

## ***SIMULATED ANNEALING* PARA A ALOCAÇÃO DE CANAIS EM REDES MÓVEIS CELULARES**

**Bruno Dias, Rosiane de Freitas**  
Instituto de Computação  
Universidade Federal do Amazonas  
Manaus - AM  
{bruno.dias, rosiane}@icomp.ufam.edu.br

**Nelson Maculan**  
Programa de Engenharia de Sistemas e Computação/COPPE  
Universidade Federal do Rio de Janeiro  
Rio de Janeiro - RJ  
maculan@cos.ufrj.br

### **RESUMO**

Este artigo apresenta um algoritmo baseado em recozimento e arrefecimento simulado (*simulated annealing*) para solucionar o problema de alocação fixa de canais com minimização de extensão usada. Neste problema, é dada uma rede celular e as chamadas de cada célula, para as quais deve-se alocar canais respeitando restrições de interferência. Para análise empírica, foram usadas instâncias baseadas em cenários das cidades da Filadélfia e Helsinque, além de instâncias artificiais propostas por outros autores. O algoritmo proposto mostrou-se competitivo, gerando boas soluções em um tempo computacional aceitável. Em praticamente todos os cenários testados, as soluções obtidas são iguais ou melhores as reportadas na literatura.

**Palavras-chave:** algoritmos, metaheurísticas, telecomunicações.

**Área principal:** otimização combinatória.

### **ABSTRACT**

This paper presents a simulated annealing algorithm for solving the minimum span fixed channel assignment problem. In this problem, we are given a cellular network and the number of calls of each cell, for which channels must be assigned satisfying interference constraints. For empirical analysis, we used instances based on networks of Philadelphia and Helsinki cities, and some artificial instances proposed by other authors. The proposed algorithm is competitive, generating good solutions within reasonable computational time. In almost all tested scenarios, the obtained solutions are equal or better than the ones reported in literature.

**Keywords:** algorithms, metaheuristics, telecommunications.

**Main area:** combinatorial optimization.

## 1 Introdução

Nos últimos anos, o número de usuários de sistemas móveis tem aumentado de maneira bastante expressiva, tornando necessária a expansão das redes de serviços de conexão sem fio. Estima-se que até novembro de 2012 havia quase 4,3 bilhões de assinantes de telefonia móvel celular no mundo, o que corresponde a aproximadamente 61% da população mundial atual, sendo que o total de assinaturas (inscrições nas operadoras) é ainda maior, estimado em 9,3 bilhões (Ericsson (2012)). Tal aumento requer um uso mais eficiente do espectro eletromagnético, de forma que todos os usuários consigam acessar o serviço de telefonia móvel celular com qualidade. A determinação da melhor ocupação possível do espectro é o objetivo do Problema de Alocação de Canais.

Em uma rede móvel celular, há um conjunto de estações rádio base (ERBs) com as quais os dispositivos dos usuários se comunicam. Por sua vez, as ERBs comunicam-se com os controladores de estações base (CEBs), que possuem conexão com a central de comutação e controle (CCC) da rede, onde as chamadas são roteadas de acordo com os usuários envolvidos. Esta estrutura é exemplificada na Figura 1.

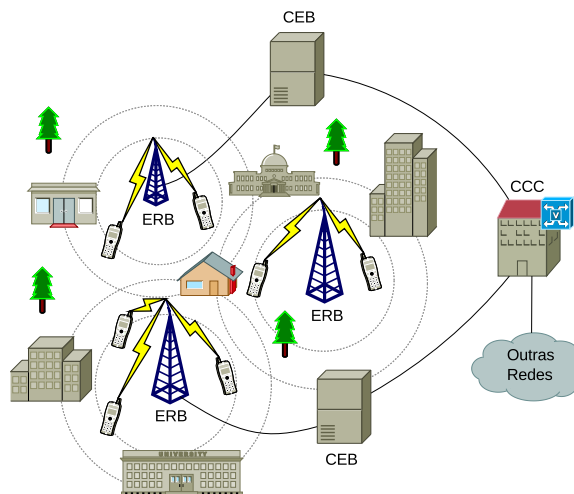


Figura 1. Exemplo de rede de telefonia móvel celular.

Cada ERB define uma célula hexagonal, que é a área de cobertura do dispositivo. A célula correspondente a uma chamada dependerá de sua localização. A comunicação é feita através de um canal, que consiste em uma porção discreta do espectro eletromagnético. Entretanto, os canais não devem ser alocados de qualquer forma, já que existe o problema de interferência, sendo três os principais tipos:

- **Co-canal:** interferência entre chamadas de ERBs distintas usando o mesmo canal;
- **Canal adjacente:** interferência entre chamadas de diferentes ERBs utilizando canais adjacentes entre si;
- **Co-site:** interferência entre chamadas situadas na mesma célula.

