

Meta-heurística ILS para o problema de posicionamento automático de pontos de acesso em ambientes internos

Anderson Rufino², Thiago Gouveia da Silva^{1,3}, Eduardo Vieira Queiroga²,
 Nailson dos Santos Cunha², Lucidio dos Anjos F. Cabral²

¹Instituto Federal de Educação, Ciência e Tecnologia da Paraíba(IFPB)

²Centro de Informática – Universidade Federal da Paraíba(UFPB)
 João Pessoa – PB – Brasil

³Instituto de Computação – Universidade Federal Fluminense(UFF)
 Niterói – RJ – Brasil

thiago.gouveia@ifpb.edu.br, eduardo.queiroga@ci.ufpb.br
 nailsoncunha@gmail.com, lucidio@ci.ufpb.br, andersonrufino2007@gmail.com

RESUMO

As redes Wi-Fi têm se tornado cada vez mais presentes em ambientes internos. O projeto de redes desse tipo, contudo, requer que o posicionamento dos pontos de acesso atenda a diversas restrições relacionadas ao custo do projeto, à área de cobertura, à qualidade do sinal, entre outros critérios. Em virtude da dificuldade do problema quando tratado de forma manual, surgem os métodos que realizam o posicionamento automaticamente. Este trabalho propõe um método baseado na meta-heurística *Iterated Local Search* (ILS) para o Problema de Posicionamento de Pontos de Acesso (APPP), considerando restrições de capacidade, cobertura interna e externa, e demanda por qualidade. Resultados promissores foram obtidos em experimentos com instâncias sintéticas e reais.

PALAVRAS-CHAVE. Projeto de redes. Wireless. ILS-RVND.

Área Principal: MH - Meta-heurísticas

ABSTRACT

The Wi-Fi networks have been becoming even more present in indoor environments. The Wi-Fi network design, however, requires that the Wi-Fi access point placement respect several constraints related to the project cost, the covering area, the signal quality and the energy expenditure, above others. Due to the hardness of handmade designing a Wi-Fi network, the automatic access point placement methods appear. This paper proposes a method based on the metaheuristic *Iterated Local Search* (ILS) to address the Access Point Positioning Problem, considering capacity constraints, internal and external coverage and demand for quality. Promising results were obtained in experiments with synthetic and real instances.

KEYWORDS. Network Design. Wireless. ILS-RVND.

Main Area: MH - Metaheuristics

1. Introdução

As redes locais sem fio (WLANs, do inglês *Wireless Local Area Networks*) têm se tornado cada vez mais importantes para a conectividade de ambientes internos. O crescimento da utilização das redes Wi-Fi (redes WLAN baseadas nos padrões IEEE 802.11) pode ser creditado, principalmente, à diminuição do custo dos equipamentos, ao aumento da confiabilidade e à praticidade da implantação destas redes.

O projeto de redes Wi-Fi de médio e grande porte é uma tarefa complexa que requer, normalmente, que o posicionamento dos roteadores sem fio ou pontos de acesso (APs, do inglês *Access Points*) siga as diversas restrições relacionadas ao custo do projeto, à área de cobertura, à qualidade do sinal, à minimização da interferência e/ou ao consumo energético, etc. Neste contexto, surgem problemas de otimização combinatória da classe NP-Difícil que podem ser modelados matematicamente e/ou tratados por abordagens exatas e heurísticas.

Neste trabalho, tratamos do Problema de Posicionamento de Pontos de Acesso WLAN (APPP), considerando diferentes tipos de APs, a demanda por qualidade de serviço nos pontos de usuário, a demanda por banda passante e a proibição de cobertura em áreas externas. O APPP é modelado matematicamente como um problema de programação linear inteira baseado no problema de localização de facilidades, provado ser NP-Difícil por Megiddo e Tamir (1982), minimizando os custos dos APs e penalizando a cobertura de pontos proibidos (área externa). Uma meta-heurística baseada no ILS (do inglês, *Iterated Local Search*) é proposta para resolver instâncias sintéticas e reais de diferentes dimensões. Para avaliar o método proposto, experimentos computacionais envolvendo adaptações de meta-heurísticas da literatura foram realizados.

O trabalho está organizado como segue. Na Seção 2, uma revisão da literatura com trabalhos relacionados ao projeto de redes WLAN em ambientes internos é apresentado. Na Seção 3, o problema é descrito formalmente através de um modelo de programação linear inteira. Na Seção 4, a meta-heurística proposta é apresentada em detalhes. Na Seção 5, o processo de geração de instâncias sintéticas é descrito e justificado. Na Seção 6, os experimentos computacionais são descritos e os resultados analisados.

