

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Abordagens heurísticas para o problema de atribuição de potência e de canais de transmissão em redes cognitivas

Lucas Rossini Ferreira

JUIZ DE FORA
JULHO, 2015

Abordagens heurísticas para o problema de atribuição de potência e de canais de transmissão em redes cognitivas

LUCAS ROSSINI FERREIRA

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Stênio São Rosário Furtado Soares

JUIZ DE FORA

JULHO, 2015

ABORDAGENS HEURÍSTICAS PARA O PROBLEMA DE
ATRIBUIÇÃO DE POTÊNCIA E DE CANAIS DE TRANSMISSÃO
EM REDES COGNITIVAS

Lucas Rossini Ferreira

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Stênio Sã Rosário Furtado Soares
Doutor em Ciência da Computação (UFF)

Luciana Brugiolo Gonçalves
Doutora em Ciência da Computação (UFF)

Lorenza Leão Oliveira Moreno
Doutora em Informática (PUC-Rio)

JUIZ DE FORA
6 DE JULHO, 2015

À todos que acreditaram que um dia eu chegaria aqui!

Resumo

Atualmente, em redes de comunicação sem fio, a crescente quantidade de dispositivos simultaneamente conectados exige grande utilização do espectro. Tendo em vista a limitação do número de canais disponíveis no espectro, novas tecnologias de balanceamento de carga e redução de interferência são necessárias. Neste cenário, um importante problema que ocorre em redes sem fio que deve ser trabalhado pelas entidades que definem a arquitetura do padrão IEEE 802.11 é a interferência causada pelos múltiplos Pontos de Acesso (AP, do inglês *Access Point*) dentro de uma mesma região compartilhada por estes dispositivos. Com o objetivo de reduzir a interferência gerada dentro na rede e garantir métricas de qualidade de serviço para os dispositivos conectados, abordagens heurísticas foram desenvolvidas para identificar o melhor canal e potência de transmissão a serem definidos para cada AP. Tais heurísticas foram combinadas em um algoritmo GRASP (*Greedy Randomized Adaptative Search Procedure*) e submetidas a testes comparativos com uma abordagem da literatura. Os resultados demonstram a eficiência e eficácia das soluções propostas para o problema.

Palavras-chave: Redes cognitivas, coexistência, otimização, heurísticas, GRASP, IEEE 802.11, seleção de canais, controle de potência.

Abstract

Currently, in wireless communication networks, the growing number of simultaneously connected devices requires great use of spectrum. With a view to limiting the number of available channels in the spectrum, new load-balancing technology and noise reduction are necessary. In this scenario, a major problem that occurs in wireless networks that must be worked by the entities that define the IEEE 802.11 standard architecture is the interference caused by multiple Access Points (AP) within the same shared region by these devices. In order to reduce the interference generated within the network and ensure service quality metrics for connected devices, heuristic approaches have been developed to identify the best channel and power transmission to be defined for each AP. Such heuristics were combined in a GRASP (Greedy Randomized Adaptive Search Procedure) and subjected to comparative tests with an approach of the literature. The results demonstrate the efficiency and effectiveness of the proposed solutions to the problem.

Keywords: Cognitive networks, coexistence, optimization, heuristics, GRASP, IEEE 802.1, channel selection, power control.

Agradecimentos

A Deus, pela vida, saúde e conquistas concedidas. Sem isso nada seria possível;

À minha família, que sempre fez de tudo para me dar a oportunidade de chegar onde estou agora;

À Camila, meu amor eterno, que está sempre ao meu lado e é meu incentivo para evoluir cada dia mais;

Aos amigos, por toda a ajuda concedida e esforço compartilhado para crescermos juntos;

Ao professor Stênio pela orientação, paciência e confiança que foram fundamentais para realização deste trabalho;

À Universidade Federal de Juiz de Fora e ao Departamento de Ciência da Computação, por toda a estrutura, conhecimento e apoio fornecidos para minha formação.

“Don’t comment bad code, rewrite it.”

B. W. Kernighan, P. J. Plauger

Sumário

Lista de Figuras	7
Lista de Tabelas	8
Lista de Abreviações	9
1 Introdução	10
2 Fundamentação teórica	12
2.1 Coexistência em redes cognitivas	13
2.1.1 Técnicas de implementação da coexistência	14
2.1.2 Desafios em redes compartilhadas	15
3 Problema de atribuição de potência e de canais de transmissão em redes cognitivas	18
3.1 Representação do problema	19
3.2 Trabalhos relacionados	20
3.2.1 Algoritmo SCIFI	21
4 Abordagens desenvolvidas	24
4.1 Heurística de construção	25
4.2 Heurística de busca local	28
4.3 Algoritmo de refinamento de potências	30
4.4 Abordagens GRASP propostas	31
5 Experimentos e resultados	34
5.1 Comparação entre heurísticas de construção e a literatura	35
5.2 Comparação entre as abordagens GRASP propostas	38
5.3 Comparação entre GRASP e literatura	40
6 Conclusão	42
Referências Bibliográficas	44

Lista de Figuras

2.1	Coexistência não-colaborativa versus coexistência colaborativa (Lansford, 2004)	14
3.1	Canais de transmissão no espectro de 2.4 GHz (Chieochan et al, 2010) . . .	19
3.2	Exemplo de rede sem fio controlada pelo SCIFI (Balbi et al, 2012)	21
4.1	Comportamento da heurística de busca local em um espaço de busca (Talbi, 2009)	29
5.1	Distribuição de APs e clientes em octantes a partir de uma instância do problema	35

Lista de Tabelas

5.1	Comparação entre os resultados dos algoritmos de construção da abordagem proposta com o SCIFI	37
5.2	Comparação entre os resultados das metaheurísticas GRASP desenvolvidas	39
5.3	Comparação entre os resultados do GRASP 3 com o SCIFI	41

Lista de Abreviações

ANATEL	Agência Nacional de Telecomunicações
AP	Access Point
CR	Cognitive Radio
CSMA	Carrier Sense Multiple Access
FCC	Federal Communications Commission
GRASP	Greedy Randomized Adaptive Search Procedure
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
OSI	Open Systems Interconnection
QoS	Quality of Service
RAM	Random Access Memory
RDI	Relative Deviation Index
RF	Rádiofrequência
RNP	Rede Nacional de Ensino e Pesquisa
SCIFI	Sistema de Controle Inteligente para redes sem FIo
TDMA	Time Division Multiple Access
UFF	Universidade Federal Fluminense

1 Introdução

Em redes de comunicação por rádio, o espectro (meio utilizado para a transmissão de ondas) é um recurso muito limitado. Informações medidas pelo *Federal Communications Commission* (FCC) em 2002, o órgão regulador da área de telecomunicações dos Estados Unidos e equivalente à Agência Nacional de Telecomunicações (ANATEL) no Brasil, indicam que, em média, as faixas de frequência licenciadas permanecem inutilizadas durante 90% do tempo.

Com o crescimento desenfreado de redes e dispositivos sem fio, novos sistemas *wireless* necessitam de uma largura de banda maior que as convencionais e, atualmente, não existem faixas de frequência suficientes para serem utilizadas futuramente por sistemas de banda larga sem fio. Além disso, vários sistemas de comunicação por rádio devem coexistir em uma faixa de frequência comum para que haja um melhor aproveitamento desse espectro (Harada, 2010).

Para suportar um tráfego de informações pesado, tecnologias de balanceamento de carga entre múltiplos esquemas de acesso a dados são necessárias, incluindo, também, a redução da contaminação por interferência entre sistemas de comunicação por rádio. Um dos problemas mais recorrentes em redes do padrão IEEE 802.11 é a interferência causada entre Pontos de Acesso dentro de uma mesma região compartilhada por esses dispositivos.

Abordagens para solucionar o problema da interferência entre APs visam minimizar o somatório das interferências causadas pelos Pontos de Acesso da rede sobre os demais e, assim, melhorar a Qualidade de Serviço – QoS, do inglês *Quality of Service* – oferecido para seus respectivos clientes (dispositivos móveis conectados), efetuando a realocação dos canais de operação e das potências de transmissão de cada AP.

Para que não haja interferência em uma rede sem fio, os Pontos de Acesso que a compõem devem ser configurados para operar sobre canais não sobrepostos, isto é, canais que não causam interferência entre si. Porém, o reuso de canais acaba sendo necessário devido à escassez dos mesmos. As bandas de espectro disponíveis para o padrão 802.11

no Brasil são 2.4 GHz e 5 GHz, cujos canais de operação estão contidos no intervalo de 1 a 11, sendo os canais 1, 6 e 11 denominados ortogonais ou não sobrepostos. No caso em que dois ou mais APs que operam em canais não ortogonais, quanto maior a proximidade dos mesmos, maior é a intensidade da interferência entre os mesmos.

Uma rede sem fio para o problema abordado pode ser modelada a partir de um digrafo ponderado $G = (V, E)$ tal que $V = A \cup D$ é o conjunto de vértices representado pelos APs (conjunto A) e clientes (conjunto D) em uma rede e E é o conjunto de arcos $(i, j) \mid i, j \in A$, cujos pesos c_{ij} indicam a intensidade da interferência causada pelo AP i no AP j . Destaca-se, entretanto, que os pesos associados aos arcos estão relacionados a dois fatores inerentes aos nós: o canal utilizado e a potência de transmissão em que o mesmo opera.

O problema tratado neste trabalho consiste em definir uma atribuição de canais de espectro e uma atribuição de potências de transmissão aos nós de A de forma que o somatório dos pesos dos arcos entre estes seja minimizado, garantindo que cada nó de D esteja sempre coberto pelo sinal de transmissão dos nós de A .

O presente trabalho está assim estruturado: além deste capítulo de introdução, o Capítulo 2 apresenta a fundamentação teórica do tema trabalhado; no Capítulo 3 é apresentado o problema de forma mais detalhada e a principal abordagem da literatura para o mesmo. O Capítulo 4 explica as abordagens desenvolvidas para o problema. O Capítulo 5 apresenta a análise dos resultados computacionais, enquanto o Capítulo 6 apresenta as conclusões e propostas de trabalhos futuros.