A classe de problemas de alocação de canais é bastante estudada na literatura, com diversos algoritmos e modelos propostos por vários autores. Um dos trabalhos pioneiros foi realizado por Hale (1980), que formulou modelos teóricos de otimização para situações de planejamento do uso do espectro eletromagnético. Kunz (1991) aplicou um algoritmo de redes neurais para efetuar a alocação de canais em uma rede homogênea (com

demandas iguais) e outra rede não-homogênea baseada em um cenário real da cidade de Helsinque, Finlândia. Chakraborty (2001) propôs uma heurística simples que constrói um conjunto de soluções factíveis e retorna a melhor dentre elas. O algoritmo foi testado utilizando instâncias baseadas em uma rede da cidade da Filadélfia, Estados Unidos, além das instâncias de Helsinque e outras artificiais. Kendall e Mohamad (2005) desenvolveram uma busca local cujo critério de aceitação é um algoritmo Monte Carlo. Chaves-González et al. (2008) usaram um algoritmo de Aprendizagem Incremental Baseado em População (*Population-Based Incremental Learning* - PBIL) para obter soluções para o problema, e aplicaram o método nas instâncias Philadelphia. Narayanaswamy (2012) utilizou uma abordagem hiper-heurística consistindo de seis heurísticas básicas, obtendo soluções melhores em comparação aos resultados de Chakraborty.

O restante deste artigo está organizado como segue. Na Seção 2, o problema, para o qual o algoritmo é aplicado, é definido formalmente. Na Seção 3, é apresentado o algoritmo de recozimento e arrefecimento simulado (*simulated annealing*). A Seção 4 mostra os resultados obtidos pelos experimentos computacionais com os algoritmos e as instâncias utilizadas. Por fim, na Seção 5, são feitas as considerações finais e os trabalhos relacionados em andamento são brevemente comentados.

## 2 Definições do problema

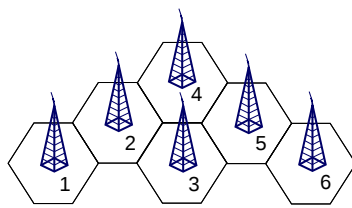
Conforme Koster (1999) e Audhya et al. (2011), pode-se definir formalmente o problema como se segue. Existe um conjunto de  $n$  ERBs, onde cada uma define uma célula hexagonal. A  $i$ -ésima ERB tem uma demanda de  $d_i$  canais, que corresponde à quantidade de chamadas daquela estação. Cada chamada necessita de um canal para sua realização. Com isto, tem-se que para todo par  $(i, j)$  de ERBs que possam interferir entre si, onde  $f$  é algum dos canais alocados para  $i$ , e  $g$  é algum dos alocados para  $j$ , existe um valor  $c_{ij}$  (a distância de separação) para o qual a condição  $|f - g| \geq c_{ij}$  deve ser respeitada. Os valores  $c_{ij}$  definem a **matriz de interferências** da rede. O conjunto de todos os canais disponíveis para alocação na rede é denotado por  $F$ .

Com essas características, deseja-se então **minimizar o maior canal usado** dentre todos os canais usados pelas chamadas. Este é um problema de **min-max**, portanto, o problema de alocação de canais tratado neste trabalho pode ser denominado de **Min-Max-CAP** (Dias et al. (2012)).

Baseado no modelo de programação inteira de Koster (1999), este problema de alocação de canais pode ser formulado matematicamente como segue. São usadas as seguintes variáveis de decisão:

- $x_{if} = \begin{cases} 1 & \text{, se o canal } f \text{ for alocado à célula } i \\ 0 & \text{, caso contrário} \end{cases}$
- $y_f = \begin{cases} 1 & \text{se o canal } f \text{ for alocado a alguma célula} \\ 0 & \text{caso contrário} \end{cases}$
- $z_{max} = \text{valor do maior canal utilizado.}$

Assim, tem-se o seguinte modelo de programação inteira:



Matriz de interferências	Demandas
$\begin{bmatrix} 5 & 2 & 1 & 1 & 0 & 0 \\ 2 & 5 & 2 & 2 & 1 & 0 \\ 1 & 2 & 5 & 2 & 2 & 1 \\ 1 & 2 & 2 & 5 & 2 & 1 \\ 0 & 1 & 2 & 2 & 5 & 2 \\ 0 & 0 & 1 & 1 & 2 & 5 \end{bmatrix}$	1, 1, 2, 1, 3, 2

Figura 2. Rede celular de 6 células usada como exemplo.

Tabela 1. Alocação factível de custo 16 para a instância de 6 células da Figura 2.

