamente, vários objetivos: apresentar uma sequência compatível com a disponibilidade de tempo do aluno, os matérias da sequência estejam em conformidade com o estilo de aprendizagem e com o nível de habilidade do aluno nos conceitos abordados, entre outros. Estes objetivos são muitas vezes conflitantes, ou seja, não é possível obter uma única solução que otimize-os simultaneamente. Essa característica torna o problema propício para a aplicação de técnicas de otimização multiobjetivo.

A finalidade desse artigo é propor uma abordagem de otimização multiobjetivo ao problema do SCA por meio da utilização do algoritmo *Nondominated Sorting in Genetic Algorithms III* (NSGA-III). O estudo de caso nesta pesquisa considera cinco objetivos, que levam em conta os diferentes aspectos e produz um conjunto de possíveis sequências de conteúdo que atendam aos múltiplos objetivos e permite ao estudante encontrar uma solução mais apropriada por meio de articulação posterior de suas preferências.

O trabalho está organizado da seguinte forma: a Seção 2 descreve o problema do SCA e apresenta os trabalhos relacionados. A metodologia utilizada e os experimentos são descritos na Seção 3 e 4, respectivamente. Por fim, a Seção 5 apresenta as considerações finais.

2. Sequenciamento Curricular Adaptativo: um problema com muitos objetivos

O SCA é a tarefa de organizar o conteúdo, exercícios e atividades para atingir os objetivos de aprendizado. Como cada aluno tem características particulares, são necessárias sequências diferentes para cada perfil [Vanitha et al. 2019]. O objetivo do SCA é substituir a estrutura de curso rígida, geral e de tamanho único por um caminho de aprendizado mais flexível e personalizado [Al-Muhaideb and Menai 2011]. O SCA é um problema NP-difícil, como demonstrado por [Acampora et al. 2011], e pode ser considerado como um problema de *Constraint Satisfaction* ou como um problema de otimização multiobjetivo [Dwivedi et al. 2018, Al-Muhaideb and Menai 2011].

As abordagens baseadas em Computação Evolutiva (CE) são amplamente utilizadas na literatura para tratar o problema do SCA, como o uso do Algoritmo Presa-Predador [Machado et al. 2018], Colônias de Formigas [Vanitha and Krishnan 2019, Kamsa et al. 2018], Algoritmo Genético [Christudas et al. 2018] ou uma combinação destes, como em [Vanitha et al. 2019], onde os autores aplicaram um algoritmo de otimização colaborativa combinando o Algoritmo de Colônias de Formigas com Algoritmo Genético.

Em [Barrére et al. 2017], foi proposto um modelo para comparação dos trabalhos sobre o SCA baseado em Computação Evolutiva, onde as seguintes variáveis foram consideradas: tipo de sequenciamento (TS), podendo ser o sequenciamento individual (SI) ou

o sequenciamento social (SS); tipo de representação do caminho de aprendizagem (CA) e informações intrínsecas e extrínsecas dos alunos e materiais. Entre os parâmetros intrínsecos do aluno (PIA) estão as características que definem as preferências de aprendizagem de cada aluno (A) e a intenção do aluno ao acessar o *e-Learning* (I). Dentre os parâmetros do material didático (PMD) estão: o grau de complexidade de um conceito (D); o tempo médio de aprendizado (TM); a relação entre os conceitos (C); e o tipo do material didático (M). Já entre os parâmetros extrínsecos dos alunos (PEA) estão: o nível de conhecimento de um aluno perante um conceito (N); e a relação entre material didático e aluno (R). A Tabela 1 apresenta uma comparação entre os trabalhos relacionados utilizando o modelo apresentado.

Tabela 1. Comparação entre os trabalhos sobre o SCA.

