

# A Novel One-to-Many Matching Method for the Assignment Problem: An ENEM Case Study

Giuseppe F. Neto<sup>1</sup>, Péricles B.C. Miranda<sup>1</sup>, Rafael Ferreira Mello<sup>1</sup>,  
André C.A. Nascimento<sup>1</sup>

<sup>1</sup>Departamento de Estatística e Informática  
Universidade Federal Rural de Pernambuco (UFRPE)

**Abstract.** *Distributing candidates in test locations is a relevant logistics problem and affects different countries, including Brazil, that conduct selection exams through face-to-face assessments. Defining an appropriate distribution considering criteria such as distance, cost, and occupation is a challenging task. This work treats the task in question as a combinatorial problem of stable marriage and proposes a new optimization algorithm (O2MSM) for the automatic distribution of candidates to test sites. The O2MSM aims to find a stable correspondence between candidates and test locations, minimizing the distance between them and the number of test locations and maximizing the occupancy rate of these locations. The results showed that O2MSM surpassed the baseline approach, being more efficient, performing the distribution of candidates in seconds, and more effective, reducing the number of test locations, vacancies, and distance to candidates as much as possible.*

**Resumo.** *A distribuição de candidatos em locais de provas é um problema logístico relevante e afeta diversos países, inclusive o Brasil, que realizam exames de seleção por meio de avaliações presenciais. Definir uma distribuição adequada considerando critérios como distância, custo e ocupação é uma tarefa desafiadora. Este trabalho trata a tarefa em questão como um problema combinatório de casamento estável e propõe um novo algoritmo de otimização (O2MSM) para a distribuição automática de candidatos a locais de teste. O O2MSM visa encontrar uma correspondência estável entre candidatos e locais de prova, minimizando a distância entre eles e o número de locais de prova e maximizando a taxa de ocupação desses locais. Os resultados mostraram que o O2MSM superou a abordagem baseline, sendo mais eficiente, realizando a distribuição dos candidatos em segundos, e mais eficaz, reduzindo ao máximo o número de locais de prova, vagas e distância dos candidatos.*

## 1. Introdução

O Exame Nacional do Ensino Médio (ENEM) é um sistema centralizado de avaliação, de abrangência nacional, que seleciona candidatos para o ensino superior [Lima et al. 2019]. O ENEM é o maior processo seletivo do Brasil recebendo, em média, sete milhões de inscrições. Para a realização das provas, o ENEM precisa distribuir os candidatos em locais previamente autorizados, tentando garantir a ocupação máxima de candidatos e o número mínimo de locais de prova possível. O intuito é atender a todos, reduzindo o custo [Rocha et al. 2019, Lima et al. 2019]. O atual sistema de distribuição de candidatos segue, desde 2016, regras básicas, definidas pelo *Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira* (INEP) [INEP], que são:

- Respeitar o município e unidade da Federação indicados na inscrição pelo participante;
- Distribuir os candidatos em um raio de até 30 km do local de sua residência, quando o município da realização da prova selecionado pelo participante no momento da inscrição for o mesmo de seu domicílio;
- Agrupá-los em salas preferencialmente em número múltiplo de quatro, considerando, em média, 40 participantes por sala;
- Garantir que pessoas com deficiência sejam distribuídas em locais com acessibilidade.

O Problema de Distribuição de Candidatos em Locais de prova (PDCL) é um desafio clássico de exames de seleção, no mundo inteiro, baseados em provas presenciais [Lima et al. 2019, Rocha et al. 2019], que também acontece no ENEM. Levando-se em conta as restrições acima, e as dimensões continentais do território brasileiro, a distribuição de candidatos do ENEM torna-se uma tarefa desafiadora [Rocha et al. 2019]. Devido a sua relevância e recorrência, o problema da distribuição passou a ser encarado como um problema de otimização combinatória [Ranjan and Singh 2018]. Com isso, algoritmos de otimização passaram a ser aplicados com o objetivo de tentar encontrar a melhor configuração de candidatos-locais de provas de forma automática [Rocha et al. 2019].

