

Artificial Immune Systems for Solving the Uncapacitated Single-Allocation p -Hub Median Problem

Gabriel L. e Nobrega¹, Vinicius J. Tasso², Allan G. Souza²,
Stephanie A. Fernandez², Daniel G. Silva²

¹ Instituto de Ciências Exatas – Universidade de Brasília (UnB)

² Faculdade de Tecnologia – Universidade de Brasília (UnB)

***Abstract.** Optimization problems such as the Uncapacitated Single-Allocation p -Hub Median Problem represent good models for real network design issues, hence an increasing research interest has emerged. A good hub location reduces costs and improves the quality of delivered services on network-based systems. In this work, two artificial immune systems are employed in order to address the problem, where the numerical results indicate good quality of solutions.*

1. Introdução

Os problemas de localização tem recebido muita atenção nos últimos anos devido à necessidade de se projetar redes de baixo custo e alta eficiência em diferentes áreas como computação, redes de telecomunicações, logística e transporte [Resende and Pardalos 2008]. É um tema de amplo interesse para diversas aplicações como o caso de localização de centros de abastecimento, subestações de redes de energia elétricas, hospitais [Rosário et al. 2002], escolas [Pizzolato et al. 2004], antenas [Lorena and Pereira 2002], entre outros.

Cada uma dessas áreas apresenta interpretações, aplicações e problemas diferentes, no entanto, o termo instalação (*hub* ou *facility*) é comum a todas elas e é geralmente utilizado para fazer referência a fábricas, empresas, antenas, depósitos, uma cidade, uma escola, entre outros. Neste sentido, um dos problemas de maior importância e clássico dentro da teoria da localização de *hubs* é o problema das p -medianas. Seu objetivo é determinar os locais de p facilidades (denominadas medianas) em uma rede de n nós de modo a minimizar a soma das distâncias entre cada nó de demanda e a mediana mais próxima.

Com relação a este problema, é importante destacar que, à medida que o seu tamanho aumenta, cresce também a complexidade da resolução devido à explosão combinatorial de soluções factíveis, tornando-se em certos casos inviável encontrar a respectiva solução ótima em um tempo considerado aceitável. Consequentemente, o problema das p -medianas é considerado de difícil solução e pertence à classe NP-difícil. Ademais, ele pode ser classificado conforme a capacidade de cada *hub* ou facilidade de atender os pontos de demanda próximos a ele. Neste trabalho, consideraremos o problema das p -medianas não capacitado (PPMNC), em que cada *hub* possui capacidade ilimitada para o atendimento dos pontos de demanda.

Ao longo dos anos, várias meta-heurísticas têm sido sugeridas para este problema. Algoritmos genéticos e variações [Topcuoglu et al. 2005, Pankratz 2005,

Naeem and Ombuki-Berman 2010], bem como métodos híbridos combinando algoritmos genéticos e busca tabu [Abdinnour-Helm 1998] e métodos de arrefecimento simulado (*simulated annealing*) e busca tabu [Chen 2007]. Para a versão capacitada do problema também tem sido propostas técnicas *branch-and-bound* [Ernst and Krishnamoorthy 1999], combinação das meta-heurísticas Busca Dispersa e Caminho de Reconexão (*path relinking*) [Díaz and Fernandez 2006], busca em agrupamentos (*clustering search*) [Chaves et al. 2007], algoritmo de busca local iterada [Fernandez et al. 2018] e otimização por colônia de formigas [Randall 2008].

Embora o PPMNC tenha recebido tal atenção, existem poucos trabalhos que empreguem sistemas imunológicos artificiais em problemas de localização em geral e, particularmente, nenhum conhecido pelos autores no caso do problema sob análise deste artigo. Dado que recentes trabalhos demonstram com sucesso a aplicação de SIAs na solução de problemas de otimização de natureza contínua [Dias et al. 2009, Wada et al. 2009, Silva et al. 2015] ou combinatória/discreta [Silva et al. 2011, Silva et al. 2014], acredita-se que problemas como o PPMNC, em que há grandes espaços de solução e/ou superfícies de otimização multimodais, possam ser abordados de maneira eficiente graças aos princípios imunológicos de descentralização e de manutenção de diversidade.

