

Micael António da Silva Rama

# Árvores de Steiner Euclidianas

Setembro de 2013



UNIVERSIDADE DE COIMBRA





UNIVERSIDADE DE COIMBRA  
FACULDADE DE CIÊNCIAS E TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

# Árvores de Steiner Euclidianas

Micael António da Silva Rama

Coimbra, 2013





UNIVERSIDADE DE COIMBRA  
FACULDADE DE CIÊNCIAS E TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

Dissertação submetida para obtenção do grau de Mestre em Engenharia  
Electrotécnica e de Computadores

## Árvores de Steiner Euclidianas

Orientador(a):

Dra. Lúcia Maria dos Reis Albuquerque Martins

Júri:

Dr. Carlos Alberto Henggeler de Carvalho Antunes (Presidente)

Dra. Lúcia Maria dos Reis Albuquerque Martins

Dra. Rita Cristina Girão Coelho da Silva

Setembro 2013



# Agradecimentos

Em primeiro lugar, à Professora Doutora Lúcia Martins agradeço o apoio e a confiança depositada, ao ter aceite o desafio de orientar esta dissertação.

Aos meus pais e ao meu irmão que, apesar das dificuldades, sempre me apoiaram e incentivaram a concluir esta fase importante da minha vida, mesmo quando as esperanças escasseavam.

À minha namorada pelo seu companheirismo, apoio moral e ajuda a todos os níveis, principalmente por entender a falta de atenção e de tempo.

Ao meu amigo Luís Garrote pelas sugestões, interesse e ânimo ao longo desta dissertação.

Ao Departamento de Engenharia Electrotécnica e de Computadores (DEEC) que disponibilizaram as suas instalações para a realização desta dissertação.

À Universidade de Coimbra, por me ter permitido crescer enquanto pessoa.

A todos aqueles que de uma forma ou de outra, me ajudaram nesta caminhada.

**A todos os meus sinceros agradecimentos.**





# Resumo

Nesta dissertação é estudado o problema de Steiner Euclidiano com e sem obstáculos. Para a resolução destes problemas existem heurísticas bem como algoritmos exatos complexos. Tendo em vista a sua posterior adaptação ao traçado de redes de telecomunicações é ainda proposta uma nova heurística para resolução do problema de Steiner com obstáculos.

Palavras chaves: Caminhos mais curtos; Árvore abrangente mínima; Árvore de Steiner Euclidiana; Árvore de Steiner Euclidiana com obstáculos; Planeamento de redes de telecomunicações.



# Abstract

In this master thesis the Euclidean Steiner problem and the obstacle-avoiding Euclidean Steiner problem are studied. To solve these problems some heuristics and more complex exact algorithms had been proposed. In order to be applied into topological design of telecommunication networks a new heuristic for obstacle-avoiding Euclidean Steiner problem is herein proposed.

Keywords: Shortest paths; Minimal spanning tree; Euclidean Steiner tree; Obstacle-avoiding Euclidean Steiner tree; Design of telecommunication networks.



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>O problema de determinação de árvores de Steiner Euclidianas</b>	<b>3</b>
2.1	Introdução . . . . .	3
2.2	Definições e Propriedades . . . . .	4
2.3	Algoritmo de Melzak . . . . .	7
2.4	Algoritmo proposto por P. Winter e M. Zachariasen . . . . .	9
2.4.1	Projeções . . . . .	10
2.4.2	Propriedade <i>lune</i> . . . . .	11
<b>3</b>	<b>Árvores de Steiner Euclidianas com obstáculos</b>	<b>13</b>
3.1	Introdução . . . . .	13
3.2	Árvore abrangente mínima . . . . .	14
3.3	Árvore de Steiner Euclidiana . . . . .	15
3.4	Obstáculos . . . . .	17
<b>4</b>	<b>Análise de desempenho dos algoritmos desenvolvidos</b>	<b>23</b>
4.1	Introdução . . . . .	23
4.2	Resultados obtidos para o problema de Steiner sem obstáculos . . . . .	23
4.3	Resultados obtidos para o problema de Steiner com obstáculos . . . . .	25
<b>5</b>	<b>Conclusões</b>	<b>35</b>
5.1	Trabalho futuro . . . . .	35



# Lista de Tabelas

2.1	Número de FSTs de acordo com fórmula 2.3. . . . .	8
3.1	Casos possíveis de análise dos ângulos e soluções. . . . .	17
4.1	Comparação de resultados para os 46 casos disponíveis em [11], relativos ao problema de Steiner sem obstáculos. . . . .	25
4.2	Resultados numéricos relativos às figuras 4.3, 4.4 e 4.5. . . . .	28





# Lista de Figuras

2.1	<i>Ponto de Torricelli</i> proposto por Torricelli e Simpson. . . . .	4
2.2	Condição de ângulo. . . . .	7
2.3	FSTs possíveis para $K = 4$ . . . . .	8
2.4	Construção de uma FST para $K = 5$ . . . . .	9
2.5	Projeções [12]. . . . .	10
2.6	Propriedade <i>lune</i> . . . . .	11
3.1	Evolução do algoritmo de Prim. . . . .	14
3.2	Exemplo de duas MSTs para a mesma árvore. . . . .	15
3.3	Exemplo de MST única. . . . .	15
3.4	Exemplo de MSTO. . . . .	18
3.5	Exemplos do problema de Steiner com obstáculos. . . . .	19
3.6	Exemplo de localização de obstáculos na SMT. . . . .	20
3.7	Escolha de localização do nó de Steiner. . . . .	22
4.1	SMT construída através da heurística e do algoritmo exato para o caso 11 de [11]. . . . .	26
4.2	SMT construída através da heurística e do algoritmo exato para o caso 30 de [11]. . . . .	27
4.3	Árvores de Steiner com obstáculos [14]. . . . .	29
4.4	Árvores obtidas através da heurística para os primeiros seis exemplos da figura 4.3. . . . .	30
4.5	Árvores obtidas através da heurística para os últimos cinco exemplos da figura 4.3. . . . .	31
4.6	Carta topográfica com sobreposição da SMT. . . . .	32
4.7	Carta topográfica com proposta inicial de um traçado. . . . .	33
4.8	Carta topográfica com proposta de um novo traçado através da MST e SMT. . . . .	33
4.9	Carta topográfica com proposta de um novo traçado através da MSTO e SMTO. . . . .	34



# Lista de Acrónimos

- **ESMT** - Euclidean Steiner Minimal Tree
- **FST** - Full Steiner Tree
- **MST** - Minimum Spanning Tree
- **MSTO** - Obstacles-Avoiding Minimum Spanning Tree
- **SMT** - Steiner Minimal Tree
- **SMTO** - Obstacles-Avoiding Steiner Minimal Tree
- **ST** - Steiner Tree
- **STP** - Steiner Tree Problem



# Capítulo 1

## Introdução

O problema de Steiner tem como objetivo encontrar a rede mínima (uma árvore) para interligação de determinados nós, designados por nós terminais. Informalmente, o objetivo é que a soma dos custos dos caminhos (ou das distâncias) entre todos os nós dessa rede seja a menor possível. Para permitir a obtenção dessa árvore de custo mínimo podem ser usados outros nós, designados por nós de Steiner.

Nesta dissertação é estudado um caso particular do problema de Steiner que é o problema de determinação de árvores de Steiner Euclidianas. Neste caso a árvore pretendida é obtida no plano cartesiano sendo a distância Euclidiana a métrica a minimizar. Este problema é considerado como sendo *NP-hard* [4], não sendo conhecido um algoritmo que permita encontrar a solução ótima em tempo polinomial. Outros tipos de árvores de Steiner são as árvores de Steiner retilíneas [13] e as árvores de Steiner em grafos.

As árvores de Steiner Euclidianas têm aplicação em vários campos tais como o planeamento de redes de comunicação e de redes elétricas, o desenho de circuitos elétricos, a definição de traçados de vias de comunicação, entre outros. No entanto, no âmbito desta dissertação, o estudo deste problema teve como motivação o desenvolvimento de um sistema de apoio ao planeamento de redes de comunicação, nomeadamente ao nível da escolha de novos traçados de fibra ótica, ou de qualquer outro meio físico utilizado nas interligações destas redes. A utilização deste sistema pode então conduzir, no caso de utilização da fibra ótica, ao traçado que necessita da menor quantidade de fibra para a interligação de um determinado conjunto de pontos extremos na rede como é o caso de novos clientes, por exemplo.