Artigo	TS	CA	PIA	PMD	PEA
[Christudas et al. 2018]	SS	Vetor Binário	A	D, M	N, R
[Kamsa et al. 2018]	SS	Grafo Ponderado	A	D, M	N, R
[Machado et al. 2018]	SI	Vetor Binário	A, I	D, TM, M	N, R
[Vanitha and Krishnan 2019]	SS	Grafo Ponderado	A, I	D, TM, C	N, R
[Vanitha et al. 2019]	SS	Grafo Ponderado	A, I	D, TM, C	N, R

Em todos os trabalhos analisados, os objetivos para a personalização da entrega do conteúdo foram avaliados de maneira conjunta, seja utilizando a soma ponderada dos objetivos [Machado et al. 2018, Kamsa et al. 2018] ou utilizando a programação por metas para encontrar soluções que possam alcançar uma meta pré-determinada [Vanitha and Krishnan 2019, Vanitha et al. 2019, Christudas et al. 2018]. Essas técnicas reduzem o problema de otimização multiobjetivo a um problema de otimização mono-objetivo.

O estudo realizado por [Machado et al. 2019] levantou os parâmetros de materiais e alunos mais comuns nos trabalhos de SCA. Ainda, verificou quais os objetivos mais utilizados na literatura, chegando a um total de cinco objetivos. Neste artigo, será verificado como esses objetivos são, na verdade, conflitantes. Como o número de objetivos a ser considerado na formulação do problema é alto, serão utilizadas técnicas de otimização que lidam com muitos objetivos. Para tanto, seja m_i o i -ésimo material didático, c_j o j -ésimo conceito de um curso e x_i indica um valor binário da i -ésima posição de um vetor solução. Além disso, dado o aluno a_k e o material m_i , $E_{c_j}^{a_k}$ indica se o conceito c_j é esperado pelo aluno a_k ; $H_{c_j}^{a_k}$ indica o grau de habilidade deste aluno em relação ao conceito c_j ; $T_{inf}^{a_k}$ e $T_{sup}^{a_k}$ indicam os limites inferior e superior do tempo esperado de duração do curso para o aluno a_k ; E^{a_k} , o conjunto de conceitos que integra o objetivo de aprendizagem do aluno a_k e; $L_l^{a_k}$, que assinala, para cada estilo de aprendizagem l do *Felder and Silverman Learning Style Model* (FSLSM) a intensidade do aluno. Quanto aos materiais, dado o material m_i considerou-se: D^{m_i} , grau de dificuldade; $R_{c_j}^{m_i}$, que indica se o material contém um conceito c_j ; $T_{est}^{m_i}$, tempo estimado de aprendizagem do material; R^{m_i} , conjunto de conceitos abordados pelo material; e $\theta_l^{m_i}$ que indica a intensidade do estilos de aprendizagem de cada uma das dimensões l do FSLSM o material apresenta.

Todos os objetivos foram modelados como funções a serem minimizadas e são descritas a seguir.

1. Equilíbrio entre os conceitos: analisa se os conceitos cobertos pela sequência selecionada estão equilibrados com a meta de aprendizagem do aluno.

$$\sum_{j=1}^{|C|} E_{c_j}^{a_k} \left| \sum_{i=1}^{|M|} x_i R_{c_j}^{m_i} - \frac{\sum_{i=1}^{|M|} \sum_{j'=1}^{|C|} x_i R_{c_{j'}}^{m_i}}{\sum_{j'=1}^{|C|} E_{c_{j'}}^{a_k}} \right| \quad (1)$$

2. Concordância quanto à dificuldade: averigua se a dificuldade do material é adequada a capacidade média do aluno nos conceitos abordados pelos materiais.

$$\sum_{i=1}^{|M|} x_i \left| D^{m_i} - \frac{\sum_{j=1}^{|C|} R_{c_j}^{m_i} E_{c_j}^{a_k} H_{c_j}^{a_k}}{\sum_{j=1}^{|C|} R_{c_j}^{m_i} E_{c_j}^{a_k}} \right| \frac{1}{\sum_{i=1}^{|M|} x_i} \quad (2)$$