Este trabalho define o PDCL como sendo uma instância do problema combinatório de casamento estável (do inglês, *Stable Marriage Problem* - SMP) [Biró 2017], chamada de Problema da Admissão na Faculdade (PAF). No PAF, alunos e escolas têm preferências, escolas podem matricular muitos alunos, e um aluno só pode ser matriculado em uma escola. Esta característica restringe o PAF a um tipo específico de casamento estável, chamado *one-to-many stable marriage*. Esta natureza do PAF se molda às características do PDCL; uma vez que as preferências podem estar relacionadas à distância, locais de prova podem receber vários candidatos, e os candidatos só podem ser alocados a um local de prova.

Diate deste contexto, este trabalho propõe um novo algoritmo de otimização para a distribuição automática de candidatos a pontos de prova. A abordagem proposta é inspirada no algoritmo de casamento estável de *Gale-Shapley*, muito usado no PAF [Gale and Shapley 2013]. No entanto, a proposta se trata de uma versão otimizada e adaptada para atender aos requisitos básicos do ENEM e do INEP, sendo chamado de O2MSM (do inglês, *Optimized One-to-Many Stable Marriage*). O algoritmo proposto tem como objetivo encontrar uma correspondência estável entre dois conjuntos de elementos, candidatos e locais de prova, minimizando a distância entre eles. No entanto, além disso, o algoritmo tenta minimizar o número de pontos de prova, e maximizar a taxa de ocupação destes pontos.

O algoritmo proposto (O2MSM) foi comparado com duas outras abordagens (i) uma abordagem aleatória de alocação, (ii) uma abordagem de agrupamento usando o *K-means*. Para isso, foi criada uma base de dados de coordenadas com 88 mil candidatos, 100 locais de prova (com 1200 vagas cada) dentro do território de Salvador-BA. Estes números são relativos ao ano de 2019. Os resultados mostraram que o O2MSM superou as demais abordagens, sendo mais eficiente e mais eficaz, reduzindo ao máximo possível o número de locais de prova, a suas vacâncias, e a distância para os candidatos.

## 2. Pergunta de Pesquisa

Esse trabalho propõe um novo algoritmo para automatizar o processo de alocação de candidatos em locais de prova, visando contribuir para a realização de exames de seleção presencial, tal como o ENEM. O método proposto visa realizar essa tarefa maximizando a taxa de ocupação de candidatos por local de prova, bem como minimizar o número de locais de prova envolvidos. O intuito é minimizar o custo de operação de realização do exame, evitando que locais de prova sejam subutilizados.

**PERGUNTA DE PESQUISA:** *Como aumentar a taxa de ocupação média de candidatos em locais de prova, minimizando o número total de locais de prova, e garantindo que todos os alunos sejam alocados?*

## 3. Trabalhos Relacionados

O problema de distribuição de candidatos em locais de prova (PDCL) é um problema relevante e é realidade em países que costumam fazer seleções através de avaliações presenciais, como o Brasil [Rocha et al. 2019]. Escolher uma distribuição adequada levando-se em conta diferentes critérios como: distância do candidato para o local de prova; minimização de locais de prova envolvidas para a redução de custo; e maximização da taxa de ocupação; é uma tarefa desafiadora. O PDCL pode ser encarado como uma instância de problemas combinatórios, tais como o *p-median* [Prima and Arymurthy 2019] e PAF [Fleiner et al. 2019]. Deste modo, alguns trabalhos, correlacionados ao PDCL, serviram de inspiração para a criação de soluções que automatizassem o processo de distribuição/alocação de candidatos em locais de prova.

Os trabalhos desenvolvidos em [Diebold and Bichler 2017, Biró 2017, Kawase and Iwasaki 2017, Ranjan and Singh 2018, Fleiner et al. 2019] adotaram algoritmos de casamento estável, variações do desenvolvido por Gale-Shapley [Gale and Shapley 2013], para o PAF. Neste caso, estas pesquisas contribuíram para a automatização da admissão de alunos em faculdades, minimizando a taxa de vacância. O trabalho desenvolvido por [Kawase and Iwasaki 2017] também usou algoritmos de casamento estável, só que para o problema de admissão de médicos em hospitais, que também é uma instância do PAF. Neste problema, o intuito é alocar médicos em hospitais, respeitando a capacidade financeira de cada hospital e os perfis médicos necessários.