No projeto de um novo traçado de fibra ótica, por exemplo, o problema de Steiner Euclidiano não tem, em geral, uma aplicação prática dado que existem muitas vezes restrições à implementação da solução obtida. Os obstáculos são uma dessas restrições, pois na construção de novos traçados poderão existir locais que terão que ser forçosamente contornados por esses traçados, como sejam edifícios, cursos de água, etc.. Assim, nesta dissertação será estudado o problema de árvores de Steiner Euclidianas com obstáculos (*Euclidean Steiner Minimum Tree with Obstacles - ESMTO*).

Este trabalho teve como principal motivação a atividade profissional do autor desta dissertação numa empresa <sup>1</sup> de prestação de serviços de telecomunicações cuja atividade inclui a instalação e manutenção de traçados de redes.

---

<sup>1</sup>Telecomunicações Rama & Silva Lda.

**Objetivos**

Assim, os objetivos desta dissertação são os seguintes:

1. Estudo do Problema de Steiner Euclidiano, em particular dos algoritmos exatos para a sua resolução;
2. Implementação de uma heurística simples para obtenção de árvores de Steiner Euclidianas e para posterior adaptação ao problema de Steiner com obstáculos;
3. Estudo do problema de Steiner com obstáculos e desenvolvimento de uma heurística para a resolução deste problema.

**Conteúdo**

No capítulo 2 são apresentados os conceitos associados ao problema de Steiner Euclidiano nomeadamente os principais conceitos envolvidos na resolução exata deste problema. No capítulo 3 é apresentada uma heurística que resolve o problema de Steiner Euclidiano de forma aproximada. Essa heurística é de seguida usada para a resolução do problema de Steiner Euclidiano com obstáculos apresentando-se ainda neste capítulo uma nova heurística para a resolução deste problema. No capítulo 4 são apresentados alguns resultados que mostram o desempenho dos algoritmos desenvolvidos. Por fim, no capítulo 5 são apresentadas as conclusões deste trabalho, bem como algumas propostas de trabalho futuro.

## Capítulo 2

# O problema de determinação de árvores de Steiner Euclidianas

### 2.1 Introdução

O problema das árvores de Steiner Euclidianas, designado por *Euclidean Steiner Tree Problem (ESTP)*, resulta de um problema formulado inicialmente por Fermat no século XVII, que consistia em, dados três pontos no plano, encontrar um quarto ponto de forma que a soma das distâncias desse ponto adicional aos três pontos iniciais fosse mínima. Este problema foi designado como *Problema de Fermat*. Torricelli e Simpson encontraram uma solução para o *Problema de Fermat*.

Atribui-se a Torricelli (por volta do ano de 1640) a primeira solução do problema. Esta solução propunha a criação de um triângulo com os três pontos dados. Para cada lado do triângulo era proposta ainda a criação de um novo triângulo equilátero inscrito numa circunferência. Assim no final existiam três triângulos equiláteros inscritos em três circunferências. O ponto de interseção das três circunferências ficou designado como *ponto de Torricelli* [3] (figura 2.1). Mais tarde em 1750, Simpson [10] propôs uma alternativa à solução de Torricelli. Esta consistia em ligar os vértices exteriores dos triângulos equiláteros com o vértice do triângulo inicial. Estas três linhas são chamadas de *linhas de Simpson* (figura 2.1). A interseção das três linhas coincide com o *ponto de Torricelli*.

Entretanto, em 1647, Cavalieri mostrou que as linhas que ligam o *ponto de Torricelli* com os pontos dados formam  $120^\circ$  entre si.

Heinen, em 1834, provou que as *linhas de Simpson* têm o mesmo comprimento e que é igual à soma das distâncias entre o *ponto de Torricelli* e os três pontos dados inicialmente. Demonstrou ainda que esta construção falharia caso um dos ângulos internos do triângulo fosse superior a  $120^\circ$ . Neste caso, o método de Torricelli iria construir um ponto fora do triângulo e as *linhas de Simpson* não se intersectariam no mesmo ponto. Neste caso, o ponto que minimiza a soma das distâncias aos três pontos iniciais é num desses pontos (o vértice cujo ângulo é maior ou igual a  $120^\circ$ ).

Assim conclui-se que a primeira solução completa para o *Problema de Fermat* foi descoberta por Heinen e consiste no seguinte:

1. Se um dos ângulos internos do triângulo formado pelos pontos iniciais for superior ou igual a  $120^\circ$ , então o ponto que minimiza a soma das distâncias aos pontos iniciais é um desses pontos, ou seja, é o vértice do triângulo cujo ângulo é obtuso;

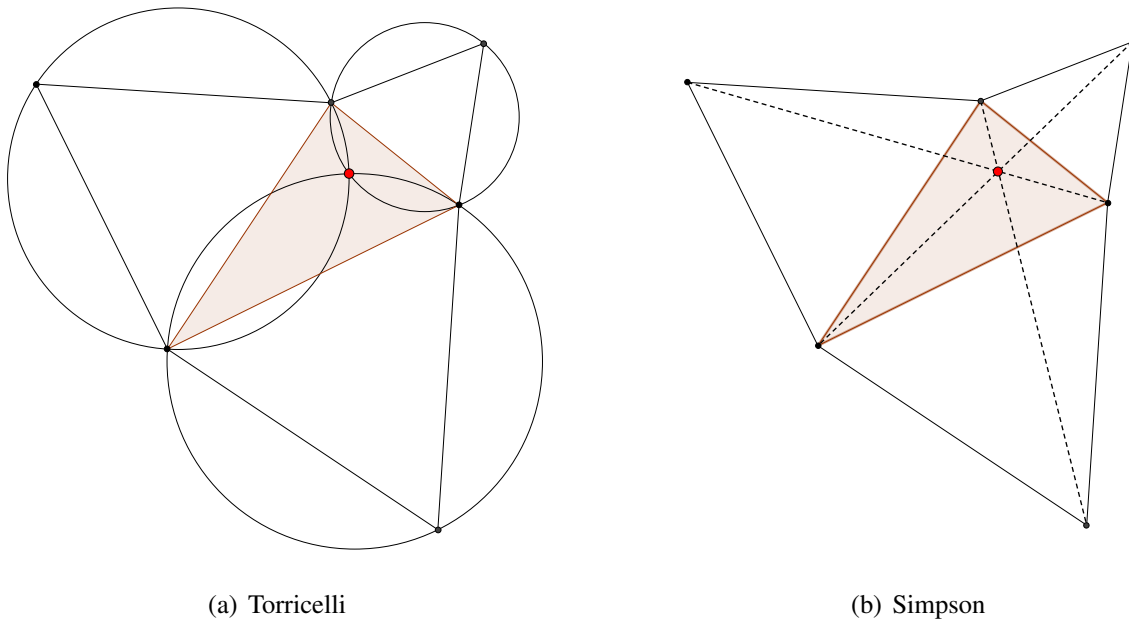


Figura 2.1: *Ponto de Torricelli* proposto por Torricelli e Simpson.

2. Se todos os ângulos internos do triângulo formado pelos pontos iniciais forem inferiores a  $120^\circ$  então o ponto que minimiza a soma das distâncias desse ponto aos pontos iniciais poderá ser obtido através de um dos métodos já referidos.

A generalização do problema de Fermat, isto é, determinação de um ponto num plano para o qual a soma das distâncias entre esse ponto e um conjunto de pontos pré-determinado é mínima veio dar origem mais tarde à formulação do problema conhecido por problema de Steiner<sup>1</sup> Euclidiano em que se pretende determinar a rede mínima para a interligação de  $n$  pontos num plano. O problema de Fermat é então o caso particular do problema de Steiner Euclidiano em que o conjunto de pontos a interligar tem apenas três pontos [3].

Melzak [9], em 1961, propôs a primeira abordagem para um algoritmo que permita resolver o problema das árvores de Steiner Euclidianas (*Euclidean Steiner Tree Problem* - ESTP) de forma exata.