Neste sentido, este artigo propõe uma análise dos algoritmos imuno-inspirados CLONALG e optAiNet — aplicados recentemente com sucesso nos trabalhos previamente citados — como estratégias de otimização para se determinar a solução ótima do PPMNC. Para isto, o resto do trabalho se organiza nas seguintes seções: uma descrição formal do PPMNC é apresentada na Seção 2, assim como a metodologia de solução. Na Seção 3 são analisados um conjunto de resultados numéricos. Por último, as considerações finais deste trabalho são apresentadas na Seção 4.

2. Metodologia

2.1. Problema de otimização

O PPMNC é um problema de formulação matemática simples, mas de difícil solução computacional. A formulação de programação linear inteira mista proposta por M. O’Kelly [O’Kelly 1987] é usada neste artigo. Dado um conjunto $N = \{1, \dots, n\}$ de n nós e um determinado número de p medianas ou *hubs* em que cada nó ou ponto de demanda é considerado uma possível mediana, assim $p \leq n$, deve-se encontrar o conjunto de p medianas na determinada rede e alocar os nós restantes a eles, de forma que cada nó não-mediana seja alocado para uma única mediana, e o custo geral de transporte seja minimizado. Para todos os pares de nós i e j ($i, j \in N$), W_{ij} representa a quantidade de fluxo (dados, pacotes, entre outros) do nó origem i até o nó destino j que será roteado por uma ou duas medianas. A distância (custo) do nó de origem i ao nó de destino j é C_{ij} . Dado que as p medianas estão totalmente interconectadas, cada *link* entre um nó de origem e um nó de destino conterà pelo menos uma mediana e no máximo duas. Portanto, os parâmetros \mathcal{X} , τ e δ representam as taxas unitárias (custos) para a coleta (origem-para-mediana), transferência (mediana-para-mediana) e distribuição (mediana-para-destino) ao longo do caminho. Geralmente, τ é usado como um fator de desconto para fornecer custos unitários reduzidos nos arcos entre medianas de forma a refletir a economia de escala, por isso $\tau < \mathcal{X}$ e $\tau < \delta$.

A possível larga escala do problema e o fato de ser NP-difícil são pontos que chamam a atenção por serem características que se combinam de forma a dificultar qualquer tipo de solução computacionalmente pesada/exaustiva. Assim, meta-heurísticas inteligentes são necessárias para se atingir a solução ótima ou minimamente boa. Em especial para o PPMNC, as seguintes restrições devem ser obedecidas: (i) o número de medianas deve ser exatamente p ; (ii) um ponto de demanda ou nó deve estar alocado apenas a uma mediana. Portanto tem-se:

$$\text{minimize} \quad \sum_{i,j,k,l \in N} W_{ij} (\mathcal{X}C_{ik}H_{ik} + \tau C_{kl}H_{ik}H_{jl} + \delta C_{jl}H_{jl}) \quad (1a)$$

$$\text{sujeito a:} \quad \sum_{k=1}^n H_{kk} = p, \quad (1b)$$

$$\sum_{k=1}^n H_{ik} = 1, \quad \forall i = 1, \dots, n, \quad (1c)$$

$$H_{ik} \leq H_{kk} \quad \forall i, k = 1, \dots, n, \quad (1d)$$

$$H_{ik} \in \{0, 1\} \quad \forall i, k = 1, \dots, n, \quad (1e)$$

com $H_{ij} \in \{0, 1\}$ valendo 1 se o nó i for alocado numa mediana j e 0 caso contrário. A condição $H_{kk} = 1$ implica que o nó k é uma mediana.

2.2. Otimização via Sistemas Imunológicos Artificiais

Os Sistemas Imunológicos Artificiais (SIAs) são uma classe de algoritmos inspirados no sistema imunológico de mamíferos. Aplicados no contexto da engenharia e da computação, os princípios imunológicos de descentralização e de manutenção de diversidade são frequentemente úteis na resolução de problemas com grandes espaços de soluções, como é o caso do PPMNC.

No escopo biológico, os *linfócitos* ou simplesmente células de defesa são estruturas que reagem de maneira específica ao entrar em contato com alguma entidade forasteira — *patógeno*, em que as substâncias que o caracterizam se denominam *antígenos*—, enquanto o sistema imunológico é o resultado do trabalho em conjunto dos seus órgãos e células. Assim, um sistema imunológico é tão eficaz quanto as entidades que o compõem. Analogamente, o mesmo é válido para o custo energético do sistema, também proporcional à quantidade de células de defesa e de *anticorpos* que as mesmas produzem.