Os trabalhos desenvolvidos por [Prima and Arymurthy 2018, Prima and Arymurthy 2019] usaram meta-heurísticas para apoiar o sistema de admissão de novos alunos da Indonésia. O objetivo de ambos era obter uma distribuição ideal dos alunos, de modo a minimizar o total de distâncias percorridas pelos estudantes. Esse problema foi modelado como problema de *p-median* e as meta-heurísticas usadas foram a baseada em vaga-lume, e algoritmo genético, respectivamente. Em [de Oliveira et al. 2013], um algoritmo genético também foi utilizado. No entanto, o objetivo foi otimizar a localização de seções eleitorais e a alocação de eleitores.

Todos os trabalhos citados acima estão correlacionados com o PDCL, e as soluções adotadas podem ser adaptadas e usadas para tentar encontrar soluções para o mesmo. No entanto, há uma carência de trabalhos que são aplicados diretamente ao PDCL. De acordo com o nosso conhecimento, apenas o trabalho desenvolvido por [Rocha et al. 2019] investigou o problema do PDCL no contexto brasileiro, que é o ENEM. Em [Rocha et al. 2019] o PDCL foi mapeado como o *p-median* e um algoritmo genético foi

adotado para tentar encontrar a distribuição ótima de alunos em locais de provas para o ENEM, levando em consideração o critério de distância exigido pelo Inep. Neste trabalho, foram considerados dados proporcionais à cidade de Santa Luzia – MG. Vale ressaltar que algoritmo proposto fez a alocação de candidatos, mas não tentou minimizar o número de locais de prova, visando uma economia maior para execução do exame.

O presente trabalho objetiva contribuir com a literatura de duas formas. A primeira, é a proposição de uma nova forma de modelar computacionalmente o problema PDCL. Diferentemente de [Rocha et al. 2019], aqui o PDCL é mapeado como um PAF. A segunda contribuição é a implementação de um novo algoritmo de otimização para o PDCL focado no ENEM. Serão considerados três objetivos durante a otimização: minimização da distância entre candidatos e locais de prova; minimização do número de locais de prova; e maximização da taxa de ocupação dos locais de prova. O algoritmo proposto será avaliado usando dados proporcionais à cidade de Salvador-BA.

## 4. Proposta

Nesta seção serão detalhados a modelagem computacional do PDCL no formato do PAF, e o novo algoritmo de otimização proposto para o PDCL no ENEM.

### 4.1. Modelagem

O PAF foi estudado inicialmente em [Gale and Shapley 2013], e é uma instância do problema combinatório da atribuição. O PAF pode ser definido como uma tupla de quatro elementos  $(L, C, q, r)$ , onde  $L = \{l_1, \dots, l_m\}$  é um conjunto de  $m$  faculdades,  $C = \{c_1, \dots, c_n\}$  é um conjunto de  $n$  estudantes,  $q = (q_{l_1}, \dots, q_{l_m})$  é um vetor com as capacidades (número de vagas) de cada faculdade, e  $r = (r_{l_1}, \dots, r_{l_m}, r_{c_1}, \dots, r_{c_n})$  é uma matriz de preferências. Esta matriz armazena as preferências dos alunos em relação às faculdades e vice-versa. Deste modo, o intuito é encontrar o casamento estável entre faculdades e alunos (definir que aluno vai para qual faculdade), levando em consideração as preferências de ambas as partes e a capacidade de cada faculdade.

Neste trabalho, modelamos o PDCL como sendo um PAF. Para isso,  $L$  passou a ser um conjunto de *locais de prova*,  $C$  tornou-se um conjunto de candidatos e,  $q$  passa a ser a capacidade que cada local de prova tem. No caso da matriz de preferências,  $r$ , esta passa a ser uma matriz de distâncias entre locais de provas e candidatos, e vice-versa. Uma vez definidos, os elementos da tupla  $(L, C, q, r)$  são passados como entrada para o algoritmo proposto encontrar o casamento estável entre candidatos e locais de prova. Os detalhes do algoritmo, sua implementação e os critérios de otimização são detalhados a seguir.