2 Fundamentação teórica

Rede cognitiva é uma rede capaz de perceber as condições atuais da mesma e planejar, decidir e atuar sobre essas condições de forma a melhor atender critérios de *QoS*. A rede pode aprender a partir dessas adaptações e utilizar essas informações para tomar futuras decisões, enquanto leva em consideração os objetivos de transmissão fim-a-fim (Thomas et al, 2007).

Segundo Sousa et al (2010) e Akyildiz et al (2006), em um ambiente de redes cognitivas, algumas porções do espectro estão licenciadas para diferentes propósitos, enquanto outras permanecem não licenciadas. Para evitar interferência entre os sinais de rádio transmitidos, políticas de alocação do espectro de rádiofrequência (RF) são estabelecidas por agências governamentais reguladoras (ex.: ANATEL no Brasil, FCC nos Estados Unidos), geralmente vinculando-as ao pagamento de licenças para seu uso.

A arquitetura de redes cognitivas cobre todas as camadas do modelo ISO/OSI e, de acordo com Song e Hatzinakos (2009), pode ser interpretada como uma rede que é capaz de utilizar de forma oportunística tanto o espectro de rádio quanto recursos de estações *wireless* baseadas em aprendizado passado adquirido pelo processo cognitivo.

A tecnologia de uma rede cognitiva aumenta a eficiência da alocação espectral por meio do acesso oportunista às faixas de frequência, pois fornece aos usuários móveis uma grande largura de banda através de técnicas de acesso dinâmico ao espectro (Sousa et al, 2010).

Os usuários que utilizam o rádio cognitivo (CR) sobre esse tipo de rede podem ser divididos em dois tipos (Sousa et al, 2010):

- **Primário:** possuem licença para operar em uma determinada faixa do espectro de frequência e não deve sofrer interferências causadas por transmissões de outros usuários não licenciados. Além disso, esse tipo de usuário não tem a necessidade de modificar suas configurações de operação e infraestrutura para coexistir com demais usuários. Alguns exemplos conhecidos de usuários primários podem ser redes de

telefonia celular e de televisão;

- **Secundário:** Também chamados de usuários cognitivos, esse tipo de usuário não possui nenhuma licença para uso dos recursos do espectro. Como essas faixas são reservadas para os usuários primários, a utilização delas pelos usuários secundários se dá a partir do compartilhamento do espectro e opera de forma oportunista, ou seja, enquanto está ociosa. Sua comunicação com outros usuários cognitivos não deve causar interferências aos usuários primários da faixa.

A partir do momento em que o estado de cognição da rede seja atingido de forma satisfatória, os dispositivos que formam sua estrutura passam a coexistir, trabalhando de forma compartilhada e realizando trocas de informações para seu melhor funcionamento.

2.1 Coexistência em redes cognitivas

De acordo com a definição dada pelo IEEE, coexistência em sistemas de redes sem fio significa "a habilidade de um sistema executar tarefas em um ambiente compartilhado onde outros sistemas possuem a habilidade de executar suas tarefas utilizando, ou não, o mesmo conjunto de regras", e sua medida pode ser calculada através da taxa de sucesso de entrega de mensagens fim-a-fim entre dispositivos.

Quanto melhor o nível de coexistência entre os dispositivos da rede, maior é a vazão de dados transmitidos por estes devido à interferência reduzida, aumentando a taxa de sucesso de entrega de pacotes. Caso contrário, a falta de interação e comunicação entre os emissores pode resultar em uma elevada taxa de interferência na rede, prejudicando a qualidade das informações transmitidas.

Peha (2000) diz que, para que dispositivos inteligentes possam coexistir, compartilhar o mesmo espectro de utilização e até mesmo interoperarem, alguns objetivos devem ser alcançados:

- Para todos os dispositivos da rede deve-se assegurar uma qualidade de serviço, no mínimo, suficiente para sua utilização;
- Nenhum dispositivo deve passar por uma situação de *starvation*, ou seja, permanecer

aguardando um recurso indefinidamente devido ao seu sinal bloqueado por longos períodos. Quando isto ocorre, tem-se uma baixa qualidade de serviço e deve-se tratar o problema de forma a assegurar que não existam dispositivos na rede capazes de transmitir dados ininterruptamente, prejudicando os demais;

- O espectro é limitado e, por esse motivo, deve ser utilizado com consciência e eficiência pelos dispositivos.

2.1.1 Técnicas de implementação da coexistência

Como dado pelo padrão IEEE 802.15.2, técnicas de como a coexistência pode ser aprimorada são divididas em duas categorias, que são descritas a seguir:

- **Colaborativa:** um mecanismo deve existir para realizar a comunicação direta entre sistemas de redes sem fio;
- **Não-colaborativa:** não existe mecanismo direto para coordenação dos sistemas, sendo assim necessário que um ou ambos os sistemas devam inferir o ambiente e tomarem uma decisão unilateral, ou seja, sem se preocupar com seus vizinhos.

As duas técnicas de aprimoramento da coexistência são ilustradas pela Figura 2.1 como um modelo de tráfego.

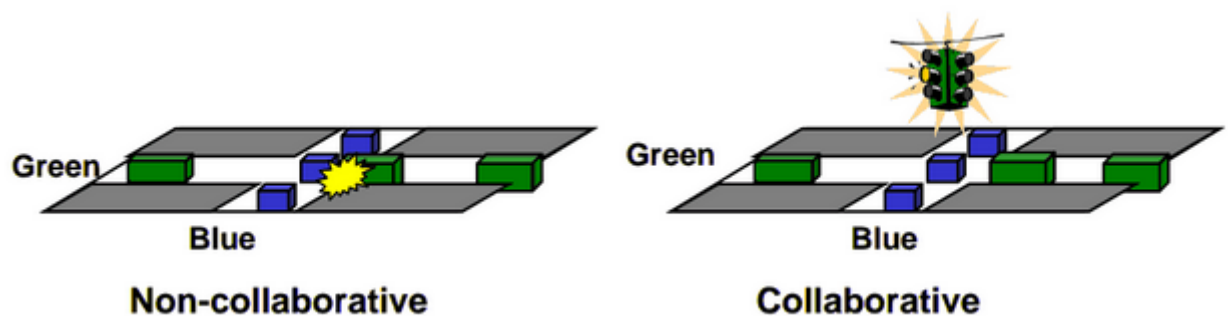


Figura 2.1: Coexistência não-colaborativa versus coexistência colaborativa (Lansford, 2004)

Na forma de coexistência não-colaborativa, o tráfego flui de ambos os sistemas e colisões (interferência) só podem ser evitadas através de técnicas de sensoriamento médio como CSMA ("olhe antes de cruzar") ou salto adaptativo de frequência. Esta última pode

ser vista como uma analogia aos sistemas de controle de tráfego de veículos, que sugerem uma mudança de rota sempre que se verifica que em uma determinada via estão ocorrendo muitos acidentes. Em qualquer uma destas técnicas, as vias (canais de transmissão) não estão sendo utilizadas em sua capacidade máxima.

Já na técnica de coexistência colaborativa, as colisões podem ser completamente mitigadas se os sistemas sem fio forem capazes de compartilhar, em tempo real, informações detalhadas sobre seu tempo de uso do espaço desejado, faixa de frequência utilizada e potência de transmissão.

Na analogia com o tráfego, a Figura 2.1 mostra um sinal de trânsito (TDMA) que, claramente, divide a utilização do espaço através do tempo. Em uma implementação mais completa de rádios cognitivos, os sistemas podem otimizar sua performance (vazão, latência, taxa de erros, etc) apenas compartilhando o conhecimento dessas variáveis de cada dispositivo.

Baykas et al (2012) afirmam que os mecanismos colaborativos dependem da troca de informações entre redes heterogêneas¹ e de utilizar essas informações para negociar o particionamento do canal compartilhado. Quando os parâmetros de funcionamento são aceitos por ambas as partes, o canal é dividido cooperativamente.

O trabalho aqui proposto se define na implementação de técnicas colaborativas para garantir a coexistência dos dispositivos de transmissão da rede, onde o sistema controlador desenvolvido é aplicado de forma centralizada e realiza a divisão do espectro de uso através do tempo, bem como o método TDMA.

2.1.2 Desafios em redes compartilhadas

A coexistência ocorre quando dois ou mais dispositivos da rede compartilham recursos de um espectro, e o nível, ou qualidade dessa coexistência, está relacionado diretamente com a experiência sofrida por esses dispositivos (Baykas et al, 2012). Simplificando, uma boa coexistência é caracterizada por níveis aceitáveis de interferência, enquanto uma coexistência ruim é caracterizada por interferências obstrutivas.

Segundo IEEE (2010), melhorar a coexistência entre redes e dispositivos geral-

¹Definidas por Delphinanto et al (2009) como redes que conectam dispositivos que operam sobre diferentes sistemas e/ou protocolos.

mente é uma tarefa complexa e que envolve múltiplos passos, que devem ser efetuados de forma individual ou concorrente. Para uma melhor compreensão, os mesmos são apresentados a seguir:

1. **Descoberta:** A descoberta em coexistência possui duas fases: descoberta de interferência e descoberta do vizinho que a esteja causando. Para a detecção de interferência, um bom indicador é a queda da eficiência da rede, podendo ser esta medida pela diminuição da vazão de dados ou pelo aumento da latência. Assim, o desafio é isolar os elementos da rede com o intuito de identificar qual(is) dos nós está(ão) sendo mais afetado(s) pela interferência e passar para a próxima fase.

Já a segunda fase da descoberta identifica qual rede próxima pode estar causando a interferência. Para isso, é necessária uma análise da propagação de rádio de todas as redes vizinhas, incluindo dados como potência da transmissão, localização, sensibilidade do receptor e informações topográficas. Logo que os responsáveis pela interferência sejam descobertos, uma comunicação deve ser estabelecida entre essas redes para diminuí-la e melhorar a coexistência.