Célula	Demanda	Canais alocados
1	1	1
2	1	10
3	2	3, 8
4	1	5
5	3	1, 11, 16
6	2	4, 9

Minimizar  $z_{max}$

Sujeito a:

$$\sum_{f \in F} x_{if} = d_i \quad (\forall i = 1, 2, \dots, n) \quad (1)$$

$$x_{if} + x_{jg} \leq 1 \quad (\forall i, j = 1, 2, \dots, n; \forall f, g \in F : |f - g| < c_{ij}) \quad (2)$$

$$x_{if} \leq y_f \quad (\forall i = 1, 2, \dots, n; f \in F_i) \quad (3)$$

$$z_{max} \geq fy_f \quad (\forall f \in F) \quad (4)$$

$$z_{max} \in \mathbb{Z}^+ \quad (5)$$

$$x_{if} \in \{0, 1\} \quad (\forall i = 1, 2, \dots, n; \forall f \in F) \quad (6)$$

$$y_f \in \{0, 1\} \quad (\forall f \in F) \quad (7)$$

Nessa formulação, o conjunto de restrições (1) garante que as demandas de canais de cada célula sejam satisfeitas. As restrições (2) asseguram que as separações necessárias entre canais sejam respeitadas. O conjunto de restrições (3) faz com que, se alguma variável  $x_{if}$  tem valor 1 para um determinado canal  $f$ , a variável  $y_f$  também deverá ter o mesmo valor. As restrições (4) fazem com que a variável  $z_{max}$  tenha, como valor, o maior canal utilizado. Por fim, as restrições (5), (6) e (7) determinam a integralidade das variáveis.

Como um exemplo do problema, será considerada a rede de 6 células mostrada na Figura 2, onde são dadas as demandas de canais e a matriz de interferências para a mesma. A dimensão da matriz é  $6 \times 6$  e o vetor de demandas possui 6 elementos, onde seu  $i$ -ésimo elemento é a demanda da  $i$ -ésima célula. Uma alocação de canais factível para este exemplo é dada na Tabela 1, cujo valor do maior canal usado é 16.

### 3 Algoritmo *simulated annealing* proposto

A metaheurística de recozimento/arrefecimento simulado (*simulated annealing*) é baseada em processos metalúrgicos, nos quais um metal é aquecido de forma a tornar-

se mais maleável (recozimento), sendo em seguida lentamente resfriado para que suas partículas se tornem estáveis (arrefecimento). A cada temperatura, o metal deve atingir o equilíbrio termodinâmico antes que a temperatura decresça mais. Ao final do processo, o metal, já sólido novamente, terá uma estrutura cristalina.

O método foi proposto por Kirkpatrick et al. (1983) baseado no trabalho de mecânica estatística de Metropolis et al. (1953), sendo aplicado com sucesso em problemas de particionamento em grafos e desenvolvimento de circuitos digitais.

Para a aplicação do método ao problema Min-Max-CAP, deve-se definir uma representação para a solução, um gerador de solução inicial, a vizinhança para geração de novas soluções e os parâmetros do algoritmo, como será visto adiante.

### 3.1 Representação da solução

Para o problema de alocação de canais, uma solução pode ser codificada como uma sequência de pares  $(v, k)$ , onde  $v$  é uma célula e  $k$  é o índice da chamada. Os canais serão alocados de acordo com a ordem de ocorrência na sequência. Uma sequência inicial de chamadas para a inicialização é obtida ordenando as células em ordem decrescente de demandas de canais, onde as chamadas das células serão todas adicionadas na frente da sequência. O Algoritmo 1 fornece o pseudocódigo para isso.

Para alocar os canais de acordo com a sequência de chamadas, foi utilizada a regra de **alocação exaustiva por frequência** (*frequency exhaustive allocation* - FEA) (Sivarajan et al. (1989); Beckmann e Killat (1999)), que consiste em atribuir o menor canal que não viole restrições de interferência para a chamada. O canal candidato é obtido da seguinte maneira:

1. O primeiro candidato é o menor canal possível, de valor 1;
2. Se a ERB usada pela chamada possuir outras chamadas com canais já alocados, o menor canal possível é incrementado em  $\lambda_i d_{ii}$ , onde  $\lambda_i$  é o número de chamadas de  $i$  já com canais e  $d_{ii}$  é a separação de canais entre chamadas na mesma célula;
3. Finalmente, é feita uma verificação para garantir que o canal candidato não viole restrições de interferência com os canais atribuídos a células adjacentes. Se alguma violação for encontrada, o canal candidato é incrementado em 1 e a verificação é repetida. Quando nenhuma violação for encontrada, o canal candidato é o mínimo alocável para a chamada.