3. Estilo de aprendizagem: examina se as características do material são adequadas ao estilo de aprendizagem do aluno.

$$\sum_{l=1}^4 \frac{\sum_{i=1}^{|M|} x_i |3sgn(\theta_l^{m_i} - L_l^{a_k})|}{4 \sum_{i=1}^{|M|} x_i} \quad (3)$$

4. Cobertura dos conceitos: verifica se os conceitos cobertos pela sequência selecionada atendem às metas de aprendizado do aluno.

$$(|R^{m_i}| - |R^{m_i} \cap E^{a_k}|) + (|E^{a_k}| - |R^{m_i} \cap E^{a_k}|) \quad (4)$$

5. Compatibilidade de tempo: observa se o tempo total do curso está ou não entre o limite inferior e superior de tempo esperado pelo aluno.

$$\left(\max \left(T_{inf}^{a_k} - \sum_{i=1}^{|M|} T_{est}^{m_i}, 0 \right) + \left(0, \sum_{i=1}^{|M|} T_{est}^{m_i} x_i - T_{sup}^{a_k} \right) \right) \quad (5)$$

A aplicação de um algoritmo evolutivo (AE) multiobjetivo apresenta várias vantagens em relação às técnicas de redução, dentre elas, pode-se citar que um AE multiobjetivo não cria parâmetros adicionais ao problema; opera diretamente com várias funções-objetivo usando o conceito de Fronteira de Pareto; em apenas uma execução do algoritmo, um conjunto diversificado de soluções pode ser encontrado; é menos suscetível à forma e continuidade da Fronteira de Pareto [Azuma 2011]. Essas vantagens tornam o uso de um AE multiobjetivo propício para tratar o problema do SCA, onde a diversidade de soluções e a necessidade de se considerar objetivos conflitantes são características bastante desejáveis. O diferencial deste trabalho, em relação às propostas já existentes na literatura, está no tratamento do problema do SCA efetivamente como um problema multiobjetivo, usando algoritmos e técnicas concebidas especificamente para esse fim.

3. O Algoritmo NSGA-III para o problema de SCA

A otimalidade em problemas de otimização multiobjetivo é definida em termos de fronteira de Pareto, composta por um conjunto de soluções não-dominantes entre si. Considerando um conjunto de objetivos $F = \{f_1, f_2, \dots, f_n\}$, dadas duas soluções s_1 e s_2 , diz-se que s_1 domina s_2 se, para todo objetivo f_j , $f_j(s_1)$ é melhor ou igual a $f_j(s_2)$, e estritamente melhor em pelo menos um dos objetivos de F . No processo de otimização multiobjetivo o que se busca é determinar as soluções pertencentes ao conjunto Pareto.

Existem várias propostas de algoritmos evolutivos voltados para otimização multiobjetivo. Neste trabalho, é apresentada uma abordagem baseada no NSGA-III para o SCA. Atualmente, esse algoritmo é amplamente estudado e aplicado na solução de problemas multiobjetivo [Zhou et al. 2020]. Contudo, não foi explorado na área de SCA e foi escolhido por se adequar bem à resolução de problemas com muitos objetivos, com a vantagem de manter a diversidade das soluções em relação ao seu predecessor, NSGA-II.

O algoritmo NSGA-III foi proposto por [Deb and Jain 2014], sendo baseado em pontos de referência e seguindo a mesma estrutura de seu predecessor, o NSGA-II. Ambos utilizam operador de cruzamento e mutação para gerar indivíduos descendentes e aplicam uma abordagem rápida de classificação não dominada para dividir os indivíduos em diferentes fronteiras não dominadas. Nessa abordagem, todos os indivíduos do nível 1 ao nível l da fronteira não dominada são primeiramente incluídos na população parental P_t . Se $|P_t| = N$, sendo N o tamanho da população, então a próxima geração é iniciada com P_t . Caso $|P_t| > N$, os indivíduos da fronteira $l - 1$ já estão selecionados e os demais indivíduos da população são escolhidos da última fronteira F_l [Deb and Jain 2014].

