



UNIVERSIDADE FEDERAL DO MARANHÃO

Curso de Ciência da Computação

Artur Albuquerque Silva

**Implementação Capacitada do *Growing Neural Gas* aplicada à Logística de Leitura em  
Companhias de Distribuição de Energia Elétrica**

São Luís - MA

Janeiro - 2019

Artur Albuquerque Silva

**Implementação Capacitada do *Growing Neural Gas*  
aplicada à Logística de Leitura em Companhias de  
Distribuição de Energia Elétrica**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão como parte dos requisitos necessários para obtenção do título de bacharel em Ciência da Computação.

Curso de Ciência da Computação

UFMA

Orientador: Prof. Dr. Geraldo Braz Júnior

Coorientador: Prof. Dr. Anselmo Cardoso de Paiva

São Luís - MA

Janeiro - 2019

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).  
Núcleo Integrado de Bibliotecas/UFMA

Albuquerque Silva, Artur.

Implementação Capacitada do Growing Neural Gas aplicada à Logística de Leitura em Companhias de Distribuição de Energia Elétrica / Artur Albuquerque Silva. - 2019.

45 f.

Coorientador(a): Anselmo Cardoso de Paiva.

Orientador(a): Geraldo Braz Júnior.

Monografia (Graduação) - Curso de Ciência da Computação, Universidade Federal do Maranhão, São Luís, 2019.

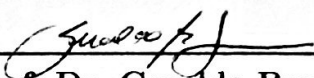
1. Agrupamento Capacitado. 2. Growing Neural Gas. 3. Problema de Distritamento. I. Braz Júnior, Geraldo. II. Cardoso de Paiva, Anselmo. III. Título.


Artur Albuquerque Silva

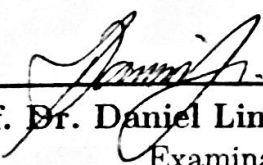
## **Implementação Capacitada do *Growing Neural Gas* aplicada à Logística de Leitura em Companhias de Distribuição de Energia Elétrica**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão como parte dos requisitos necessários para obtenção do título de bacharel em Ciência da Computação.

Monografia Aprovada em 11 de Janeiro de 2019

  
\_\_\_\_\_  
Prof. Dr. Geraldo Braz Júnior  
Orientador

  
\_\_\_\_\_  
Profa. Dra. Simara Vieira da Rocha  
Examinador  
Universidade Federal do Maranhão

  
\_\_\_\_\_  
Prof. Dr. Daniel Lima Gomes Júnior  
Examinador  
Instituto Federal do Maranhão

São Luís - MA  
Janeiro - 2019

# Agradecimentos

Agradeço aos meus pais, Patrícia Maia Correia de Albuquerque e Antônio Augusto Moura da Silva, por todo o suporte e ajuda que me deram durante toda a minha vida.

A minha família, que me ajudaram a formar quem sou hoje.

A todos os amigos que fiz durante minha graduação, pela ajuda no decorrer do curso e na vida.

A Geraldo Braz Júnior e Anselmo Cardoso de Paiva, por serem meus orientadores e me ajudarem a concluir mais uma etapa da minha vida.

A Simara Viera da Rocha e Daniel Lima Gomes Júnior, pela oportunidade de defesa e por auxiliar a melhorar este trabalho.

Aos professores Alexandre César Muniz Oliveira e Aristófanés Correa Silva, pela tutoria nos programas de pesquisa dos quais participei.

Por fim, agradeço a aqueles que não mencionei mas que me ajudaram de alguma forma no decorrer da minha vida.

*"What a beautiful Duwang!."*  
*(Kira Yoshikage, JoJo's Bizzare Adventure - part 4)*

# Resumo

A leitura dos medidores de energia e entrega de contas para cada cliente são as atividades principais relacionadas ao processo de contas de uma empresa de distribuição de energia elétrica. As companhias de distribuição de energia elétrica têm a tarefa de organizar o processo de medição por meio de grupos de consumidores. Portanto, cada funcionário responsável pela leitura tem de ter rotas para percorrer na medição e entrega de contas de energia em cada grupo. Para resolver esse problema do agrupamento, o desenvolvimento de métodos computacionais são importantes, de forma a melhorar o processo. Portanto, este trabalho propõe o *Capacitated Growing Neural Gas* (C-GNG), uma alteração no *Growing Neural Gas* (GNG) para utilizar os conceitos do agrupamento capacitado. Aplicado no contexto de distribuição de energia das companhias elétricas do Brasil. Visando obter um agrupamento melhor que o atual, que foi gerado manualmente pelos especialistas das companhias. Os resultados obtidos para as cidades usadas na simulação, demonstram uma melhoria nos aspectos de coesão, separação e coeficiente de silhueta gerados pela solução otimizada.

**Palavras-chaves:** Agrupamento Capacitado, Growing Neural Gas, Problema de Distritamento.

# Abstract

The main tasks related to the billing process of an electrical energy distribution company are the measuring of client energy meters and invoice delivery. The energy distribution companies have the job of organizing the measuring process by creating groups of consumers. Thus, each employee must have routes to perform the measuring and delivering invoices in each group. To solve this clustering problem, the development of computational methods is important, so that the process can be improved. Therefore, this work proposes Capacitated Growing Neural Gas (C-GNG), a modified Growing Neural Gas using capacitated clustering concepts. Applied to the electric distribution companies context in Brazil. Aiming to obtain a better clustering than the current one, that was manually generated by company experts. The obtained results, demonstrates improvements in terms of cohesion, separation and silhouette using the proposed C-GNG when compared with the manually crafted solution.

**Keywords:** Capacitated Clustering, Growing Neural Gas, Districting Problem.



# Lista de ilustrações

Figura 1 – Inicialização de $a$ (vermelho) e $b$ (amarelo) . . . . .	16
Figura 2 – Determinando $s_1$ e $s_2$ a partir do ponto da entrada . . . . .	17
Figura 3 – Aproximação de $s_1$ (Vermelho) e vizinhos em direção a $\xi$ . . . . .	18
Figura 4 – Determinando $r$ (Azul) a partir de $q$ (Vermelho) e $f$ (Amarelo) . . . . .	19
Figura 5 – Exemplo do GNG. a)0 sinais b)2800 sinais c) 9400 sinais d) 20000 sinais.	19
Figura 6 – Resultado após 20000 sinais . . . . .	20
Figura 7 – Triangulação de Delaunay de 16 pontos (linhas pontilhadas) . . . . .	22
Figura 8 – Diagrama de Voronoi . . . . .	22
Figura 9 – Etapas da Metodologia Proposta . . . . .	23
Figura 10 – Exemplo de resultado do GNG (Vermelho: Dados, Azul: Nós da rede) .	25
Figura 11 – Agrupamento por Componentes Conectados (grupos separados pela curva) . . . . .	26
Figura 12 – Separação nó a nó do grupo 3 da Figura 11 . . . . .	27
Figura 13 – Resultado da separação nó a nó do grupo 3 da Figura 11 . . . . .	27
Figura 14 – Triangulação de Delaunay dos grupos da Figura 13 . . . . .	30
Figura 15 – Triangulação de Delaunay dos grupos da Figura 13 após o corte das arestas acima da média . . . . .	31
Figura 16 – Grupos finais após a união . . . . .	32
Figura 17 – Resultados do C-GNG em Imperatriz . . . . .	36
Figura 18 – Agrupamento atual feito manualmente em Imperatriz . . . . .	36
Figura 19 – Recorte da Figura 17 ilustrando o grupo 0, o grupo 17 e o grupo 16 . .	37
Figura 20 – Resultado do C-GNG em Açailândia . . . . .	39
Figura 21 – Agrupamento atual feito manualmente em Açailândia . . . . .	39
Figura 22 – Recorte da Figura 20 ilustrando os grupos mais de perto . . . . .	40

# Lista de abreviaturas e siglas

ANEEL	Agência Nacional de Energia Elétrica
CSV	Arquivo de valores separados por virgula (Comma Separated Value file)
BFS	Breadth-First Search
C-GNG	Capacitated Growing Neural Gas
GRASP	Greedy Randomized Adaptative Search
GNG	Growing Neural Gas
PPMC	Problema das P-Medianas Capacitado
PACC	Problema do Agrupamento Centrado Capacitado
QGIS	Quantum Geographic Information System

# Lista de símbolos

$\alpha$	Alfa
$\beta$	Beta
$\lambda$	Lambda
$\xi$	Xi
$\Delta$	Delta
$\leq$	Menor ou igual a
$\geq$	Maior ou igual a
$\forall$	Para todo
$\in$	Pertence
$\setminus$	Que não é
$\cup$	União

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
<b>1.1</b>	<b>Objetivo</b>	<b>13</b>
1.1.1	Objetivos Específicos	14
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>15</b>
<b>2.1</b>	<b>Agrupamento Capacitado</b>	<b>15</b>
<b>2.2</b>	<b><i>Growing Neural Gas (GNG)</i></b>	<b>16</b>
<b>2.3</b>	<b>Breadth-First Search (BFS)</b>	<b>20</b>
<b>2.4</b>	<b>Triangulação de Delaunay</b>	<b>21</b>
2.4.1	Diagrama de Voronoi	22
<b>3</b>	<b>METODOLOGIA</b>	<b>23</b>
<b>3.1</b>	<b>Aquisição dos dados</b>	<b>23</b>
<b>3.2</b>	<b>Execução do GNG</b>	<b>24</b>
<b>3.3</b>	<b>Agrupamento a partir dos componentes conectados</b>	<b>25</b>
<b>3.4</b>	<b>Redução dos grupos acima da capacidade</b>	<b>26</b>
<b>3.5</b>	<b>União dos grupos abaixo da capacidade</b>	<b>29</b>
<b>3.6</b>	<b>Métricas de avaliação</b>	<b>33</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>35</b>
<b>5</b>	<b>CONCLUSÃO</b>	<b>42</b>
	<b>REFERÊNCIAS</b>	<b>43</b>

# 1 Introdução

A leitura dos medidores de energia e entrega das contas para cada cliente são as atividades principais relacionadas ao processo de contas de uma empresa de distribuição de energia elétrica. No Brasil, a regulamentação das condições de distribuição de energia elétrica é feita pela Agência Nacional de Energia Elétrica (ANEEL). Essa regulamentação tenta melhorar a relação entre os provedores da energia e os consumidores. Dentre elas, a ANEEL estabelece que uma provedora tem certos intervalos para terminar o processo de medição (Agência Nacional de Energia Elétrica, 2010).

As companhias de distribuição de energia elétrica, tem a tarefa de organizar o processo de medição por meio de grupos de consumidores. Portanto, cada funcionário responsável pela leitura tem de ter rotas para percorrer na medição e entrega de contas de energia em cada grupo. Na prática cada município possui uma organização distinta dos grupos e subgrupos.