### 4.2. *Optimized One-to-Many Stable Marriage - O2MSM*

O algoritmo de Gale-Shapley [Gale and Shapley 2013] é um dos principais métodos de otimização para problemas de casamento estável, tais como o PAF [Fleiner et al. 2019]. Este método tenta exaustivamente encontrar a melhor combinação de pares aluno-local, tentando satisfazer as preferências de ambas as partes. Diferentes variações deste algoritmo foram criadas para atender requisitos de diferentes tipos de problemas correlacionados ao PAF [Diebold and Bichler 2017, Biró 2017, Kawase and Iwasaki 2017, Ranjan and Singh 2018, Fleiner et al. 2019].

Este trabalho propõe um novo algoritmo de casamento estável, do tipo um-para-muitos, para o PDCL. Este algoritmo é chamado de O2MSM, e tem como objetivo encontrar uma correspondência estável entre candidatos e locais de prova, minimizando a distância entre eles. Além disso, o algoritmo tenta minimizar o número de pontos de prova, e maximizar a taxa de ocupação destes pontos. O intuito é que a solução resultante encontrada pelo O2MSM atenda aos requisitos do ENEM e Inep, e ainda tenha um custo reduzido em termos de locais de prova.

Como se pode ver no Algoritmo 1, o O2MSM é um método iterativo que recebe como entrada  $(L, C, q, r)$ ,  $max\_it$  e  $max\_oc$ . O parâmetro  $max\_it$  é o número de iterações máximo que o algoritmo pode atingir, e o  $max\_oc$  diz respeito ao valor máximo que a taxa de ocupação média pode atingir. No processo iterativo, o primeiro passo é realizar a alocação dos candidatos contidos em  $C$  nos devidos locais de provais pertencentes a  $L$  levando em consideração as capacidades de cada local  $q$  e as preferências  $r$ .

---

**Algorithm 1** Pseudo-código do algoritmo O2MSM.

---

**Function** O2MSM( $L, C, q, r, max\_it, max\_oc$ )

```

for iteração = 0 até  $max\_it$  do
    ( $L, C, q, r$ ) = Alocar candidatos  $\in C$  em locais de prova  $\in L$  usando o Algoritmo
    2 Calcular a taxa de ocupação média dos locais em  $L$  pós-alocação
    Desabilitar de  $L$  locais de prova com taxa de ocupação abaixo (iteração*0.1)
    Atualizar  $L, C, q$  e  $r$ 
    if Taxa de ocupação média for maior que  $max\_oc$  then
        | Encerra o laço;
    end
end
Return:  $L, C$ 

```

---

Esta alocação é realizada através do algoritmo de Gale-Shapley (GS) adaptado para o PDCL, como mostra o Algoritmo 2. O algoritmo de GS tenta encontrar o local  $l$  de maior preferência, que esteja disponível, para o candidato  $c$ . Após a alocação,  $L, C, q$  e  $r$  são devidamente atualizados. O passo seguinte do algoritmo O2MSM é calcular a taxa de ocupação média dos locais em  $L$ , após a alocação do passo anterior. Com essa informação, o algoritmo desabilita todos aqueles locais de prova em  $L$  cuja taxa de ocupação é menor que a  $iterao \times 10\%$ . Este limiar varia com o passar das iterações para ajudar a controlar a remoção de locais de prova. Na primeira iteração ( $= 0$ ), nenhuma redução ocorre. Se a  $iterao = 1$  for considerada como exemplo, apenas aqueles locais de prova com taxa de ocupação abaixo de  $10\%$  são desabilitados. Esse limiar aumenta gradativamente com o passar das iterações ( $10\%$  por iteração), e o intuito é garantir que os locais de prova consigam atingir, em média, uma taxa  $max\_oc$  de suas capacidades de forma controlada. Após a desabilitação dos locais de prova,  $L, C, q$  e  $r$  são devidamente atualizados. Todos os passos do O2MSM são executados iterativamente até que  $max\_it$  iterações sejam alcançadas ou que a taxa de ocupação média seja maior que  $max\_oc$ . A final do algoritmo, as alocações resultantes envolvendo  $L$  e  $C$  são retornadas.