2. Trabalhos relacionados

A literatura apresenta diversos trabalhos que tratam sobre a resolução do APPP. Contudo, a definição do problema possui alta variabilidade, devido à adequação dos métodos para estudos de casos de projetos reais, dificultando a estruturação das pesquisas na área. Apesar de existirem muitos trabalhos propondo métodos exatos para o APPP, por restrições de espaço, o foco desta seção será apenas sobre métodos heurísticos.

Kamenetsky e Unbehaun (2002) propuseram uma combinação de duas abordagens utilizando um algoritmo de remoções sucessivas chamado *pruning* para obter uma solução inicial, além de refiná-la utilizando a meta-heurística *Simulated Annealing* (SA). O método é avaliado através de uma função objetivo sobre um espaço discreto, com o intuito de melhorar a área de cobertura e a qualidade do sinal recebido.

Arroyo e Marques (2006) propuseram dois métodos baseados na meta-heurística GRASP para resolver o problema de alocação de antenas de transmissão em uma região. A função objetivo do modelo busca maximizar a cobertura do sinal utilizando um número

mínimo de antenas e instalando-as o mais próximo possível dos pontos de demanda, levando em consideração restrições como o alcance do sinal de transmissão e a presença de obstáculos.

O mesmo autor propôs em Marques (2007) outra análise comparativa para o mesmo modelo, desta vez, utilizando algoritmos baseados nas meta-heurísticas GRASP e Algoritmos Genéticos (AG). A função objetivo neste trabalho é definida como a soma dos custos das antenas utilizadas e a soma das distâncias mínimas entre cada ponto de demanda e a antena instalada.

Lu *et al.* (2006) propuseram uma abordagem baseada na meta-heurística Busca Tabu (TS, do inglês *Tabu Search*) para a definição da quantidade de APs e o seu posicionamento com respeito a um critério que considera a cobertura, a interferência e a qualidade de serviço (QoS, do inglês *Quality of Service*). Uma cadeia de Markov é utilizada para avaliar a largura de banda em cada célula da área e a cobertura é estimada por um simulador de propagação de rádio multiresolução.

Vanhatupa *et al.* (2007) apresentam um novo algoritmo baseado em AG para a criação de redes WLAN de acordo com os requisitos desejados e faz uma estimativa da qualidade de serviço (cobertura, custos de implantação e capacidade da rede), provendo um *feedback* para o próprio algoritmo e também para o construtor da rede. O método leva em consideração requisitos como consumo de energia, possíveis locais para posicionamento de APs e quantidade de APs permitidos.

Em Barbosa e Gouvea (2012), uma heurística gulosa e um abordagem *steady-state* AG foram propostos para definir o posicionamento de APs, maximizando a cobertura da área e o número de usuários atendido.

Farkas *et al.* (2013) propuseram um método baseado em SA com o objetivo de encontrar o posicionamento e o número ideal de pontos de acesso Wi-Fi de tal forma que cada ponto na área seja coberto por pelo menos 3 APs.

Por fim, Capdeville e Vianna (2013) propõem duas implementações diferentes do GRASP para resolver o APPP em um ambiente interno de uma instituição federal de ensino. O modelo é baseado no problema de localização de facilidades, levando em consideração a atenuação do sinal por obstáculos. Os autores também propuseram duas heurísticas de busca local para complementar a meta-heurística.

3. Formulação matemática proposta

Nesta seção, a versão do APPP tratada é formalizada através de um modelo de programação linear inteira. Para discretizar o problema, uma grade de pontos candidatos sobre a área é considerada. Dentre esses pontos, P é o conjunto de todos os pontos que precisam ser cobertos, R o conjunto dos pontos onde é possível alocar um AP, T o conjunto dos diferentes tipos de APs e S o conjunto dos pontos proibidos.