A homogeneidade e compacidade são características utilizadas para medir os grupos e seleção das unidades. Homogeneidade se refere a criação de grupos mais homogêneos possíveis em relação a tempo de trabalho, o que reduz os custos. Já a compacidade se refere a forma geográfica dos grupos, que devem ser os mais compactos possíveis. Essa segunda característica contribui para a geração de rotas efetivas, e outras atividades comerciais da companhia também podem usar essas definições de grupo. O que reduz o tempo total de medição dos funcionários, além de cortar gastos.

O processo de criação desses grupos é denominado o problema de distritamento. O objetivo do problema de distritamento é agrupar pequenos espaços territoriais em distritos contíguos não sobrepostos, dado uma função objetivo e restrições.

As aplicações clássicas deste problema são em casos de alocação de distritos eleitorais (RICCA; SCOZZARI; SIMEONE, 2011), (GARFINKEL; NEMHAUSER, 1970) e (HOJATI, 1996). Outras aplicações são na rede de transporte público, visto em (TAVARES-PEREIRA et al., 2007), a definição de áreas de venda e local de trabalho de vendedores (RÍOS-MERCADO; FERNÁNDEZ, 2009), (EASINGWOOD, 1973), (SHANKER; TURNER; ZOLTNERS, 1975) e (HESS; SAMUELS, 1971) e a definição de áreas de fabricantes e consumidores (FLEISCHMANN; PARASCHIS, 1988).

Alguns métodos propostos para solucionar o problema são o *tabu search* (BOZKAYA; ERKUT; LAPORTE, 2003), algoritmo evolucionário (AHMADI; OSMAN, 2005), *simulated annealing* (RICCA; SIMEONE, 2008) e GRASP (RÍOS-MERCADO; SALAZAR-ACOSTA, 2011).

O problema do agrupamento capacitado/distritamento capacitado é uma generalização do problema de distritamento, onde os distritos têm capacidade limitada. Esse problema possui diversas variações, como o Problema do Agrupamento Centrado Capacitado (CARVALHO; SALDANHA; MARTINS, 2015) e problema das P-Mediana Capacitado (STEFANELLO; MÜLLER, 2009).

O agrupamento é um problema vastamente estudado em diversas aplicações, incluindo redes neurais, inteligência artificial e estatística. Diversos algoritmos foram propostos na literatura para o agrupamento: K-Means (JAIN; DUBES, 1988), ISODATA (KAUFMAN; ROUSSEEUW, 2009), CLARA (KAUFMAN; ROUSSEEUW, 2009), CLARANS (NG; HAN, 2002), *Focusing Techniques* (ESTER; KRIEGEL; XU, 1995), P-CLUSTER (JUDD; MCKINLEY; JAIN, 1998), DBSCAN (ESTER et al., 1996) e *Growing Neural Gas* (FRITZKE, 1997).

O *Growing Neural Gas* (GNG) tem o objetivo de gerar uma estrutura de grafo que reflete a topologia dos dados de entrada. Como um gás preenche uma área, o GNG preenche o espaço dos dados de entrada, gerando uma estrutura topologicamente similar. As suas aplicações são em agrupamento, quantização de vetores, modelagem e *tracking* de objetos.

Esses aspectos reforçam a importância de desenvolver métodos computacionais eficientes que consigam melhorar o processo de medição de unidades de leitura e entrega de contas. Além de proporcionar uma melhora na qualidade de serviço, fortalecendo a satisfação dos clientes.

Este trabalho propõe uma alteração no algoritmo original do GNG, criando o *Capacitated Growing Neural Gas* (C-GNG), um método para criação de grupos capacitados aplicado no contexto de distribuição de energia das companhias do Brasil. Os resultados e métricas são então comparados com o agrupamento manual feito pelas companhias. O GNG foi escolhido por ter a característica de refletir topologicamente os dados e por isso almeja-se que os grupos gerados por ele mantenham as características topológicas dos dados.

## 1.1 Objetivo

Este trabalho tem por objetivo o desenvolvimento de um método que agrupa áreas de leitura de instalações de energia, utilizando o Capacitated Growing Neural Gas (C-GNG). Visando obter um agrupamento melhor que o atual, que foi gerado manualmente.

### 1.1.1 Objetivos Específicos

- Implementar o Growing Neural Gas (GNG) para particionar dados de maneira capacitada;
- Propôr uma solução para o problema de medição de consumo de energia de maneira capacitada;
- Analisar e comparar a solução proposta usando os dados com o estado atual e real das companhias.

## 2 Fundamentação Teórica

Este capítulo descreve as técnicas mais importantes para o entendimento do trabalho. Iniciando pelo Agrupamento Capacitado que será usado na etapa de agrupamento do método, de forma a medir e garantir que a capacidade máxima seja seguida, descrito na seção 2.1. Em seguida detalhamos o *Growing Neural Gas*, que é o algoritmo de agrupamento utilizado, descrito na seção 2.2. O *Breadth-First Search*, algoritmo utilizado para realizar operações no grafo gerado pelo GNG, descrito na seção 2.3. E por fim a triangulação de Delaunay que é uma técnica importante utilizada na etapa de união de grupos, descrito na seção 2.4.

### 2.1 Agrupamento Capacitado

O agrupamento capacitado é um problema de agrupamento que consiste em um número de indivíduos com dados pesos que devem ser particionados em  $p$  grupos com capacidades limitadas. Cada indivíduo deve ser atribuído somente a um grupo. No qual a soma dos pesos ou da capacidade do indivíduo não extrapole a capacidade máxima do problema. O objetivo é encontrar  $p$  indivíduos chamados de medianas, de onde a soma das distâncias para todos os outros indivíduos no grupo seja minimizada (FRANÇA; SOSA; PUREZA, 1999).

O agrupamento capacitado é um problema combinatorial complexo que foi provado ser NP-completo (MICHAEL; DAVID, 1979). Dada a sua complexidade não é esperado que algoritmos exatos lidem com problemas de dimensões encontradas no mundo real, e a maioria dos métodos propostos para resolvê-lo são baseados em heurísticas.

Considerando um conjunto de indivíduos  $I = \{1, 2, \dots, n\}$ , com uma matriz  $n \times n$  simétrica  $d_{ij}$ , que representa a distância euclidiana entre um par de indivíduos. Assumindo que  $d_{ij} > 0$  e que  $d_{ii} = 0, \forall i, j \in I$ . Um peso positivo  $w_i$  é atribuído a cada indivíduo  $i$ . Sendo  $J = \{1, 2, \dots, m\}$  o conjunto de grupos e  $W_j$  suas capacidades. A mediana do grupo  $j$  é definida pelo indivíduo no qual a soma das distâncias a todos os outros indivíduos do grupo  $j$  é minimizada. Para a simplicidade da notação, a capacidade de cada grupo,  $W$ , é assumida como igual.

A variável binária  $y_j$  é 1 se o indivíduo  $j$  é atribuído como mediana do grupo  $j \in J$  do contrário o valor é 0. A variável binária  $x_{ij}$  indica se o indivíduo  $i$  pertence ao grupo  $j$  ou não, 1 caso verdadeiro e 0 caso falso.

$$F(P) = \text{Min} \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \quad (2.1)$$



Sujeito a:

$$\sum_{j \in J} y_j = p \quad (2.2)$$

$$\sum_{j \in J} x_{ij} = 1 \quad i \in I \quad (2.3)$$

$$x_{ij} \leq y_j \quad i \in I, j \in J \quad (2.4)$$

$$\sum_{i \in I} w_i x_{ij} \leq W \quad j \in J \quad (2.5)$$

$$x_{ij}, y_j \in \{0, 1\} \quad i \in I, j \in J \quad (2.6)$$

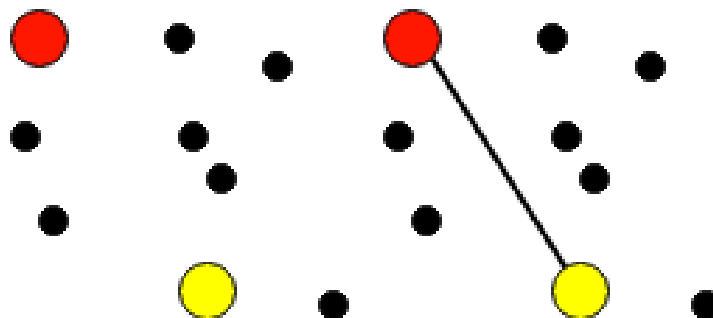
A restrição 2.2 garante que somente  $p$  grupos serão selecionados. A restrição 2.3 garante que todos os indivíduos serão atribuídos a um grupo, já a restrição 2.4 previne a atribuição de indivíduos que não foram selecionados como medianas. A restrição 2.5 garante que a soma dos pesos atribuídos ao grupo não extrapole a capacidade máxima.

## 2.2 Growing Neural Gas (GNG)

O *Growing Neural Gas* (GNG) é um algoritmo com o objetivo de gerar uma estrutura de grafo que reflete a topologia dos dados de entrada (aprendizado da topologia) (FRITZKE, 1997). Esse grafo tem uma dimensionalidade que varia com a dimensionalidade dos dados de entrada. A estrutura resultante pode então ser utilizada para identificar grupos no conjunto de entrada. Os itens de 1 a 13 abaixo, detalham o fluxo de funcionamento do GNG.

1. Inicia o conjunto  $A$  com indivíduos  $a$  e  $b$  em posições aleatórias  $w_a$  e  $w_b$  no  $R^n$ :
  - $A = \{a, b\}$
2. Inicia o conjunto de conexões  $C$  para conter a aresta entre  $a$  e  $b$  e atribui 0 à variável  $age$  dessa conexão:
  - $C = \{(a, b)\} \quad age_{(a,b)} = 0$

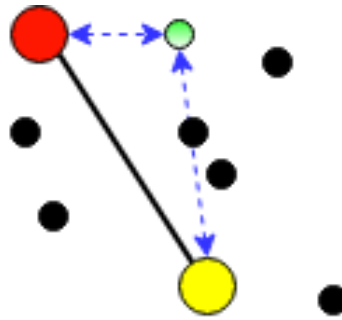
Figura 1 – Inicialização de  $a$  (vermelho) e  $b$  (amarelo)



Fonte: Acervo do autor

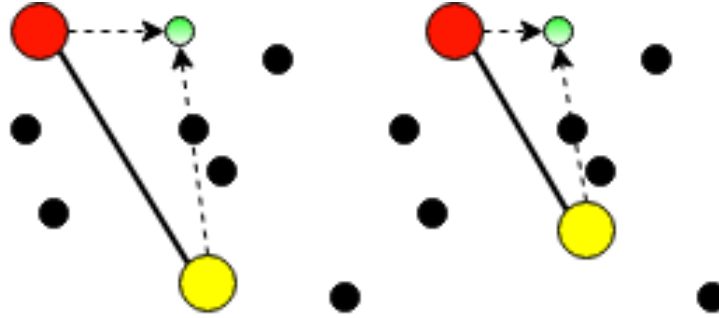
3. Gerar um sinal de entrada  $\xi$  de acordo com a função de densidade da probabilidade  $P(\xi)$ .
4. Determinar  $s_1$  e  $s_2$ ,  $(s_1, s_2) \in A$  de forma que:
  - $\|w_{s_1} - \xi\| \leq \|w_c - \xi\|, \forall c \in A$  e
  - $\|w_{s_2} - \xi\| \leq \|w_c - \xi\|, \forall c \in A \setminus s_1$