No escopo computacional, porém, *antígenos* se traduzem em algum problema que necessita solução (neste caso, o PPMNC) enquanto as células e seus *anticorpos* são análogos à uma solução-candidata para o problema em questão. Dessa forma, é possível simular imunologicamente um grande leque de problemas da vida real, buscando soluções com eficácia.

Dois meta-heurísticas imuno-inspiradas serão trabalhadas para o PPMNC, neste artigo: (i) o algoritmo de Seleção Clonal (CLONALG), que simula a resposta rápida de sistemas imunológicos quando apresentados a um agente patogênico e (ii) o algoritmo de otimização de Rede Imunológica (optAiNet) inspirado na Teoria da Rede Imunológica, que simula uma rede auto-regulatória de anticorpos em que cada membro tem capacidade para afetar o todo.

2.2.1. Algoritmo CLONALG

O CLONALG é um SIA proposto por [de Castro, L. N. and Von Zuben, F. J. 2002] que se baseia na Teoria da Seleção Clonal, ou seja, na dinâmica das células B e na capacidade de responderem à exposição de um antígeno por meio da secreção de anticorpos. Estas células estimuladas pela exposição antigênica passam por processos de clonagem e hipermutação a fim de se tornarem os mais eficientes possíveis contra tais antígenos, tendo também a capacidade de gerar memória para uma melhor resposta a invasões futuras.

O algoritmo de seleção clonal é uma meta-heurística populacional e sua abordagem é parecida com meta-heurísticas evolutivas. Entre suas principais características estão a manutenção de conjunto de memória (opcional), a seleção dos anticorpos de melhor afinidade, a maturação da afinidade (hipermutação) e a geração e manutenção de diversidade entre os anticorpos¹.

O Algoritmo 1 apresenta o pseudo-código do CLONALG. Ele é caracterizado por uma população de anticorpos Ab , cuja afinidade em relação ao antígeno é representada pela função objetivo. O algoritmo é inicializado com um conjunto de tamanho fixo n , em que cada anticorpo de Ab representa possivelmente uma solução ótima. Primeiro, a função de afinidade avalia todos os anticorpos, então, prossegue selecionando um subconjunto de Ab que possua as maiores afinidades, que são subsequentemente clonados. O conjunto de clones nc experimenta então um processo de maturação por afinidade, em que a intensidade das modificações é inversamente proporcional à afinidade (valor da função objetivo) de seus pais. Na sequência, os clones são comparados aos seus pais para selecionar aquele com a maior afinidade e conseqüentemente os piores anticorpos são eliminados tendo em conta o percentual de eliminação pe . O laço principal é concluído com uma geração aleatória de novos anticorpos que irá substituir aqueles previamente eliminados. O processo se repete até atingir o critério de parada. Depois disso, a solução é o melhor anticorpo de Ab , isto é, o de maior afinidade.

2.2.2. Algoritmo optAiNet

A teoria da rede imunológica (que serve de inspiração para a família *AiNet* de algoritmos) difere da teoria da seleção clonal ao propor anticorpos capazes de se identificar e de interagir entre si, estimulando e suprimindo outros indivíduos continuamente. Assim, o conjunto de anticorpos forma uma rede imunológica descentralizada e autônoma em que cada membro é capaz de influenciar o que acontecerá com a rede. Logo, além de serem capazes de reagir à antígenos, os anticorpos aqui também reagem a si mesmos [De França et al. 2010].

As relações anticorpo-anticorpo dependem apenas do desempenho individual e do grau de semelhança (afinidade) entre eles, ocorrendo da seguinte maneira: (i) sendo suficientemente semelhantes, o anticorpo de pior desempenho é eliminado; (ii) caso contrário, geram-se clones à mutar de forma inversamente proporcional ao seu desempenho, removendo os que não tiverem mutações benéficas i.e. aplica-se a teoria da seleção clonal. Por fim, de forma a manter a diversidade celular análoga à vista no sistema imune, é adicio-

¹Por simplicidade, os termos célula/anticorpo passarão a ser considerados equivalentes, dentro dos SIAs.

Algoritmo 1: Pseudo-código do algoritmo CLONALG.