---

**Algorithm 2** Algoritmo Gale-Shapley para alocação de candidatos em locais de prova.

---

**Function**  $\text{Gale-Shapley}(L, C, q, r)$

**while**  $C$  não está vazia **do**

    Selecionar o próximo candidato  $c \in C$  Buscar em  $L$  o local de maior preferencia ( $l$ ) de  $c$  que esteja disponível

    Adicionar  $c$  na lista de candidatos ( $VC$ ) de  $l$  Atualizar  $c$  como já alocado Diminuir o numero de vagas de  $l$

**end**

**Return:**  $(L, C, q, r)$  atualizados com as devidas alocações

**end**

---

## 5. Metodologia Experimental

Neste trabalho, o método proposto foi analisado em relação aos seguintes aspectos: capacidade de minimizar a distância entre candidatos e locais de prova; capacidade de maximizar a taxa de ocupação nos locais de prova; e capacidade de reduzir o número de locais de prova. O O2MSM foi comparado com um algoritmo de alocação aleatória (RA), e uma versão iterativa do *K-means*. O RA e o *K-means* foi implementado para também otimizar os mesmos objetivos do método proposto. A intenção desta comparação é usar o RA e o *K-means* como *baselines* a serem superados.

### 5.1. Dados

Para os experimentos, a base de coordenadas, de latitude e longitude, para os locais de prova e candidatos utilizada foi criada aleatoriamente, usando a malha digital ou *shapefile* de Salvador-BA fornecido pelo IBGE [IBGE 2022], referente ao ano de 2015 (o mais recente). Com isso, os dados geoespaciais foram devidamente posicionados dentro do território do estado levando em consideração cidades e municípios. Ao todo foram criadas 88 mil coordenadas para candidatos, e 100 coordenadas para locais de prova. Neste trabalho, cada local teve sua capacidade associada à 1.200 vagas para a realização da prova. Vale salientar que a quantia de candidatos e locais de prova usada condizem com os números do ENEM de 2019 disponibilizados pelo Inep [INEP 2022]. Vale salientar que não foram considerados locais com acessibilidade, nem o interesse de local de prova do candidato.

Uma vez que a base de dados de coordenadas foi definida, a matriz de preferências (distâncias) foi criada calculando-se a distância entre cada candidato e local de prova através da fórmula de *Haversine* [Winarno et al. 2017]. A fórmula de *Haversine* é uma importante equação usada em navegação, fornecendo distâncias entre dois pontos de uma esfera a partir de suas latitudes e longitudes.

### 5.2. Configurações

Os algoritmos usados no experimento foram implementados na linguagem C++ usando a versão C++14, e as ferramentas disponíveis tanto na linguagem quanto na biblioteca *Armadillo: C++ library for linear algebra scientific computing* [Sanderson and Curtin 2016]. Tanto o O2MSM quanto a abordagem aleatória usaram os valores  $max\_it = 9$  e  $max\_oc = 70\%$ . Este valores foram definidos empiricamente, objetivando uma remoção gradual de locais de prova. A adoção de um limiar alto desde o início pode causar a

remoção de locais de prova importantes para a configuração final da distribuição, podendo causar até a não alocação de todos os candidatos. Por isso a decisão de um limiar variante e gradativo.

Todas as execuções foram realizadas em um laptop com Intel i5-7300HQ, memória cache de 4M, velocidade de clock de 2,5 GHz e turbo de até 3,5 GHz, com dois threads lógicos por núcleo físico e 8 GB de RAM. Como GPU, foi utilizada NVIDIA GeForce GTX 1050, além da GPU on-board Intel HD Graphics 630. O sistema operacional usado foi o Manjaro Linux com Kernel 5.4.34-1-MANJARO.

## 6. Resultados

A Figura 1 apresenta o desempenho dos algoritmos no processo de alocação de candidatos e redução do número de locais de prova com o passar das iterações. O intuito é mostrar a capacidade das abordagens em maximizar a taxa de ocupação nos locais de prova, reduzir o número de locais de prova, e ainda assim, garantir que, se não todos, a maioria dos candidatos estejam devidamente alocados. Na Figura 1, a barra cinza representa o número de candidatos alocados, associada ao eixo-y esquerdo; já a linha representa o número de locais de prova utilizados, associada ao eixo-y direito. Antes da execução dos algoritmos, o número de candidatos é igual a 88 mil, e o número de locais de prova igual a 100, cada um com 1200 vagas. Em outras palavras, há 1,36 vagas por candidato.