Temos que p_k é o custo de utilizar o roteador do tipo $k \in T$; q_s a penalidade pelo ponto proibido s estar na área de cobertura de pelo menos um roteador ativo; c_k a capacidade do roteador do tipo k e d_i a demanda por largura de banda do ponto i . A variável de decisão x_{ijk} indica se o ponto i está ligado ao roteador j do tipo k . A variável de decisão r_{jk} assume o valor 1 sse o roteador j do tipo k está sendo utilizado na solução. A variável de decisão s_h assume o valor 1 sse o ponto proibido h está na área de

cobertura de algum roteador utilizado. Seja $cob(r_{jk})$ uma função que retorna o conjunto de todos os pontos que estão na área de cobertura do roteador r_{jk} , e $cobp(r_{jk})$ uma função análoga para os pontos proibidos. A formulação matemática proposta é apresentada nas expressões (1 - 6):

$$\min \sum_{j \in R} \sum_{k \in T} p_k r_{jk} + \sum_{h \in S} q_h s_h \quad (1)$$

$$\text{s.a.} \quad \sum_{j \in R} \sum_{k \in T} x_{ijk} = 1, \quad \forall i \in P, \quad (2)$$

$$r_{jk} \geq x_{ijk}, \quad \forall i \in cob(r_{jk}), \forall j \in R, \forall k \in T, \quad (3)$$

$$\sum_{i \in P} d_i x_{ijk} \leq c_k, \quad \forall j \in R, \forall k \in T, \quad (4)$$

$$s_h \geq r_{jk}, \quad \forall h \in cobp(r_{jk}), \forall j \in R, \forall k \in T, \quad (5)$$

$$r_{ik}, x_{ijk}, s_h \in \{0, 1\}, \quad \forall i \in P, \forall j \in R, \forall k \in T, \forall h \in S. \quad (6)$$

A função objetivo (1) busca minimizar o custo total da utilização dos roteadores, juntamente com a minimização da penalização de cobertura de pontos proibidos. As restrições (2) estabelecem que todos os pontos clientes estejam associados a um roteador. As restrições (3) indicam que cada ponto só pode ser ligado a um roteador ativo. As restrições (4) definem que a demanda por largura de banda dos pontos conectados ao roteador k não pode ultrapassar a capacidade do mesmo. A restrição (5) indica se o ponto proibido s_h está dentro da área de cobertura de algum roteador. Por fim, as restrições (6) definem o domínio das variáveis de decisão.

4. Método ILS-RVND proposto

Para a resolução do APPP como definido na Seção 3, propomos um algoritmo baseado na meta-heurística *Iterated Local Search* (ILS) em conjunto com uma busca local *Randomized Variable Neighborhood Descent* (RVND). Uma característica importante do método proposto é que ele permite a exploração do espaço de soluções inviáveis do problema. A busca local, por sua vez, tem a responsabilidade de retornar apenas soluções viáveis. Os procedimentos de geração da solução inicial, busca local e perturbação são descritos nas subseções subsequentes.

Segundo Lourenço *et al.* (2003), a meta-heurística ILS é composta por quatro procedimentos básicos: **GeraSoluçãoInicial()**, que constrói a primeira solução do problema; **BuscaLocal()**, que melhora a solução de entrada fazendo-a convergir para um ótimo local; **CritérioDeAceitação()**, que determina se a solução corrente será aceita ou não como a nova solução e **Perturbação()**, que modifica a solução para a próxima iteração do algoritmo. O ILS-RVND proposto é detalhado no Algoritmo 1.

Temos a geração de uma solução inicial na linha 1, sendo S a solução atual, S^* a melhor solução da iteração e S^{**} a melhor solução encontrada pelo método. Na linha 2, o procedimento de iteração do método inicia-se, o qual possui um tempo limite de duração delimitado pela comparação das variáveis *tempoExecução* e *tempoLimite*. Na linha 3, são inicializadas as variáveis k , responsável pela intensidade das perturbações e

Algoritmo 1: ILS-RVND()

```

1  $S^{**} \leftarrow S^* \leftarrow S \leftarrow \text{GeraSoluçãoInicial}();$ 
2 enquanto  $\text{tempoExecução} \leq \text{tempoLimite}$  faça
3    $k \leftarrow 1; \text{iterSemMelhora} \leftarrow 0;$ 
4   enquanto  $\text{iterSemMelhora}++ < \text{maxIterSemMelhora}$  faça
5      $S \leftarrow \text{RVND}(S);$ 
6     se  $fo(S) < fo(S^*)$  então
7        $S^* \leftarrow S; k \leftarrow 1; \text{iterSemMelhora} \leftarrow 0;$ 
8        $S \leftarrow S^*;$ 
9       para cada  $i = 1, 2 \dots \text{Min}(k, \text{max}K)$  faça
10         $S \leftarrow \text{Perturbação}(S);$ 
11    $S^{**} \leftarrow \text{Best}(S^{**}, S^*);$ 
12    $S \leftarrow S^* \leftarrow \text{GeraSoluçãoInicial}();$ 
13 retorne  $S^{**}$ 

```

iterSemMelhora, que monitora a quantidade de iterações sem melhora da solução. Entre as linhas 4 e 10 temos o principal laço do algoritmo, o qual possui os procedimentos de busca local e perturbação. Este laço encerra-se quando o número de iterações sem melhora da solução atingir um valor máximo, este último representado pela constante *maxIterSemMelhora*. Neste trabalho utilizamos *maxIterSemMelhora*=1000.