2. **Reconfiguração de rede autônoma:** Em alguns casos, dependendo da quantidade de tráfego gerado na rede e dos recursos disponíveis, uma solução independente para resolver o problema de interferência pode ser tomada e o gerente da rede pode reconfigurá-la para operar em uma nova faixa de frequência, deixando a frequência antiga com recursos suficientes. Essa é a chamada reconfiguração de rede autônoma, que modifica alguns ou todos os parâmetros de transmissão.
3. **Reconfiguração de rede negociada:** Se a reconfiguração de rede autônoma não for capaz de resolver o problema de interferência, a rede precisa negociar o compartilhamento de espectro diretamente com a rede que está causando esse problema.

Para isso, as duas redes devem estabelecer uma conexão direta e decidir em conjunto quais são os melhores parâmetros de configuração necessários para diminuir a interferência. Mesmo que um algoritmo padronizado seja capaz de sugerir quais são os melhores valores a serem utilizados, ambas as redes devem concordar com

a solução e serem monitoradas de forma com que haja a possibilidade de avaliar a efetividade negociação realizada.

3 Problema de atribuição de potência e de canais de transmissão em redes cognitivas

Em redes sem fio, os Pontos de Acesso são as entidades responsáveis pela transmissão de dados, troca de informações e conexão entre os dispositivos móveis que utilizam a rede. Os dados transmitidos nesse tipo de rede são codificados em ondas de rádio especificadas pelo padrão IEEE 802.11 que trafegam, no Brasil, dentro dos espectros de frequência de 2.4 GHz e 5 GHz.

Alguns parâmetros são necessários para que um AP funcione corretamente e co-exista com outros dispositivos na mesma região, como a potência e o canal de transmissão. No entanto, para minimizar a interferência causada entre os Pontos de Acesso e otimizar a qualidade do serviço oferecido para os clientes conectados, os valores desses atributos devem ser definidos de forma calculada, analisando o ambiente e as demais entidades da rede.

As faixas de frequência de 2.4 GHz e 5 GHz utilizadas pelos pelo padrão 802.11 são subdivididas em canais que definem o meio de transmissão de dados das redes sem fio. Existem 14 canais de operação disponíveis em diferentes regiões do mundo, sendo que apenas três deles não se sobrepõem, isto é, não causam interferência mútua, os chamados canais ortogonais. Os canais ortogonais disponíveis no Brasil e licenciados pela ANATEL são o 1, 6 e 11, dentro do intervalo de 1 a 11, como pode ser visualizado na Figura 3.1.

Quando dispositivos operam em canais sobrepostos, quanto maior a proximidade dos mesmos maior é o peso da interferência entre eles. Para que não haja interferência em uma rede sem fio, os APs que a compõem devem ser preferencialmente configurados para operar sobre canais não sobrepostos, porém o reuso de canais acaba sendo necessário devido à escassez dos mesmos.

Deve-se destacar que a interferência entre dois ou mais APs que operam em canais não ortogonais está relacionada também à potência de transmissão dos mesmos, já que, se o sinal de transmissão de um não incide no outro, não ocorre interferência. Além

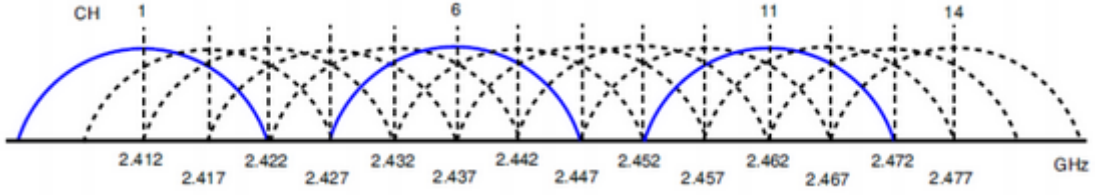


Figura 3.1: Canais de transmissão no espectro de 2.4 GHz (Chieochan et al, 2010)

disso, uma vez que a potência do sinal cai não linearmente com a distância do emissor, a interferência está relacionada também com a distância entre estes APs.

A potência de transmissão de um AP é a métrica que define o alcance do sinal transmitido por ele. Todavia, caso o Ponto de Acesso não esteja operando em um canal ortogonal não ocupado, sua intensidade deve ser controlada com o intuito de atingir um valor ótimo, pois se o valor for muito baixo os dispositivos conectados deixarão de ser atendidos, e se for muito alto seu sinal poderá alcançar um número maior de Pontos de Acesso vizinhos operando no mesmo canal ou em canais sobrepostos, reduzindo a qualidade da transmissão destes para seus respectivos clientes e aumentando a interferência da rede.

3.1 Representação do problema

Seja $G = (V, E)$ um grafo orientado ponderado tal que $V = A \cup D$ é o conjunto de vértices representado por APs (conjunto A) e clientes (conjunto D) em uma rede sem fio e E é o conjunto de arcos $(i, j) \mid i, j \in A$.

Seja p_i a potência de transmissão associada ao nó $i \in A$, que define o conjunto de vértices de V que são alcançados pelo seu sinal de transmissão. O conjunto E é tal que, se dois vértices $i, j \in A$ operam em canais que geram interferência entre si e p_i é suficiente para alcançar o nó j , então o arco $(i, j) \in E$.

Seja $c : E \mapsto \mathbb{R}$ uma função que define, para todo arco $(i, j) \in E$, a interferência da transmissão de i em j . O custo c , atribuído como peso de cada arco (i, j) do grafo, pode ser calculado como o valor excedente do sinal incidido de i sobre j com relação à quantidade mínima necessária para alcançar o último de acordo com sua localização, isto

é, o valor da potência p_i subtraído do quadrado da distância euclidiana d_{ij} entre os APs.

O problema de atribuição de potência e de canais de transmissão em redes cognitivas consiste em, dado o grafo G e um conjunto C de canais de transmissão, atribuir a cada vértice de A uma potência de transmissão e um canal tal que minimize o somatório de $c(i, j)$ garantindo a conexidade dos vértices de D aos vértices de A da rede.

Neste trabalho, considera-se uma rede heterogênea, com transmissão de dados de forma assíncrona e em tempo real². Além disso, possui estrutura pré-estabelecida e imóvel, onde um sistema controlador deve operar de forma centralizada, gerenciando todos os Pontos de Acesso da rede no que tange a alocação de canais e de potências de transmissão.

3.2 Trabalhos relacionados

Para Balbi et al (2012), problemas de escala começaram a surgir com a crescente popularização das redes sem fio IEEE 802.11, citando como uma das dificuldades a gerência de cada Ponto de Acesso individualmente quando estes se encontram em um mesmo grupo.

É necessário evitar a interferência excessiva na rede para que bons índices de transmissão sejam atingidos pelos dispositivos pertencentes. A princípio, os sistemas padrão de gerenciamento dos APs disponibilizam a escolha do canal de operação utilizado pelos mesmos, mas isso pode não ser suficiente em um ambiente dinâmico, onde novas redes entram em operação e outras saem, tornando a escolha obsoleta.

A partir deste cenário, foi criado, na Universidade Federal Fluminense - UFF, com o apoio da Rede Nacional de Ensino e Pesquisa (RNP), o projeto denominado SCIFI (Sistema de Controle Inteligente para redes sem Fio), com o objetivo de funcionar como um controlador central de Pontos de Acesso que operam através de um sistema operacional de código aberto (ex.: OpenWrt), coordenando e configurando os APs de forma automática.

A Figura 3.2 apresenta a organização de uma rede gerenciada pelo SCIFI. Note que o módulo controlador opera através de um equipamento *switch*, também conectado

²Em sua aplicação em ambientes reais, clientes e APs podem ser inseridos ou removidos da rede a qualquer momento.

aos APs. Isto possibilita a obtenção de informações da rede e permite que o sistema aja sobre a mesma.

Deve-se destacar que em uma rede sem fio pode haver alguns APs não controlados pelo sistema. Estes APs deverão, portanto, ser considerados como elementos cujos canais e potências de transmissão não são passivos de controle.

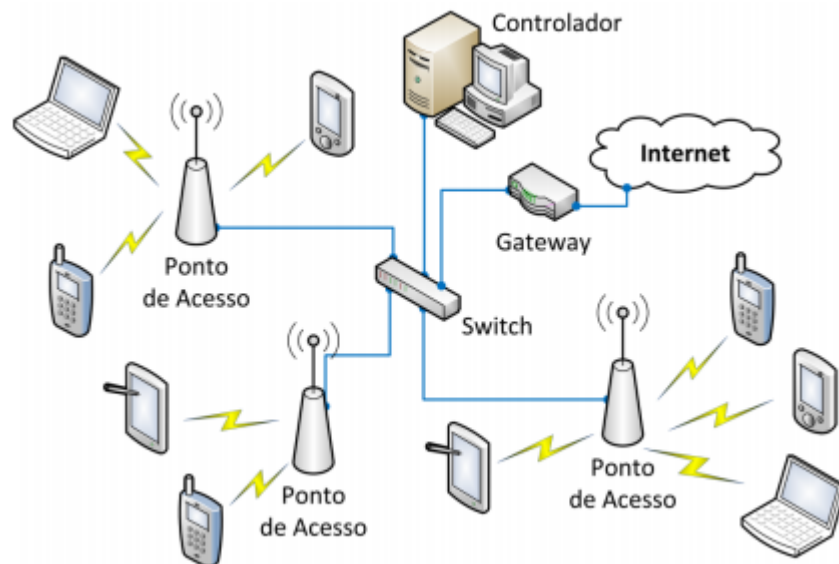


Figura 3.2: Exemplo de rede sem fio controlada pelo SCIFI (Balbi et al, 2012)

Através do sistema SCIFI é realizada a seleção dinâmica dos canais de operação dos APs gerenciados. O objetivo é reduzir a interferência causada em Pontos de Acesso da rede, sendo eles controlados ou não.

3.2.1 Algoritmo SCIFI

O SCIFI (Balbi et al, 2012) realiza a seleção automática de canais dos APs em uma rede no padrão IEEE 802.11, facilitando o processo de gerência de redes sem fio. Para isso, informações adquiridas após uma varredura espectral realizada pelos Pontos de Acesso são enviadas para um controlador central (SCIFI), que as utiliza para configurar os canais de operação de forma que a interferência seja reduzida.