Como se pode ver na Figura 1-A, o O2MSM, na primeira iteração, alocou todos os candidatos em todas as 100 locais de prova, não havendo redução no número de vagas por candidato. No entanto, na quarta iteração, o O2MSM foi capaz de reduzir o número de locais para 89, e ainda assim garantindo todos os candidatos alocados. Este resultado reduz de 1,36 para 1,01 vagas por candidato, atingindo uma taxa de ocupação média de 98,86%. A partir da quarta iteração o número de locais de prova diminui ainda mais, porém, há candidatos desalocados.

A Figura 1-B mostra o desempenho da abordagem aleatória. Desde a primeira iteração, este algoritmo não é capaz de fazer as alocações de candidatos em locais de prova garantido o requisito da distância. Por este motivo, muitos candidatos ficaram de fora, mesmo havendo vagas suficientes. Ao final das iterações, ainda há uma redução do número de locais de prova, mas o número de candidatos desalocados permaneceu elevado. A Figura 1-C apresenta o desempenho do *K-means* iterativo. O resultado mostra que essa abordagem consegue reduzir o número de locais e manter um número elevado de candidatos alocados, nas primeiras quatro iterações. No entanto, desde a segunda iteração que já há presença de candidatos desalocados. Mesmo tendo utilizado mais iterações que o O2MSM, o *K-means* iterativo não foi capaz de melhorar o desempenho. Os resultados mostram a capacidade do O2MSM em balancear os objetivos de maximizar a taxa de ocupação e minimizar o número de locais de prova, superando os dois *baselines* apresentados.

Para complementar a análise do desempenho da abordagem proposta, a Figura 2 mostra um histograma que representa a distribuição de candidatos levando-se em consideração a distância para os locais de prova aos quais foram alocados. O número de candidatos está associado ao eixo-y e a distância para o local de prova ao eixo-x. O histograma preto representa a distribuição de candidatos na primeira iteração (iteração 0) do

O2MSM, e o histograma cinza claro representa o histograma na quarta iteração (iteração 3), quando há o menor número de locais de prova com alocação de 100% dos candidatos.

Como se pode ver, mesmo na quarta iteração, após ter uma redução de 11% no número de locais de prova (de 100 para 89), o histograma apresenta um comportamento similar ao da primeira iteração. Isso significa que mesmo com um número reduzido de locais, o O2MSM foi capaz de realizar uma realocação de candidatos sem aumentar muito a distância dos candidatos para os locais de prova, alcançando uma distribuição similar (99,1% de similaridade) a de quando havia mais flexibilidade na alocação (primeira iteração). Na Figura 2, a dissimilaridade entre os histogramas fica clara, ao se perceber uma pequena redução no número de candidatos alocados em locais entre 5 e 20km (*bins* 3, 4, 5 e 6 do histograma) em relação à primeira iteração (histograma preto). Essa redução, gerou um aumento nos *bins* 9 e 10 (os dois últimos) do histograma da quarta iteração (cinza). Isso significa que com a redução de locais, após uma nova alocação realizada pelo O2MSM, candidatos tiveram que ser alocados em locais mais distantes. Embora tenha havido esse comportamento, o mesmo representa apenas 0,9%. Os resultados apresentados respondem a *Pergunta de Pesquisa*, mostrando que o O2MSM é capaz de atender a todos os critérios para os quais foi projetado. Os resultados apresentados pelo O2MSM são promissores, e superaram as abordagens *baseline*. No entanto, ainda há alguns requisitos do FINEP/ENEM que precisam ser considerados, e novas análises a serem realizadas. Dentre o requisitos não considerados neste trabalho destacam-se: (i) Candidatos com de-