O pseudocódigo para a regra FEA é dado no Algoritmo 2. Um exemplo de sequência de chamadas e a alocação de canais correspondente utilizando a instância fornecida pela Figura 2 (matriz de interferências e demandas) é mostrado na Figura 3.

---

#### Algoritmo 1 Geração da sequência inicial

---

$V \leftarrow$  conjunto de células

$S \leftarrow$  sequência de chamadas (inicialmente vazia)

**enquanto**  $V \neq \emptyset$  **faça**

$u \leftarrow$  célula de  $V$  de maior demanda

**para**  $k \leftarrow 1$  **até**  $u.numChamadas$  **faça**

Adicionar a  $k$ -ésima chamada de  $u$  à frente de  $S$

$V \leftarrow V - \{u\}$

---

---

**Algoritmo 2 FEA:** Alocação exaustiva por frequência

---

```

numCanaisAlocados ← vetor de numCelulas elementos, todos inicializados com 0
maiorCanal ← 0
para  $c \leftarrow 1$  até numChamadas faça
     $i \leftarrow \text{ordemChamadas}[c].\text{celula}$ 
    canalCandidato ←  $(\text{numCanaisAlocados}[i] * d[i][i]) + 1$ 
    flagOK ← falso
    enquanto flagOK = falso faça
        flagOK ← verdadeiro
        para cada célula  $j$  adjacente a  $i$  faça
            para  $k \leftarrow 1$  até numCanaisAlocados[ $k$ ] faça
                se  $| \text{canalCandidato} - \text{canaisAlocados}[j][k] | < \text{matrizInterf}[i][j]$  então
                    flagOK ← verdadeiro
                    interrompa
            se flagOK = falso então
                canalCandidato ← canalCandidato + 1
                interrompa
    numCanaisAlocados[ $i$ ] ← numCanaisAlocados[ $i$ ] + 1
    canaisAlocados[ $i$ ][numCanaisAlocados[ $i$ ]] ← canalCandidato
    se canalCandidato > maiorCanal então
        maiorCanal ← canalCandidato

```

---

### 3.2 Vizinhaça de soluções

Para modificar a solução, uma vizinhaça  $N : S \rightarrow 2^S$  com movimentos de troca entre pares é usada. Dada uma sequência  $S$  definindo uma solução, uma nova sequência  $S'$  é derivada efetuando a troca entre duas posições de chamadas em  $S$ . O Algoritmo 3 fornece o pseudocódigo para este procedimento. Na Figura 4, a sequência da Figura 3 é modificada trocando as posições das duas chamadas marcadas, o que, após o uso da regra FEA, fornece uma alocação de menor custo em relação à sequência original.

### 3.3 Algoritmo proposto

O algoritmo *simulated annealing* proposto funciona da seguinte forma: partindo de uma sequência inicial  $S$  e uma temperatura inicial  $t_0$ , é feita uma cópia de  $S$  para uma outra sequência  $R$  que será modificada (e  $S$  será considerada a melhor solução obtida na

5,1 | 3,1 | 1,1 | 3,2 | 6,2 | 6,1 | 4,1 | 5,2 | 2,1 | 5,3

Célula	Canal(is)
1	1
2	10
3	3, 8
4	5
5	1, 11, 16
6	4, 9

Figura 3. Exemplo de sequência de chamadas e a alocação de canais correspondente obtida pelo algoritmo FEA.

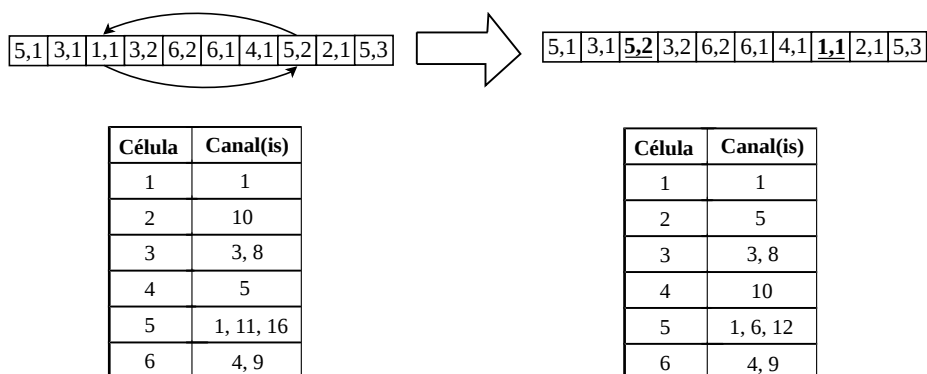


Figura 4. Operação de troca em uma sequência de chamadas. A sequência original possui uma alocação de custo 16. Após trocar as posições dos elementos (5,2) e (1,1) entre si, é obtida uma nova alocação de custo 12, que é melhor.