O Algoritmo 1 apresenta o funcionamento do SCIFI. Seu mecanismo de seleção de canais interpreta e modela a rede como um grafo de interferências, também conhecido como grafo de conflitos. A partir da modelagem da rede como um grafo, o problema de

alocação de canais pode ser resolvido através da redução à um problema de coloração de vértices. No problema da coloração em grafos, tem-se como objetivo colorir todos os vértices de um grafo, mas com a restrição de que vértices adjacentes não possuam a mesma cor.

Uma rede no padrão IEEE 802.11 pode ser representada por um digrafo ponderado em que os vértices indicam os APs, as cores são associadas aos diferentes canais de operação e, por fim, os pesos atribuídos aos arcos indicam a interferência causada entre os Pontos de Acesso.

Uma vez que existem APs não controlados, o SCIFI fixa as cores atribuídas a estes nós e procede a atribuição de cores relaxando a restrição do problema caso a interferência seja verificada. De qualquer forma, o SCIFI também considera estes vértices ao procurar atribuir cores aos APs controlados, já que estes podem gerar interferência.

A heurística utilizada no SCIFI para a alocação de canais se baseia no Grau de Saturação ou DSATUR, definido por Brélaz (1979), que consiste em determinar quantas cores adjacentes cada vértice do grafo possui (grau de saturação do vértice) e ordenar a coloração do maior para o menor grau de saturação. Em caso de empate, o AP que possuir o maior número de clientes associados a ele tem prioridade na escolha do canal e, em caso de nova ocorrência, a prioridade é dada aos APs de menor endereço.

O algoritmo de controle de potência utilizado no SCIFI é executado após o algoritmo de seleção de canal e detecta a interferência dos APs a partir de varreduras espectrais realizadas em cada um. As potências de transmissão dos Pontos de Acesso que geram interferência mútua na rede são reduzidas gradualmente até que o sinal de transmissão deixe de ser detectado pelo vizinho ou que a potência mínima previamente configurada seja atingida.

O algoritmo também determina que os APs que deixem de ser identificados pelos demais tenham suas potências aumentadas gradativamente e, caso um AP seja o único em um canal, sua potência é aumentada ao máximo.

Algoritmo 1: Pseudocódigo do sistema SCIFI

Entrada: Instância I **Saída:** Solução final S^*

```
1 início
2    $S^* \leftarrow \infty$ ;
3    $S_{aux} \leftarrow SeleccionaCanais(I)$ ;
4   se ( $S_{aux}.custo \leq S^*.custo$ ) então
5      $S^* \leftarrow S_{aux}$ ;
6   fim se
7    $S^* \leftarrow ControlaPotencias(S_{aux})$ ;
8   retorna  $S^*$ ;
9 fim
```

4 Abordagens desenvolvidas

Com o objetivo de reduzir a interferência gerada dentro na rede e garantir métricas de *QoS* para os dispositivos conectados, abordagens heurísticas foram desenvolvidas para identificar o melhor canal e potência de transmissão a serem definidos para cada AP. Tais heurísticas foram combinadas em um algoritmo GRASP.

Neste trabalho, uma solução é representada por dois vetores de tamanho dado pelo número de APs: no que se refere a alocação de canais o vetor X , onde cada elemento de X indica qual o canal atribuído ao AP correspondente; já para a atribuição de potência, um vetor Y indica, para cada posição, a respectiva potência de transmissão atribuída ao AP.

As abordagens desenvolvidas neste trabalho foram projetadas levando-se em conta as técnicas colaborativas para implementação da coexistência e os desafios de descoberta da interferência/entidades interferentes e da reconfiguração da rede, descritos na Seção 2.1.

Para a aplicação dos algoritmos propostos, considerou-se a rede como um grafo, onde os nós ou vértices definem os Pontos de Acesso integrantes. Os dispositivos clientes são associados aos APs mais próximos, ou seja, de menor distância euclidiana. Em seguida, o procedimento inicial para controle de potências é executado, definindo a potência de cada AP com a intensidade mínima necessária para alcançar e servir todos os clientes conectados.

A última etapa de pré-processamento realizada antes de iniciar os procedimentos heurísticos é a criação dos arcos do grafo a partir do cálculo da interferência gerada na rede, em que os APs que sofrem incidência são interligados aos interferentes, sendo o peso do arco dado pelo o valor da interferência gerada.

4.1 Heurística de construção

Segundo Cormen (2009), algoritmos gulosos são heurísticas de solução de problemas que possuem a característica de selecionar a melhor escolha local de uma lista de candidatos a cada passo para adicioná-lo ao conjunto solução, minimizando a função objetivo naquele momento.

Em um algoritmo guloso construtivo, inicia-se uma solução como vazia, que será construída iterativamente escolhendo-se elementos candidatos a entrar na solução até que esta esteja completa. A cada iteração, uma heurística local define o critério de seleção de um novo elemento para ser inserido ao conjunto da solução, que uma vez parte desse conjunto, nunca mais será substituído por outro, ou seja, não há retrocesso (*backtracking*) de decisões já tomadas. Normalmente, algoritmos gulosos são determinísticos e possuem complexidade reduzida (Talbi, 2009).

Heurísticas de construção são algoritmos que encontram uma solução inicial para o problema, geralmente de forma gulosa, satisfazendo as regras e objetivos definidos conforme o problema proposto. Esse tipo de abordagem é um caminho rápido para se encontrar soluções viáveis, porém não tão boas quando comparadas com os resultados de outras heurísticas ditas de intensificação.

Deste modo, algoritmos de construção são principalmente utilizados quando a solução do problema deve ser razoavelmente boa e encontrada rapidamente ou para gerar soluções iniciais que serão passadas como entrada para heurísticas de refinamento (Schneider e Kirkpatrick, 2007).

Schneider e Kirkpatrick (2007) também descrevem a divisão em etapas de uma heurística de construção, começando com a fase de inicialização da solução, continuada pelo laço sobre a seleção de elementos até que as condições pré-estabelecidas sejam atendidas.

O Algoritmo 2 descreve o procedimento básico da heurística de construção desenvolvida. O algoritmo recebe como entrada a instância do problema, uma função f , usada para definir o critério a ser utilizado para ordenar os elementos candidatos a entrar na solução, e, por fim, um fator de aleatoriedade $\alpha \in [0, 1]$, que determina quão guloso é o algoritmo.

Inicialmente, na linha 3, as componentes conexas do grafo que representa a instância I são encontradas e, para cada uma das componentes, uma lista L de elementos candidatos a entrar na solução é criada e ordenada conforme a função critério f . A solução é iniciada como um conjunto vazio, indicando que nenhum elemento foi o escolhido. O laço compreendido entre as linhas 7 e 11 definem o processo iterativo de construção propriamente dita da solução.

Observe que, na linha 8 do Algoritmo 2, um índice i é definido aleatoriamente entre zero e o valor dado pelo tamanho da lista L multiplicado pelo valor de α . Assim, quando $\alpha = 0$, o melhor elemento (posição zero da lista L) segundo a função f é selecionado e inserido na solução (linha 9). Após a inclusão de um elemento de L na solução, um procedimento de atualização da lista é chamado de forma a remover os elementos que não podem mais entrar na solução após a escolha do último elemento inserido.

Algoritmo 2: Modelo de algoritmo construtivo

Entrada: Instância I , critério f , aleatoriedade $\alpha \in [0, 1]$
Saída: Solução final S

```

1 início
2    $S \leftarrow \emptyset$ ;
3    $C \leftarrow \text{ComponentesConexas}(I)$ ;
4   enquanto  $C \neq \emptyset$  faça
5      $c \leftarrow \text{ProximaComponente}(C)$ ;
6      $L \leftarrow \text{Ordena}(c, f)$ ;
7     repita
8        $i \leftarrow \text{RandomRange}(0, \alpha \times |L|)$ ;
9        $S \leftarrow S \cup L[i]$ ;
10       $L \leftarrow \text{AtualizaLista}(S, L)$ ;
11    até  $|L| = 0$ ;
12     $C \leftarrow C \setminus \{c\}$ ;
13  fim enquanto
14  retorna  $S$ ;
15 fim
```

Para a construção da solução inicial no algoritmo proposto, um grafo de conflito é gerado. Para tanto, a partir da localização de cada AP, calcula-se a distância euclidiana entre pares de APs. Para cada AP i , considera-se a menor distância entre i e os demais APs. O grafo de conflito é gerado tomando-se como limiar de proximidade (o que possivelmente pode gerar interferência) a média das menores distâncias entre os pares de APs, denotada por ρ . Assim, para cada par (i, j) , insere-se a aresta se $d_{i,j} \leq \rho$, onde $d_{i,j}$ é a

distância euclidiana entre os APs i e j .

A ideia principal para usar uma medida de limiar de proximidade entre APs é permitir que o algoritmo identifique as regiões da rede onde há a maior concentração de APs. Isto é possível tomando-se como entrada o grafo de conflitos gerados por esta métrica e identificando-se as componentes conexas neste grafo.

Espera-se que nestas regiões sejam observadas maiores taxas de interferência de sinal entre APs, o que requer um tratamento especial do algoritmo no início do processo de alocação de canais, em vez de considerar a rede como um todo. As componentes conexas do grafo podem ser obtidas facilmente a partir da aplicação de algoritmos de busca em profundidade no grafo.

Para cada uma das componentes conexas encontradas, uma lista L de candidatos é criada e preenchida com os Pontos de Acesso de maneira ordenada decrescentemente pela prioridade P de cada um, cujo cálculo é baseado na interferência sofrida e na quantidade de clientes conectados, o que pode ser verificado na Equação 4.1,

$$P_i = (\beta \times \sum_{j=1}^{|A|} I(i, j)) + (\lambda \times C_i); \beta + \lambda = 1; i, j \in A \quad (4.1)$$

onde β representa o peso do somatório das interferências sofridas por um AP pelos vizinhos, λ indica o peso do número de dispositivos associados ao Ponto de Acesso, a função I calcula o valor da interferência sofrida por um AP i pelo interferente j e C é a quantidade de clientes conectados ao Ponto de Acesso.

A partir da lista L de cada componente conexa, uma solução para o problema de alocação de canais é construída conforme o Algoritmo 2 de forma gulosa randomizada reativa, o que significa que o fator de aleatoriedade α do algoritmo é reajustado com o decorrer das iterações.