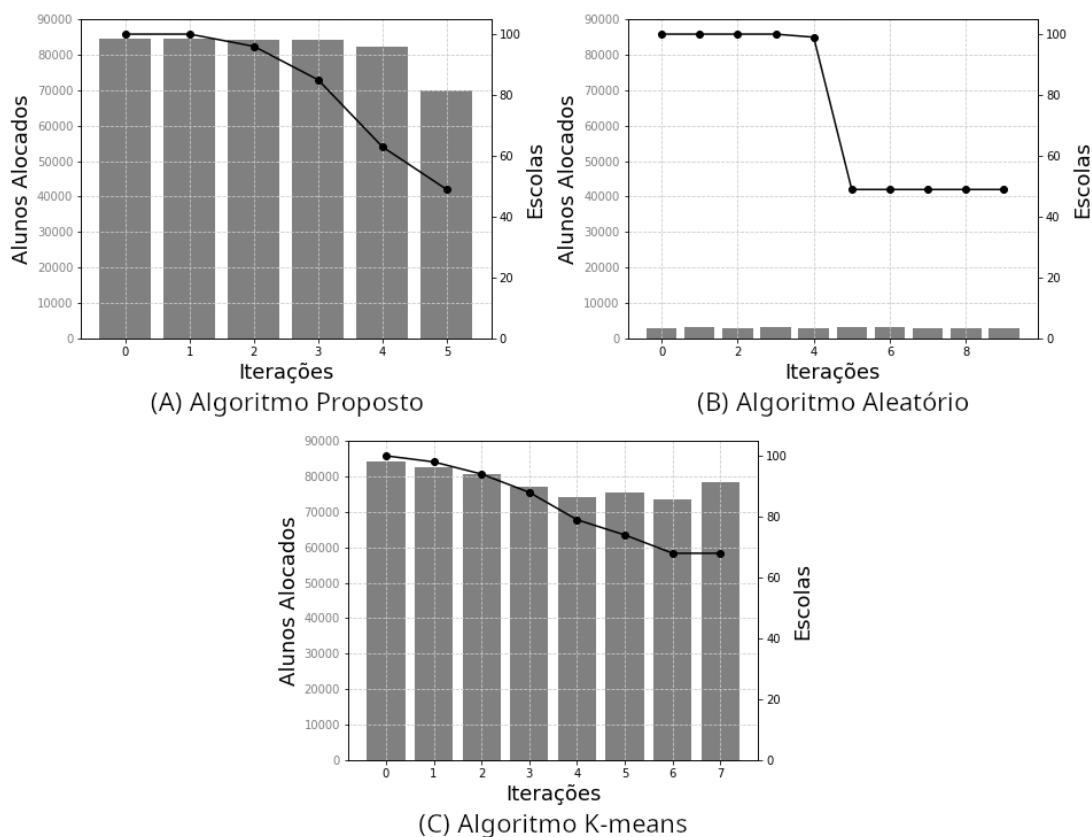
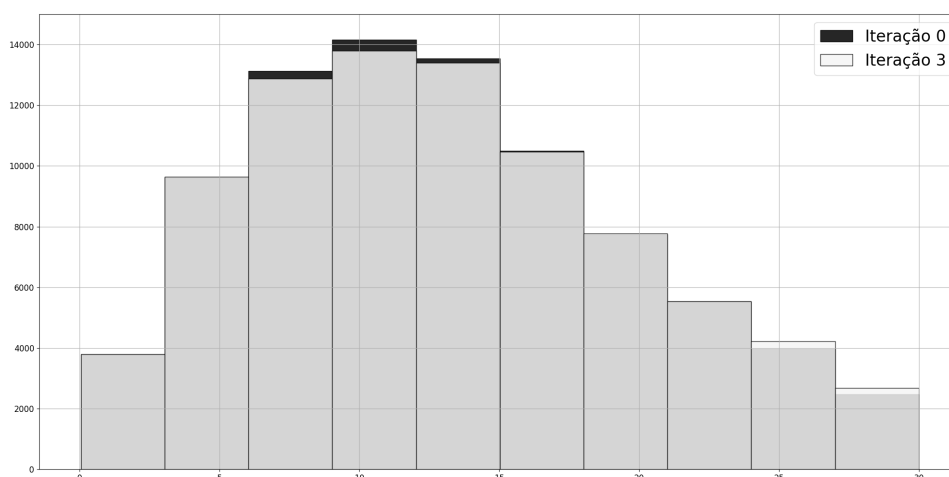


Figura 1. Desempenho dos algoritmos em termos de taxa de alocação de candidatos versus número de locais de prova.





**Figura 2. Histograma de candidatos em relação à distância para os respectivos locais de prova.**

ficiência que necessitam de locais com acessibilidade; (ii) Candidatos de um município não devem realizar prova em outro.

Com relação às análises a serem realizadas, destacam-se:

- Executar considerando múltiplas cidades e estados;
- Custo de execução do algoritmo.