---

### Algoritmo 3 Geração de um vizinho aleatório de uma sequência $S$

---

$S \leftarrow$  sequência de chamadas de entrada

$p_1 \leftarrow$  inteiro aleatório entre 0 e o número de chamadas

$p_2 \leftarrow$  inteiro aleatório entre 0 e o número de chamadas

Trocar as chamadas das posições  $p_1$  e  $p_2$  em  $S$

---

temperatura). Enquanto o equilíbrio termodinâmico não for atingido para a temperatura, duas posições aleatórias na sequência  $R$  são escolhidas e trocadas de posição entre si. O algoritmo FEA é aplicado e, então:

- Se o maior canal utilizado para a nova sequência for menor que o usado na sequência antiga, então  $R$  é copiado para  $S$ ;
- Caso contrário, com probabilidade  $e^{(-\Delta/t)}$  (onde  $\Delta$  é a diferença entre o maior canal utilizado em  $R$  e em  $S$ ),  $R$  é copiado para  $S$ .

Quando o equilíbrio termodinâmico é atingido, a solução  $S$  é comparada à melhor obtida durante todo o processo até o momento. Se o custo de  $S$  for menor, então ela se tornará a nova melhor solução corrente. A temperatura então é decrementada. No algoritmo proposto, foi utilizado o esquema de resfriamento **linear**. Além disso, foi considerado que o equilíbrio termodinâmico é atingido após um número fixo  $L$  de iterações a cada temperatura. Esta abordagem, apesar de simples, teve bons resultados. O pseudocódigo desta rotina é dado pelo Algoritmo 4.

## 4 Experimentos computacionais

Para avaliar empiricamente o desempenho do algoritmo, foram utilizadas os conjuntos *Philadelphia*, *Helsinki* e as instâncias artificiais de 55 células.

As instâncias *Philadelphia* são baseadas em uma rede de 21 células hexagonais da cidade da Filadélfia, Estados Unidos, com demandas de canais variadas. Os três tipos de interferências estão presentes nas mesmas. Os requisitos de separação entre canais são dadas por meio de um conjunto  $R$  de distâncias de reuso, onde cada elemento  $r_i$  do mesmo indica que, entre chamadas de células cuja distância geográfica seja  $i$ , os canais alocados para as mesmas devem distar  $i$  unidades no espectro eletromagnético.

---

**Algoritmo 4** *Simulated annealing* para alocação de canais em redes celulares

---

```

S ← sequência inicial de chamadas
t0 ← temperatura inicial
tf ← temperatura final (tf < t0)
α ← fator de resfriamento
L ← iterações para o equilíbrio termodinâmico
Melhor ← S
R ← S
t ← t0
enquanto t > tf faça
  para p ← 1 até L faça
    // Gerar novo vizinho aleatório
    R ← R.NovoVizinho();
    // Alocar canais usando a regra FEA
    R.Alocar()
    Δ ← R.maiorCanal – S.maiorCanal
    se Δ < 0 então
      S ← R
    senão
      r ← número real aleatório entre 0 e 1
      se r < e(-Δ/t) então
        S ← R
    se S.maiorCanal < Best.maiorCanal então
      Melhor ← S
  t ← t – α

```

---

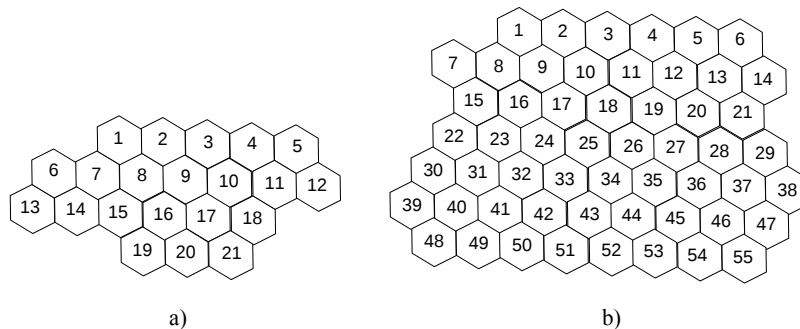


Figura 5. Redes das instâncias: a) Philadelphia e, b) Artificiais.

Outros conjuntos de instâncias de testes similares aos problemas *Philadelphia* também estão disponíveis. Kunz (1991) modelou uma rede real da cidade de Helsinque, Finlândia, contendo 25 células. Para cada uma delas, são dadas as outras células que causam interferências. As separações de canais entre chamadas de duas células que possam interferir entre si é 1, enquanto a separação entre chamadas na mesma célula é 5. Chakraborty (2001) forneceu instâncias artificiais de 55 células baseadas nas instâncias *Philadelphia*. As disposições das células das instâncias *Philadelphia* e artificiais são mostradas na Figura 5.