Neste capítulo são apresentados de seguida os principais conceitos associados à resolução exata deste problema bem como as principais características de dois algoritmos exatos relevantes.

## 2.2 Definições e Propriedades

### O problema de Steiner Euclidiano

Como já foi referido, no problema de Steiner Euclidiano pretende-se encontrar a rede mínima no plano cartesiano para interligar um conjunto  $P$  de pontos  $a_i$ , com  $i = 1, \dots, n$  sendo  $n$  o número

<sup>1</sup>A designação de **Steiner** resulta da popularidade do livro "What is Mathematics" publicado em 1941 por Courant e Robbins [2] onde o problema de Fermat foi reconhecido como conduzindo à rede mínima para a interligação de três pontos e se designou esse problema como problema de Steiner (independentemente do número de pontos a interligar). Steiner foi um matemático famoso que se tinha interessado anteriormente pelo problema de Fermat generalizado.



total de pontos, designados no contexto deste problema por pontos (ou nós) terminais. Estes pontos são definidos no plano cartesiano  $\mathbb{R}^2$  e, por conseguinte, cada ponto é definido pelas suas coordenadas no plano,  $a_{ix}$  e  $a_{iy}$ . Considerando que a distância Euclidiana<sup>2</sup> entre os pontos  $a_i$  e  $a_j$  é dada por  $\|a_i - a_j\|_2 = (|a_{ix} - a_{jx}|^2 + |a_{iy} - a_{jy}|^2)^{1/2}$ , a rede é caracterizada pelas ligações no plano que interligam os pontos de  $P$  e eventualmente outros pontos ou nós adicionais designados por pontos (ou nós) de Steiner,  $S_1, \dots, S_{n-2}$  de tal forma que a soma das distâncias das ligações consideradas seja mínima.

Quando não se consideram pontos adicionais o problema é de muito mais fácil resolução e existem algoritmos exatos para o resolver em tempo polinomial, como é o caso do algoritmo de Prim [1], partindo do conhecimento das distâncias Euclidianas entre todos os pares de pontos. Quer neste problema, quer no problema de Steiner a rede resultante é uma árvore.

Seguem-se algumas definições que permitem clarificar os conceitos introduzidos posteriormente.

**Grafo:** Seja  $G = (N, E)$  o grafo não dirigido representativo de uma rede de tal modo que  $N = \{a_1, a_2, \dots, a_n\}$  é o conjunto de vértices (ou nós) e  $E = \{(a_i, a_j) : a_i, a_j \in N\}$  é o conjunto de arestas (ou arcos) que interligam pares de nós do grafo.

**Grafo conexo:** Um grafo diz-se conexo se não contiver nós isolados, isto é, todos os nós pertencentes ao grafo estão interligados por um caminho (pelo menos).

**Sub-grafo:**  $G' = (N', E')$  é um sub-grafo de  $G = (N, E)$  se  $N' \subseteq N$  e  $E' \subseteq E$ .  $G'$  é um sub-grafo abrangente de  $G$  se  $N' = N$  e  $E' \subseteq E$ .

**Caminho:** Um caminho  $r_{ij}$  entre o nó  $a_i$  e o nó  $a_j$  é um sub-grafo de  $G$  constituído por uma sequência de arcos e de nós desde  $a_i$  até  $a_j$  definido por  $\langle (a_i, a_l), (a_l, a_m), \dots, (a_k, a_j) \rangle$ .

**Ciclo:** Um ciclo é um caminho (ou parte de um caminho) tal que, considerando os nós  $i, j, k, l$  pertencentes a esse caminho, se verifica a seguinte sequência de nós e arcos:

$$\langle (a_i, a_j), (a_j, a_k), \dots, (a_l, a_i) \rangle$$

**Grau de um nó:** O grau de um nó é o número total de arcos incidentes nesse nó.

**Árvore:** Uma árvore  $T$  é um grafo conexo sem ciclos com as seguintes propriedades [1]:

- Uma árvore com  $n$  nós contém exatamente  $n - 1$  arcos;
- Uma árvore possui pelo menos dois nós de grau 1 (ou nós folhas);
- Cada par de nós de uma árvore é interligado por um único caminho.

<sup>2</sup>Define-se distância entre  $u, v \in \mathbb{R}^n$  como sendo a métrica  $L_p$ , com  $1 \leq p \leq \infty$  tal que:  $\|u - v\|_p = \left( \sum_{i=1}^n |u_i - v_i|^p \right)^{(1/p)}$

**Árvore abrangente (ou *spanning tree*):** Uma árvore abrangente é um sub-grafo abrangente conexo sem ciclos.

O algoritmo de Prim encontra então uma árvore abrangente mínima (ou *Minimum Spanning Tree - MST*) dado que encontra o sub-grafo abrangente mínimo (ou o mais curto) conectando todos os nós terminais (e não considera nós adicionais). Se esse sub-grafo contivesse ciclos, não sendo nesse caso uma árvore, não seria mínimo dado que se eliminássemos um arco (o de maior comprimento no ciclo) não se desconectava nenhum nó e o comprimento total do sub-grafo seria menor. Com um raciocínio semelhante conclui-se que, no problema de Steiner, a rede pretendida também tem que ser uma árvore.

No problema de Steiner, para que uma árvore possa ser mínima - *Steiner Minimum Tree (SMT)* - é necessário que se verifiquem as seguintes propriedades:

1. Os nós de Steiner têm exatamente três arestas incidentes (ou seja, têm grau 3), formando ângulos de  $120^\circ$  entre si. A esta propriedade dá-se o nome de **condição de ângulo** (figura 2.2). O fato de nós de Steiner não poderem ter grau inferior a 3 é fácil de provar dado que se tivessem grau 1 podiam ser eliminados sem desconectar nenhum nó terminal, diminuindo assim o comprimento da árvore. Se tivessem grau 2 também podiam ser eliminados bastando para isso ligar diretamente os extremos dos seus arcos incidentes permitindo encontrar-se eventualmente uma ligação mais curta. Por outro lado, se um determinado nó de Steiner tivesse grau superior a 3 então existiriam arcos incidentes que fariam entre si ângulos inferiores a  $120^\circ$ . No entanto sempre que dois arcos incidentes num determinado nó façam entre si um ângulo inferior a  $120^\circ$  dão origem à definição de um triângulo, com os 3 pontos que são os extremos dos arcos em causa, podendo-se encontrar o respetivo ponto de Torricelli através do qual se minimiza o comprimento da interligação entre esses 3 pontos, permitindo assim a substituição dos 2 arcos iniciais.
2. Nenhum vértice, numa árvore de Steiner, poderá ter mais que três arestas (consequência da condição anterior);
3. Numa árvore de Steiner não há cruzamento de arestas dado que os ângulos menores teriam, no máximo,  $90^\circ$  para além de conduzir à existência de ciclos;
4. Para  $n$  nós terminais, uma árvore de Steiner contém no máximo  $n - 2$  nós de Steiner (equação 2.2).

Para provar o último item, considerando  $n$  nós terminais e  $S$  nós de Steiner, então a árvore deverá ter  $n + S - 1$  arestas. Como cada nó de Steiner tem três arestas e cada nó terminal tem pelo menos uma, o número de arestas será  $(3S + n)/2$  (a divisão por 2 ocorre devido ao fato de que cada aresta conter dois vértices). Assim:

$$n + S - 1 \geq (3S + n)/2 \quad (2.1)$$

ou

$$n - 2 \geq S \quad (2.2)$$

Uma árvore de Steiner é considerada uma árvore de Steiner completa (*Full Steiner Tree - (FST)*) quando se verificar a **condição de grau**, ou seja, quando todos os nós terminais tiverem grau 1 e os

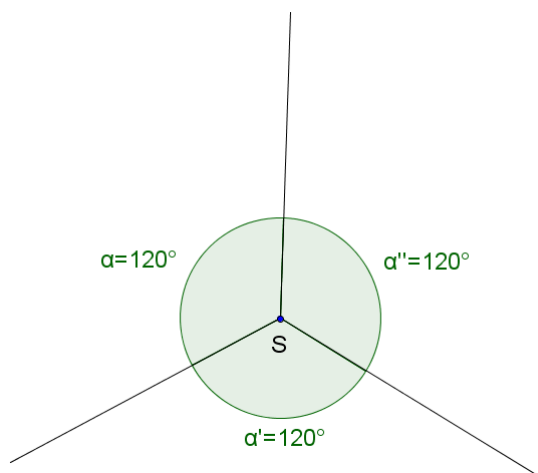


Figura 2.2: Condição de ângulo.

nós de Steiner tiverem grau 3. Assim desde que esta condição se verifique, poder-se-ão encontrar, para o mesmo conjunto de nós terminais, várias FSTs. Entre todas as FSTs encontradas poder-se-á então determinar a de menor comprimento.