Resultado: Melhor Anticorpo Ab
Dados: Tamanho de Ab , ic ;
Intensidade de mutação, β ;
Percentual de Eliminação, pAb ;
Número de clones, nc ;

- 1 Iniciar ic anticorpos em Ab ;
- 2 **Enquanto** critério de parada não atingido **faça**
- 3 | Contabilização e normalização da afinidade de cada Ab ;
- 4 | **Para** cada anticorpo Ab em n **faça**
- 5 | | Gerar nc clones;
- 6 | | **Para** cada clone gerado **faça**
- 7 | | | Mutar inversamente proporcional a afinidade e ajustado por β ;
- 8 | | **fim**
- 9 | | Selecionar melhor Ab dessa família;
- 10 | **fim**
- 11 | Retirada de $ic \cdot pAb$ piores Ab ;
- 12 | Gerar $ic \cdot pAb$ novos Ab ;
- 13 **fim**
- 14 **retorna** melhor anticorpo

nada uma quantidade percentual de anticorpos aleatórios ao conjunto. Com isso, o algoritmo incentiva os aspectos de exploração global do espaço enquanto evita convergências prematuras para ótimos locais.

Isto posto, a *optAiNet* é uma variação do algoritmo de rede imunológica com foco especial para otimização de funções, em que as principais características se encontram na forma em que novos anticorpos são introduzidos ao sistema e nos mecanismos de mutação e de avaliação de anticorpos. O Algoritmo 2 apresenta o seu pseudo-código.

Vale ressaltar que, como o PPMNC é um problema de natureza combinatória, não é adequado usar uma abordagem típica como a distância euclidiana para se medir o grau de similaridade entre soluções. Assim, o operador de similaridade utilizado deve ser personalizado de acordo, funcionando conforme se segue.

2.3. Operadores especiais

O desempenho na execução de um SIA depende da maneira escolhida para representar o problema de otimização no âmbito computacional. Assim, segue uma breve descrição dos operadores utilizados para o PPMNC.

2.3.1. Mutação

Responsável pela dinâmica de evolução dos anticorpos e construção de novas soluções, o algoritmo de (hiper)mutação assume um papel de importância na operação de busca. Levando em conta que um anticorpo irá representar uma matriz de alocação das medianas e dos nós i.e. a matriz H na Equação 1, o Algoritmo 3 apresenta o pseudo-código para

Algoritmo 2: optAiNet

Resultado: Melhor anticorpo em Ab
Dados: Contagem inicial de Ab , ic ;
Número de clones a serem gerados, nc ;
Intensidade da mutação, mv ;
Limiar de performance, pth ;
Limite de semelhança, sth ;
Porcentagem de Ab a serem adicionados, pAb ;

- 1 Inicializar ic anticorpos ;
- 2 **Enquanto** critério de parada não atingido **faça**
- 3 Contabilização e normalização da afinidade de cada Ab ;
- 4 Retirada de anticorpos cuja performance seja inferior que o limite pth ;
- 5 **Para** cada Ab no conjunto total de anticorpos **faça**
- 6 Gerar nc clones;
- 7 **Para** cada Ab gerado **faça**
- 8 Mutações inversamente proporcional à performance e proporcional à mv ;
- 9 **fim**
- 10 Seleção do melhor Ab nessa família;
- 11 **fim**
- 12 Remoção de anticorpos mais similares que o limite sth ;
- 13 Adição percentual pAb de anticorpos aleatórios;
- 14 **fim**
- 15 **retorna** melhor Ab ;

esse operador.

Vale frisar que a operação de mutação aqui desenhada obedece às restrições impostas pela formulação do problema, vistas na Equação 1, e que ela é usada tanto pelo método CLONALG como pela optAiNet.

2.3.2. Definição do grau de similaridade

Para cada *mediana* de uma dada solução-candidata (anticorpo), calcula-se o centro geométrico não ponderado entre suas conexões. Então, com todos os centros obtidos, soma-se a distância entre os centros mais próximos para ambas as soluções. Dessa maneira, espera-se que o operador retorne uma medida em que soluções semelhantes sejam, de fato, semelhantes em termos do problema i.e. medianas e nós alocados parecidos. O Algoritmo 4 apresenta o pseudo-código para esse operador, empregado no contexto da optAiNet.