Figura 2 – Determinando  $s_1$  e  $s_2$  a partir do ponto da entrada



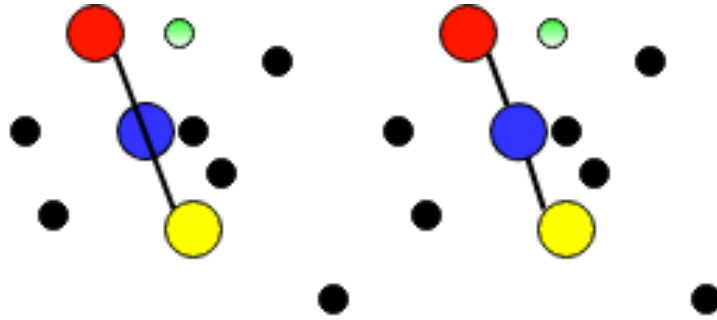
Fonte: Acervo do autor

5. Se ainda não existir, inserir uma conexão entre  $s_1$  e  $s_2$  em  $C$ :
  - $C = C \cup \{(s_1, s_2)\}$
6. De qualquer forma, atribui a variável *age* da conexão entre  $s_1$  e  $s_2$  para 0 ('atualizar' a aresta):
  - $age_{(s_1, s_2)} = 0$
7. Adicionar a distância ao quadrado entre o sinal de entrada e o indivíduo mais próximo para uma variável de erro local:
  - $\Delta E_{s_1} = \|w_{s_1} - \xi\|^2$
8. Mover  $s_1$  e seus vizinhos topológicos diretos em direção a  $\xi$  pelas frações  $\epsilon_b$  e  $\epsilon_n$ , respectivamente, a distância total:
  - $\Delta w_{s_1} = \epsilon_b (\xi - w_{s_1})$
  - $\Delta w_i = \epsilon_n (\xi - w_i) \quad (\forall i \in N_{s_1})$

Figura 3 – Aproximação de  $s_1$  (Vermelho) e vizinhos em direção a  $\xi$ 

Fonte: Acervo do autor

9. Incrementar a variável *age* de todas as arestas de  $s_1$ :
  - $age_{(s_1,i)} = age_{(s_1,i)} + 1 \quad (\forall i \in N_{s_1})$
  - onde  $N_c$  é um conjunto de vizinhos topológicos diretos de  $c$
10. Remover todas as arestas com *age* maior que  $a_{max}$ . Se isso resultar em indivíduos sem vizinhos, removê-los também.
11. Se o número de sinais de entrada gerados é um múltiplo inteiro do parâmetro  $\lambda$ , inserir um novo indivíduo da seguinte forma:
  - Determinar o indivíduo  $q$  com o erro acumulado máximo:
    - $E_q \geq E_c \quad (\forall c \in A)$
  - Interpolar o novo indivíduo  $r$  de  $q$  e seu vizinho  $f$  com a variável de erro maior:
    - $A = A \cup \{r\} \quad w_r = 0.5(w_q + w_f)$
  - Inserir arestas conectando o novo indivíduo  $r$  com  $q$  e  $f$  e remover a aresta entre  $q$  e  $f$ :
    - $C = C \cup \{(r, q), (r, f)\} \quad C = C \setminus (q, f)$
  - Reduzir as variáveis de erro de  $q$  e  $f$ :
    - $\Delta E_q = -\alpha E_q$
    - $\Delta E_f = -\alpha E_f$
  - Interpolar as variáveis de erro de  $r$  a partir de  $q$  e  $f$ :
    - $E_r = 0.5(E_q + E_f)$

Figura 4 – Determinando  $r$  (Azul) a partir de  $q$  (Vermelho) e  $f$  (Amarelo)

Fonte: Acervo do autor

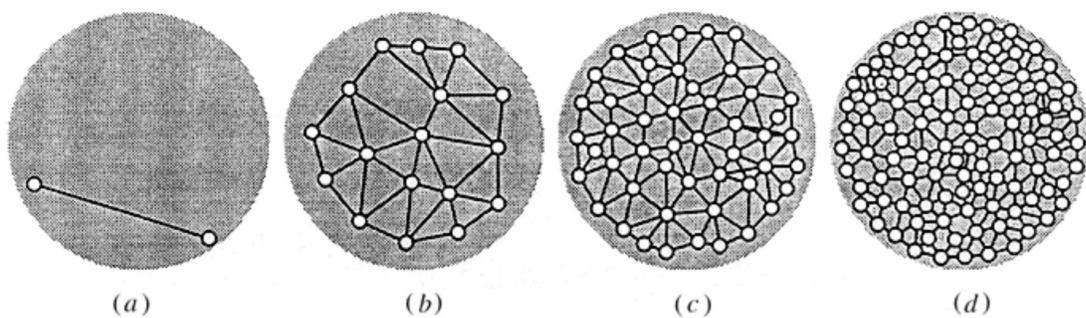
12. Reduzir as variáveis de erro de todos os indivíduos:

- $\Delta E_c = -\beta E_c \quad (\forall c \in A)$

13. Se o critério de parada ainda não foi satisfeito (ex.: tamanho da rede ou alguma medida de performance), continuar com o passo 3

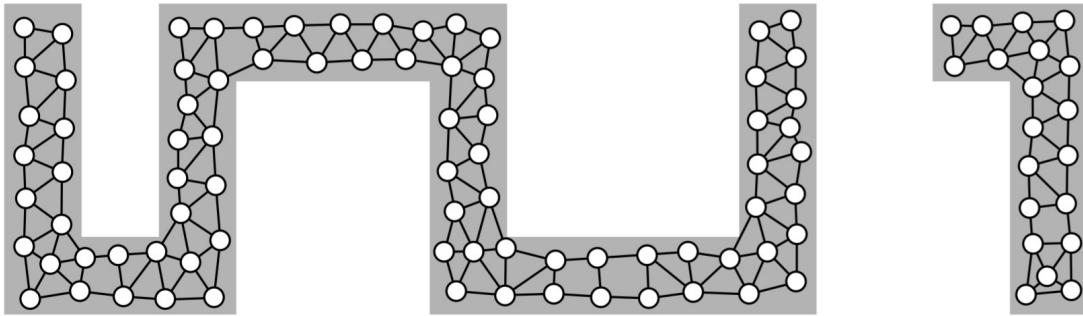
O algoritmo apresenta diversas vantagens: o tamanho da rede não precisa ser predefinido, todos os parâmetros são constantes e o número total de iterações de adaptação não precisam ser definidos *a priori* e o processo pode ser interrompido quando um critério for satisfeito. A desvantagem é que a visualização da rede é limitada a dimensão da entrada, visto que, a redução da dimensionalidade não é prevista em redes com aprendizado topológico.

Figura 5 – Exemplo do GNG. a) 0 sinais b) 2800 sinais c) 9400 sinais d) 20000 sinais.



Fonte: (FRITZKE, 1997)

Figura 6 – Resultado após 20000 sinais



Fonte: (FRITZKE, 1997)

## 2.3 Breadth-First Search (BFS)

O Breadth-First Search é um algoritmo simples de busca em um grafo e é base para diversos algoritmos.

Dado um grafo  $G = (V, E)$  e um vértice de origem  $s$  o BFS sistematicamente percorre os vértices de  $G$  para “descobrir” cada vértice que é alcançado a partir de  $s$ . Calcula a distância de todos os vértices alcançáveis até  $s$  (menor número de vértices) e produz uma árvore que possui como raiz o vértice  $s$ , contendo todos os vértices alcançáveis.

Para manter o progresso, o BFS (Algoritmo 1) utiliza três cores: branco, cinza e preto. Todos os vértices começam com a cor branca, os vértices pretos representam vértices já percorridos e os vértices cinza representam os vértices que foram vistos. Dessa forma

um vértice cinza pode vir a ter um vértice branco adjacente, mas um vértice preto não.

---

**Algoritmo 1:** Breadth-First Search
 

---

```

1 Function Breadth-First Search( $G, s$ )
2   for cada vértice  $u \in G.V - \{s\}$  do
3      $u.cor = BRANCO$ 
4      $u.d = \infty$ 
5      $u.\pi = NIL$ 
6    $s.cor = CINZA$ 
7    $s.d = 0$ 
8    $s.\pi = NIL$ 
9    $Q = []$ 
10  Enqueue( $Q, s$ )
11  while  $Q \neq []$  do
12     $u = Dequeue(Q)$ 
13    for cada  $v \in G.Adj[u]$  do
14      if  $v.cor == BRANCO$  then
15         $v.cor = CINZA$ 
16         $v.d = u.d + 1$ 
17         $v.\pi = u$ 
18        Enqueue( $Q, v$ )
19     $u.cor = PRETO$ 

```

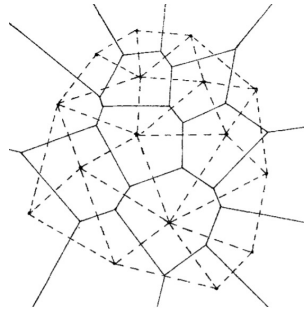
---

O algoritmo tem complexidade  $O(V + E)$  onde  $V$  é o número de vértices e  $E$  o número de arestas. Como apresentado em (CORMEN et al., 2009), o BFS descobre todos os vértices na distância  $k$  de um vértice antes de descobrir qualquer vértice na distância  $k + 1$ . Por isso os vizinhos mais próximos de um vértice sempre serão visitados primeiro.

## 2.4 Triangulação de Delaunay

A triangulação de Delaunay utiliza a estrutura formada pelo diagrama de Voronoi, explicado na seção 2.4.1, e constrói triângulos a partir dos pontos dos planos. Considerando um conjunto  $V$  com  $n \geq 3$  pontos distintos e assumindo que estes não são colineares uma triangulação é um grafo onde nenhuma aresta se intersecta (LEE; SCHACHTER, 1980). A triangulação de Delaunay pode ser feita utilizando os polígonos formados pelo diagrama de Voronoi, para cada vizinho do polígono, uma aresta é formada entre seus centros. Outra definição da triangulação de Delaunay é que cada círculo circunscrito a um triângulo é um círculo vazio (OKABE et al., 2009). Ou seja, os círculos não contêm nenhum dos pontos do conjunto.