Todas as instâncias citadas são baseadas em um sistema de *two-band buffering*,



Tabela 2. Separações entre canais para cada instância.

Num. células	Matriz interf.	Dist. entre células $i$ e $j$	Valor de $c_{ij}$	Separação <i>co-site</i> ( $c_{ii}$ )
21	$C_{21}^1$	1	1	7
		2	1	
		> 2	0	
	$C_{21}^2$	1	2	7
		2	1	
		> 2	0	
	$C_{21}^3$	1	1	6
		2	1	
		> 2	0	
	$C_{21}^4$	1	2	6
		2	1	
		> 2	0	
	$C_{21}^5$	1	1	5
		2	1	
		> 2	0	
	$C_{21}^6$	1	2	5
		2	1	
		> 2	0	
	$C_{21}^7$	1	1	4
		2	1	
		> 2	0	
	$C_{21}^8$	1	2	4
		2	1	
		> 2	0	
25	$C_{25}^1$	1	1	2
		2	1	
		> 2	0	
55	$C_{55}^1$	1	1	7
		2	1	
		> 2	0	

onde cada célula só interfere em outras cuja distância geográfica seja, no máximo de duas unidades. Assim, o conjunto  $R$  possui três elementos ( $R = (r_1, r_2, r_3)$ ). As demandas de canais para todas as instâncias são dadas na Tabela 3.

A matriz de interferências de cada instância pode ser obtida como se segue. Como todas as instâncias utilizam *two-band buffering*, há um valor de separação entre células adjacentes e entre células cuja distância seja 2. Quando a distância é maior que 2, as células não interferem entre si. Nas instâncias *Helsinki*, a separação necessária entre os canais de células que interfiram entre si é 1. A distância de separação para cancelar a interferência *co-site*, ou seja, entre chamadas na mesma célula, é dada pelo valor  $c_{ii}$ . Como este tipo de interferência é o mais forte, tais valores são os maiores. Os dados para obtenção de cada matriz de interferência é dada na Tabela 2.

O algoritmo foi implementado em linguagem C e executado em um computador com processador AMD Athlon II X2 (2,9GHz), 2GB de memória RAM e sistema operacional Ubuntu Linux 12.10 (64 bits). Para cada instância, o algoritmo foi executado dez vezes e a melhor solução obtida para a instância foi extraída para comparações. Os parâmetros do *simulated annealing* foram configurados da seguinte forma:

- Temperatura inicial  $t_0 = 100$ ;

**Tabela 3. Vetores de demanda**

#Células	Vetor	Demandas
21	$D_{21}^1$	8 25 8 8 8 15 18 52 77 28 13 15 31 15 36 57 28 8 10 13 8
	$D_{21}^2$	5 5 5 8 12 25 30 25 30 40 40 45 20 30 25 15 15 30 20 20 25
25	$D_{25}^3$	10 11 9 5 9 4 5 7 4 8 8 9 10 7 7 6 4 5 5 7 6 4 5 7 5
	$D_{25}^4$	5 5 5 8 12 25 30 25 30 40 40 45 20 30 25 15 15 30 20 20 25 8 5 5 5
55	$D_{55}^5$	5 5 5 8 12 25 30 25 30 40 40 45 20 30 25 15 15 30 20 20 25 8 5 5 5 5 5 8 12 25 30 25 30 40 40 45 20 30 25 15 15 30 20 20 25 8 5 5 5 25 8 5 5 5
	$D_{55}^6$	10 11 9 5 9 4 5 7 4 8 8 9 10 7 7 6 4 5 5 7 6 4 5 7 5 10 11 9 5 9 4 5 7 4 8 8 9 10 7 7 6 4 5 5 7 6 4 5 7 5 6 4 5 7 5

- Temperatura final  $t_f = 0$ ;
- Número de iterações para o equilíbrio termodinâmico = 40;
- Esquema de resfriamento: linear, com decrementos de 0,5.

A Tabela 4 mostra os resultados obtidos pelo algoritmo proposto (com o desvio padrão dos resultados de cada execução) e os resultados obtidos por Chakraborty, Kendall e Mohamad e Chaves-González et al.. A Figura 6 mostra os resultados em formato de gráfico para os trabalhos que usaram todas as instâncias. Para cada problema, a tabela informa a matriz de interferências, o vetor de demandas e o limite inferior dado nos trabalhos supracitados, definido como o maior produto da distância *co-site* pela demanda de canais dentre todos os vértices, isto é,  $\max_{i \in V} [(c_{ii} \times (d_i - 1)) + 1]$ , onde  $V$  é o conjunto de células.