Na linha 5, temos a obtenção de um ótimo local após a aplicação do método de busca local RVND sobre a solução de entrada. Na linha 6 é verificado se o novo ótimo local será aceito através do valor da função objetivo (novamente, na primeira iteração sempre haverá uma melhora pelo fato da solução de entrada ser inviável) que, em caso positivo, é armazenado e os valores das variáveis *k* e *iterSemMelhora* são alterados para os valores iniciais.

Na linha 8, o algoritmo garante que o procedimento de perturbação será aplicado ao ótimo local sem comprometer sua integridade. Nas linhas 9 e 10, o procedimento de perturbação com intensidade *k* é aplicado ao ótimo local, gerando uma solução (possivelmente inviável) que servirá de entrada para a próxima iteração do laço. Se o número de iterações sem melhora atingir o valor máximo antes do tempo de execução atingir o tempo limite, a melhor solução encontrada até então será armazenada e outra solução será construída.

4.1. Procedimento de geração da solução inicial

Na fase de construção, a solução inicial é gerada de forma gulosa. Cada roteador é posicionado de forma a atingir a maior quantidade de pontos clientes cobertos possível, ou seja, dentro do raio de cobertura do roteador. Os empates são quebrados de forma equívoca. Pelo fato das restrições de capacidade não serem levadas em consideração, soluções inviáveis serão construídas e penalizadas, levando a um alto valor de função objetivo.

4.2. Heurísticas de busca local

Nesta seção descrevemos o procedimento de busca local implementado. Este consiste nas cinco estruturas de vizinhança descritas a seguir, que são executadas conforme o *framework* RVND padrão.

allChangeLink(): Este procedimento verifica, para todos os pontos clientes da solução que estão conectados a algum roteador, se este último está com uma capacidade de banda passante acima do limite. Em caso positivo, será feita uma tentativa de reconectar o ponto corrente para algum outro roteador que ainda não esteja na sua capacidade máxima de banda.

allDryRouter(): Procura por roteadores ativos que estejam com uma exigência de banda passante acima do limite. Após identificá-los, um a um, tenta distribuir todos os pontos conectados ao roteador corrente para outros roteadores ativos que ainda não estão na sua capacidade máxima de banda. Se o procedimento for realizado com sucesso, ou seja, se todos os pontos do roteador analisado forem conectados a outros roteadores, o mesmo será retirado da solução, diminuindo o custo da função objetivo.

bestAddRouter(): Dados os pontos clientes ainda não conectados a nenhum roteador da solução, procura a posição onde, posicionando-se um roteador, este último atenderá a maior quantidade possível de pontos clientes não conectados. Encontrada tal posição, um roteador é adicionado à solução e todos os pontos clientes dentro do raio de cobertura do roteador serão conectados ao mesmo, desde que não ultrapassem sua capacidade.

allBestRadiusType(): Analisa todos os roteadores da solução. Se um roteador possuir algum ponto cliente conectado com uma demanda por qualidade de sinal acima/abaixo da necessária (ponto cliente próximo/distante do roteador), este roteador será substituído por outro com um raio de cobertura menor ou maior, se disponível, a fim de satisfazer a demanda por qualidade de sinal do ponto cliente com eficiência.

bestLink(): Verifica, um por um, os pontos clientes ainda não conectados da solução. Se o ponto cliente corrente está dentro do raio de cobertura de um roteador ativo e este possui uma quantidade de banda suficiente para receber o ponto cliente, o procedimento fará a ligação do ponto ao roteador em questão.

Após o método de busca local RVND, um ótimo local foi encontrado. O critério de aceitação do algoritmo é encontrar um novo ótimo local e continuar a busca por melhores soluções partindo do mesmo. Se isto acontecer, este ótimo local de melhor qualidade é armazenado (aceito) e o procedimento de perturbação é aplicado sobre este.