3. Análise Experimental

Analisemos o desempenho da metodologia proposta. Os algoritmos CLONALG e optAiNet foram implementados usando Python e todos os resultados foram obtidos executando os algoritmos em um processador Intel i7 7700HQ, na frequência base de 2.8GHz. Não houve nenhum tratamento para paralelizar a execução, sem inclusão de GPU em qualquer parte do processo.

Algoritmo 3: Operador de Mutação

Resultado: Anticorpo mutado
Dados: Anticorpo, Ab ;

- 1 **Enquanto** critério de parada não atingido **faça**
- 2 Selecionar modo *mediana* ou *nó*;
- 3 **Se** *mediana* **faça**
- 4 Selecionar uma *mediana* de forma aleatória;
- 5 Desconectar todos os *nós* alocados a essa *mediana*;
- 6 A antiga *mediana* transforma-se em *nó*;
- 7 Selecionar um dos *nós* que estava ligado à antiga *mediana* para ser a nova *mediana*;
- 8 Conectar todos os *nós* não alocados a nenhuma *mediana* à nova *mediana*;
- 9 **fim**
- 10 **Se** *nó* **faça**
- 11 Selecionar um *nó* de forma aleatória;
- 12 Desconectar *nó* do antiga *mediana*;
- 13 Conectar *nó* à uma nova *mediana* selecionada aleatoriamente;
- 14 **fim**
- 15 **fim**

O problema-teste utilizado é o conjunto de instâncias Australian Post (AP) [Ernst and Krishnamoorthy 1998], baseado no sistema postal de Sidney, Austrália, e composto por 28 instâncias envolvendo até 200 nós que representam os distritos postais e suas respectivas coordenadas, assim como o volume de fluxo e o número p de medianas variando de 2 a 20. Assim, o problema do PPMNC se traduz na escolha dos melhores distritos (*nós*) como centros de distribuição e das melhores conexões (*medianas*) como vias de transporte, visando minimizar o custo total do conjunto de envios. Simulando de forma completa o PPMNC, a instância AP fornece as seguintes informações: (i) localização de cada nó; (ii) fluxo entre todos os nós; (iii) custo de coleta; (iv) custo de transferência e (v) custo de distribuição.

Antes, entender os efeitos de cada um dos parâmetros de execução é vital para a compreensão dos algoritmos, bem como para sua reprodução e/ou aperfeiçoamento. Assim, são apresentadas as consequências na mudança das configurações de cada parâmetro, com a ressalva que os dois últimos itens são exclusivos do algoritmo optAiNet.

- Contagem inicial de anticorpos, ic : abrangência primária do domínio de soluções; se excessivo, pode gerar sobrecarga de anticorpos redundantes, que serão eliminados pelo limiar de semelhança. O valor utilizado foi 20.
- Percentual de reinicialização, pAb : percentual de novos anticorpos aleatórios a serem adicionados ao conjunto de soluções; adição excessiva de anticorpos aumenta o custo computacional sem trazer benefícios equivalentes; por outro lado, adicionar poucos anticorpos prejudica a busca pelo ótimo global. O valor utilizado foi 0,1.
- Número de clones a serem mutados, nc : número de clones a serem gerados para cada anticorpo no grupo de soluções; muitos clones podem ser redundantes, enquanto poucos clones podem ser inefetivos na busca por uma mutação favorável. O valor utilizado foi 4.

Algoritmo 4: Operador de similaridade entre anticorpos

Resultado: Distância total, dt
Dados:
Anticorpo 1, $Ab1$;
Anticorpo 2, $Ab2$

- 1 Inicializar dt como 0;
- 2 **Para** cada mediana em $Ab1$ **faça**
- 3 | Calcular centro geométrico entre conexões da *mediana*;
- 4 **fim**
- 5 **Para** cada mediana em $Ab2$ **faça**
- 6 | Calcular centro geométrico entre conexões da *mediana*;
- 7 **fim**
- 8 **Para** cada centro geométrico de $Ab1$ **faça**
- 9 | Calcular centro geométrico de $Ab2$ mais próximo;
- 10 | Somar distância entre centros à dt ;
- 11 **fim**
- 12 **retorna** Distância total, dt ;

- Condição de parada: ocorre após o decorrimto de 200 iterações.
- Limiar de semelhança, sth : determina o limite máximo de semelhança para que dois anticorpos possam coexistir no conjunto de soluções sem que sejam eliminados; demasiadamente pequeno perde-se poder computacional em soluções no mesmo ótimo local; excessivamente grande e ótimos locais serão ignorados. O valor utilizado foi 5.
- Limiar de performance, pth : limite mínimo para a performance relativa de um anticorpo antes que este seja eliminado; valores altos diminuem a capacidade de reinicialização do algoritmo, custando possíveis soluções; valores baixos permitem a existência de soluções inefetivas. O valor utilizado foi 0,05.