O processo de seleção dos indivíduos na última fronteira F_l é a principal diferença entre o NSGA-II e o NSGA-III. O NSGA-II aplica a distância de agrupamento, enquanto que o NSGA-III utiliza o mecanismo de seleção baseado em ponto de referência, onde o vetor objetivo de cada membro da população F_l é normalizado e associado aos pontos de referência Z^r gerados. Cada ponto de referência representa uma direção de pesquisa. Assim, os K membros apropriados são selecionados de F_l com base na contagem de nichos de Z^r . Os pontos de referência distribuídos uniformemente garantem a uniformidade da busca ótima e, assim, o algoritmo evita cair em um ótimo local. Logo, o NSGA-III pode encontrar com êxito uma solução ótima de Pareto, convergente e diversificada, na solução de problemas com muitos objetivos [Cao et al. 2019].

O algoritmo NSGA-III não requer a definição de nenhum novo parâmetro além dos parâmetros usuais do Algoritmo Genético, como tamanho da população, critério de parada, probabilidades de cruzamento e mutação e seus parâmetros associados. A quantidade de pontos de referência H é calculada com base no número de objetivos e no número de divisões p ao longo de cada eixo objetivo. O tamanho da população N depende de H , onde $H \approx N$ [Deb and Jain 2014].

No contexto desse trabalho, cada indivíduo representa uma solução, que é uma sequência de materiais didáticos. O indivíduo foi modelado como um vetor de números binários $s = [x_1, \dots, x_{|M|-1}, x_{|M|}]$, onde M representa o número de materiais no repositório e x_i um valor binário que indica se o material m_i está presente na solução ou não. Para geração da população inicial P_0 , os valores x_i são gerados aleatoriamente e toda a população inicial é gerada por esse processo. Cada indivíduo de P_0 , bem como os das gerações seguintes, tem calculado o valor para cada uma das funções objetivo. A população parental P_t de tamanho N é utilizada para criar uma nova população de descendentes Q_t , também de tamanho N , aplicando-se os operadores genéticos de cruzamento e mutação. O mecanismo de classificação não-dominada baseada em pontos de referência seleciona a nova população de pais R_t com os N melhores membros de $P_t \cup Q_t$ (de tamanho $2N$). Esse processo se repete até que o critério de parada seja atingido [Deb and Jain 2014].

O operador de mutação consiste em alterar aleatoriamente uma parte específica

da representação do indivíduo no sentido de aumentar a variabilidade do genótipo da população. No esquema proposto, são realizados testes de mutação para cada elemento do cromossomo: o valor do bit é invertido caso a condição estabelecida no teste seja satisfeita. Para realização da recombinação dos genótipos entre dois indivíduos, utilizou-se o cruzamento com dois pontos de corte, onde dois descendentes são formados pela recombinação de subcadeias binárias definidas com base em dois pontos de corte gerados de forma aleatória e fixados no vetor solução de cada um dos dois indivíduos pais. Já os pontos de referência foram predefinidos de maneira estruturada, colocados em um hiperplano normalizado, igualmente inclinado para todos os eixos objetivos.

4. Experimentos, Resultados e Discussões

A configuração dos parâmetros de um algoritmo de busca heurística influencia significativamente seu desempenho. Neste sentido, na condução dos testes realizados, alguns experimentos com diferentes valores de parâmetros foram necessários para se obter as melhores configurações para o algoritmo. Assim, utilizou-se no ajuste de parâmetros os seguintes valores: p : {4, 5, 6}, para 5 objetivos; quantidade de pontos de referência H : {70, 126, 210}, respectivamente. Como $H \approx N$, os valores {72, 128, 212} também foram utilizados para a variação no tamanho da população N . Analisou-se ainda o número de gerações G , variando entre {100, 200 e 300}; a probabilidade de cruzamento C_x : {0.8, 0.9, 1.0} e a probabilidade de mutação M_u : {0.01, 0.02, 0.03}. Para avaliação do modelo, foram utilizados dados de 1/3 dos alunos, escolhidos de forma aleatória, da base¹ que contém dados de 284 materiais de aprendizagem e 24 alunos. Foram realizadas 15 execuções com cada um desses alunos.