O comportamento reativo do algoritmo é obtido através da utilização de vetores de controle da qualidade e probabilidade de uso dos possíveis valores de α como fator de aleatoriedade na seleção de elementos da lista L de candidatos. Seu valor está contido no intervalo $[0, 1]$, com saltos de 0.1, ou seja, consideram-se 11 valores distintos de α . O cálculo da qualidade dos valores do fator α , que consequentemente aumenta sua probabilidade, é realizado a cada bloco de iterações cujo tamanho é passado por parâmetro, e

está relacionado proporcionalmente ao seu uso. A cada iteração do algoritmo, o valor de α a ser utilizado naquele momento é selecionado de forma aleatória, porém ponderada, havendo a tendência de selecionar valores com maior probabilidade.

Durante o processo de construção, cada AP selecionado tem seu canal alocado de forma a reduzir a interferência, considerando os canais dos APs vizinhos e priorizando a escolha de um canal ortogonal livre. Caso não seja possível a alocação de um canal não sobreposto para um dado AP, o que significa que todos os canais ortogonais já estão em uso por nós adjacentes, o algoritmo verifica o canal menos interferente através do cálculo da previsão do somatório das incidências sofridas para cada um dos canais restantes.

4.2 Heurística de busca local

Para Talbi (2009), a busca local é a metaheurística mais antiga e simples, que começa com uma solução dada inicialmente e, a cada iteração do algoritmo, realiza a troca da solução atual por outra solução vizinha que melhora a função objetivo. Para uma dada solução inicial S e um dado movimento M , definido na estrutura que representa S , a vizinhança de S , denotada $N(S, M)$, é o conjunto de todas as soluções obtidas a partir de S pela aplicação do movimento M .

O critério de parada de uma heurística de busca local é alcançado após uma varredura de todo o conjunto $N(S, M)$. O melhor vizinho é retornado como saída do algoritmo. A Figura 4.1 ilustra a ideia de uma heurística de busca local para um problema de minimização. Observe que, a partir da solução inicial, novas soluções foram verificadas até que uma solução seja obtida sem que nenhum vizinho seja melhor, o que define a solução final como mínimo local.

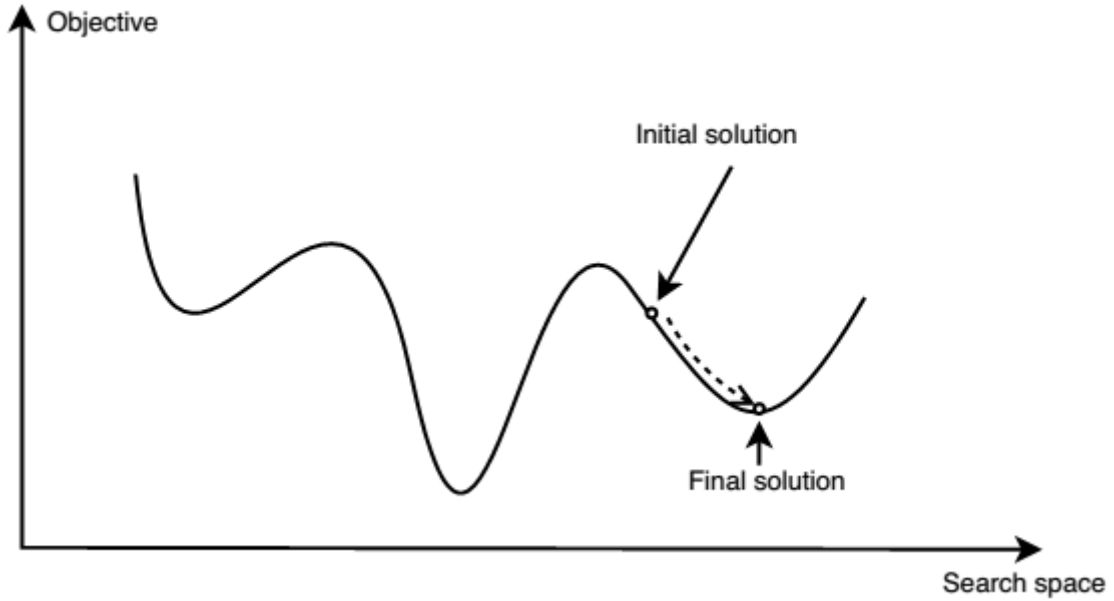


Figura 4.1: Comportamento da heurística de busca local em um espaço de busca (Talbi, 2009)

O Algoritmo 3 ilustra o funcionamento de uma heurística de busca local de caráter geral. O algoritmo recebe como entrada uma solução inicial S_0 e uma estrutura de vizinhança definida pelo tipo de movimento M .

A partir de uma solução inicial obtida pelo algoritmo de construção, a heurística de busca local é aplicada para tentar encontrar uma solução de melhor qualidade através de um tratamento no vetor de canais X . A estrutura de movimento utilizada foi a troca do canal de cada AP pelo canal de cada um dos demais. Assim, a solução sofre sucessivas trocas para soluções vizinhas, de modo que todos os pares de APs do grafo tenham executado todas as possibilidades existentes de alternância de seus canais, o que é definido no Algoritmo 3 pelo laço compreendido entre as linhas 5 e 11.

A cada troca realizada, o somatório das interferências na rede é recalculado e verifica-se a qualidade da solução gerada quando comparada à solução original (linha 8). A melhor troca identificada durante as iterações é armazenada e dada como resultado ao final do procedimento (linha 13).

Algoritmo 3: Pseudocódigo de heurística de busca local

Entrada: Solução inicial S_0 , movimento M
Saída: Solução final S^*

```

1 início
2    $S^* \leftarrow S_0$ ;
3   enquanto existirem soluções vizinhas e não houver melhoria faça
4      $N \leftarrow \text{ConstroiVizinhanca}(S_0, M)$ ;
5     enquanto  $N \neq \emptyset$  faça
6        $S_{aux} \leftarrow \text{ProximoVizinho}(N)$ ;
7        $N \leftarrow N \setminus \{S_{aux}\}$ ;
8       se  $(S_{aux}.custo \leq S^*.custo)$  então
9          $S^* \leftarrow S_{aux}$ ;
10      fim se
11    fim enqto
12  fim enqto
13  retorna  $S^*$ ;
14 fim

```

4.3 Algoritmo de refinamento de potências

Existem duas técnicas que podem ser utilizadas para reduzir a interferência causada dentro de uma rede sem fio controlada: seleção de canais e controle de potência. As heurísticas abordadas anteriormente tratam exclusivamente da alocação de canais para os Pontos de Acesso e já apresentam resultados positivos. No entanto, também é necessário o ajuste das potências de transmissão para atingir índices ainda menores de interferência na rede.

Para permitir que a rede apresente um comportamento cognitivo considerando também o conhecimento a respeito das potências de transmissão dos seus nós, foi desenvolvido um procedimento (Algoritmo 4) para melhor calibrar a potência de transmissão, denotado neste trabalho como refinamento de potências.

A ideia principal do algoritmo é realizar reduções sucessivas de níveis de potência de transmissão sequencialmente, sempre verificando a qualidade do sinal para os clientes da rede e provendo, eventualmente, a realocação destes a novos Pontos de Acesso, quando necessário.

Após qualquer redução da potência de algum AP, caso uma melhora no nível de interferência seja constatada, a modificação é mantida e os dispositivos móveis consequentemente desprovidos de sinal são identificados e reconectados em outros APs mais próximos a partir de um processo executado de forma gulosa e aleatória.

No processo de redução de potências, para a realocação de um dado cliente a um novo AP, a exemplo do que fora apresentado para a heurística de busca local, uma lista de candidatos é composta pelos Pontos de Acesso, neste caso, ordenada pela menor distância com relação a esse cliente. Para cada cliente descoberto após a redução da potência de transmissão de um determinado AP, o Ponto de Acesso que deve se conectar a ele é selecionado randomicamente por um algoritmo guloso sobre a lista restrita de APs candidatos gerada através da utilização de um fator α de aleatoriedade.

Quando a tentativa de redução de potência de um AP resultar em aumento da interferência, a redução é descartada e um outro AP é selecionado para redução. Concluído o refinamento para todos os APs, a solução retornada é composta pela lista de nós do grafo com as potências devidamente ajustadas.

Algoritmo 4: Algoritmo de refinamento de potências

Entrada: Solução inicial S_0 , taxa de redução r

Saída: Solução final S^*

```

1 início
2    $S^* \leftarrow S_0$ ;
3    $A_{aux} \leftarrow A$ ;
4   enquanto  $A_{aux} \neq \emptyset$  faça
5      $ap \leftarrow ProximoAP(A_{aux})$ ;
6      $S_{aux} \leftarrow ReduzPotencia(S^*, ap, r)$ ;
7      $RealocaClientes(S_{aux}, ap)$ ;
8     se  $(S_{aux}.custo \leq S^*.custo)$  então
9        $S^* \leftarrow S_{aux}$ ;
10    fim se
11     $A_{aux} \leftarrow A_{aux} \setminus \{ap\}$ ;
12  fim enqto
13  retorna  $S^*$ ;
14 fim
  
```

4.4 Abordagens GRASP propostas

Em sua tradução direta, GRASP é um procedimento de busca guloso, aleatório e adaptativo, uma metaheurística utilizada para resolver problemas de otimização combinatória. Foi primeiramente introduzido por Feo e Resende (1989) e seu algoritmo baseia-se simplesmente na combinação de outras duas heurísticas conhecidas: construção e busca local, vide exemplo no Algoritmo 5.

Cada iteração do GRASP executa um método de construção de uma solução inicial, gerada através da utilização de um algoritmo guloso randomizado e adaptativo, seguido por uma busca local que é aplicada sobre a solução construída anteriormente. Esse processo é repetido até o número de iterações dado em sua entrada ser atingido e a melhor solução encontrada dentre todas é considerada como resultado final (Talbi, 2009).

É importante observar que o algoritmo guloso utilizado na etapa de construção deve ser aleatório para ser capaz de gerar diferentes soluções a cada execução, pois de outra forma o procedimento de busca local poderia ser utilizado apenas uma única vez.