Figura 7 – Triangulação de Delaunay de 16 pontos (linhas pontilhadas)

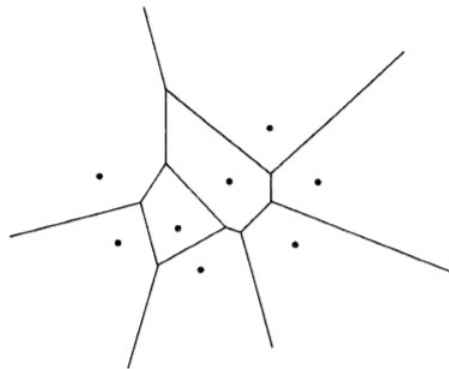


Fonte: (LEE; SCHACHTER, 1980)

### 2.4.1 Diagrama de Voronoi

Dado um conjunto  $S$  de  $n$  pontos no plano queremos determinar para cada ponto  $p$  de  $S$  qual é a região  $V(p)$  dos pontos do plano que estão mais próximos de  $p$  do que de qualquer outro ponto em  $S$  (AURENHAMMER; EDELSBRUNNER, 1984). As regiões determinadas por cada ponto formam o Diagrama de Voronoi como pode ser visto na Figura 8.

Figura 8 – Diagrama de Voronoi



Fonte: (AURENHAMMER, 1991)

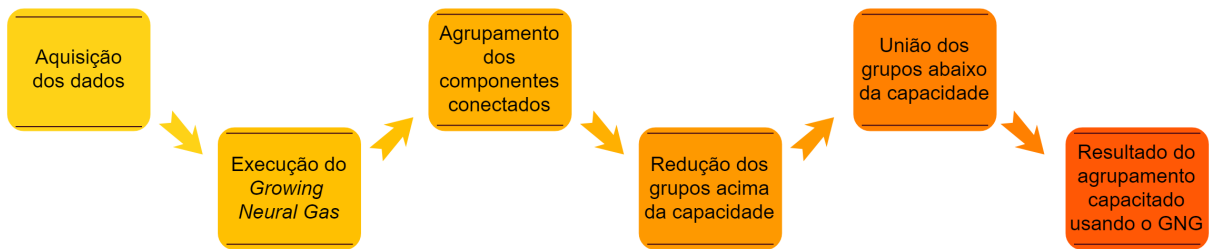
Definição formal: seja  $X$  um conjunto não vazio com função de distância  $d$ . Seja  $K$  um conjunto de índices e seja  $(P_k)_{k \in K}$  uma tupla de subconjuntos não vazios no espaço  $X$ . A célula de Voronoi ou região de Voronoi,  $R_k$ , associada com o ponto  $P_k$  é o conjunto de todos os pontos em  $X$  cuja distância a  $P_k$  não é maior que a suas distâncias a outros pontos  $P_j$ , onde  $j$  é qualquer índice diferente de  $k$ . Em outras palavras, se  $d(x, A) = \inf\{d(x, a) \mid a \in A\}$  denota a distância do ponto  $x$  e o subconjunto  $A$ , então:  $R_k = \{x \in X \mid d(x, P_k) \leq d(x, P_j) \forall j \neq k\}$ .

O diagrama de Voronoi é simplesmente a tupla de elementos  $(R_k)_{k \in K}$  (AURENHAMMER, 1991).

## 3 Metodologia

Este capítulo descreve o método proposto para o agrupamento das etapas de leitura usando o GNG Capacitado. A Figura 9 apresenta a sequência de etapas propostas neste trabalho.

Figura 9 – Etapas da Metodologia Proposta



Fonte: Acervo do autor

Os pontos utilizados para o agrupamento são as instalações de medida. Cada instalação tem um aparelho com o objetivo de medir a energia elétrica utilizada em uma residência. Um grupo de instalações recebe a denominação Etapa pelas companhias de energia, que consiste na área na qual os funcionários realizam a leitura em um dia.

### 3.1 Aquisição dos dados

As bases de dados utilizadas neste trabalho foram fornecidas por duas empresas de distribuição de energia elétrica. As bases utilizadas possuem informações das cidades de Imperatriz e Açailândia no Maranhão. Os dados se encontram em um arquivo csv, onde cada linha possui a informação de uma instalação. As informações de cada instalação são a latitude, longitude, etapa e tempo, representada na Tabela 3.1.

A latitude e longitude são as coordenadas geográficas de cada instalação. A etapa é a denominação dos grupos de instalações que irão ser medidas em um dia, estes grupos foram definidos manualmente por especialistas da companhia. O tempo indica o tempo necessário em minutos para que um funcionário consiga ler o medidor.



LAT	LONG	ETAPA	TEMPO
-5.4927806	-47.4836327	10	1.13
-5.5106509	-47.4862713	8	0.55
-5.5182095	-47.4355126	18	0.58
-5.5188013	-47.4514449	16	0.59
-5.4999448	-47.4669273	13	1.28
-5.5008595	-47.4804245	10	0.53
-5.523097227	-47.49366387	5	1.01
-5.523097227	-47.49366387	6	0.52

Tabela 1 – Configuração da base de dados

## 3.2 Execução do GNG

O próximo passo da metodologia é a aplicação do *Growing Neural Gas*, para isso é necessário realizar uma normalização dos valores de latitude e longitude para possibilitar o cálculo da distância euclidiana. Consideramos o raio da terra como 6371 km e aplicamos as seguintes transformações nas coordenadas:

$$X = R * \cos(\text{latitude}) * \cos(\text{longitude}) \quad (3.1)$$

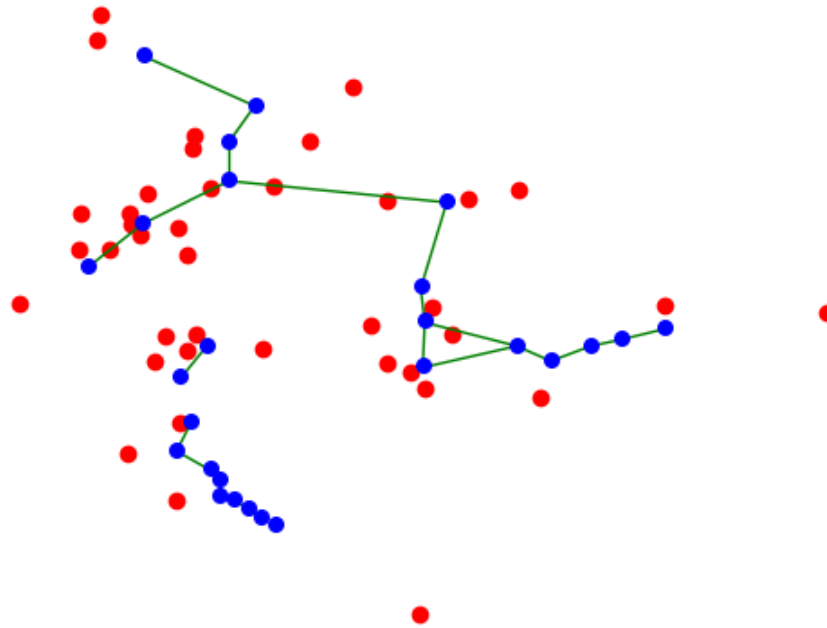
$$Y = R * \cos(\text{latitude}) * \sin(\text{longitude}) \quad (3.2)$$

Devido ao elevado número de elementos, de forma a reduzir o tempo necessário para a execução do algoritmo, foi adicionado um limite de nós como critério de parada. Utilizamos como capacidade máxima o tempo total de todos os elementos, dividido pelo número de etapas e adicionamos uma tolerância de 10%.

O GNG é executado na base de dados utilizando os seguintes parâmetros  $\epsilon_b = 0.1$ ,  $\epsilon_n = 0.006$ ,  $a_{max} = 5$ ,  $\lambda = 20$ ,  $\alpha = 0.5$ ,  $\beta = 0.995$ , Número máximo de nós = 3000, Capacidade máxima =  $\frac{\text{tempo dos elementos}}{\text{número de etapas}} * 1.1\%$ . Como saída do GNG obtemos um grafo que representa topologicamente os elementos da base de dados.

A Figura 10 apresenta um exemplo da execução do GNG em uma base de 39 elementos (em vermelho). Após o término do algoritmo, foi gerado um grafo com 26 nós (em azul).

Figura 10 – Exemplo de resultado do GNG (Vermelho: Dados, Azul: Nós da rede)



Fonte: Acervo do autor

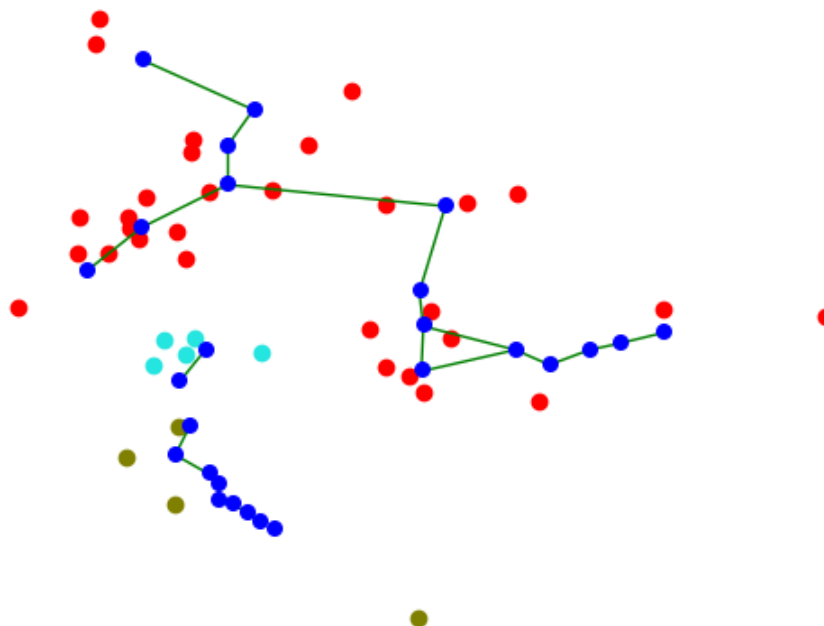
Pontos a agrupar (Vermelho)	Nós da rede (Azul)
39	26

Tabela 2 – Informação da Figura 10

### 3.3 Agrupamento a partir dos componentes conectados

Após o término da execução do GNG cada componente conectado da rede é atribuído um número começando de 0, que representa o grupo. Para cada elemento da base é checado a distância do elemento para cada nó do grafo e para o nó com a menor distância ao elemento é atribuído ao elemento o número do grupo do componente conectado a qual aquele nó pertence. Na Figura 11 ilustramos um exemplo de resultado do agrupamento de componentes conectados, utilizando os dados apresentados na Figura 10. Após o agrupamento, o Grupo 0 (em vermelho) ficou com 30 elementos, o Grupo 1 (em azul claro) ficou com 5 e o Grupo 2 ficou com 4 elementos.