## 7. Conclusão

A alocação de candidatos em locais de provas é um problema de logística relevante, e presente em diferentes locais do mundo que realizam seleções presenciais. Este desafio foi tratado aqui como uma instância do problema de admissão em faculdade. Neste trabalho, um novo método de casamento estável é proposto visando realizar a tarefa maximizando a taxa de ocupação de candidatos, e minimizando o número de locais de prova, considerando os requisitos do FINEP/ENEM. Os resultados mostraram que o método proposto possui melhores resultados que os *baselines*, em termos de redução do número de locais de prova, aumento da taxa de ocupação, e a garantia de cobertura de todos os candidatos envolvidos. Como trabalhos futuros, pretende-se implementar no O2MSM outros requisitos do FINEP/ENEM, tais como: candidatos que necessitam de locais com acessibilidade, e restrição de alocação de candidatos pelo município aos quais pertencem. Além disso, deseja-se realizar novas análises: quanto ao custo computacional do algoritmo proposto quando comparado aos *baselines*, bem como considerar dados de diferentes cidade e estados do país.

## Referências

- Biró, P. (2017). *Applications of matching models under preferences*. AI Access.
- de Oliveira, F. M., Aloise, D. J., de Lima Júnior, F. C., Aloise, D., and do Nascimento, H. A. D. (2013). Problema de localização de seções eleitorais e alocação de eleitores. In *Anais do Simpósio Brasileiro de Pesquisa Operacional*, pages 586–597.
- Diebold, F. and Bichler, M. (2017). Matching with indifferences: A comparison of algorithms in the context of course allocation. *European Journal of Operational Research*, 260(1):268–282.

- Fleiner, R., Ferkai, A., and Biró, P. (2019). College admission problem for university dual education. In 2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMI), pages 31–36. IEEE.
- Gale, D. and Shapley, L. S. (2013). College admissions and the stability of marriage. The American Mathematical Monthly, 120(5):386–391.
- IBGE 2022. Portal de mapas. <https://mapas.ibge.gov.br/bases-e-referenciais/bases-cartograficas/malhas-digitais>. Online; acessado em 01 de maio de 2022.
- INEP. Escolha e adaptação das salas para provas é processo complexo. <http://portal.mec.gov.br/component/content/article/418-noticias/enem-946573306/41251-entenda-o-complexo-e-demorado-processo-de-escolha-e-adaptacao-das-salas-para-provas>. Acessado: 08-05-2022.
- INEP 2022. Dados Abertos INEP. <http://inep.gov.br/dados>. Online; acessado em 01 de maio de 2022.
- Kawase, Y. and Iwasaki, A. (2017). Near-feasible stable matchings with budget constraints. arXiv preprint arXiv:1705.07643.
- Lima, P. d. S. N., Ambrósio, A. P. L., Ferreira, D. J., and Brancher, J. D. (2019). Análise de dados do enade e enem: uma revisão sistemática da literatura. Avaliação: Revista da Avaliação da Educação Superior, 24(1):89–107.
- Prima, P. and Arymurthy, A. M. (2018). Optimization of school location-allocation using genetic algorithm. In 2018 8th International Workshop on Computer Science and Engineering, WCSE 2018, pages 750–755. International Workshop on Computer Science and Engineering (WCSE).
- Prima, P. and Arymurthy, A. M. (2019). Optimization of school location-allocation using firefly algorithm. In Journal of Physics: Conference Series, volume 1235, page 012002. IOP Publishing.
- Ranjan, P. and Singh, S. (2018). A new algorithm for student-optimal matching: A framework for management institution’s admission in india. Indian Institute of Management Calcutta-Working Paper Series-No, 805.
- Rocha, E., Cunha Jr, J., Duarte, G., and Fernandes, G. (2019). Otimização para o problema de deslocamento dos candidatos a prova do enem. In Anais do Simpósio Brasileiro de Pesquisa Operacional.
- Sanderson, C. and Curtin, R. (2016). Armadillo: a template-based c++ library for linear algebra. Journal of Open Source Software, 1(2):26.
- Winarno, E., Hadikurniawati, W., and Rosso, R. N. (2017). Location based service for presence system using haversine method. In 2017 International Conference on Innovative and Creative Information Technology (ICITech), pages 1–4. IEEE.