Algoritmo 5: Modelo da metaheurística GRASP desenvolvida

Entrada: Instância I , Quantidade de iterações k , bloco de iterações b
Saída: Solução S^*

```

1 início
2    $S^* \leftarrow \infty$ ;
3    $PreProcessamento(I)$ ;
4   para  $i \leftarrow 0$  até  $k$  faça
5      $\alpha \leftarrow SelecionaFatorAleatoriedade()$ ;
6      $S \leftarrow Construtivo(I, \alpha)$ ;
7      $S \leftarrow BuscaLocal(S)$ ;
8     se  $S < S^*$  então
9        $S^* \leftarrow S$ ;
10    fim se
11     $S^* \leftarrow RefinamentoPotencias(S)$ ;
12    se  $i \% b = 0$  então
13       $RecalculaProbabilidadesAlpha()$ ;
14    fim se
15  fim para
16  retorna  $S^*$ ;
17 fim

```

Apesar de sua simplicidade e facilidade de implementação, a metaheurística GRASP demonstra resultados eficazes e muitas vezes os melhores conhecidos quando aplicada em problemas de otimização combinatória da literatura (Mateus et al, 2010). Baseando-se nessa constatação, as heurísticas apresentadas anteriormente foram organizadas de maneira a gerar três diferentes combinações de procedimentos do tipo GRASP:

- **GRASP 1:** Construção + busca local;
- **GRASP 2:** Construção + refinamento de potências;
- **GRASP 3:** Construção + refinamento de potências + busca local.

A combinação desejada a ser executada pelo programa principal é definida como um de seus parâmetros de entrada. Cada algoritmo GRASP realiza k iterações dos procedimentos combinados em sequência, sendo essa sequência de heurísticas definidas executadas a cada iteração. A cada passo, a solução gerada é comparada com a melhor encontrada até o momento e, caso seja mais satisfatória no intuito de minimizar a função objetivo do problema, ocupa o posto de melhor solução.

Também deve-se observar que, a cada iteração do algoritmo, o processo de construção da solução inicial é realizado novamente, o que significa que as etapas de alocação dos clientes e seleção das componentes conexas do grafo são executadas k vezes.

Após k iterações, a alocação de canais e de potências de transmissão dos Pontos de Acesso da rede está concluída e o valor da função objetivo, dada pelo somatório das interferências mútuas dos APs que a compõem a rede, é retornado com os vetores de canais X e de potências Y .

5 Experimentos e resultados

Neste capítulo serão demonstrados os resultados obtidos após a realização de testes construídos a partir de instâncias criadas pelo programa gerador de instâncias que foi desenvolvido e através da execução dos algoritmos descritos nas seções anteriores.

As instâncias com os dados necessários para execução do algoritmo são compostas simplesmente de pontos no plano cartesiano que definem a posição geográfica dos Pontos de Acesso da rede controlada e dos dispositivos móveis existentes naquele momento. Para diferenciar as entidades representadas pelos pontos, a primeira linha do arquivo de entrada é composta pela quantidade de APs e pela quantidade de clientes que serão apontados nas linhas posteriores.

No intuito de facilitar a execução dos testes de aferição do algoritmo, um programa complementar foi desenvolvido com a função de gerar instâncias para o programa principal de forma aleatória. A linguagem de programação utilizada para sua implementação foi a C++, com auxílio da IDE Code::Blocks. Seus parâmetros de entrada são definidos pelo nome do arquivo de instância a ser gerado e a quantidade desejada de APs controlados e clientes na rede. A distribuição dos pontos no plano, ilustrada na Figura 5.1, é realizada de forma uniforme e limitada em uma área de tamanho 500 x 500 dividida em octantes (sendo 2 no eixo x e 4 no eixo y).

O algoritmo para solução do problema de atribuição de potência e canais de transmissão em redes cognitivas foi implementado utilizando a linguagem de programação Java, com suporte e compilação pela IDE NetBeans, sob o sistema operacional Microsoft Windows 7 Ultimate 64 *bits* executado em um computador com processador Intel Core i7 4790K 4.5 GHz e 16 GB de memória RAM.

Também foi implementado, em linguagem Java, um algoritmo para simular o funcionamento do SCIFI, composto de suas funcionalidades principais de alocação de canais, realizada através da coloração dos vértices do grafo, e controle de potências, executada de forma a reduzi-las gradualmente. Portanto, esse algoritmo é preparado para interpretar as instâncias no formato utilizado já descritas anteriormente e seu resultado indica o valor

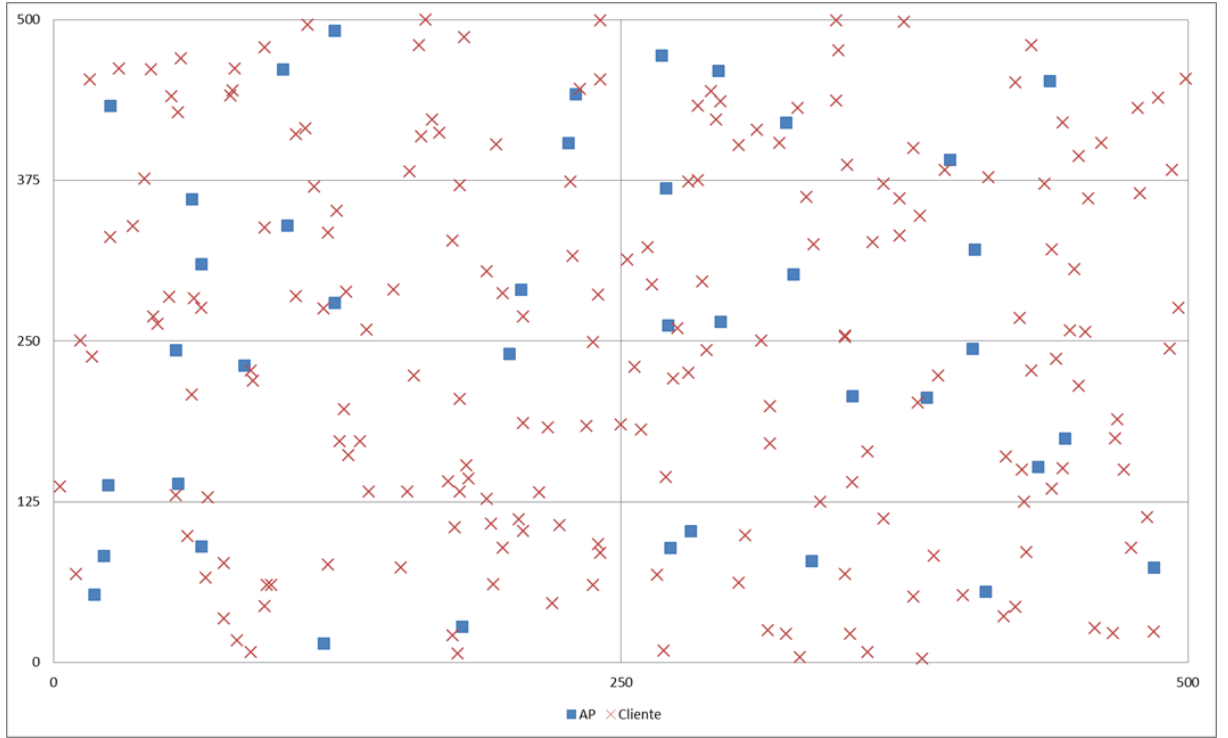


Figura 5.1: Distribuição de APs e clientes em octantes a partir de uma instância do problema

da interferência acumulada na rede, assim como o código proposto.

Para a realização e análise dos experimentos, foi desenvolvido um programa independente que se comporta como um *script*, efetuando múltiplas execuções do programa principal e construindo uma tabela comparativa dos resultados obtidos por cada algoritmo GRASP para cada uma das instâncias utilizadas. Os tempos apresentados nas tabelas a seguir estão definidos em segundos.

5.1 Comparação entre heurísticas de construção e a literatura

Para fins de comparação com a abordagem da literatura, inicialmente procedeu-se um experimento considerando apenas a heurística de construção proposta. O objetivo principal desta comparação é avaliar o impacto da abordagem que considera a coloração do grafo iniciada após a construção das componentes conexas.

Uma vez que o SCIFI consiste em uma heurística de construção muito similar ao

algoritmo proposto, as contribuições do algoritmo podem ser verificadas sem a influência das heurísticas de busca local e de refinamento de potências.

Assim, as heurísticas de construção foram executadas para cada instância em duas diferentes formas: gulosas e gulosas randomizadas reativas. A Tabela 5.1 exibe os resultados gerados pelos algoritmos de construção guloso (colunas 3 e 4), o guloso randomizado reativo (colunas 5 e 6) em comparação com as soluções produzidas pelo algoritmo SCIFI (colunas 7 e 8) para as instâncias. Os valores destacados em negrito indicam os melhores resultados.

Com o objetivo de realizar os testes comparativos, foi utilizado um conjunto de 30 instâncias, sendo as 9 primeiras consideradas pequenas (10 APs e 50 clientes); as instâncias de 10 a 21 consideradas médias (40 APs e 200 clientes); e as 9 últimas, numeradas de 22 a 30, consideradas grandes (70 APs e 350 clientes).

Nos resultados apresentados na Tabela 5.1, a primeira coluna indica o número da instância utilizada, a segunda coluna mostra o valor da melhor média das soluções encontrada dentre todas as execuções realizadas para os diferentes algoritmos apresentados.

As demais colunas da tabela fornecem detalhadamente as métricas resultantes das execuções de cada abordagem, como o desvio relativo da média das soluções do algoritmo com relação à melhor média para a instância (coluna 2), e o tempo médio de execução, dado em segundos.

A princípio, a diferença percentual média (*gap*) das soluções poderia ser calculada conforme a Equação 5.1 descrita a seguir, onde $Media(a)$ é o valor médio das soluções obtidas pelo algoritmo a e b é o valor da melhor média obtida dentre todos os algoritmos comparados (*best*):

$$gap(a) = \frac{Media(a) - b}{b} \times 100 \quad (5.1)$$

No entanto, casos em que $b = 0$, ou seja, a melhor solução encontrada possui custo igual a 0, o cálculo do *gap* realiza uma divisão por zero causando um erro matemático. Para resolver esse problema, Vallada et al (2008) propõem o uso da métrica denominada Índice de Desvio Relativo (RDI, do inglês *Relative Deviation Index*), cujo cálculo é apresentado pela Equação 5.2 e define a qualidade de uma determinada solução:

$$RDI(a) = \frac{Media(a) - b}{c - b} \times 100 \quad (5.2)$$

onde c é o valor da pior solução dentre os algoritmos utilizados, e nesse caso específico, a pior média encontrada. Note que $RDI(a) \in [0, 100]$ e quanto menor o valor resultante, maior é a qualidade da solução do algoritmo a .