Naturalmente, busca-se uma análise acurada e completa dos resultados. Dessa forma, montou-se um *testbench* que roda ambos os algoritmos um total de dez vezes, coletando informações relativas à performance do algoritmo. Para avaliar o desempenho obtido, define-se *gap* como a diferença percentual entre a solução ótima, previamente conhecida, e a encontrada pelo algoritmo em questão. Além disso, informações relativas ao tempo de execução serão utilizadas para comparar a eficiência dos métodos apresentados anteriormente.

Com isso em mente, a análise se inicia pela Figura 1, que mostra a curva de custo para a melhor solução em uma simulação teste com 5 medianas, 50 nós e 100 iterações até a condição de parada. Nota-se, como esperado, a presença de *steps* na curva de custo. Tal fenômeno ocorre quando a busca é capaz de localizar um novo ponto de ótimo, seja por mutação ou por reinicialização. Ainda, percebe-se que a largura dos *steps* aumenta com a época atual de execução, significando maior tempo de busca entre o encontro de novos ótimos e maior dificuldade em encontrá-los. É uma consequência do processo de exploração do espaço de busca pelas populações dos dois métodos.

Falhando em abranger maior espaço de soluções para otimizar o problema, o CLONALG encontrou adversidades para sair de ótimos locais. Tais dificuldades

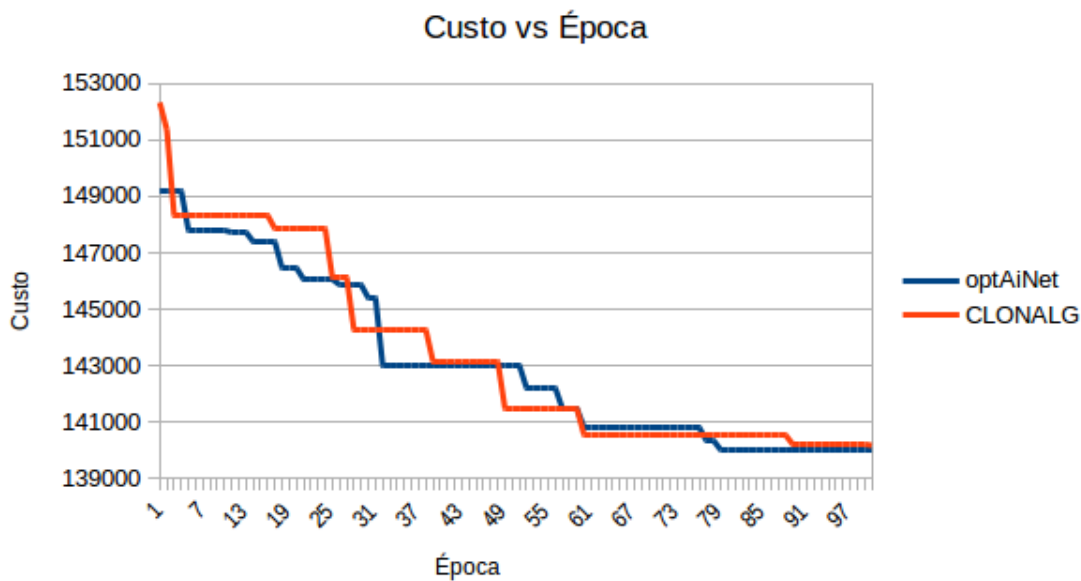


Figura 1. Custo da melhor solução em relação à época, algoritmo optAiNet.

eram esperadas [de Moraes et al. 2011], visto que o algoritmo não possui capacidade de controle populacional como a optAiNet e depende exclusivamente dos processos de clonagem/seleção/mutação para escapar de regiões de busca já exploradas suficientemente. Por outro lado, a quantidade de recursos computacionais para a execução da técnica foi inferior.