4.3. Mecanismos de perturbação

Nesta seção, descrevemos os procedimentos utilizados para perturbar as soluções geradas. As perturbações modificam o ótimo local gerando uma nova solução, com o objetivo de explorar outras regiões do espaço de soluções. Uma variável k controla o nível de perturbação. Cada vez que o procedimento é executado, são escolhidas k heurísticas de perturbação para serem executadas. As heurísticas utilizadas são descritas a seguir:

randomDelRouter(): Retira aleatoriamente um roteador da solução atual. Após a deleção, os pontos clientes que estavam conectados a este roteador ficarão desconectados e os pontos proibidos que estavam sendo cobertos ficarão descobertos.

randomDelLink(): Exclui a conexão entre um ponto cliente escolhido aleatoriamente e o roteador ao qual está conectado. Esta operação é executada cinco vezes, com o objetivo de garantir a eficácia da perturbação na solução atual.

randomChangeRouterType(): Escolhe aleatoriamente um roteador da solução e modifica o seu tipo. Este último também é escolhido aleatoriamente. Como consequências da execução desta perturbação, raio, limite de banda e custo do roteador modificado serão alterados, podendo acarretar em pontos clientes com demanda por qualidade de sinal insatisfeita e roteadores com limite de banda excedida.

randomMoveRouter(): Move um roteador da solução para uma outra posição escolhida aleatoriamente. Isso fará com que os pontos conectados ao roteador, dependendo da posição escolhida, passem a possuir uma demanda por qualidade de sinal insatisfeita ou satisfeita, além de cobrir e descobrir pontos proibidos.

randomJoinRouter(): Dados dois roteadores $r1$ e $r2$ escolhidos aleatoriamente, este procedimento transfere todos os pontos conectados ao roteador $r1$ para o roteador $r2$, sem se preocupar com a demanda por qualidade de sinal dos pontos clientes ou com o limite de banda do roteador alvo. Isto causará a existência de pontos clientes com demanda por qualidade de sinal insatisfeita ou satisfeita e de roteadores com seu limite de capacidade de banda excedido.

5. Geração de instâncias

Uma vez que a literatura na área do APPP é bastante fragmentada e possui distintos tipos de objetivos, se fez necessária a criação de instâncias para avaliar a eficiência dos métodos propostos. Para tal, foi utilizada a discretização do espaço em uma grade de pontos, conforme exibido na Figura 1. Cada ponto é posicionado (horizontal e verticalmente) há um metro dos outros.

O modelo de propagação utilizado para a geração de instâncias foi baseado nos estudos de Ogunjemilua *et al.* (2009), Androne e Palade (2010) e Sohail *et al.* (2013), sobre o comportamento do sinal das redes Wi-Fi na frequência de 2.4 Ghz. A Tabela 1 descreve os valores de atenuação utilizados nas instâncias geradas.

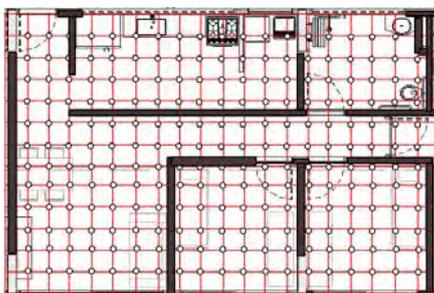


Figura 1. Discretização do espaço

Tipo de Anteparo	Atenuação
Divisória de Madeira (4cm)	2db
Divisória de Vidro (3cm)	3db
Drywall (5-10cm)	4db
Parede de Tijolo (15cm)	6db
Parede de Concreto (15-20cm)	9db
Parede de Concreto (25-30cm)	12db

Tabela 1. Modelo de atenuação

Foram gerados dois tipos de instâncias: randômicas e reais, todas utilizando os três tipos de roteadores descritos na Tabela 2. Foram gerados 7 grupos de instâncias randômicas, cada um contendo 10 casos de teste em ordem crescente de dificuldade. Cada grupo possui o mesmo conjunto de pontos e de anteparos, sendo o tipo de anteparo selecionado aleatoriamente, assim como a demanda por qualidade e por largura de banda de cada ponto. Além disso, cada grupo é identificado pela largura e altura (em metros) da área

onde foram gerados os pontos, sendo o grupo 8x14 o menor e o 50x50 o maior.

Tabela 2. Tipos de roteadores utilizados

Tipo de Roteador	Custo	Alcance	Banda Passante
Roteador de Entrada	150 R\$	100m	150 Mbps
Roteador Médio	200 R\$	100m	300 Mbps
Roteador Alta Qualidade	300 R\$	400m	400 Mbps

Por fim, foram geradas 5 instâncias baseadas em plantas-baixas reais: me_100x40, referente a uma estação de metrô; ho_60x32, referente a um andar de hotel; of_100x35, referente a um escritório; sc_70x45, referente a uma escola; e sh_100x60, referente a um pequeno *shopping center*. Equivalente às instâncias randômicas, as instâncias reais são identificadas pelas largura e altura das plantas-baixas utilizadas como base.