Figura 11 – Agrupamento por Componentes Conectados (grupos separados pela curva)



Fonte: Acervo do autor

	Grupo 0	Grupo 1	Grupo 2
N#	30	5	4

Tabela 3 – Informação da Figura 11

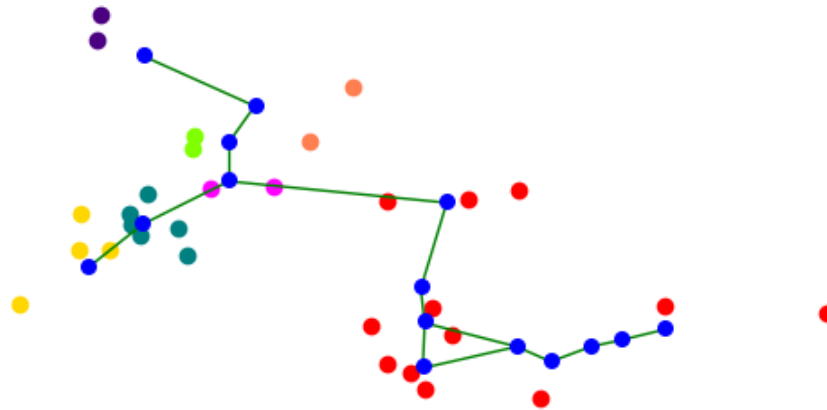
### 3.4 Redução dos grupos acima da capacidade

Depois do agrupamento inicial é necessário checar a capacidade de cada grupo. Como o GNG não utiliza nenhum parâmetro para checar a capacidade dos grupos não é possível garantir que os grupos iniciais possuem a capacidade menor ou igual a capacidade máxima. Logo é necessário reduzir os grupos que estão acima da capacidade máxima e unir os grupos que estão abaixo da capacidade máxima. Para isso executamos primeiramente o algoritmo de redução dos grupos.

A redução dos grupos é realizada selecionando cada elemento da base, achando o nó mais próximo a ele, adicionando então o elemento em um novo grupo identificado com o índice do nó. Para cada grupo que excede a capacidade máxima, executamos o BFS iniciado no primeiro nó ao qual aquele grupo pertence.

Percorremos o grafo pelo BFS e checamos a capacidade de cada nó, representada pelos grupos identificados com os índices dos nós, se essa capacidade mais a capacidade total atual for menor que a máxima, adicionamos a capacidade à capacidade total e adicionamos os elementos no grupo, se não, assumimos a capacidade total como a capacidade do nó e criamos um novo grupo adicionando os elementos nele. Até que não sobre mais nenhum nó que não pertença a um grupo.

Figura 12 – Separação nó a nó do grupo 3 da Figura 11



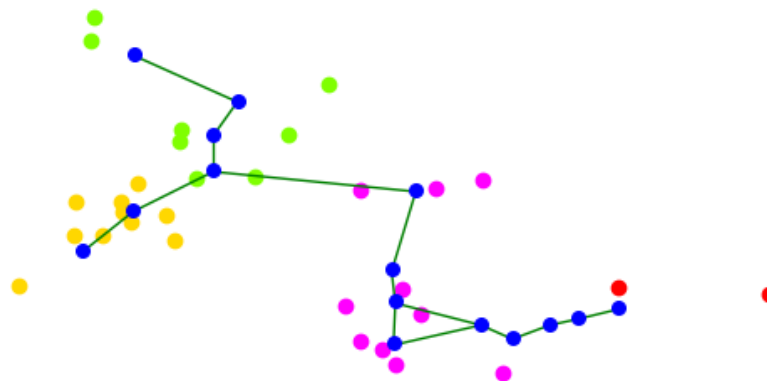
Fonte: Acervo do autor

	Grupo a	Grupo b	Grupo c	Grupo d	Grupo e	Grupo f	Restante
N#	4	6	2	2	2	2	12

Tabela 4 – Informação da Figura 12

Para exemplificar o funcionamento da redução dos grupos acima da capacidade selecionamos somente o grupo 0 da Figura 11. Assumindo que a capacidade máxima é 10 e que cada ponto tem a capacidade 1, o grupo 0 então tem capacidade total de 30. Para cada nó da rede checamos os pontos mais próximos e os agrupamos. Após todo esse processo o que nos resta são mais de 7 grupos como podemos ver na Figura 12.

Figura 13 – Resultado da separação nó a nó do grupo 3 da Figura 11



Fonte: Acervo do autor

	Grupo 0	Grupo 1	Grupo 2	Grupo 3
N#	10	8	10	2

Tabela 5 – Informação da Figura 13

O próximo passo é selecionar o primeiro nó que pertence ao grupo acima da capacidade. Dessa forma selecionamos o nó que contém o grupo a (na Figura 12 em

amarelo) e aplicamos o BFS com origem nele. Como saída do BFS temos:  $a \rightarrow b \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow \text{Restante}$ . O grupo  $a$  tem a capacidade igual a 4, checamos se a sua capacidade mais a capacidade do próximo grupo, que é o grupo  $b$ , ultrapassa a capacidade máxima ( $4 + 6$ ). Como isso não ocorre, adicionamos os elementos de  $b$  no grupo 0 e então checamos a capacidade do grupo 0 mais a capacidade do grupo  $c$  ( $10 + 2$ ), como ela ultrapassa a capacidade máxima, incrementamos o contador dos grupos e adicionamos os elementos de  $c$  no grupo 1 e assim percorremos todo o grafo até que os grupos  $d$ ,  $e$  e  $f$  se unam ao grupo 1 e os 12 elementos restantes formem o grupo 2 e 3, Como ilustra a Figura 13. O pseudocódigo deste método é apresentado no Algoritmo 2.

**Algoritmo 2:** Redução dos grupos

---

**entrada:** conjunto de grupos acima da capacidade  
**saída:** conjunto de grupos novos

```

1 Function ReduzirCluster(grupos_acima)
2   inicializar matriz grupo_temp[x][y], x = num de nós e y = [ ]
3   for cada elemento da base do
4     achar o id do nó mais próximo id_no
5     adicionar o elemento ao vetor grupo_temp na posição id_no
6   inicializar o vetor no_visitado com zeros com tamanho igual ao número de nós
7   grupo_atual = zero
8   for cada grupo n em grupos_acima do
9     cap_total = 0
10    bfs = resultado do BFS iniciado no primeiro nó de n
11    for cada nó x de bfs do
12      if no_visitado[x]  $\neq$  1 then
13        cap_atual = capacidade de cada elemento de grupo_temp[x] if
14          capacidade_total + capacidade_atual < capacidade_maxima then
15            cap_total = cap_total + cap_atual
16            for cada elemento i de grupo_temp[x] do
17              adiciona ao vetor novos_grupos na posição grupo_atual cada
18              elemento i
19            else
20              cap_total = cap_atual
21              grupo_atual = grupo_atual + 1
22              for cada elemento i de grupo_temp[x] do
23                adiciona ao vetor novos_grupos na posição grupo_atual cada
24                elemento i
25      no_visitado[x] = 1
26   retornar novos_grupos

```

---

### 3.5 União dos grupos abaixo da capacidade

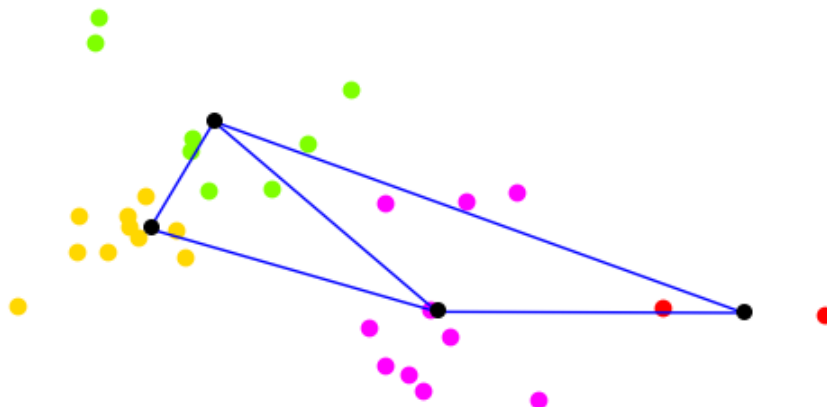
Depois da redução dos grupos acima da capacidade podemos garantir que os grupos estejam com a capacidade abaixo do máximo. Mas não temos como garantir que estes estejam próximos ou iguais à capacidade máxima para o problema, por isso, é preciso fazer uma união dos grupos com baixa capacidade. Para isso utilizamos o método de união proposto em (COSTA et al., 2018).

Além da capacidade, é necessário que um grupo seja unido com outros grupos próximos a ele. Para isso é criado um grafo que os conectam, cada nó do grafo será o centroide de um grupo. Para a definição das arestas do grupo foi utilizado a triangulação de Delaunay. Gerar um grafo completamente conectado exigiria de muito poder computacional e não seria prático por causa do tempo necessário para calcular os pesos entre cada nó, que é a distância euclidiana. Também vale ressaltar que conectar cada nó entre si resultariam em uniões de grupos muito distantes. Por isso um grafo não completo é o suficiente. Chew (1986) discute que é possível um grafo tão bom quanto um grafo completo que pode ser gerado pela triangulação de Delaunay.

O método consiste em selecionar o conjunto de grupos, encontrar seus centroides e fazer uma triangulação de Delaunay. Cortar arestas com distância maior que a distância média de todas as arestas. Para cada vértice do grafo, checar se sua capacidade está abaixo da capacidade máxima, caso verdadeiro, realizar um BFS a partir desse vértice. Percorrer o grafo usando o resultado do BFS e checar se a capacidade do vértice  $b$  somada ao vértice inicial  $a$  é igual ou menor que a capacidade máxima. Caso verdadeiro, unir os dois vértices em um grupo e marcar o vértice  $b$  como já visitado. Caso o valor da soma seja maior que a capacidade máxima retirar os  $n$  elementos do vértice  $b$  que estão mais próximos ao vértice inicial e satisfaçam a capacidade máxima e adicionar ao grupo do vértice  $a$ .

Continuando o exemplo da seção 3.4. A Figura 14 mostra o resultado da triangulação de Delaunay usando os mesmos parâmetros do exemplo anterior da seção 3.4, os pontos pretos representam os centroides e as arestas azuis representam as conexões do grafo resultante.

Figura 14 – Triangulação de Delaunay dos grupos da Figura 13



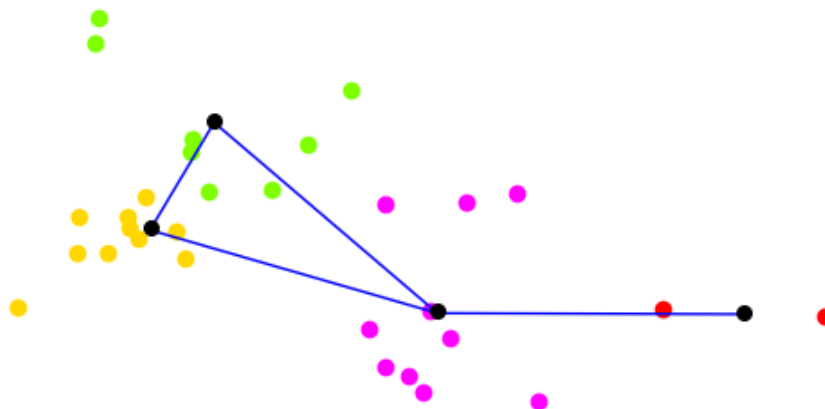
Fonte: Acervo do autor

	Grupo 0	Grupo 1	Grupo 2	Grupo 3
N#	10	8	10	2

Tabela 6 – Informação da Figura 14

O próximo passo após a triangulação é checar a distância das arestas e calcular a distância média. A aresta entre o grupo 1 e o grupo 2 tem uma distância maior que a distância média encontrada e é então removida, como apresentado na Figura 15.