Nos resultados obtidos neste trabalho, a média de tempo das dez execuções também é fornecida. Para os resultados de Chakraborty, são fornecidos o valor da melhor solução e o tempo de geração de 1000 soluções, e para os trabalhos de Kendall e Mohamad e Chaves-González et al., são dados os valores obtidos. Todos esses dados foram obtidos diretamente dos respectivos artigos. A melhor solução para o problema é destacada na tabela (negrito e itálico), exceto nos casos onde todos os métodos obtiveram os mesmos valores. Se o valor de um dos trabalhos para algum problema for –, tem-se que o respectivo artigo não executou testes com o problema.

O algoritmo proposto mostrou-se competitivo, apesar de sua simplicidade: foram obtidas soluções iguais ou melhores para praticamente todos os problemas. Em especial, para o problema 20 (com 55 células), o algoritmo foi capaz de obter o limite inferior da instância, o que não ocorreu com os outros métodos. Os únicos problemas nos quais o algoritmo proposto não obteve as melhores soluções foram as instâncias 11 e 18, sendo que neste último, houve empate com o método de Kendall e Mohamad, mas inferior a Chakraborty, o que sugere que uma heurística capaz de explorar características específicas da instância deve ser usada.

## 5 Considerações finais

Neste artigo, foi apresentado um algoritmo baseado em *simulated annealing* para um problema de alocação de canais em redes celulares, que é um problema muito impor-

Tabela 4. Resultados obtidos com a heurística *simulated annealing* (SA).

#Células	Algoritmo proposto							Chakraborty (2001)		Kendall e Mohamad (2005)	Chaves-González et al. (2008)
	Inst.	Matr.	Demand.	Lim.Inf.	Obtido	Tempo total	Desvio padrão	Obtido	Tempo (1000 sol.)	Obtido	Obtido
21	1	$C_{21}^1$	$D_{21}^1$	533	533	9.77s	0.00	533	8.2s	533	533
	2	$C_{21}^1$	$D_{21}^2$	309	309	8.49s	0.00	309	6.0s	309	309
	3	$C_{21}^2$	$D_{21}^1$	533	533	20.51s	0.00	533	11.1s	533	-
	4	$C_{21}^2$	$D_{21}^2$	309	309	15.64s	0.00	309	10.2s	309	-
	5	$C_{21}^3$	$D_{21}^1$	457	457	13.23s	0.00	457	8.9s	457	-
	6	$C_{21}^3$	$D_{21}^2$	265	265	10.35s	0.00	265	8.1s	265	-
	7	$C_{21}^4$	$D_{21}^1$	457	457	25.25s	0.00	457	9.8s	457	-
	8	$C_{21}^4$	$D_{21}^2$	265	<b>267</b>	18.30s	1.41	280	7.9s	273	-
	9	$C_{21}^5$	$D_{21}^1$	381	<b>381</b>	18.68s	0.00	<b>381</b>	7.5s	<b>381</b>	383
	10	$C_{21}^5$	$D_{21}^2$	221	<b>221</b>	13.15s	0.00	<b>221</b>	6.9s	<b>221</b>	223
	11	$C_{21}^6$	$D_{21}^1$	381	442	29.05s	2.06	463	9.5s	<b>435</b>	-
	12	$C_{21}^6$	$D_{21}^2$	221	<b>265</b>	20.93s	1.37	273	7.7s	268	-
	13	$C_{21}^7$	$D_{21}^1$	305	305	25.43s	0.00	305	7.3s	305	-
	14	$C_{21}^7$	$D_{21}^2$	177	<b>181</b>	16.67s	0.49	197	6.8s	185	-
	15	$C_{21}^8$	$D_{21}^1$	305	<b>443</b>	34.67s	1.20	465	8.4s	444	-
	16	$C_{21}^8$	$D_{21}^2$	177	<b>269</b>	24.53s	1.20	278	7.5s	271	-
25	17	$C_{25}^1$	$D_{25}^3$	21	73	4.18s	0.00	73	1.9s	73	-
	18	$C_{25}^1$	$D_{25}^4$	89	200	29.24s	0.00	<b>121</b>	6.3s	200	-
55	19	$C_{55}^1$	$D_{55}^5$	309	309	33.64s	0.00	309	24.5s	309	-
	20	$C_{55}^1$	$D_{55}^6$	71	<b>71</b>	7.54s	0.40	79	16.7s	72	-