6. Experimentos computacionais

Os experimentos computacionais foram realizados em um computador Intel Core i7 1.80 GHz com 4GB de RAM e sistema operacional Ubuntu 14.04 LTS. Todos os métodos foram implementados na linguagem C++ e compilados com o g++ na versão 4.8.4, utilizando a opção -O3. O método exato foi implementado utilizando a linguagem Concert e os testes foram executados utilizando o *solver* CPLEX 12.51 com tempo limite de 2 horas.

O ILS proposto foi comparado com adaptações das meta-heurísticas SA (Farkas *et al.*, 2013), AG (Barbosa e Gouvea, 2012) e TS (Lu *et al.*, 2006). Estas adaptações possuem a representação da solução e, conseqüentemente, o cálculo da função objetivo adequados para a versão do problema tratado neste trabalho. Cada método foi executado 10 vezes para cada instância, com um tempo limite de 10s para instâncias de tamanho 8x14 e 13x13, 30s para instâncias de tamanho 11x23, 25x25 e 30x30 e 60s para as instâncias reais, de tamanho 50x25 e 50x50. A escolha do tempo limite baseou-se na quantidade de execuções necessárias para se obter boas soluções para determinada instância.

Os resultados apresentados nas tabelas (3 - 10) reportam o melhor valor da função objetivo e a média aritmética, obtidos entre 10 execuções para cada instância do grupo. Para o grupo r_8x14 (tabela 3), o ILS e o GA empatam na solução mínima encontrada, porém o GA obteve uma média melhor na instância 7. Para o grupo r_13x13 (tabela 4), o ILS consegue leve vantagem na solução mínima, com destaque para a instância 7, e perde na solução média apenas nas instâncias 0 e 6. Para o grupo r_11x23 (tabela 5), o ILS possui desempenho competitivo com relação ao AG, com destaque para a instância 1, onde houve uma diferença considerável.

Para o grupo r_25x25 (tabela 6), o ILS demonstrou maior capacidade de resolução e robustez que os demais métodos, obtendo apenas uma derrota na instância 8. Nos grupos r_30x30 e r_50x25 (tabelas 7 e 8), fica evidente a vantagem obtida pelo ILS, uma vez que o mesmo conseguiu melhores resultados em quase todas as instâncias. Para o grupo r_50x50 (tabela 9), o ILS conseguiu os melhores resultados para todas as instâncias.

Dentro do tempo limite de 2 horas, a formulação matemática proposta foi capaz de resolver todas as instâncias do grupo r_8x14, as 6 primeiras instâncias do grupo r_13x13 e

apenas a primeira instância do grupo r_11x23. Quando os métodos heurísticos alcançam o resultado ótimo obtido pelo método exato, este é grafado em negrito e com um asterisco.

Os resultados obtidos para o conjunto de instâncias reais é reportado na tabela 10. Nesta, k significa 1000 vezes. Para este grupo, o ILS novamente conseguiu os melhores resultados para todas as instâncias. Os métodos da literatura, no entanto, enfrentaram dificuldades neste grupo, alcançando valores de função objetivo muito altos. Acreditamos que o tamanho das instâncias reais não permite que estes métodos executem suas buscas locais dentro do tempo limite estabelecido, reportando apenas soluções iniciais construídas aleatoriamente.

De acordo com os experimentos realizados, percebe-se que o método abordado neste trabalho mostra-se bastante competitivo em relação aos outros métodos da literatura implementados para instâncias pequenas. Porém, à medida que as instâncias ficam maiores, ou seja, as quantidades de pontos e de obstáculos aumenta, o ILS-RVND consegue obter resultados melhores.

Tabela 3. Resultados Computacionais para o Grupo de Instâncias r_8x14

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	339*	339	339*	339.9	339*	339	339*	339
1	181*	181	181*	290.3	181*	181	181*	255.5
2	181*	181	181*	229.9	181*	181	181*	242.2
3	333*	333	333*	351	333*	333	333*	333.4
4	326*	326	339	339.5	326*	326	326*	335.6
5	338*	338	340	340	338*	338	338*	369.6
6	222*	222	223	302.9	222*	222	222*	263.7
7	336*	336.6	337	353.4	336*	336	338	377.4
8	229*	229	330	338.6	229*	229	330	331.8
9	339*	339	389	439.9	339*	339	339*	394