Figura 15 – Triangulação de Delaunay dos grupos da Figura 13 após o corte das arestas acima da média



Fonte: Acervo do autor

	Grupo 0	Grupo 1	Grupo 2	Grupo 3
N#	10	8	10	2

Tabela 7 – Informação da Figura 15

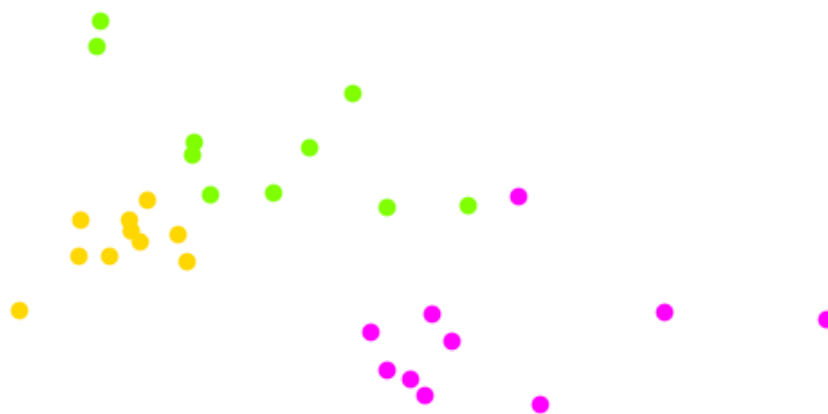
Checamos então a capacidade do grupo 0 e vimos que ele possui a capacidade igual a máxima. Passamos para o grupo 1 e checamos que ele possui capacidade igual a 8, iniciamos então o BFS começando no seu centroide, e temos como resultado:  $1 \rightarrow 0 \rightarrow 2 \rightarrow 3$ .

Verificamos então o primeiro nó após o nó inicial, o centroide do grupo 1, checamos que ele já foi visitado e então passamos para o próximo nó, o centroide do grupo 2. Como ele ainda não foi visitado, checamos sua capacidade e somamos com a capacidade do grupo 1 ( $8 + 10$ ) e vimos que o resultado é maior que a capacidade máxima. Seleccionamos então o primeiro ponto mais próximo do centroide de 1, checamos se a capacidade do ponto mais a capacidade do grupo não ultrapassa a capacidade máxima ( $8 + 1$ ). Seleccionamos o próximo ponto mais próximo do centroide de 1 e verificamos que sua capacidade, somada com a capacidade do grupo 1, não ultrapassa a capacidade máxima ( $9 + 1$ ). Fazemos o mesmo para outro ponto e vemos que agora a soma das capacidades ultrapassa a capacidade máxima, por isso marcamos o grupo 1 como visitado e vamos então checar o grupo 2.

A capacidade do grupo 2 somada com a capacidade do grupo 3 não ultrapassa a capacidade máxima, marcamos então 2 e 3 como visitados. Como não restam mais nós a união dos grupos é completada, a Figura 16 mostra o resultado final. O pseudocódigo deste método é apresentado Algoritmo 3.



Figura 16 – Grupos finais após a união



Fonte: Acervo do autor

	Grupo 0	Grupo 1	Grupo 2
N#	10	10	10

Tabela 8 – Informação da Figura 16

**Algoritmo 3:** União dos grupos

---

```

entrada: conjunto de grupos
saída: conjunto de grupos novos
1 Function UnirCluster(grupos)
2   for cada elemento i de grupos do
3     | calcular os centroides de cada grupo e colocar no vetor centroides
4     executar uma triangulação de Delaunay usando centroides
5     calcular a distância média entre cada aresta da triangulação
6     remover arestas com distância maior que a distância média
7   for cada elemento i de grupos do
8     | if capacidade do grupo i < capacidade_maxima e i não visitado then
9       | bfs = BFS do grafo resultante da triangulação de Delaunay iniciado em i
10      | for cada elemento j de bfs sem contar com i do
11        | if capacidade de i + capacidade de j then
12          | adicionar todos os elementos de j a i
13          | marcar j como visitado
14        | else
15          | if j não foi visitado then
16            | achar elementos de j mais próximos de i e ordená-los em um
17              | vetor chamado ordenado
18            | for cada elemento k de ordenado do
19              | if capacidade de k + capacidade de i  $\leq$ 
20                | capacidade_maxima then
21                  | adicionar k ao grupo i
22                | else
23                  | break
24            | marcar i como visitado
25      | retornar novos_grupos

```

---

Após esse passo já temos os grupos finais do GNG Capacitado.

### 3.6 Métricas de avaliação

Para validar o método utilizado, foram selecionadas métricas que calculam a eficiência dos grupos em relação ao estado atual.

O que se busca em um agrupamento capacitado, além dos grupos se manterem abaixo ou iguais a capacidade máxima, é o balanceamento dos tamanhos. Pelo desvio

padrão podemos calcular se os grupos estão balanceados ou não, quanto maior o desvio, mais desbalanceados e quanto menor, mais balanceados.

A validação geral de um grupo pode ser expressa em termos da coesão e separação (TAN et al., 2007). A coesão quantifica o quão próximo os objetos dentro de um grupo estão e a separação determina o quão isolado um grupo está de outro. A coesão de um grupo é calculada pelo somatório da distância euclidiana de cada ponto do grupo com o seu centroide. Em relação a coesão, valores elevados são melhores, já para a separação, o que se busca são valores baixos. A coesão e separação podem ser definidas pelas seguintes expressões:

$$Coesão(C_i) = \sum_{x \in C_i} distância(x, c_i) \quad (3.3)$$

$$Separação(C_i, C_j) = distância(c_i, c_j) \quad (3.4)$$

onde  $C_i$  é o  $i$ -ésimo grupo e  $c_i$  é o seu centroide. A função  $distância(a, b)$  retorna o resultado da distância euclidiana entre o  $a$  e  $b$ .

Outra forma de medir a qualidade de um grupo individualmente é pelo coeficiente de silhueta (SC). O SC pode variar de -1 a 1 quanto mais próximo de 1, maior a compacidade do grupo.

$$SC(C_i) = \frac{b_x - a_x}{max(a_x, b_x)} \quad (3.5)$$

onde  $a_x$  é o resultado da média das distâncias entre um elemento aleatório  $x$  e os outros elementos do grupo  $C_i$ .  $b_x$  é a média das distâncias do elemento  $x$  até os pontos do grupo mais próximo de  $C_i$  que não contém  $x$ . A função  $max(a, b)$  retorna o valor máximo entre  $a$  e  $b$ .

Calculamos então a média da coesão de todos os grupos, a média da separação entre cada grupo com o grupo mais próximo dele e a média do SC de cada grupo.

## 4 Resultados e Discussão

Para a validação dos métodos foram realizados testes nas bases de Imperatriz - MA e Açailândia - MA. Foi utilizado o programa *QGIS* ([QGIS Development Team, 2019](#)) para a plotagem dos dados e visualização da informação geoespacial.

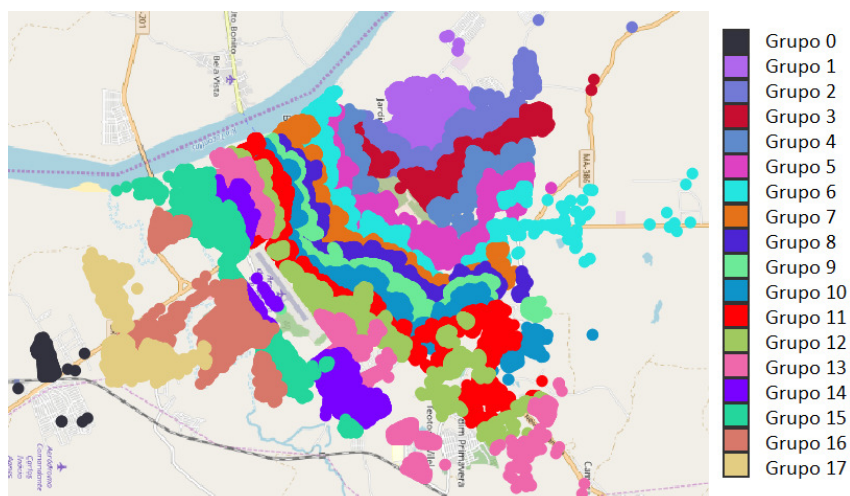
Para o agrupamento realizado na base de Imperatriz temos uma capacidade máxima de 8135.07, 182297 instalações e um total de 18 etapas, numeradas de 2-18 e 31. A Tabela 9 apresenta o tempo, a média do tempo e o desvio padrão de cada grupo para os dados atuais e os dados encontrados. Os parâmetros utilizados foram os mesmos que podem ser encontrados no Capítulo 3 na Seção 3.2. A Tabela 9 mostra que grupo com o maior tempo não ultrapassou a capacidade máxima. Mas o grupo com o menor tempo tem a capacidade 12 vezes menor que capacidade máxima. Também observamos que o desvio padrão encontrado, quando comparado ao estado atual, foi menor. Isso mostra que o agrupamento encontrado pelo GNG Capacitado é melhor que o agrupamento atual.

Tabela 9 – Dados resumidos Imperatriz - MA

	<b>ATUAL</b>	<b>ENCONTRADO</b>	<b>DIFERENÇA</b>
<b>Média</b>	7395.52	<b>7395.52</b>	0
<b>Desvio Padrão</b>	2563.62	<b>2030.86</b>	532.75
<b>Menor Tempo (min)</b>	449.9	<b>645.21</b>	195.3
<b>Maior Tempo (min)</b>	10869.88	<b>8135.06</b>	2734.82

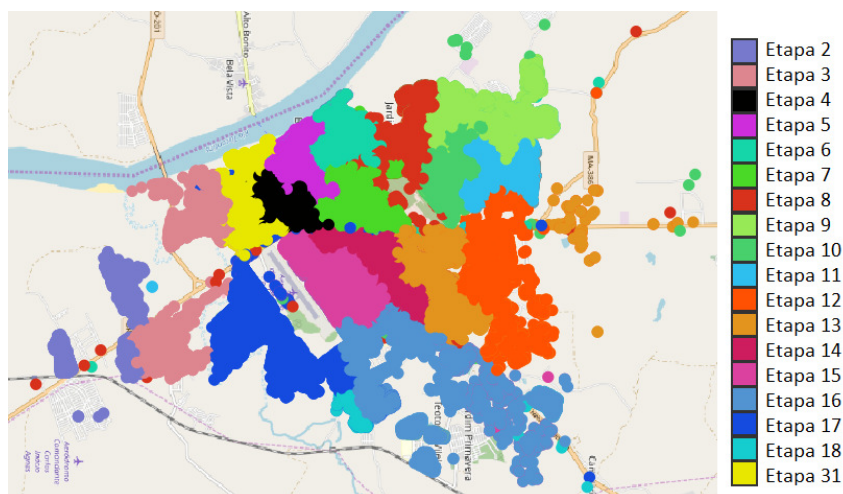
A Figura 17 apresenta os grupos gerados pelo C-GNG e a Figura 18 apresenta as etapas definidas manualmente. Podemos observar que os grupos gerados pelo C-GNG não possuem uma forma circular, diferente de métodos de agrupamento convencionais como K-Means. Os grupos mantêm essa característica apesar do método para a união utilizar os centroides dos grupos.