## 2.3 Algoritmo de Melzak

Como já foi referido anteriormente Melzak [9] deu origem ao primeiro algoritmo exato para a resolução do problema de Steiner Euclidiano. Assim, dado um conjunto de nós terminais  $n$ , o algoritmo de Melzak consiste em gerar todas as FSTs para cada subconjunto de  $K$  nós terminais e depois selecionar a FST de menor custo para cada um desses subconjuntos. As FSTs resultantes serão concatenadas de todas as formas possíveis até encontrar a SMT.

Assim para encontrar a menor FST é necessário considerar todas as FSTs, isto é, todas as maneiras de se ligar  $K$  nós terminais a  $K - 2$  nós de Steiner. O número de FSTs para  $K$  nós terminais é dado por:

$$f(K) = \frac{(2K - 4)!}{2^{K-2}(K - 2)!} \quad (2.3)$$

Esta fórmula resulta da resolução da equação seguinte sabendo que  $f(2) = 1$  (i. e. para dois nós terminais existe uma única FST constituída por esses nós e pela ligação direta entre eles).

$$f(K + 1) = (2K - 3)f(K) \quad (2.4)$$

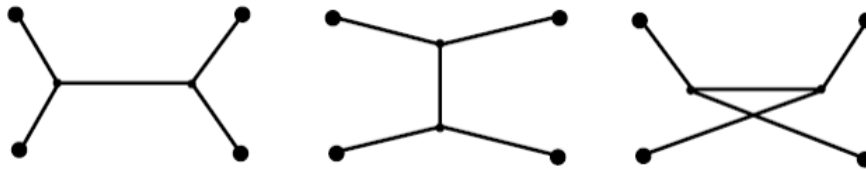
Para numa determinada FST, com  $(K + 1)$  nós terminais, se se remover um nó terminal e um nó de Steiner que lhe é adjacente obtém-se uma FST com  $K$  nós terminais. Isto prova que uma FST com  $K + 1$  nós terminais pode ser obtida a partir da FST com  $K$  nós terminais adicionando um nó de Steiner entre os  $2K - 3$  arcos e adicionando-lhe um nó terminal.

Na tabela 2.1 está ilustrado o número de FSTs, para  $3 \leq K \leq 10$  nós, a serem geradas para encontrar o caminho de menor comprimento.

K	3	4	5	6	7	8	9	10
f(K)	1	3	15	105	945	10395	135135	2027025

Tabela 2.1: Número de FSTs de acordo com fórmula 2.3.

Um subconjunto com 4 nós terminais dá origem a três FSTs (figura 2.3).

Figura 2.3: FSTs possíveis para  $K = 4$ .

Como seria de esperar, devido à natureza combinatória da expressão 2.3, quanto maior o número de terminais, maior será o número de FSTs que serão geradas.

O número total de FSTs para  $n$  nós terminais é dado por:

$$f^*(n) = \sum_{K=2}^n \binom{n}{K} f(K) \quad (2.5)$$

Par  $K \leq 3$  uma FST tem pelo menos um par de nós terminais adjacentes a um nó de Steiner. Esse nó de Steiner tem que estar localizado sobre um dos arcos de circunferência que passa por esses dois nós terminais e por um terceiro ponto que é um dos vértices dos dois triângulos equiláteros definidos pelos dois nós terminais referidos. A posição do terceiro nó terminal (para  $K = 3$ ) é que irá determinar qual o triângulo que irá ser considerado e qual a posição do nó de Steiner no arco de circunferência respectivo. A posição relativa dos pontos terminais permite escolher qual dos dois triângulos equiláteros deve ser escolhido para cada par de nós terminais [7]. É de notar que para um determinado valor de  $K$  podem não existir FSTs e nesse caso também não existem FSTs para  $K + 1$ .

A construção de uma FST, através do algoritmo de Melzak, é ilustrada através da figura 2.4 para  $K = 5$ . Sejam os nós  $a_1$  e  $a_K$  designados de raiz e terminal, respetivamente. A determinação dos nós de Steiner ( $S_1, S_2, S_3$ ) e da FST resultante são descrito de seguida.

A partir da raiz, até ao próximo nó ( $a_2$ ) traça-se a mediatriz dessa reta e cria-se um triângulo equilátero (todos os ângulos internos iguais a  $60^\circ$ ) em que um dos vértices fica sobre a mediatriz. Depois de encontrado o novo vértice do triângulo,  $e_1$ , considera-se uma circunferência  $C_1 = (a_1, a_2, e_1)$  circunscrevendo o triângulo equilátero. O arco entre  $a_1$  e  $a_2$  é o arco de Steiner ( $\widehat{a_1 a_2}$ ). O primeiro nó de Steiner ir-se-á encontrar nesse arco. Para encontrar o arco de Steiner seguinte, irão ser considerados os nós  $e_1$  e  $a_3$ . Utilizando o raciocínio anterior, traça-se a mediatriz do segmento de reta  $\overline{e_1 a_3}$  e respetivo triângulo equilátero. De seguida considera-se a circunferência  $C_1 = (e_1, a_3, e_2)$ . O arco entre  $e_1$  e  $a_3$  é o arco de Steiner  $\widehat{e_1 a_3}$  e o segundo nó de Steiner encontrar-se-á nesse arco. O arco de Steiner para o nó de Steiner  $S_3$  é determinado de forma análoga. Caso existissem mais nós terminais, ou seja,  $K > 5$ , os arcos de Steiner seriam determinados da mesma

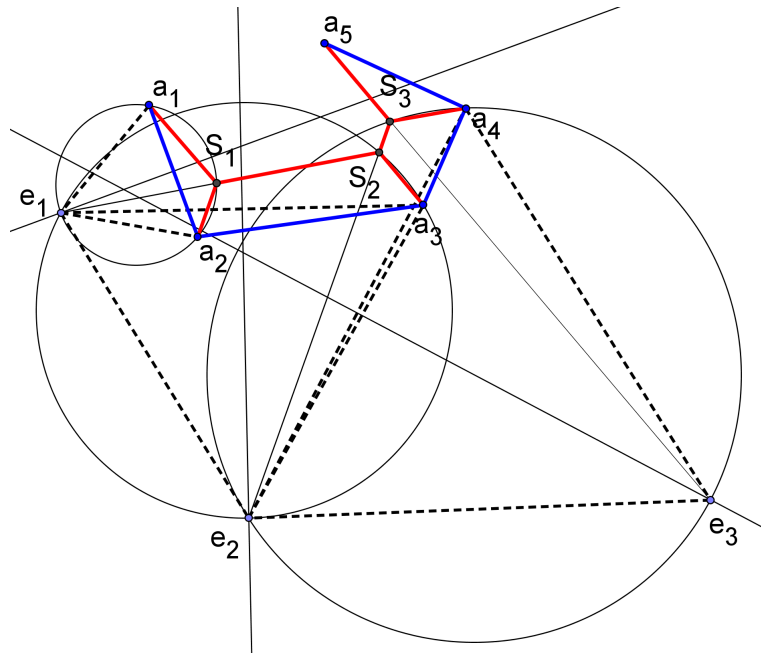


Figura 2.4: Construção de uma FST para  $K = 5$ .

forma. Depois de determinados os arcos de Steiner possíveis iremos encontrar os nós de Steiner respetivos. Assim a intersecção do segmento de reta  $\overline{a_{K-2}e_{K-2}}$  com  $e_{K-3}\widehat{a_{K-1}}$  é a localização de  $S_{K-2}$ . Dada a localização de  $S_{K-2}$ , a localização de  $S_{K-3}$  é dada pela intersecção do segmento de reta  $\overline{S_{K-2}e_{K-3}}$  com  $e_{K-4}\widehat{a_{K-2}}$ . A localização de  $S_{K-4}$ , ou seja,  $S_1$ , é determinada da mesma maneira. Esta é uma das quinze FSTs que se podem obter com este conjunto de nós terminais. Para encontrar as outras FSTs ter-se-ia que alterar a ordem pela qual os nós terminais vão sendo considerados na obtenção de cada FST.