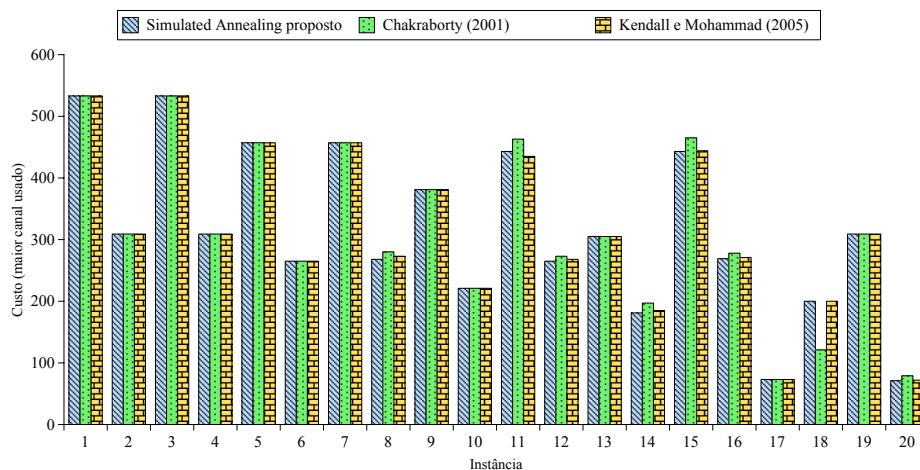


Figura 6. Gráfico de resultados da Tabela 4.

tante das redes de telecomunicações. Como o espectro eletromagnético é um recurso limitado e controlado, seu uso deve ser otimizado. O algoritmo aloca canais para as chamadas das células de forma a evitar interferências e minimizar o uso dos canais, sendo competitivo com outros da literatura, como visto nos experimentos realizados com instâncias de *benchmark*.

Trabalhos em andamento envolvem a utilização deste algoritmo e outros no cenário de alocação dinâmica de canais, onde as instâncias mudam no decorrer do tempo, exigindo

que uma estratégia de adaptação ao tráfego seja utilizada, além da aplicação de abordagens de otimização na alocação de canais em redes de rádios cognitivos, onde os dispositivos podem fazer uso oportunista de canais licenciados, respeitando a prioridade dos usuários que possuem a devida licença e detectando quais porções do espectro eletromagnético podem ser usadas.

**Agradecimento:** Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

## Referências

- Audhya, G. K., Sinha, K., Ghosh, S. C., e Sinha, B. P.** (2011). A survey on the channel assignment problem in wireless networks. *Wireless Communications and Mobile Computing*, 11(5):583–609.
- Beckmann, D. e Killat, U.** (1999). A new strategy for the application of genetic algorithms to the channel assignment problem. *IEEE Transactions on Vehicular Technology*, 48(4):1261–1269.
- Chakraborty, G.** (2001). An Efficient Heuristic Algorithm for Channel Assignment Problem in Cellular Radio Networks. *IEEE Transactions on Vehicular Technology*, 50(6):1528–1539.
- Chaves-González, J. M., Vega-Rodríguez, M. A., Domínguez-González, D., e Juan A. Gómez-Pulido, J. M. S.-P.** (2008). Population-Based Incremental Learning to Solve the FAP Problem. In *Proceedings of the 2th International Conference on Advanced Engineering Computing and Applications in Sciences*.
- Dias, B. R. C., Freitas, R., e Maculan, N.** (2012). Alocação de canais em redes celulares sem fio: algoritmos e modelos teóricos em grafos e escalonamento. In *Anais do XLIV Simpósio Brasileiro de Pesquisa Operacional*.
- Ericsson** (2012). Mobility Report - November 2012. <http://www.ericsson.com/traffic-market-report>.
- Hale, W. K.** (1980). Frequency assignment: Theory and applications. *Proceedings of the IEEE*, 25:1497–1514.
- Kendall, G. e Mohamad, M.** (2005). Solving the Fixed Channel Assignment Problem in Cellular Communications Using an Adaptive Local Search. In Burke, E. e Trick, M., editors, *5th International Conference for the Practice and Theory of Automated Timetabling (PATAT 2004)*, Lecture Notes in Computer Science, vol. 3616. Springer, Heidelberg.
- Kirkpatrick, S., Jr., C. D. G., e Vecchi, M.** (1983). Optimization by Simulated Annealing. *Science*, 220(4598):671–680.
- Koster, A. M. C. A.** (1999). *Frequency assignment: models and algorithms*. Universiteit Maastricht.
- Kunz, D.** (1991). Channel Assignment for Cellular Radio Using Neural Networks. *IEEE Transactions on Vehicular Technology*, 40(1):188–193.
- Metropolis, N., Rosenbluth, A., e Rosenbluth, M.** (1953). Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*, 21(6):1087–1092.
- Narayanaswamy, K.** (2012). Analysis and Optimization of Channel Allocation Strategies in Cellular Network. *International Journal of Computer Networks and Wireless Communications*, 2(6):739–743.
- Sivarajan, K. N., McEliece, R. J., e Ketchum, J. W.** (1989). Channel assignment in cellular radio. In *IEEE 39th Vehicular Technology Conference*, volume 2, pp. 846–850.