Para avaliação de algoritmos multiobjetivos, uma métrica bastante utilizada é o indicador de hipervolume. Um alto valor de hipervolume indica que houve um elevado espalhamento entre as soluções e uma melhor convergência [Azuma 2011]. Neste trabalho, o vetor objetivo de cada membro da população final foi normalizado, então calculou-se o volume da região coberta por essas soluções e o ponto de referência (1.1, 1.1, 1.1, 1.1, 1.1). A Figura 1 apresenta o gráfico *boxplot* dos hipervolumes obtidos em cada combinação de parâmetro do algoritmo. Cada grupo apresenta uma comparação entre os três diferentes valores para cada parâmetro e a legenda informa quais os valores fixados para os demais parâmetros. Observa-se que há pouca variabilidade dentre dos grupos. Assim, para a escolha das diferentes configurações, priorizou-se um valor maior para a mediana e para o limite inferior. Dessa forma, chegou-se à seguinte configuração: tamanho da população N igual a 212; número de gerações G sendo 300; probabilidade de cruzamento C_x como 1 e probabilidade de mutação M_u igual a 0.01.

Os problemas de otimização com muitos objetivos requerem métodos adequados para visualização do conjunto de soluções. Uma alternativa para visualização dos resultados é a utilização de um gráfico de coordenadas paralelas, que permite exibir dados multidimensionais em um gráfico bidimensional. Neste gráfico, as coordenadas paralelas permitem conhecer o conflito entre os objetivos. Quando há muitas linhas cruzando entre dois objetivos adjacentes, isso indica que há conflito entre esses objetivos [Li et al. 2017]. A Figura 2 apresenta três gráficos de coordenadas paralelas para um mesmo conjunto de

¹<https://github.com/ufjf-dcc/LAPIC3-benchmark/tree/master/LaboratoryExperiments/Databases>

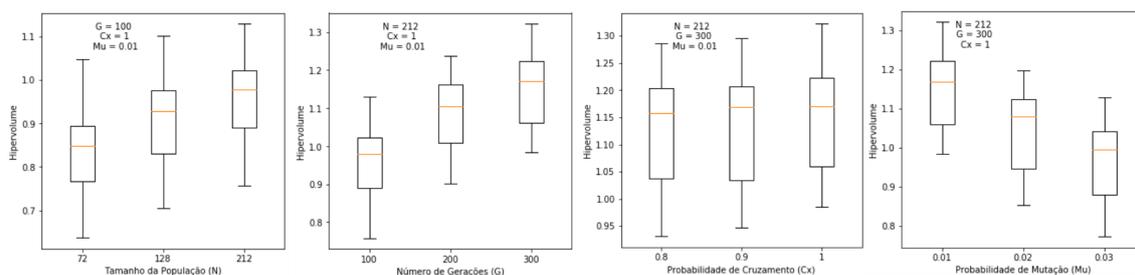


Figura 1. Hipervolume para cada configuração de parâmetros.

soluções retornadas pelo algoritmo para um caso de teste referente a um aluno selecionado de forma aleatória dentre os 24 alunos que integraram o experimento realizado. Cada gráfico apresenta uma ordem dos objetivos e permite comparar os conflitos entre cada par. Como muitas linhas se cruzam entre todos os pares, os objetivos são altamente conflitantes. É possível também observar que o conjunto de soluções falha em cobrir a região entre 0 e 0.2, exceto no quinto objetivo. Ao analisar os gráficos referentes aos experimentos com os demais alunos da base, observou-se o mesmo comportamento do exemplo apresentado.