## 2.4 Algoritmo proposto por P. Winter e M. Zachariasen

Este algoritmo em vez de tentar criar todas as FSTs, gera todos os pontos equilaterais para todos os pares de nós terminais existentes.

Um ponto equilateral é o terceiro ponto definido pelo triângulo equilátero construído a partir de um par de nós terminais, já utilizado pelo algoritmo de Melzak.

Através de testes específicos, são de seguida eliminados todos os pontos equilaterais que não conduzem à construção de FSTs. Os testes considerados nesta abordagem são bastante exaustivos reduzindo o número de FSTs a considerar e tornando o algoritmo bastante eficiente quando comparado com o anterior, sendo capaz de resolver problemas de maior dimensão. O código correspondente a este algoritmo encontra-se disponível em <http://www.cs.sunysb.edu/~algorithm/implementation/geosteiner/distrib/>, e foi usado na obtenção das soluções exatas para cada um dos problemas considerados neste trabalho.

Apresenta-se de seguida dois dos testes usados para eliminar pontos equilaterais e reduzir o arco onde se poderá localizar um nó de Steiner.

### 2.4.1 Projeções

Por uma questão de simplicidade de notação os nós terminais  $a_1$  e  $a_2$  são agora designados por  $p$  e  $q$ . Assim  $p$  e  $q$  são dois pontos no plano e o ponto equilateral  $e_{pq}$  é o terceiro ponto de um triângulo equilátero, de tal modo que a sequência  $p, e_{pq}, q$  faz curva à direita de  $e_{pq}$ . Os pontos  $p$  e  $q$  são chamados de pontos base de  $e_{pq}$ . De realçar que os pontos  $e_{pq}$  e  $e_{qp}$  são pontos distintos.

Se os pontos  $p$  e  $q$  forem terminais, então  $e_{pq}$  é um ponto equilateral de primeira ordem. Se os pontos base,  $p$  ou  $q$  de  $e_{pq}$  são pontos equilaterais de ordem  $ORD(p)$  e  $ORD(q)$ , respectivamente, então a ordem de  $e_{pq}$  é dada por:  $ORD(e_{pq}) = \max(ORD(p), ORD(q)) + 1$ .

Supondo agora que  $p$  e  $q$  são dois pontos equilaterais, sendo que  $p$  não tem ordem zero e que os pontos  $a$  e  $c$  são pontos base de  $e_p$ , isto é,  $p = e_{ac}$ . Agora serão analisadas como as localizações de  $a$  e  $c$  podem ser usadas para identificar localizações inviáveis de pontos de Steiner ( $S_{pq}$ ) no arco de Steiner  $\widehat{pq}$ . Os seis casos necessários à compreensão desta análise estão apresentados na figura 2.5. Analisando os casos da figura 2.5 é possível concluir que existem casos em que são violadas

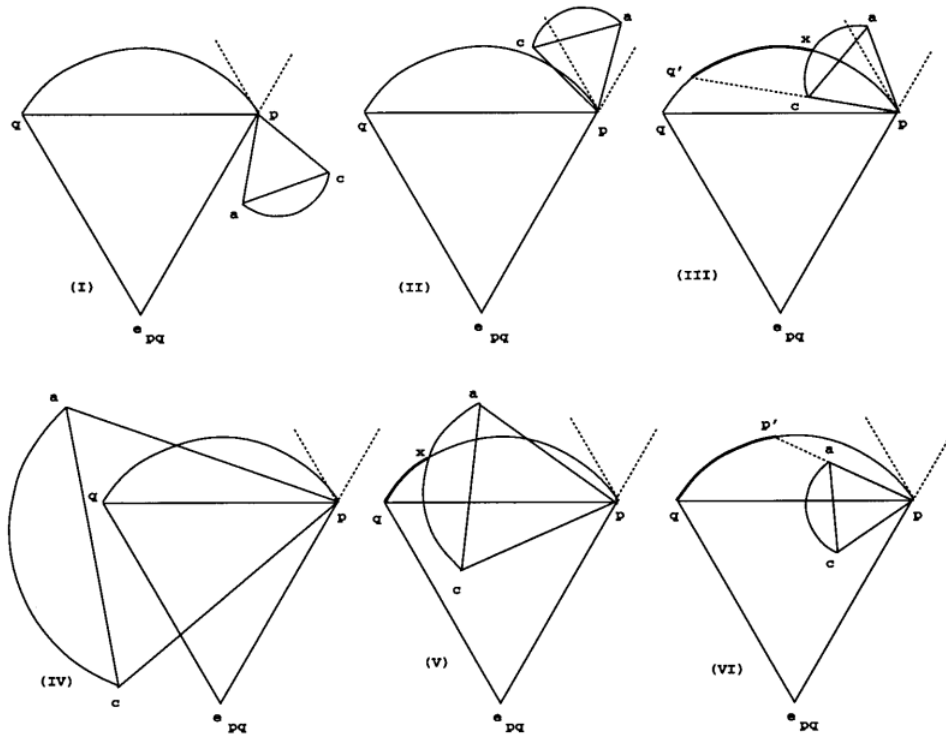


Figura 2.5: Projeções [12].

a condição de ângulos e daí serem inviáveis podendo os pontos equilaterais  $e_{pq}$  serem excluídos. Nesta situação encontram-se os casos *I, II* e *IV*. Para os restantes casos (*III, V* e *VI*) o arco de Steiner poderá ser reduzido encurtando assim a localização do ponto de Steiner.

### 2.4.2 Propriedade *lune*

Uma *lune* de um segmento de reta  $uv$  é a intersecção de dois círculos ambos com raio  $\|u - v\|_2$  e centro em  $u$  e  $v$  respectivamente (figura 2.6). Uma condição necessária para o segmento de reta  $uv$  estar em qualquer SMT é que a sua *lune* não contenha terminais.

Esta propriedade permite reduzir as regiões dos arcos onde podem existir pontos de Steiner.

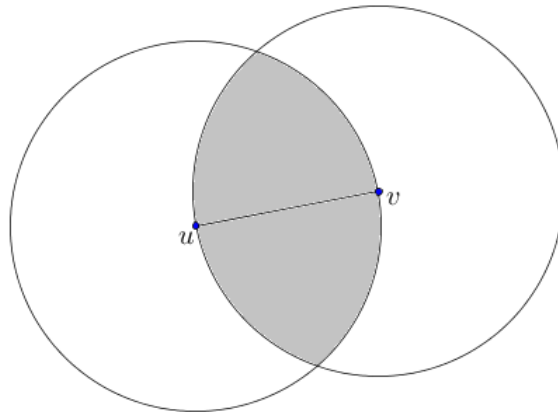


Figura 2.6: Propriedade *lune*.





## Capítulo 3

# Árvores de Steiner Euclidianas com obstáculos

### 3.1 Introdução

Apesar de Winter ter apresentado um algoritmo exato, em [14], para o problema de Steiner Euclidiano com obstáculos, optou-se pelo desenvolvimento de uma heurística mais simples, por ser computacionalmente mais leve dada a complexidade do problema a resolver, que é NP-Hard. Pretendia-se que o sistema que viesse a ser desenvolvido no âmbito desta dissertação pudesse vir a ser integrado num sistema mais abrangente de apoio ao desenvolvimento de projetos de novos traçados de redes de telecomunicações.

O algoritmo exato proposto em [14] para a resolução do problema de Steiner com obstáculos usa a mesma abordagem do algoritmo exato sem obstáculos, sendo neste caso mais sofisticados os testes que permitem a redução do número de FSTs a considerar, para levarem em conta a presença de obstáculos.