Tabela 4. Resultados Computacionais para o Grupo de Instâncias r_13x13

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	300*	301	300*	349.8	300*	300	300*	315.8
1	305*	305	359	401.5	305*	305	305*	320.3
2	513	513	563	692.8	513	513	513	538
3	512*	512	563	658	512*	512.1	513	553
4	613*	613	663	778	613*	613	613*	633
5	613*	613	713	813	613*	613	613*	673
6	703	703.4	856	908.5	703	703	703	743.1
7	812	812.4	913	1088	813	813	813	898
8	963	963	1063	1243	963	977.8	1012	1072.4
9	1212	1212	1413	1543	1212	1212.6	1213	1317.8

Tabela 5. Resultados Computacionais para o Grupo de Instâncias r_11x23

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	457*	457	507	600.6	457*	457	457*	523.8
1	522	522	573	702.9	569	569.2	569	640.3
2	618	619.2	723	842.9	617	617.5	617	704.5
3	750	753.6	859	933.1	750	756.3	800	842.6
4	823	823	1023	1163	872	872	872	907.4
5	1019	1094	1217	1332.4	1112	1114.6	1017	1167.3
6	1113	1113	1373	1533	1113	1113.6	1167	1198.3
7	1155	1156.1	1362	1475.9	1150	1150	1150	1211.7
8	1323	1323	1673	1753	1322	1322.2	1372	1442.6
9	1523	1523	1873	2043	1522	1532.4	1623	1662.8

Tabela 6. Resultados Computacionais para o Grupo de Instâncias r_25x25

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	1300	1300	1700	1780	1300	1345	1300	1375
1	1300	1300	1600	1665	1300	1325	1300	1360
2	1600	1600	1850	2135	1600	1615	1600	1745
3	1800	1865	2100	2395	1900	1940	1800	1940
4	2000	2060	2500	2670	2050	2135	2000	2165
5	2300	2380	2650	3065	2400	2495	2350	2470
6	2500	2580	3050	3300	2600	2660	2600	2795
7	3050	3075	3750	3985	3200	3260	3050	3190
8	3800	3830	4500	4825	3950	4005	3750	3865
9	3900	3980	4600	4955	4050	4125	4000	4195

7. Considerações Finais e Proposta de Trabalhos Futuros

Neste trabalho, propusemos um algoritmo heurístico baseado na meta-heurística *Iterated Local Search* (ILS), o qual apresentou bons resultados na resolução do Problema de Posicionamento de Pontos de Acesso (APPP). O método se mostrou bastante competitivo em relação aos outros métodos da literatura implementados para pequenas instâncias, enquanto que para instâncias maiores, conseguiu superá-los de forma promissora.

Como trabalhos futuros, pretendemos aprimorar o método heurístico proposto através da criação de algoritmos de pré-processamento e de novas heurísticas de busca local, com foco nas instâncias as quais não obtivemos os resultados desejados, além de expandir os métodos descritos para a realização de posicionamento de APs em três dimensões. Acreditamos também que o AG parece promissor e, por isso, vamos tentar melhorá-lo para conseguir bons resultados para as instâncias maiores. Ainda para o futuro da pesquisa, pretendemos, adicionalmente, desenvolver um *software* que permita a criação de plantas-baixas para a subsequente execução do posicionamento automático dos APs.

Tabela 7. Resultados Computacionais para o Grupo de Instâncias r_30x30

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	1200	1247.3	1464	1718.3	1368	1473.6	1259	1359.1
1	1333	1333	1679	1859	1442	1480.9	1392	1458.1
2	1775	1834.7	2125	2285	1875	1940	1825	1959.9
3	1600	1618.4	2193	2347.4	1894	1954.4	1600	1923.1
4	2174	2258.6	2725	2790	2375	2475	2325	2478.6
5	2451	2466.4	3325	3380.6	2623	2709.1	2551	2661.9
6	3075	3095	3375	3735	3175	3255	3125	3475
7	2924	2935.4	3625	3849.6	3125	3213.9	3175	3335.1
8	4159	4198.9	5125	5310	4375	4530	4275	4830
9	3892	4014.5	4819	5088.2	4323	4434.9	4345	5481.4

Tabela 8. Resultados Computacionais para o Grupo de Instâncias r_50x25

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	2200	2270	2750	3035	2400	2610	2200	3000
1	2600	2690	3150	3485	2800	2825	2650	2785
2	3600	3680	4450	4685	3850	3925	3700	3930
3	4000	4050	4900	5095	4100	4395	3950	4545
4	4900	4935	5600	6230	5250	5380	4900	5655
5	5000	5140	6200	6585	5350	5470	5250	5500
6	5900	6020	7100	7500	6300	6565	6050	6910
7	6400	6510	8000	8255	6800	7035	6450	7245
8	7600	7750	9350	9675	8250	8425	8200	9345
9	8500	8680	10350	10685	9250	9430	9850	10430