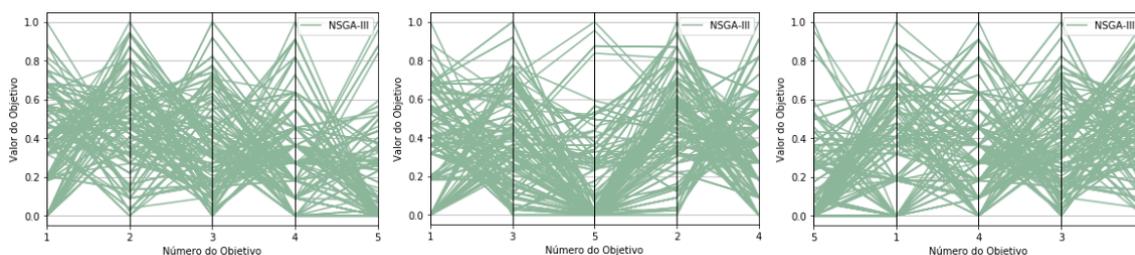


Figura 2. Conjunto de soluções mostrado em coordenadas paralelas.

Para analisar a qualidade dos resultados da otimização, foram comparados o valor médio de cada objetivo das soluções iniciais (soluções obtidas na primeira geração) com o valor médio de cada objetivo das soluções finais para todos os alunos da base. À medida que o desempenho médio melhora, a qualidade dos resultados da otimização aumenta. A diminuição percentual para cada um dos cinco objetivos é listada na Figura 3. Os resultados indicaram que os valores objetivos tiveram uma melhoria média significativa, maior que 25%. No caso do objetivo 4, cobertura dos conceitos, observou-se que 17 instâncias obtiveram melhores resultados e 7 instâncias tiveram resultados piores, sendo então o objetivo com menor ganho médio. O objetivo 5, compatibilidade de tempo, já atingia valores mínimos na solução inicial para 6 alunos, não sendo, portanto, passível de melhora nas gerações seguintes.

O algoritmo NSGA-III foi comparado com o Algoritmo Presa-Predador (APP) proposto por [Machado et al. 2018], por ser uma abordagem recente e ter utilizado a mesma base de referência, o que permite uma comparação mais justa dos resultados. Para este experimento, foram realizadas 10 execuções com cada algoritmo para cada instância do problema (aluno e conjunto de materiais didáticos). Como o NSGA-III apresenta como resultado um conjunto de soluções, a solução escolhida para a comparação com o APP foi a que apresentou o menor valor para a soma dos cinco objetivos, já que o APP

não priorizou nenhum objetivo.

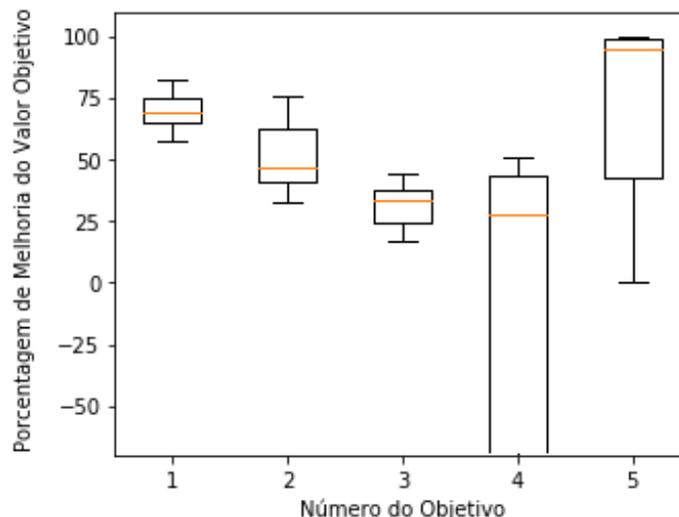


Figura 3. Diminuição percentual dos valores objetivos.