Para a resolução do problema das árvores de Steiner Euclidianas com obstáculos optou-se então pela implementação de uma heurística que tem por base a construção de uma MST. A partir dessa árvore é possível então obter uma aproximação à árvore mínima de Steiner (SMT). Em [3] é referido que Thompson foi o primeiro a sugerir a adição de pontos de Steiner à MST. Seguindo o seu raciocínio, quando dois arcos pertencentes à MST tiverem um ângulo inferior a  $120^\circ$ , é adicionado um nó de Steiner para substituição desses arcos pelas novas ligações através do ponto de Steiner adicionado.

A estratégia de obtenção de árvores de Steiner recorrendo a MSTs é também usada no caso de árvores de Steiner retilíneas [6, 5] e no caso de árvores de Steiner em grafos [8].

Neste capítulo, depois de descrito o algoritmo de Prim para a obtenção de uma MST, é descrita a heurística para a obtenção de uma árvore de Steiner que, por sua vez, serviu de base à proposta de uma nova heurística para a resolução do problema de Steiner com obstáculos, que é apresentada por último.

### 3.2 Árvore abrangente mínima

Tal como foi referido anteriormente, o algoritmo de Prim encontra uma MST para interligar um conjunto de nós. O grafo inicial é constituído pelo conjunto  $N$  de nós e pelo conjunto  $E$  de todos os arcos entre todos os pares de nós. Cada arco tem ainda associada a distância Euclidiana entre os nós que interliga. Este algoritmo recorre ao conceito de corte.

**Corte:** Um corte é uma partição do conjunto de nós  $N$  em duas partes,  $N'$  e  $\overline{N'} = N - N'$ . Cada corte define um conjunto de arcos, referido como o conjunto de arcos do corte, que contêm um ponto extremo em  $N'$  e outro em  $\overline{N'}$ .

Este algoritmo considera um sub-grafo vazio,  $G' = (N', E')$ , e adiciona-lhe um qualquer nó terminal. De seguida é escolhido para adicionar a  $G'$  o arco de menor comprimento no corte criado pelo conjunto  $N'$  (com o nó inicial adicionado) e  $\overline{N'} = N - N'$ . O nó que é a outra extremidade do arco adicionado é posteriormente retirado de  $\overline{N'}$  e inserido em  $N'$ . O novo corte entretanto criado permite escolher o arco de menor comprimento nesse corte bem como o nó seguinte a adicionar a  $N'$ . O algoritmo termina quando  $N' = N$ , ou seja, quando todos os nós de  $N$  pertencerem ao grafo  $G'$ .  $G'$  é então uma árvore abrangente mínima (MST).

Na figura 3.1 está ilustrada a explicação anterior estando os cortes representados a tracejado.

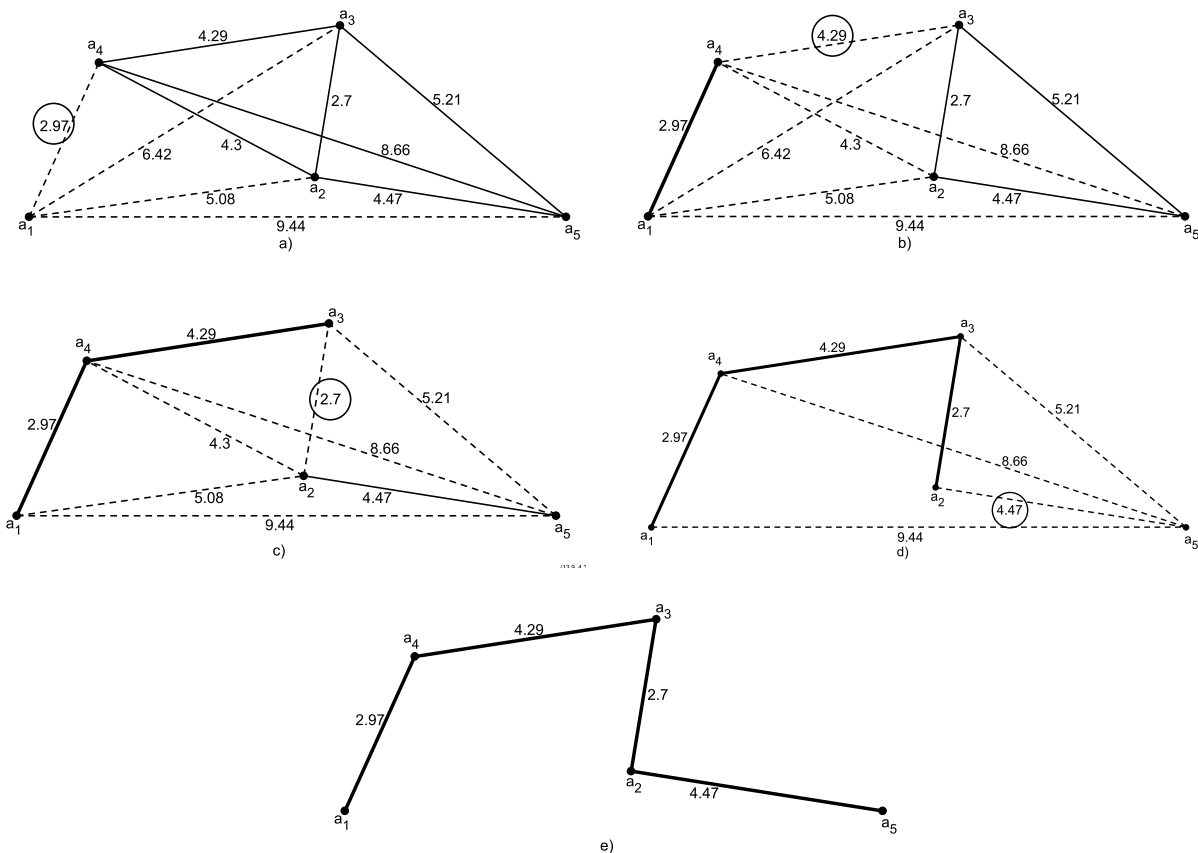


Figura 3.1: Evolução do algoritmo de Prim.

Perante a existência de dois ou mais arcos com comprimentos iguais (ver figura 3.2) a *MST*

não será única, pois o algoritmo de Prim escolhe um desses arcos, ou seja, existe mais do que uma árvore mínima. Por outro lado, se todos os arcos tiverem comprimentos diferentes (ver figura 3.3),

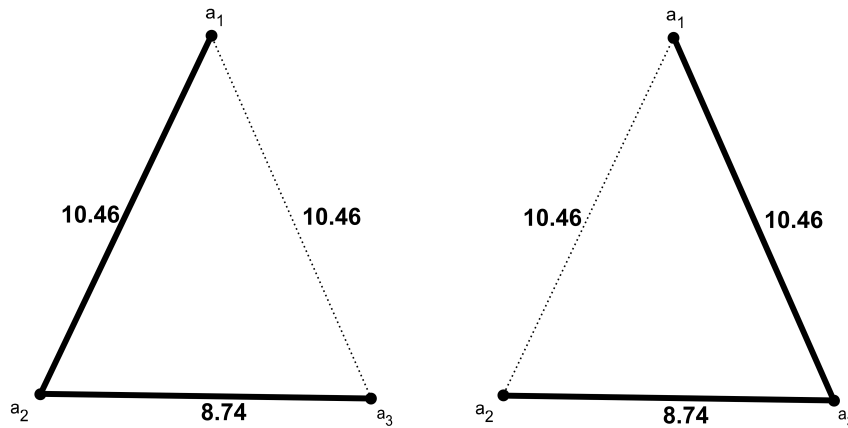


Figura 3.2: Exemplo de duas MSTs para a mesma árvore.

aí podemos afirmar que só é gerada uma única MST.

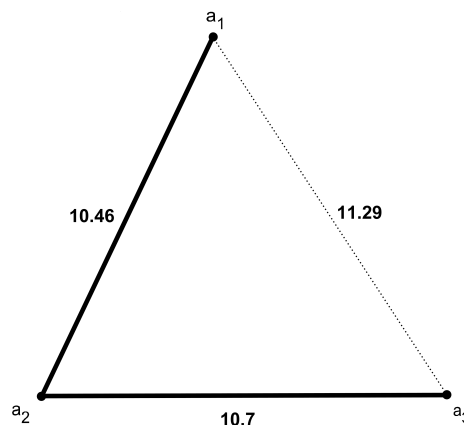


Figura 3.3: Exemplo de MST única.