Referências

- Androne, C. e Palade, T.** Radio coverage and performance analysis for local area networks. *Electronics and Telecommunications (ISETC), 2010 9th International Symposium on*, p. 213–216. IEEE, 2010.
- Arroyo, J. E. C. e Marques, T. B.** (2006), Heurística grasp aplicado ao problema de alocação de antenas de transmissão. *XXXVIII Simpósio Brasileiro de Pesquisa Operacional, Goiânia-GO*.
- Barbosa, M. A. S. e Gouvea, M. M.** Access point design with a genetic algorithm. *Genetic and Evolutionary Computing (ICGEC), 2012 Sixth International Conference on*, p. 119–123. IEEE, 2012.
- Capdeville, R. M. A. e Vianna, D. S.** (2013), Heurísticas grasp para o problema de alocação de pontos de acesso em uma rede sem fio em ambiente indoor. *Sistemas & Gestão*, v. 8, n. 1, p. 86–93.
- Farkas, K., Huszák, Á. e Gódor, G.** (2013), Optimization of wi-fi access point placement for indoor localization. *Journal IIT (Informatics & IT Today)*, v. 1, n. 1, p. 28–33.
- Kamenetsky, M. e Unbehaun, M.** Coverage planning for outdoor wireless lan systems. *Broadband Communications, 2002. Access, Transmission, Networking. 2002 Internati-*

Tabela 9. Resultados Computacionais para o Grupo de Instâncias r_50x50

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
0	4850	4930	6050	6215	5400	5650	6050	7500
1	5950	6100	7500	7800	6700	6985	7100	9050
2	7050	7190	8700	9180	7600	7935	7600	8825
3	8031	8112.8	9624	10130	9500	9784.5	10450	13784.4
4	9400	9510	11700	12080	10550	11000	11300	13095
5	10911	11015.7	13284	13695.4	12294	12687.4	13242	15593.6
6	11250	11355	13350	14135	13300	13895	15000	18065
7	12731	12960.5	15598	16300.2	15500	16180	16100	18570
8	16450	16565	19950	20395	18400	19085	20750	22210
9	17050	17230	20950	21595	19700	20170	21100	23060

Tabela 10. Resultados Computacionais para as Instâncias Reais

Inst #	ILS		SA		AG		BT	
	min	avg	min	avg	min	avg	min	avg
ho_60x32	770	770	920	970	820	850	820	880
sc_70x45	8994	9247,9	95k	997k	286k	361k	417k	458k
of_100x35	3490	3579,9	332k	338k	105k	176k	223k	396k
me_100x40	5250	5400	5750	5930	352k	376k	428k	456k
sh_100x60	1550	1625	930k	935k	4300	13485	5750	523k

onal Zurich Seminar on, p. 49–1–49–6. doi: 10.1109/IZSBC.2002.991793, 2002.

Lourenço, H. R., Martin, O. C. e Stützle, T. Iterated local search. *Handbook of metaheuristics*, p. 320–353. Springer, 2003.

Lu, J.-L., Runser, K., Gorce, J. e Valois, F. Indoor wlan planning with a qos constraint based on a markovian performance evaluation model. *2006 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, p. 152–158. doi: 10.1109/WIMOB.2006.1696372, 2006.

Marques, T. B. *Heurísticas para o problema de localização/alocação de antenas de transmissão*. Tese de doutorado, Dissertação de Mestrado em Pesquisa Operacional e Inteligência Computacional, Universidade Candido Mendes–Campos, Campos dos Goytacazes, RJ, 2007.

Megiddo, N. e Tamir, A. (1982), On the complexity of locating linear facilities in the plane. *Operations research letters*, v. 1, n. 5, p. 194–197.

Ogunjemilua, K., Davies, J. N., Picking, R. e Grout, V. (2009), An investigation into signal strength of 802.11 n wlan.

Sohail, A., Ahmad, Z. e Ali, I. (2013), Analysis and measurement of wi-fi signals in indoor environment. *International Journal of Advances in Engineering & Technology*, v. 6, n. 2, p. 678.

Vanhatupa, T., Hännikäinen, M. e Hämäläinen, T. D. Genetic algorithm to optimize node placement and configuration for wlan planning. *Wireless Communication Systems, 2007. ISWCS 2007. 4th International Symposium on*, p. 612–616. IEEE, 2007.