A Figura 4 apresenta o gráfico com o resumo dos resultados considerando os 24 alunos (eixo das ordenadas) em cada um dos cinco objetivos (eixo das abscissas). É possível observar que o NSGA-III obteve os melhores resultados em todos os 24 alunos tanto no objetivo 2 (concordância quanto à dificuldade) quanto no objetivo 3 (estilo de aprendizagem). Já quanto à cobertura dos conceitos (objetivo 4), foi melhor em 71% dos casos. Por outro lado, quanto ao objetivo 5 (compatibilidade de tempo), ambos os algoritmos zeraram a função objetivo em todas as execuções, o que considera-se empate, em 92% dos casos. O APP foi melhor em 67% dos casos para o objetivo 1 (equilíbrio entre os conceitos). Entretanto, convém destacar que as funções objetivo no APP não são normalizadas. Assim, uma diminuição nos valores obtidos pela função referente a este objetivo causa um impacto maior que a diminuição das demais funções, o que pode explicar o ganho do APP nesse objetivo específico.

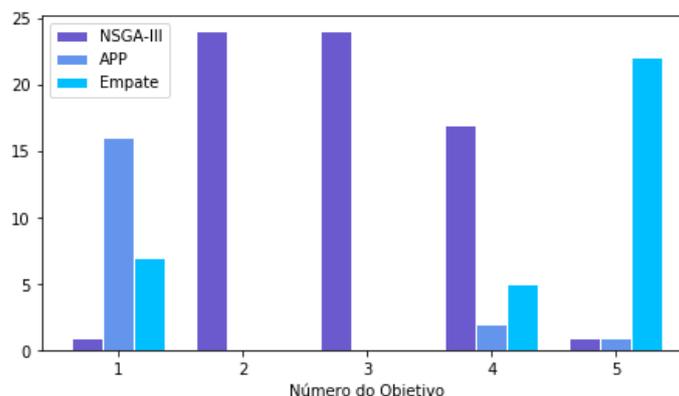


Figura 4. Comparação entre NSGA-III e APP.

5. Considerações Finais

O problema do SCA vem sendo tratado na literatura com abordagens que o reduzem a um problema mono-objetivo. Cada forma de redução adiciona novos parâmetros ao pro-

blema, sendo que a escolha desses parâmetros afeta diretamente os resultados alcançados. Para preencher essa lacuna de pesquisa, esse trabalho propôs uma abordagem de otimização com múltiplos objetivos ao problema do SCA, utilizando o algoritmo NSGA-III. Com essa abordagem, o aluno possui um protagonismo maior na tomada de decisão referente ao planejamento de seu curso, podendo articular suas preferências tendo um conjunto de soluções alternativas. Para analisar as soluções, foi utilizado o gráfico de coordenadas paralelas. A porcentagem de melhoria dos objetivos das soluções finais em relação as soluções iniciais foi empregada para verificar a eficiência do algoritmo, que obteve uma melhoria média de 25% em comparação como uma abordagem mono-objetivo clássica.

A análise da qualidade dos resultados e a comparação com os resultados apresentados por uma abordagem do tipo presa-predador, permitiu verificar que a utilização da otimização com muitos objetivos por meio do algoritmo NSGA-III apresentou melhores resultados, mesmo tomando por base cada objetivo individualmente, para a maioria dos casos de teste. Tais resultados corroboram a necessidade de novas pesquisas para a solução do SCA utilizando abordagens de otimização com muitos objetivos, as quais têm a vantagem de encontrar diversas soluções para o problema do SCA e franquear ao aluno ou gestores a escolha das soluções que melhor atendam necessidades específicas.

O estudo possui limitações que precisam ser exploradas em estudos futuros. O desempenho do algoritmo está diretamente relacionado a uma escolha adequada de parâmetros de projeto. Contudo, esses parâmetros foram selecionados de maneira empírica. Como trabalho futuro, pretende-se utilizar um *framework* como o Iterated Race (IRACE) para selecionar as melhores configurações. Pretende-se também realizar comparações do algoritmo proposto com a segunda versão do NSGA-III proposta em [Jain and Deb 2014], o qual implementou melhorias no mecanismo de seleção e no mecanismo de criação da população descendente, obtendo uma representação mais densa da fronteira de Pareto.