Tabela 5.1: Comparação entre os resultados dos algoritmos de construção da abordagem proposta com o SCIFI

Instância	Melhor	Guloso		Reativo		SCIFI	
		<i>RDI</i>	Tempo	<i>RDI</i>	Tempo	<i>RDI</i>	Tempo
I1_10AP_50C	0,0	0,00	0,282	0,00	0,141	0,00	0,114
I2_10AP_50C	0,0	0,00	0,129	0,00	0,140	0,00	0,128
I3_10AP_50C	0,0	0,00	0,112	0,00	0,131	100,00	0,128
I4_10AP_50C	88,3	58,89	0,115	0,00	0,089	100,00	0,112
I5_10AP_50C	103,0	57,45	0,111	0,00	0,092	100,00	0,144
I6_10AP_50C	42,7	49,23	0,111	0,00	0,087	100,00	0,112
I7_10AP_50C	0,0	0,00	0,098	0,00	0,115	0,00	0,159
I8_10AP_50C	82,0	93,48	0,114	0,00	0,115	100,00	0,159
I9_10AP_50C	49,1	24,05	0,124	0,00	0,104	100,00	0,127
I10_40AP_200C	222,4	18,26	0,150	0,00	0,202	100,00	0,127
I11_40AP_200C	89,0	22,66	0,155	0,00	0,182	100,00	0,110
I12_40AP_200C	158,8	28,80	0,145	0,00	0,198	100,00	0,127
I13_40AP_200C	55,4	0,68	0,164	0,00	0,193	100,00	0,128
I14_40AP_200C	80,0	0,00	0,141	0,00	0,180	100,00	0,237
I15_40AP_200C	131,4	0,72	0,145	0,00	0,187	100,00	0,143
I16_40AP_200C	48,6	4,46	0,141	0,00	0,206	100,00	0,143
I17_40AP_200C	29,1	21,65	0,141	0,00	0,198	100,00	0,143
I18_40AP_200C	109,8	3,70	0,152	0,00	0,191	100,00	0,114
I19_40AP_200C	94,6	18,59	0,141	0,00	0,184	100,00	0,116
I20_40AP_200C	305,3	34,83	0,148	0,00	0,174	100,00	0,114
I21_40AP_200C	246,6	11,51	0,157	0,00	0,199	100,00	0,105
I22_70AP_350C	350,0	5,09	0,263	0,00	0,350	100,00	0,134
I23_70AP_350C	150,4	2,02	0,269	0,00	0,310	100,00	0,159
I24_70AP_350C	200,2	6,92	0,268	0,00	0,378	100,00	0,284
I25_70AP_350C	107,7	0,71	0,260	0,00	0,337	100,00	0,175
I26_70AP_350C	147,0	2,85	0,294	0,00	0,379	100,00	0,206
I27_70AP_350C	240,7	1,02	0,234	0,00	0,366	100,00	0,237
I28_70AP_350C	72,7	12,23	0,310	0,00	0,357	100,00	0,331
I29_70AP_350C	92,9	7,70	0,251	0,00	0,290	100,00	0,190
I30_70AP_350C	194,9	1,71	0,283	0,00	0,305	100,00	0,221

Analisando a Tabela 5.1 é possível verificar que os resultados quanto à qualidade da solução apresentados pelo algoritmo proposto já em sua fase de construção são me-

lhores que aqueles obtidos pelo algoritmo SCIFI. Isto vem a confirmar a hipótese de que o processo de *clusterização* realizado pelo algoritmo através da identificação das componentes conexas do grafo, de fato, impacta na qualidade da alocação de canais com menor interferência.

O que explica este ganho na qualidade da atribuição de canais é o fato de a identificação das componentes conexas permitir ao algoritmo tratar de forma diferenciada as regiões da rede onde há uma maior incidência de APs próximos. Assim, divide-se o problema em subproblemas com as mesmas características. Isso pode ser evidenciado ao se analisar os resultados para as instâncias com maior número de APs.

Como era esperado, a heurística gulosa randomizada reativa obteve melhores resultados que versão puramente gulosa, já que esta explora melhor o espaço de soluções, embora requeira mais tempo para tal.

A Tabela 5.1 mostra ainda que o tempo médio de execução do SCIFI, quando comparado aos demandados pelos algoritmos propostos, indica que a abordagem da literatura tem melhor desempenho, já que para 17 das 30 instâncias testadas foi o mais rápido.

Entretanto, apesar das diferenças observadas, para o problema apresentado esta característica configura-se como de pouca relevância para a ordem de grandeza dos tempos de execução, pois em sua aplicação no ambiente real o algoritmo controlador é executado de forma centralizada e funciona como um processo em *background* e ininterrupto.

5.2 Comparação entre as abordagens GRASP propostas

Mesmo com os resultados satisfatórios já encontrados pelo algoritmo de construção, observou-se que ainda era possível melhorá-los com a aplicação das demais heurísticas abordadas neste trabalho, ou seja, as três versões de GRASP propostas.

A Tabela 5.2 mostra os resultados comparativos encontrados para o conjunto das 30 instâncias já apresentadas e os resultados destacados em negrito indicam a abordagem com melhor resultado.

Cada algoritmo foi submetido a 10 execuções, com cada uma realizando 500 iterações do GRASP proposto. O conjunto de parâmetros foi assim fixado: 0,5 como valor de β e λ , que definem a prioridade de escolha do AP que iniciará a alocação de canais; taxa de redução de potência no procedimento de refinamento de potências com valor de 80%; decaimento quadrático do sinal de transmissão com relação à distância dos APs; e multiplicador do limiar de proximidade igual a 100 para criação do grafo de conflito na etapa de construção da solução.

Tabela 5.2: Comparação entre os resultados das metaheurísticas GRASP desenvolvidas

Instância	Melhor	GRASP 1		GRASP 2		GRASP 3	
		<i>RDI</i>	Tempo	<i>RDI</i>	Tempo	<i>RDI</i>	Tempo
I1_10AP_50C	0,0	0,00	0,185	0,00	0,371	0,00	0,579
I2_10AP_50C	0,0	100,00	0,176	0,00	0,355	0,00	0,558
I3_10AP_50C	0,0	0,00	0,171	0,00	0,364	0,00	0,572
I4_10AP_50C	11,5	0,00	0,168	100,00	0,364	17,79	0,553
I5_10AP_50C	0,0	0,00	0,170	0,00	0,354	0,00	0,569
I6_10AP_50C	0,0	100,00	0,193	0,00	0,406	0,00	0,605
I7_10AP_50C	0,0	0,00	0,166	0,00	0,350	0,00	0,545
I8_10AP_50C	0,0	100,00	0,160	0,00	0,336	0,00	0,530
I9_10AP_50C	12,3	100,00	0,163	1,05	0,347	0,00	0,548
I10_40AP_200C	146,5	26,79	3,647	100,00	4,558	0,00	8,790
I11_40AP_200C	43,0	42,76	3,613	100,00	4,502	0,00	8,669
I12_40AP_200C	85,1	66,18	3,752	100,00	4,802	0,00	8,942
I13_40AP_200C	17,8	86,00	3,552	100,00	4,459	0,00	8,644
I14_40AP_200C	25,0	100,00	3,575	79,44	4,564	0,00	8,818
I15_40AP_200C	60,1	100,00	3,666	70,11	4,574	0,00	8,701
I16_40AP_200C	24,6	3,41	3,576	100,00	4,540	0,00	8,679
I17_40AP_200C	8,3	0,00	3,544	100,00	4,471	5,66	8,663
I18_40AP_200C	45,7	100,00	3,637	99,09	4,585	0,00	8,981
I19_40AP_200C	59,3	0,00	3,785	100,00	4,696	17,55	9,088
I20_40AP_200C	121,7	56,39	3,772	100,00	4,689	0,00	9,118
I21_40AP_200C	136,1	70,05	3,804	100,00	4,715	0,00	9,145
I22_70AP_350C	238,9	37,99	33,891	100,00	37,16	0,00	74,507
I23_70AP_350C	99,2	8,18	33,883	100,00	37,216	0,00	73,116
I24_70AP_350C	157,7	0,00	31,677	100,00	34,784	26,82	69,596
I25_70AP_350C	68,3	94,42	31,516	100,00	34,619	0,00	69,393
I26_70AP_350C	97,9	68,77	31,529	100,00	34,654	0,00	69,273
I27_70AP_350C	152,1	96,48	31,501	100,00	34,602	0,00	69,329
I28_70AP_350C	39,8	0,00	31,518	100,00	34,581	0,29	69,299
I29_70AP_350C	58,6	10,03	31,532	100,00	34,627	0,00	69,187
I30_70AP_350C	151,5	20,20	31,532	100,00	34,631	0,00	69,268

A partir da análise dos os resultados obtidos pelas abordagens GRASP desenvol-

vidas neste trabalho e apresentados na Tabela 5.2, é fácil concluir que o GRASP 3, versão que combina a construção seguida de refinamento de potências e de busca local, obteve os melhores resultados. Apenas para cinco das trinta instâncias o algoritmo não obteve as melhores soluções.

Uma outra observação que merece destaque é o fato de que a versão GRASP 2, que não emprega busca local, mas apenas a fase de refinamento de potências de transmissão, não apresentou resultados exclusivamente melhores que as demais versões que empregam busca local em nenhuma das instâncias testadas.

5.3 Comparação entre GRASP e literatura

Uma vez que a versão GRASP 3 obteve os melhores resultados dentre todas as abordagens propostas, esta foi selecionada como abordagem para a comparação em termos de eficácia e eficiência das soluções com o algoritmo da literatura, o SCIFI.