Figura 17 – Resultados do C-GNG em Imperatriz



Fonte: Acervo do autor

Figura 18 – Agrupamento atual feito manualmente em Imperatriz



Fonte: Acervo do autor

O interessante no resultado obtido pelo C-GNG é que os grupos gerados obedecem a uma forma linear que corresponde a trajetos em cidades formados pelo arruamento. Essa característica pode ser interessante quando se deseja organizar a logística de acesso a esses lugares baseando-se no arruamento da própria cidade.

Na Tabela 10 são apresentados o detalhamento dos dados de cada grupo encontrado e dos grupos feitos manualmente pelas companhias de distribuição elétrica, mais especificamente o número de instalações em cada grupo e o tempo ou a capacidade total de cada grupo.

O grupo 0 encontrado é aquele que possui o menor número de elementos e o menor tempo total, mas podemos ver que ele possui tempo e número de elementos maior que a

etapa 18, que é a etapa com o menor número de elementos e com o tempo total menor. O grupo 3 é aquele que tem o maior tempo e o maior número de elementos. Ele possui um menor número de elementos que a etapa maior, mas ainda obedece a capacidade máxima.

Tabela 10 – Dados específicos Imperatriz - MA

ATUAL			ENCONTRADO		
Etapa	Tempo (min)	Quantidade	Grupos	Tempo (min)	Quantidade
2	1863.25	3568	0	645.21	1847
3	7963.49	11961	1	8134.73	11938
4	9127.98	14989	2	8135.03	11535
5	6993.61	8750	3	8135.06	12243
6	7546.64	10243	4	8134.99	11625
7	8556.27	11325	5	8135.04	9765
8	7179.23	11392	6	8135.04	10682
9	6951.05	9260	7	8134.68	10644
10	7501.9	10315	8	8134.92	11065
11	6752.19	10060	9	8134.04	10347
12	7521.65	8063	10	8134.85	10125
13	9205.71	10726	11	8134.87	10968
14	7943.22	11980	12	8134.76	10571
15	9319.26	10647	13	8134.72	11428
16	10869.88	13137	14	8134.76	11617
17	10212.42	12243	15	8134.54	12194
18	449.91	1023	16	7088.6	9376
31	7161.8	12615	17	3363.62	4327

Para explicar o porque do grupo 0 ser o grupo com o menor tempo selecionamos uma área da Figura 17 que ilustra uma visão ampliada desse grupo.

Figura 19 – Recorte da Figura 17 ilustrando o grupo 0, o grupo 17 e o grupo 16



Fonte: Acervo do autor

Como podemos analisar através da Figura 19, o grupo 0 é o mais distante dos outros e por isso, possivelmente, ele foi o último grupo a ser percorrido no BFS na etapa da união. Então os elementos dele foram removidos e adicionados ao grupo mais próximo, como ele era o último grupo todos os outros já estavam marcados como lidos. Outra possibilidade é que as arestas que ligavam o grupo 0 aos outros grupos tinham uma distância maior que a distância média e por isso foi excluída, colocando o grupo 0 fora do grafo e assim não podendo retirar elementos de nenhum outro grupo.

A Tabela 11 apresenta os valores de coesão média, separação média e coeficiente de silhueta (SC) médio. Podemos ver que em todos os valores o C-GNG teve um desempenho melhor que o estado atual, apesar de que no coeficiente de silhueta o valor foi negativo (muito próximo de zero).

Tabela 11 – Valores de Coesão, Separação e SC médios em Imperatriz - MA

	MANUAL	C-GNG	DIFERENÇA
<b>Coesão</b>	371995.56	<b>662704.73</b>	290709.17
<b>Separação</b>	51.46	<b>30.24</b>	21.21
<b>SC</b>	-0.501419	<b>-0.006277</b>	0.495141

Para o agrupamento realizado na base de Açailândia temos uma capacidade máxima de 1092.03, um total de 34760 instalações e 17 etapas numeradas de 2-18. A Tabela 12 mostra a média e desvio padrão do tempo de cada grupo e mostra também o maior tempo e o menor tempo dos grupos. Podemos ver que a média dos grupos encontrados aumentou em relação aos grupos atuais e apresenta um valor próximo da capacidade máxima. O desvio padrão por sua vez diminuiu de 174.82 para 117.69. Com esses dois valores podemos ver que os grupos encontrados são mais balanceados que os grupos feitos manualmente.

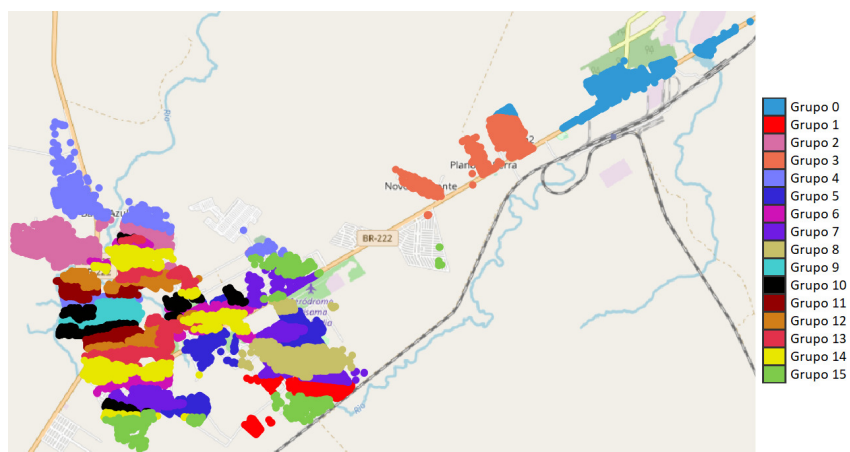
Tabela 12 – Dados resumidos Açailândia - MA

	ATUAL	ENCONTRADO	DIFERENÇA
<b>Média</b>	992.76	<b>1054.80</b>	62.04
<b>Desvio Padrão</b>	174.82	<b>117.69</b>	57.13
<b>Menor Tempo (min)</b>	677.3	<b>631.14</b>	46.16
<b>Maior Tempo (min)</b>	1439.56	<b>1092.03</b>	347.53

A Figura 20 traz o resultado do C-GNG na base de Açailândia e a Figura 21 mostra o agrupamento feito manualmente. O que mais se destaca é o fato de os grupos não terem o formato ondulado como no resultado em Imperatriz. O grupo 14 por exemplo tem vários grupos entre os seus elementos.

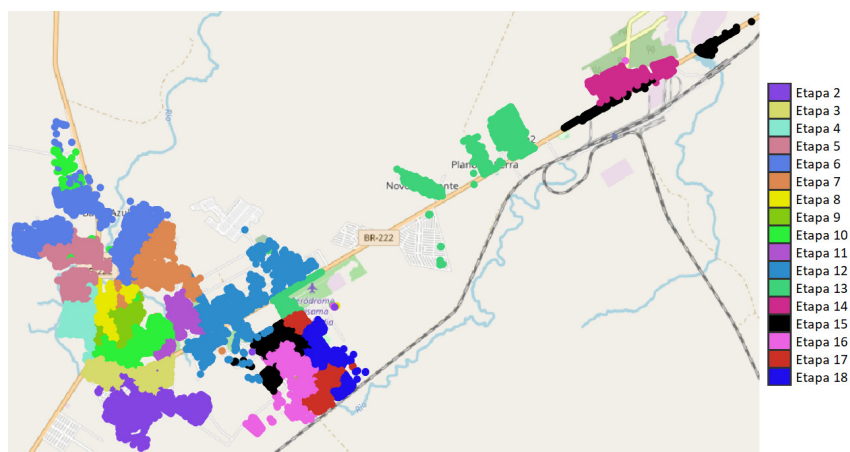
O grupo 15 é aquele que possui o menor número de elementos, e por consequência o menor tempo, como podemos verificar na Tabela 13. O grupo 15 provavelmente era o último na execução do BFS, sobrando assim nenhum outro grupo para retirar elementos.

Figura 20 – Resultado do C-GNG em Açailândia



Fonte: Acervo do autor

Figura 21 – Agrupamento atual feito manualmente em Açailândia



Fonte: Acervo do autor

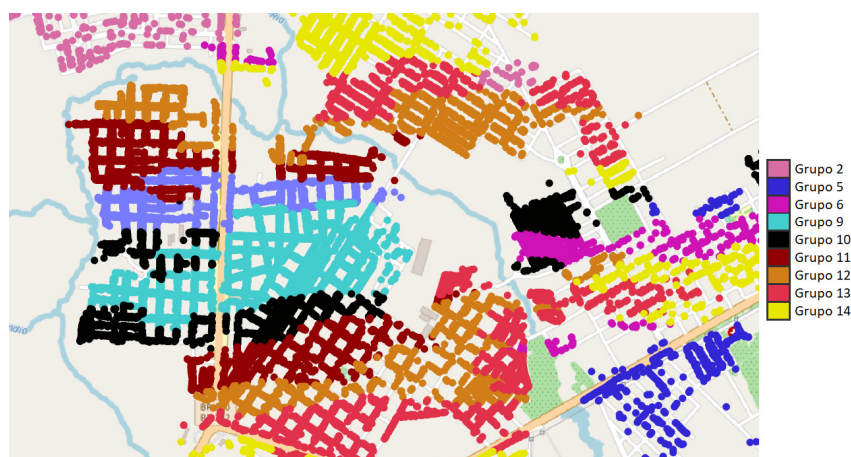
Pela Tabela 13 podemos ver que o C-GNG têm a capacidade de agrupar as instalações em menos grupos (na prática, 1 dia menos de trabalho). Porém estes grupos não possuem uma compacidade alta.