Referências

- Acampora, G., Gaeta, M., and Loia, V. (2011). Hierarchical optimization of personalized experiences for e-learning systems through evolutionary models. *Neural Computing and Applications*, 20(5):641–657.
- Al-Muhaideb, S. and Menai, M. E. B. (2011). Evolutionary computation approaches to the curriculum sequencing problem. *Natural Computing*, 10.
- Anschau, D., Marchi, J., and Rizzon, E. (2017). Smart its: um sistema tutor inteligente para flexibilização e adaptação de currículos. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 28(1):1347.
- Azuma, R. M. (2011). Otimização multiobjetivo em problema de estoque e roteamento gerenciados pelo fornecedor. Master's thesis, Universidade Estadual De Campinas.
- Barrére, E., Souza, J., and Machado, M. (2017). Geração de sequências curriculares adaptativas baseada em computação evolucionária: Estado da arte e tendências. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 28(1):1137.
- Cao, X., Xu, J., De Clercq, D., Wang, Y., and TAO, Y. (2019). Many-objective optimization of technology implementation in the industrial symbiosis system based on a modified nsga-iii. *Journal of Cleaner Production*, 245:118810.

- Christudas, B., Kirubakaran, E., and Thangaiah, P. (2018). An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials. *Telematics and Informatics*, page 520–533.
- Deb, K. and Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 18(4).
- Dwivedi, P., Kant, V., and Bharadwaj, K. (2018). Learning path recommendation based on modified variable length genetic algorithm. *Education and Information Technologies*.
- Gonçalves, A., Vivas, A., Assis, L., Pitangui, C., and Dorça, F. (2016). Avanços na modelagem automática e dinâmica de estilos de aprendizagem de estudantes em sistemas adaptativos e inteligentes para educação: uma análise experimental. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 27(1):1006.
- Jain, H. and Deb, K. (2014). An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4):602–622.
- Kamsa, I., Elouahbi, R., and Khoukhi, F. (2018). The combination between the individual factors and the collective experience for ultimate optimization learning path using ant colony algorithm. *Engineering and Information Technology*, 8:1198–1208.
- Li, M., Zhen, L., and Yao, X. (2017). How to read many-objective solution sets in parallel coordinates. *IEEE Computational Intelligence Magazine*, 12.
- Machado, M., Barrere, E., and Souza, J. (2019). Solving the adaptive curriculum sequencing problem with prey-predator algorithm. *International Journal of Distance Education Technologies*, 17:71–93.
- Machado, M., Barrére, E., and Souza, J. (2018). Uma abordagem evolutiva para o problema de sequenciamento curricular adaptativo. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 29(1):1243.
- Sani, S. and Aris, T. N. M. (2014). Computational intelligence approaches for student/tutor modelling: A review. In *2014 5th International Conference on Intelligent Systems, Modelling and Simulation*, pages 72–76.
- Shmelev, V., Karpova, M., and Dukhanov, A. (2015). An approach of learning path sequencing based on revised bloom’s taxonomy and domain ontologies with the use of genetic algorithms. *Procedia Computer Science*, page 711–719.
- Vanitha, V. and Krishnan, P. (2019). A modified ant colony algorithm for personalized learning path construction. *Journal of Intelligent & Fuzzy Systems*, 37:1–16.
- Vanitha, V., Krishnan, P., and R., E. (2019). Collaborative optimization algorithm for learning path construction in e-learning. *Computers & Electrical Engineering*, 77.
- Zhou, B., Liu, B., Yang, D., Cao, J., and Littler, T. (2020). Multi-objective optimal operation of coastal hydro-electrical energy system with seawater reverse osmosis desalination based on constrained nsga-iii. *Energy Conversion and Management*, 207:112533.