Com maior capacidade de busca e exploração da diversidade de soluções, a optAiNet atingiu resultados superiores. Posto que ela demanda, por natureza, maior gasto computacional, verificou-se um maior tempo de execução em relação ao CLONALG. Deste modo, apresenta-se na Tabela 1 as performances obtidas após a execução do *testbench*. A primeira coluna apresenta o algoritmo testado. As segunda e terceira colunas indicam o número de distritos (nós) e medianas (p) de cada instância e as três últimas colunas apresentam a diferença percentual em relação à solução ótima conhecida. Particularmente o *GAP Médio* mostra a média dos *gaps* encontrados para cada instância usando os algoritmos optAiNet e CLONALG. O *GAP Mínimo* mostra o *gap* entre a melhor solução encontrada pelos algoritmos propostos e a solução ótima e o *GAP Máximo* entre a pior solução encontrada e a solução ótima.

Ademais, os tempos médios de execução de cada algoritmo são apresentados na Tabela 2. Observando a taxa de crescimento do tempo de execução e a eficácia da técnica, nota-se a qualidade de otimização da optAiNet, a despeito do seu maior custo computacional. Ainda, fica claro que o CLONALG, embora menos efetivo, possui tempos de execução significativamente inferiores. Por fim, embora o mesmo se mostre pouco sensível ao aumento de 3 medianas para 5, no caso envolvendo 50 distritos, isto se dá porque o algoritmo possui um parâmetro de limite para o tamanho da população. Tal situação pode ser contornada ajustando tal configuração e não caracteriza um limitante teórico do algoritmo.

Tabela 1. Resultados obtidos

	Distritos	Medianas	GAP Médio (%)	GAP Mínimo (%)	GAP Máximo (%)
optAiNet	10	2	0	0	0
optAiNet	20	3	0	0	0
optAiNet	50	3	0	0	0
optAiNet	50	5	1.05	0.22	3.51
CLONALG	10	2	0	0	0
CLONALG	20	3	0.03	0	0.34
CLONALG	50	3	0.68	0	1.71
CLONALG	50	5	3.92	2.62	5.99

Tabela 2. Tempos de Execução

	Distritos	Medianas	Tempo de Execução (s)
optAiNet	10	2	4.20
optAiNet	20	3	9.90
optAiNet	50	3	61.47
optAiNet	50	5	210.97
CLONALG	10	2	1.16
CLONALG	20	3	3.48
CLONALG	50	3	18.13
CLONALG	50	5	18.44

4. Considerações Finais

Este trabalho apresenta, de forma inédita, uma formulação via SIAs para o problema das p -medianas não capacitado, um problema reconhecido de localização de *hubs* que pode ser abordado eficientemente via estratégias de busca aproximada, dada a sua natureza NP-difícil.

Destacando-se por ser capaz de buscar múltiplos ótimos simultaneamente, o algoritmo optAiNet mostrou-se capaz de solucionar 3 das 4 instâncias analisadas do PPMNC. Por sua vez, o CLONALG apresentou uma capacidade singular para convergir com velocidade, no entanto encontrando soluções de pior qualidade, para as mesmas instâncias. De maneira geral, ambas as técnicas se mostraram viáveis na resolução das instâncias menores do PPMNC, estando aberta a perspectiva de se aprofundar as análises experimentais para as instâncias maiores do problema.

Adicionalmente, pensar estratégias de diminuição do tempo de execução para a optAiNet e implementar reinicializações efetivas sem aumento significativo no custo de execução do CLONALG são questões a serem discutidas em trabalhos futuros, podendo trazer ainda mais crescimento para as meta-heurísticas discutidas e a aplicação em si. Por fim, os resultados obtidos ainda necessitam ser comparados com aqueles obtidos a partir das outras abordagens existentes na literatura.

Referências

Abdinnour-Helm, S. (1998). A hybrid heuristic for the uncapacitated hub location problem. *European Journal of Operational Research*, 106(2-3):489–499.