Tabela 13 – Dados específicos Açailândia - MA

ANTES			ENCONTRADO		
Etapa	Tempo (min)	Quantidade	Grupos	Tempo (min)	Quantidade
2	892.08	2119	0	1091.86	2437
3	964.65	1733	1	1091.94	1972
4	897.66	1944	2	1091.95	1614
5	1097.55	2084	3	959.66	1692
6	1439.56	2472	4	1091.89	1990
7	1076.22	2197	5	1091.99	2800
8	1040.76	2105	6	1091.86	2041
9	767.51	1975	7	1091.96	2666
10	1100.96	2371	8	1091.74	2328
11	677.3	1697	9	1092.02	2789
12	1095.25	1932	10	1092.03	2275
13	1162.61	2191	11	1091.69	2373
14	858.37	1859	12	1091.88	2213
15	900.8	1790	13	1091.72	2332
16	970.81	1726	14	1091.62	2255
17	1066.39	2287	15	631.14	983
18	868.47	2278			

Figura 22 – Recorte da Figura 20 ilustrando os grupos mais de perto



Fonte: Acervo do autor

Tabela 14 – Valores de Coesão, Separação e SC médios em Açailândia - MA

	MANUAL	C-GNG	DIFERENÇA
<b>Coesão</b>	54652.95	<b>69799.38</b>	15146.43
<b>Separação</b>	435.81	<b>474.71</b>	38.89
<b>SC</b>	-0.340341	<b>-0.054340</b>	0.286000

Os valores de coesão, separação e SC para Açailândia são apresentados na Tabela 14. O que se destaca é o valor da separação, que se mostra maior que o anterior. Esse valor

reflete a distribuição dos grupos, eles têm um aspecto mesclado, como ilustra a Figura 22. A coesão e SC se mostram melhores que o anterior, mas a diferença não foi muito grande.

## 5 Conclusão

O trabalho apresentou um método capaz de agrupar corretamente instalações de energia elétrica em etapas, que é o nome dado a cada dia de leitura, implementando uma adaptação do *Growing Neural Gas* para agrupar levando em conta a capacidade de cada instalação.

O método utilizado obteve um agrupamento mais eficiente que o atual para os dois casos descritos na seção 4, melhorando não só os índices de separação e coesão, mas também melhorando o coeficiente de silhueta e gerando uma versão otimizada com carga de trabalho balanceada e menor em termos de dias.

Mesmo obtendo bons resultados, verificou-se que o método apresentado possui uma limitação em não apresentar necessariamente grupos que estejam totalmente conectados. Sendo que esta pode ser uma característica desejável para uma solução final, visto que facilita a divisão de equipes de trabalho. Verificou-se que a limitação se encontra principalmente no mecanismo de refinamento de grupos, realizado através de junções.

Como trabalhos futuros, planeja-se modificar a união dos grupos com capacidade abaixo do máximo. Para gerar diversos agrupamentos com o BFS iniciando em nós aleatórios. Podendo assim comparar estes diversos agrupamentos, visando uma homogeneidade maior e por consequência um agrupamento mais eficiente. Também planejamos testar o C-GNG em uma área rural, visto que os grupos gerados nos testes se assemelham com a disposição das instalações rurais.

## Referências

Agência Nacional de Energia Elétrica. *Resolução Normativa Nº 414, de 9 de Setembro de 2010*. 2010. Disponível em: <<http://www.aneel.gov.br/documents/656877/14486448/bren2010414.pdf/3bd33297-26f9-4ddf-94c3-f01d76d6f14a?version=1.0>>. Citado na página 12.

AHMADI, S.; OSMAN, I. H. Greedy random adaptive memory programming search for the capacitated clustering problem. *European Journal of Operational Research*, Elsevier, v. 162, n. 1, p. 30–44, 2005. Citado na página 12.

AURENHAMMER, F. Voronoi diagrams&mdash;a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 23, n. 3, p. 345–405, set. 1991. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/116873.116880>>. Citado na página 22.

AURENHAMMER, F.; EDELSBRUNNER, H. An optimal algorithm for constructing the weighted voronoi diagram in the plane. *Pattern Recognition*, v. 17, n. 2, p. 251 – 257, 1984. ISSN 0031-3203. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0031320384900645>>. Citado na página 22.

BOZKAYA, B.; ERKUT, E.; LAPORTE, G. A tabu search heuristic and adaptive memory procedure for political districting. *European Journal of Operational Research*, Elsevier, v. 144, n. 1, p. 12–26, 2003. Citado na página 12.

CARVALHO, R. de; SALDANHA, R. R.; MARTINS, A. X. Heurística aplicada ao problema de agrupamento centrado capacitado. *Anais do XLVII SBPO, Simpósio Brasileiro de Pesquisa Operacional*, 2015. Citado na página 13.

CHEW, P. There is a planar graph almost as good as the complete graph. In: *Proceedings of the Second Annual Symposium on Computational Geometry*. New York, NY, USA: ACM, 1986. (SCG '86), p. 169–177. ISBN 0-89791-194-6. Disponível em: <<http://doi.acm.org/10.1145/10515.10534>>. Citado na página 30.

CORMEN, T. H. et al. *Introduction to algorithms*. [S.l.]: MIT press, 2009. Citado na página 21.

COSTA, P. et al. Rmeans: A recursive capacitated means for districting problem. *Journal on Advances in Theoretical and Applied Informatics*, v. 4, n. 1, p. 28–35, 2018. ISSN 2447-5033. Disponível em: <<http://revista.univem.edu.br/jadi/article/view/2745>>. Citado na página 29.

EASINGWOOD, C. A heuristic approach to selecting sales regions and territories. *Operational Research Quarterly (1970-1977)*, Palgrave Macmillan Journals, v. 24, n. 4, p. 527–534, 1973. ISSN 00303623. Disponível em: <<http://www.jstor.org/stable/3008330>>. Citado na página 12.

ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Knowledge Discovery and Data Mining*. [S.l.: s.n.], 1996. v. 96, n. 34, p. 226–231. Citado na página 13.

- ESTER, M.; KRIEGEL, H.-P.; XU, X. Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification. In: SPRINGER. *International Symposium on Spatial Databases*. [S.l.], 1995. p. 67–82. Citado na página 13.
- FLEISCHMANN, B.; PARASCHIS, J. N. Solving a large scale districting problem: a case report. *Computers Operations Research*, v. 15, n. 6, p. 521 – 533, 1988. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0305054888900482>>. Citado na página 12.
- FRANÇA, P. M.; SOSA, N. M.; PUREZA, V. An adaptive tabu search algorithm for the capacitated clustering problem. *International Transactions in Operational Research*, Wiley Online Library, v. 6, n. 6, p. 665–678, 1999. Citado na página 15.
- FRITZKE, B. *Unsupervised ontogenetic networks; Handbook of Neural Computation*. [S.l.]: IOP Publishing and Oxford University Press, 1997. C2.4:1-16 p. Citado 4 vezes nas páginas 13, 16, 19 e 20.
- GARFINKEL, R. S.; NEMHAUSER, G. L. Optimal political districting by implicit enumeration techniques. *Manage. Sci.*, INFORMS, Institute for Operations Research and the Management Sciences (INFORMS), Linthicum, Maryland, USA, v. 16, n. 8, p. B-495–B-508, abr. 1970. ISSN 0025-1909. Disponível em: <<http://dx.doi.org/10.1287/mnsc.16.8.B495>>. Citado na página 12.
- HESS, S. W.; SAMUELS, S. A. Experiences with a sales districting model: criteria and implementation. *Management Science*, INFORMS, v. 18, n. 4-part-ii, p. P-41, 1971. Citado na página 12.
- HOJATI, M. Optimal political districting. *Computers Operations Research*, v. 23, n. 12, p. 1147 – 1161, 1996. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054896000299>>. Citado na página 12.
- JAIN, A. K.; DUBES, R. C. Algorithms for clustering data. Prentice-Hall, Inc., 1988. Citado na página 13.
- JUDD, D.; MCKINLEY, P. K.; JAIN, A. K. Large-scale parallel data clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 20, n. 8, p. 871–876, 1998. Citado na página 13.
- KAUFMAN, L.; ROUSSEEUW, P. J. *Finding groups in data: an introduction to cluster analysis*. [S.l.]: John Wiley & Sons, 2009. v. 344. Citado na página 13.
- LEE, D.-T.; SCHACHTER, B. J. Two algorithms for constructing a delaunay triangulation. *International Journal of Computer & Information Sciences*, Springer, v. 9, n. 3, p. 219–242, 1980. Citado 2 vezes nas páginas 21 e 22.
- MICHAEL, R. G.; DAVID, S. *JOHNSON: Computers and Intractability: A Guide to the Theory of NP-Completeness*. [S.l.]: Freeman, 1979. Citado na página 15.
- NG, R. T.; HAN, J. Clarans: A method for clustering objects for spatial data mining. *IEEE transactions on knowledge and data engineering*, IEEE, v. 14, n. 5, p. 1003–1016, 2002. Citado na página 13.
- OKABE, A. et al. *Spatial tessellations: concepts and applications of Voronoi diagrams*. [S.l.]: John Wiley & Sons, 2009. v. 501. Citado na página 21.

- QGIS Development Team. *QGIS Geographic Information System*. 2019. Disponível em: <<https://qgis.org/en/site/>>. Citado na página 35.
- RICCA, F.; SCOZZARI, A.; SIMEONE, B. Political districting: from classical models to recent approaches. *4OR*, v. 9, n. 3, p. 223, Jul 2011. ISSN 1614-2411. Disponível em: <<https://doi.org/10.1007/s10288-011-0177-5>>. Citado na página 12.
- RICCA, F.; SIMEONE, B. Local search algorithms for political districting. *European Journal of Operational Research*, Elsevier, v. 189, n. 3, p. 1409–1426, 2008. Citado na página 12.
- RÍOS-MERCADO, R. Z.; SALAZAR-ACOSTA, J. C. A grasp with strategic oscillation for a commercial territory design problem with a routing budget constraint. In: SPRINGER. *Mexican International Conference on Artificial Intelligence*. [S.l.], 2011. p. 307–318. Citado na página 12.
- RÍOS-MERCADO, R. Z.; FERNÁNDEZ, E. A reactive grasp for a commercial territory design problem with multiple balancing requirements. *Computers Operations Research*, v. 36, n. 3, p. 755 – 776, 2009. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054807002249>>. Citado na página 12.
- SHANKER, R. J.; TURNER, R. E.; ZOLTNERS, A. A. Sales territory design: an integrated approach. *Management Science*, INFORMS, v. 22, n. 3, p. 309–320, 1975. Citado na página 12.
- STEFANELLO, F.; MÜLLER, F. M. Um estudo sobre problemas de agrupamento capacitado. *Anais do XLI SBPO, Simpósio Brasileiro de Pesquisa Operacional*, p. 2819–2828, 2009. Citado na página 13.
- TAN, P.-N. et al. *Introduction to data mining*. [S.l.]: Pearson Education India, 2007. Citado na página 34.
- TAVARES-PEREIRA, F. et al. Multiple criteria districting problems. *Annals of Operations Research*, v. 154, n. 1, p. 69–92, Oct 2007. ISSN 1572-9338. Disponível em: <<https://doi.org/10.1007/s10479-007-0181-5>>. Citado na página 12.