Para tanto, as 30 instâncias foram submetidas ao controlador SCIFI, cujos resultados gerados foram comparados aos obtidos pelo GRASP 3 e podem ser visualizados na Tabela 5.3.

Na realização desse teste, para cada instância foi considerado o valor da melhor solução obtida dentre todas as execuções dos algoritmos e o *gap* das médias das soluções para as dez execuções foi calculado com relação à este valor.

Tabela 5.3: Comparação entre os resultados do GRASP 3 com o SCIFI

Instância	Melhor	GRASP 3		SCIFI		
		<i>RDI</i>	Tempo	Resultado	<i>RDI</i>	Tempo
I1_10AP_50C	0	0,00	0,579	0	0,00	0,114
I2_10AP_50C	0	0,00	0,558	0	0,00	0,128
I3_10AP_50C	0	0,00	0,572	35	100,00	0,128
I4_10AP_50C	0	0,00	0,553	171	100,00	0,112
I5_10AP_50C	0	0,00	0,569	150	100,00	0,144
I6_10AP_50C	0	0,00	0,605	88	100,00	0,112
I7_10AP_50C	0	0,00	0,545	0	0,00	0,159
I8_10AP_50C	0	0,00	0,530	220	100,00	0,159
I9_10AP_50C	0	0,00	0,548	57	100,00	0,127
I10_40AP_200C	130	0,00	8,790	505	100,00	0,127
I11_40AP_200C	31	0,00	8,669	217	100,00	0,113
I12_40AP_200C	78	0,00	8,942	222	100,00	0,127
I13_40AP_200C	16	0,00	8,644	144	100,00	0,128
I14_40AP_200C	20	0,00	8,818	84	100,00	0,237
I15_40AP_200C	50	0,00	8,701	909	1718,00	0,143
I16_40AP_200C	12	0,00	8,679	775	100,00	0,143
I17_40AP_200C	0	0,00	8,663	121	100,00	0,143
I18_40AP_200C	32	0,00	8,981	1007	100,00	0,114
I19_40AP_200C	31	0,00	9,088	640	100,00	0,116
I20_40AP_200C	111	0,00	9,118	448	100,00	0,114
I21_40AP_200C	117	0,00	9,145	528	100,00	0,105
I22_70AP_350C	203	0,00	74,507	1568	100,00	0,134
I23_70AP_350C	91	0,00	73,116	1268	100,00	0,159
I24_70AP_350C	135	0,00	69,596	1497	100,00	0,284
I25_70AP_350C	53	0,00	69,393	715	100,00	0,175
I26_70AP_350C	84	0,00	69,273	463	100,00	0,206
I27_70AP_350C	139	0,00	69,329	1639	100,00	0,237
I28_70AP_350C	19	0,00	69,299	901	100,00	0,331
I29_70AP_350C	42	0,00	69,187	419	100,00	0,190
I30_70AP_350C	132	0,00	69,268	1717	100,00	0,221

Observou-se que, nas instâncias pequenas (1 a 9), o GRASP 3 foi capaz de eliminar a interferência na rede, assim como o SCIFI em alguns casos. Vê-se, também, que nas demais instâncias médias e grandes (10 a 30), o algoritmo proposto apresentou resultados muito superiores aos obtidos pela execução do SCIFI, podendo-se identificar soluções até 63 vezes melhores quando comparadas às obtidas pela abordagem da literatura.

Além disso, mesmo considerando que o SCIFI tivesse disponível a mesma quantidade de tempo utilizada pelo GRASP 3, seus resultados se manteriam inalterados por ser um algoritmo determinístico.

6 Conclusão

Com o crescimento cada vez mais acelerado do uso de redes sem fio e dispositivos móveis, enxergou-se a necessidade da construção de ambientes mais controlados e estáveis para manter a qualidade de serviço fornecida a seus utilizadores. Tratando-se desse assunto, a redução da interferência mútua entre Pontos de Acesso de uma rede no padrão 802.11 pode ser alcançada através da correta atribuição de canais e potência de transmissão aos seus componentes.

Tendo em vista o problema da interferência entre APs, técnicas colaborativas de implementação da coexistência foram utilizadas para o desenvolvimento de um sistema centralizado com base em heurísticas de construção, busca local e refinamento que, combinadas em diferentes ordens de chamada dentro de uma abordagem GRASP, foram capazes de realizar a seleção de canais e o controle das potências de forma que a interferência em diferentes redes pudesse ser reduzida ou, em alguns casos, mitigada.

Os algoritmos, apresentados com detalhes nos capítulos anteriores, foram submetidos a testes comparativos com uma outra abordagem encontrada na literatura para o mesmo problema. A partir dos experimentos realizados, foi possível analisar e concluir a eficiência e eficácia das soluções para o problema e demonstrar a qualidade superior dos algoritmos propostos.

A partir dos testes realizados, foi possível validar a hipótese de que o uso da estratégia de *clusterização* através de componentes conexas em um grafo de conflitos, onde os nós são os Pontos de Acesso, impacta em ganhos em até quase 1500% em relação a uma abordagem da literatura.

Para trabalhos futuros, sugere-se a adaptação do algoritmo para seu funcionamento de forma distribuída e paralela, podendo ser aplicado em múltiplas redes através da implementação da troca de informações entre os seus APs e sistemas controladores, tornando o programa capaz de reduzir ainda mais a interferência sobre as instâncias e, conseqüentemente, aumentar a qualidade de serviço das soluções geradas.

Também pode-se apontar uma modificação da composição das instâncias do pro-

blema para que, já no momento da entrada do algoritmo, os valores para o canal e potência de transmissão dos APs da rede estejam previamente definidos, assim como a associação dos clientes aos seus respectivos Pontos de Acesso. Dessa maneira, a abordagem iguala-se mais com a realidade e, além disso, surge-se a possibilidade de haver um menor processamento necessário da instância caso ela se encontre em uma disposição inicialmente favorável para a redução da interferência.

Ainda como sugestão de trabalhos futuros, diferentes técnicas de Inteligência Computacional podem ser empregadas a partir dos módulos de construção e de busca local aqui propostos, que apresentaram reduzidos tempos de processamento, bem como a realização da conexão dos clientes da rede aos APs de forma gulosa e aleatória na etapa de pré-processamento do algoritmo.

Por último, convém submeter a abordagem proposta a testes reais, a exemplo do que foi apresentado para o algoritmo da literatura indicado pela sua referência. Isto permitirá validar na prática os resultados aqui apresentados.

Referências Bibliográficas

- Akyildiz, I. F.; Lee, W.-Y.; Vuran, M. C. ; Mohanty, S. Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. **Computer Networks**, v.50, n.13, p. 2127–2159, 2006.
- Balbi, H. D.; Felipe, R.; Souza, R. C.; Saade, D. C.; de Albuquerque, C. V. ; Magalhães, L. C. Algoritmo de seleção de canais centralizado para redes ieee 802.11 com controlador. **XXX Simposio Brasileiro de Redes de Computadores e Sistemas Distribuidos**, 2012.
- Baykas, T.; Kasslin, M.; Cummings, M.; Kang, H.; Kwak, J.; Paine, R.; Reznik, A.; Saeed, R. ; Shellhammer, S. J. Developing a standard for tv white space coexistence: Technical challenges and solution approaches. **Wireless Communications, IEEE**, v.19, n.1, p. 10–22, 2012.
- Brélaz, D. New methods to color the vertices of a graph. **Communications of the ACM**, v.22, n.4, p. 251–256, 1979.
- Chiochan, S.; Hossain, E. ; Diamond, J. Channel assignment schemes for infrastructure-based 802.11 wlans: A survey. **Communications Surveys & Tutorials, IEEE**, v.12, n.1, p. 124–136, 2010.
- Cormen, T. H. Introduction to algorithms. 2009.
- Delphinanto, A.; Hillen, B.; Passchier, I.; van Schoonhoven, B. ; den Hartog, F. **Remote discovery and management of end-user devices in heterogeneous private networks**. In: Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE, p. 1–5. IEEE, 2009.
- Feo, T. A.; Resende, M. G. A probabilistic heuristic for a computationally difficult set covering problem. **Operations research letters**, v.8, n.2, p. 67–71, 1989.
- Harada, H. **Wireless cloud computing and cognitive wireless cloud**. In: The 3rd EU-Japan Symposium on the "Future Internet", 2010.
- Group, I. . W.; others. Ieee standard for local and metropolitan area networks. part 16: Air interface for broadband wireless access systems amendment 2: Improved coexistence mechanisms for license-exempt operation. **IEEE Standard**, 2010.
- Lansford, J. **Uwb coexistence and cognitive radio**. In: Ultra Wideband Systems, 2004. Joint with Conference on Ultrawideband Systems and Technologies. Joint UWBST & IWUWBS. 2004 International Workshop on, p. 35–39. IEEE, 2004.
- Mateus, G. R.; Resende, M. G. ; Silva, R. M. Grasp: Procedimentos de busca gulosos, aleatórios e adaptativos. **2a Escola Luso-Brasileira de Computação Evolutiva**, 2010.
- Peha, J. M. Wireless communications and coexistence for smart environments. **Personal Communications, IEEE**, v.7, n.5, p. 66–68, 2000.

- Schneider, J.; Kirkpatrick, S. Stochastic optimization. 2007.
- Song, L.; Hatzinakos, D. Cognitive networking of large scale wireless systems. **International Journal of Communication Networks and Distributed Systems**, v.2, n.4, p. 452–475, 2009.
- Sousa, M. P.; Lopes, R. F.; Lopes, W. T. A. ; Alencar, M. Redes cognitivas: Um novo paradigma para as comunicações sem fio. **Minicursos do XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos**, p. 153–197, 2010.
- Talbi, E.-G. Metaheuristics: from design to implementation. 2009.
- Thomas, R. W.; Friend, D. H.; DaSilva, L. A. ; MacKenzie, A. B. **Cognitive networks**. Springer, 2007.
- Vallada, E.; Ruiz, R. ; Minella, G. Minimising total tardiness in the m-machine flowshop problem: A review and evaluation of heuristics and metaheuristics. **Computers & Operations Research**, v.35, n.4, p. 1350–1373, 2008.