### 3.3 Árvore de Steiner Euclidiana

Com base na MST, a construção da árvore de Steiner (que é uma aproximação à SMT dado que este procedimento faz parte de uma heurística [15] e não de um algoritmo exato, como os descritos no capítulo anterior) resulta da verificação da interligação entre cada conjunto de três nós na MST,  $\langle (a_i, a_l), (a_l, a_j) \rangle$ , fazendo-se de seguida a verificação de ângulos. Das técnicas apresentadas no capítulo 2, optou-se pelo uso das linhas de Simpson para encontrar o ponto de Steiner para cada um desses conjuntos de três nós.

Assim se o ângulo entre dois arcos for inferior a  $120^\circ$ , é criado um nó de Steiner e os arcos  $(a_i, a_l)$  e  $(a_l, a_j)$  serão eliminados passando a existir novos arcos,  $(S, a_i)$ ,  $(S, a_l)$  e  $(S, a_j)$ . No caso

do ângulo entre os três nós ser superior a  $120^\circ$ , os arcos mantêm-se inalterados, sendo que não existe outro nó que minimize a distância entre eles.

---

**Algoritmo 1:** Algoritmo para obter a SMT.

---

**Input:** Coordenadas dos nós terminais( $N$ )

**Output:** SMT (aproximada)

**begin**

    Cálculo da matriz de distâncias entre todos os pares de nós em  $N$

$SMT \leftarrow$  MST obtida através do algoritmo de Prim

**forall** the  $(a_i, a_j) \in SMT$ ; **do**

$\hat{\text{Ângulo}} \leftarrow$  Menor ângulo entre  $(a_i, a_j)$  e  $(a_j, a_l) \in \{(a_j, a_l) : \forall l \in N, (a_j, a_l) \in SMT\}$ ;

**if**  $\hat{\text{Ângulo}} < 120^\circ$  **then**

            Cria Nó de Steiner ( $S_k$ ) e adiciona-o à SMT;

            Remove arcos  $(a_i, a_j)$  e  $(a_j, a_l)$  da MST;

            Adiciona os  $(a_i, S_k), (a_j, S_k)$  e  $(a_l, S_k)$  à MST;

**end**

**end**

---

O algoritmo 1 permite obter a SMT pretendida. De seguida, analisam-se com mais detalhe os tipos de ligações que se podem obter a partir da MST de modo a obter-se o menor ângulo pretendido num dos passos do algoritmo 1. Na tabela 3.1 estão apresentados graficamente todos os casos possíveis bem como a solução correspondente, ou seja, quando a SMT é percorrida encontra uma das seguintes situações:

1. Quando o ângulo entre os arcos  $(a_l, a_j)$  e  $(a_j, a_i)$  for superior a  $120^\circ$ , não será criado nenhum nó de Steiner, pelas razões já apresentadas;
2. Quando o nó  $a_j$  tem dois arcos incidentes e o ângulo entre eles é inferior a  $120^\circ$  é calculada a localização do nó de Steiner e os arcos  $(a_j, a_i)$  e  $(a_j, a_l)$  são eliminados, sendo substituídos pelos arcos  $(S, a_j), (S, a_i)$  e  $(S, a_l)$ .
3. Quando o nó  $a_j$  tem três arcos incidentes,  $(a_j, a_i), (a_j, a_l)$  e  $(a_j, a_k)$ , são calculados todos os ângulos, sendo escolhido o menor. Depois, no caso do ângulo ser inferior a  $120^\circ$ , é calculado o nó de Steiner e, tal como no caso anterior, os arcos são atualizados. O arco que não foi considerado para calcular o nó de Steiner manter-se-á inalterado e é verificado numa iteração seguinte do ciclo *for* (no caso de não ter sido já verificado).
4. Poderá ainda acontecer uma situação em que existem quatro arcos ligados a  $a_j$ . Novamente serão calculados todos os ângulos e escolhido o menor, reduzindo assim o problema ao caso 2. É de notar que neste caso existem pelo menos dois ângulos inferiores a  $120^\circ$ . Os arcos não envolvidos no cálculo do nó de Steiner, tal como no caso anterior, vão ser verificados noutras interações do ciclo *for*.

Depois de percorridos todos os arcos da SMT (aproximada) obtém-se a árvore de Steiner pretendida.

#	Situação inicial	Resultado final
1		
2		
3		
4		

Tabela 3.1: Casos possíveis de análise dos ângulos e soluções.

### 3.4 Obstáculos

Dado um conjunto  $N$  de  $n$  pontos (ou nós) terminais e um conjunto  $\omega = \{w_1, \dots, w_h\}$  de  $h$  obstáculos do tipo poligonal, encontrar a rede mais pequena (de acordo com a métrica  $L_2$  - distância Euclidiana) que interligue todos os pontos de  $N$  evitando os obstáculos  $\omega$  é conhecido como sendo o problema de Steiner Euclidiano com obstáculos (obstacles-avoiding Euclidean Steiner tree problem – STPO). Neste problema os nós terminais não podem pertencer ao interior dos obstáculos  $w_i$ , embora possam pertencer às suas fronteiras. A rede mínima resultante é, mais uma vez, uma árvore e este problema é uma generalização do problema de Steiner Euclidiano sendo também NP-Hard [14].

A solução ótima para este problema (obstacles-avoiding Euclidean Steiner minimal tree - SMT0) pode conter pontos de Steiner de grau 3 (que verificam a condição de ângulo de  $120^\circ$ ) e também pontos de Steiner degenerados, na fronteira dos obstáculos, que não respeitam a condição de ângulo.

Tal como no caso anterior do problema Euclidiano de Steiner sem obstáculos, o ponto de partida para a heurística, que é proposta nessa Dissertação para a resolução do problema com

obstáculos, é também uma árvore abrangente mínima que é obtida levando em conta os obstáculos (MSTO).

O algoritmo para a obtenção da MSTO começa por calcular a distância Euclidiana entre todos os pares de nós sem ter em conta os obstáculos. De seguida, através do algoritmo *Intersecta obstáculo*<sup>1</sup> são identificadas as ligações entre todos os pares de nós que são intersectadas por obstáculos. Para cada uma destas ligações é obtida através do algoritmo de Dijkstra a nova distância mínima que interliga esse par de nós terminais contornando o obstáculo (ver figura 3.4).

O que foi explicado anteriormente está exemplificado no algoritmo 2.

---

**Algoritmo 2:** Algoritmo para obtenção da MSTO.

---

**Input:** Coordenadas dos nós terminais ( $N$ ), Coordenadas dos pontos extremos dos obstáculos

**Output:** MSTO

**begin**

**forall the** *Cada par de nós terminais em  $N$ ; do*

    Intersecta Obstáculo  $\leftarrow$  Verifica interseção com obstáculo;

**if** *Intersecta = TRUE then*

      Calcula nova distância entre o par de nós contornando os obstáculos;

**end**

**end**

  MSTO  $\leftarrow$  Cria MSTO através do algoritmo de Prim com as novas distâncias calculadas.

**end**

---

O contorno do(s) obstáculo(s) numa determinada ligação é resolvido através de uma adaptação do algoritmo de Dijkstra considerando apenas o grafo constituído pelos dois nós terminais e os extremos do(s) obstáculo(s) que interessam para o cálculo da distância mínima entre os dois nós terminais contornando o(s) obstáculo(s). Os pontos extremos que não devem ser considerados são identificados pelo algoritmo *Intersecta obstáculos* e não entram no grafo considerado para o algoritmo de Dijkstra.

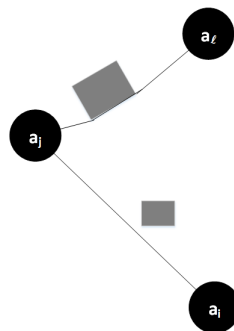


Figura 3.4: Exemplo de MSTO.