- Chaves, A. A., de Assis Correa, F., and Lorena, L. A. N. (2007). Clustering search heuristic for the capacitated p-median problem. In *Innovations in Hybrid Intelligent Systems*, pages 136–143. Springer.
- Chen, J.-F. (2007). A hybrid heuristic for the uncapacitated single allocation hub location problem. *Omega*, 35(2):211–220.
- de Castro, L. N. and Von Zuben, F. J. (2002). Learning and Optimization Using the Clonal Selection Principle. *IEEE Transactions on Evolutionary Computation*, 6(3):239–251.
- De França, F. O., Coelho, G. P., Castro, P. A. D., and Von Zuben, F. J. (2010). Conceptual and Practical Aspects of the aiNet Family of Algorithms. *International Journal of Natural Computing Research*, 1(1):1–35.
- de Moraes, V. W. C., Maravilha, A. L., de Assis, L. P., and Andrade, A. V. (2011). Abordagem Imunológica ao Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas. In *XLIII Simpósio Brasileiro de Pesquisa Operacional (SBPO)*, pages 1422–1433, Ubatuba-SP.
- Dias, T. M., Attux, R., Romano, J. M. T., and Suyama, R. (2009). Blind Source Separation of Post-Nonlinear Mixtures Using Evolutionary Computation and Gaussianization. *ICA 2009, LNCS 5441*, 5441:235–242.
- Díaz, J. A. and Fernandez, E. (2006). Hybrid scatter search and path relinking for the capacitated p-median problem. *European Journal of Operational Research*, 169(2):570–585.
- Ernst, A. T. and Krishnamoorthy, M. (1998). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104(1):100–112.
- Ernst, A. T. and Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86:141–159.
- Fernandez, S. A., Perez, A. A., de Armas, J., Silva, D. G., and Ferone, D. (2018). A two-stage biased-randomized iterated local search for the uncapacitated single allocation p-hub median problem. *Transactions on Emerging Telecommunications Technologies*.
- Lorena, L. A. and Pereira, M. A. (2002). A lagrangean/surrogate heuristic for the maximal covering location problem using hillman’s edition. *International Journal of Industrial Engineering*, 9:57–67.
- Naeem, M. and Ombuki-Berman, B. (2010). An efficient genetic algorithm for the uncapacitated single allocation hub location problem. In *Evolutionary Computation (CEC), 2010 IEEE Congress on*, pages 1–8. IEEE.
- O’Kelly, M. E. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32(3):393–404.
- Pankratz, G. (2005). Dynamic vehicle routing by means of a genetic algorithm. *International Journal of Physical Distribution & Logistics Management*, 35(5):362–383.
- Pizzolato, N. D., Barcelos, F. B., and Nogueira Lorena, L. A. (2004). School location methodology in urban areas of developing countries. *International Transactions in Operational Research*, 11(6):667–681.

- Randall, M. (2008). Solution approaches for the capacitated single allocation hub location problem using ant colony optimisation. *Computational Optimization and Applications*, 39(2):239–261.
- Resende, M. G. C. and Pardalos, P. M. (2008). *Handbook of optimization in telecommunications*. Springer Science & Business Media.
- Rosário, R., Carnieri, C., and Steiner, M. T. A. (2002). Proposta de solução para o problema das p-medianas na localização de unidades de saúde 24 horas. *XXII ENEGEP*.
- Silva, D. G., Attux, R., Nadalin, E. Z., Duarte, L. T., and Suyama, R. (2011). An Immune-Inspired Information-Theoretic Approach to the Problem of ICA Over a Galois Field. In *2011 IEEE Information Theory Workshop*, pages 618–622. IEEE.
- Silva, D. G., Montalvão, J., Attux, R., and Coradine, L. C. (2015). An immune-inspired, information-theoretic framework for blind inversion of Wiener systems. *Signal Processing*, 113:18–31.
- Silva, D. G., Nadalin, E. Z., Coelho, G. P., Duarte, L. T., Suyama, R., Attux, R., Von Zuben, F. J., and Montalvão, J. (2014). A Michigan-like immune-inspired framework for performing independent component analysis over Galois fields of prime order. *Signal Processing*, 96:153–163.
- Topcuoglu, H., Corut, F., Ermis, M., and Yilmaz, G. (2005). Solving the uncapacitated hub location problem using genetic algorithms. *Computers & Operations Research*, 32(4):967–984.
- Wada, C., Consolaro, D. M., Ferrari, R., Suyama, R., Attux, R., and Von Zuben, F. J. (2009). Nonlinear Blind Source Deconvolution Using Recurrent Prediction-Error Filters and an Artificial Immune System. *ICA 2009, LNCS 5441*, 5441:371–378.