A heurística que aqui se propõe para a resolução do problema de STPO usa então, como ponto de partida, o algoritmo 2 para obtenção da MSTO e de seguida procura encontrar pontos de Steiner

---

<sup>1</sup>Cedido pelo colega Luís Garrote, que o desenvolveu para usar com robôs

ou pontos de Steiner degenerados, à semelhança da heurística proposta para o problema STP, adaptando agora para a existência de obstáculos.

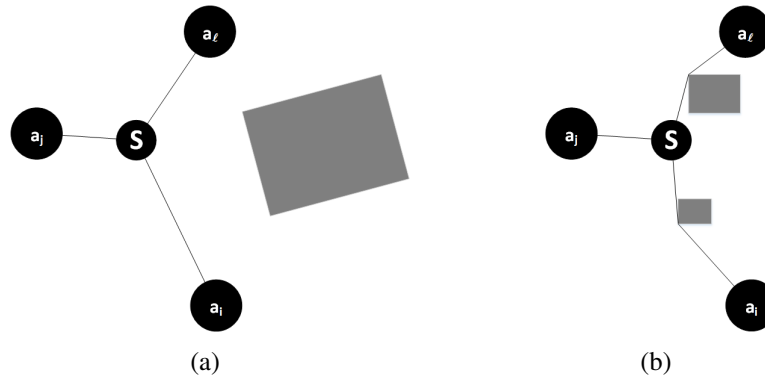


Figura 3.5: Exemplos do problema de Steiner com obstáculos.

Tendo em conta que não existem nós terminais no interior de obstáculos, as possibilidades de um obstáculo interferir na SMTO são enumeradas de seguida.

1. A presença de um obstáculo pode não interferir na SMTO (ver figura 3.5(a)) e assim a localização do nó de Steiner e os dos novos arcos seguem as regras já referidas anteriormente para o problema STP.
2. A localização de um nó de Steiner pode coincidir com o interior de um obstáculo. Nesta situação é calculada a localização de um novo nó de Steiner "puxando"o nó de Steiner calculado inicialmente para fora do obstáculo em direção ao nó  $a_j$  (nó no qual incidem os dois arcos que formam entre si um ângulo inferior a  $120^\circ$ ) (ver figura 3.6 (a)). De seguida, considerando o nó  $a_j$  e um ou dois extremos do obstáculo e eventualmente um segundo nó terminal, recalcula, para o novo conjunto de três pontos, a localização do novo nó de Steiner para que a condição de ângulo seja válida. O comprimento desta nova ligação será comparado com o comprimento dos arcos iniciais da MSTO (figura 3.6(b)) e a ligação com menor comprimento será a escolhida.
3. Um dos novos arcos associado a um ponto de Steiner intersesta o obstáculo. O arco que intersestar o obstáculo vai ser substituído por uma nova ligação contornando o obstáculo, calculada através do algoritmo de Dijkstra, usando a estratégia já descrita para obtenção da MSTO no algoritmo 2. O comprimento da nova ligação, recorrendo ao nó de Steiner, é comparado com o comprimento dos arcos da MSTO sendo escolhida a ligação de menor comprimento (ver figura 3.5(b)). É de notar que este caso pode coexistir com o caso descrito no ponto anterior.
4. Tal como no problema sem obstáculos, um nó da MSTO pode ter mais que dois arcos incidentes e nesse caso é necessário escolher os dois arcos que vão dar origem a um nó de Steiner. Desta forma, temos:

- ★ Na presença de três arcos, o ângulo escolhido será sempre o menor dos dois ângulos inferiores a  $120^\circ$ . No caso de existir um obstáculo na suposta localização do nó de Steiner (figura 3.7(b)) é usado o segundo menor ângulo para a localização de um possível novo nó de Steiner (figura 3.7(c)). Das três ligações possíveis (i. e. os arcos da MSTO (figura 3.7(a)), os arcos contornando os obstáculos usando-se o menor ângulo e os arcos do nó de Steiner calculado através do segundo menor ângulo) é escolhida a de menor comprimento.
- ★ No caso de existirem quatro arcos incidentes num nó, tal como no caso anterior, o ângulo escolhido para localização do nó de Steiner será sempre o menor. No caso de existir um obstáculo a interferir de algum modo nessa localização será analisada a possibilidade de se usar o segundo menor ângulo para a eventual escolha de um novo ponto de Steiner, e assim sucessivamente até encontrar a menor ligação que contorne o(s) obstáculo(s).
- ★ Na presença de obstáculos poderão ainda existir mais do que quatro arcos incidentes num nó da MSTO. Esta situação, embora pouco frequente, é resolvida de forma idêntica ao descrito anteriormente.

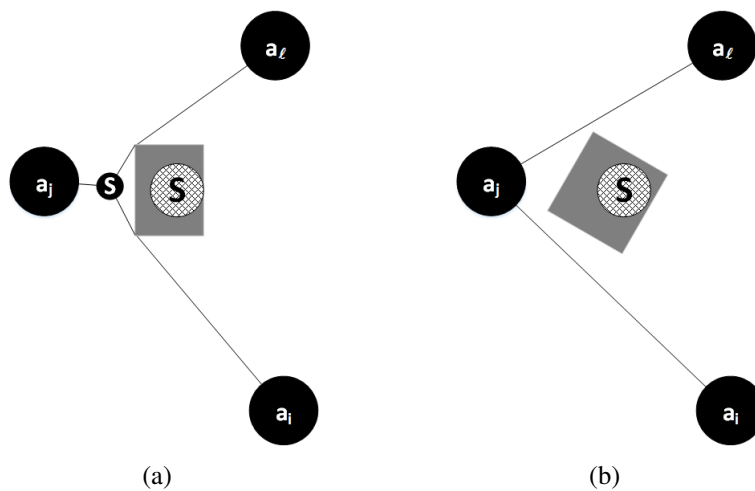


Figura 3.6: Exemplo de localização de obstáculos na SMT.

Assim, na presença de obstáculos e caso dois arcos tenham um ângulo inferior a  $120^\circ$  existem duas hipóteses a considerar optando sempre pela que tiver menor distância. Ou os arcos da MST mantêm-se intatos ou é gerado um nó de Steiner e o obstáculo é contornado. O algoritmo 3 pretende exemplificar o que é executado para a obtenção da SMTO.



**Algoritmo 3:** Algoritmo para obter a SMTO.**Input:** MSTO, Coordenadas dos pontos extremos dos obstáculos**Output:** SMTO (aproximação)**begin**SMTO  $\leftarrow$  MSTO obtida através do algoritmo 2**forall the**  $(a_i, a_j) \in SMTO$ ; **do**    Ângulo  $\leftarrow$  Menor ângulo entre  $(a_i, a_j)$  e     $(a_j, a_l) \in \{(a_j, a_l) : \forall l \in \mathbb{N}, (a_j, a_l) \in SMTO\}$ ;    **if** Ângulo  $< 120^\circ$  **then**

Cria Nó de Steiner;

**if** Nó de Steiner dentro de obstáculo || Arcos intersectam obstáculo **then**            **case 1 - Dentro Obstáculo**

Calcula novo ponto de Steiner considerando extremos de obstáculos;

Atualiza arcos;

**if** Arco intersecta obstáculo **then**

| Contorna obstáculo;

**end**                **forall the**  $(a_k, a_j) \in \{(a_j, a_k) : \forall k \in \mathbb{N} \wedge k \neq l, (a_j, a_k) \in SMTO\}$  **do**                    Ângulo  $\leftarrow$  Menor ângulo entre  $(a_i, a_j)$  e  $(a_j, a_k)$  (para todos os  $a_k$   
                    ainda não considerados no ciclo for)                    **if** Ângulo  $< 120^\circ$  **then**

Cria Nó de Steiner;

**if** Nó de Steiner dentro de obstáculo || Arcos intersectam  
                        obstáculo **then**

| Case 1 ...

| Case 2...

**end**

Compara com ligação anterior e guarda a mais curta;

**end**                **end**

Compara com ligação inicial e escolhe a mais curta;

**end**            **case 2 - Arco intersecta obstáculo**

Contorna obstáculo;

Compara com ligação inicial e escolhe a mais curta;

**end**        **end**    **end****end****end**

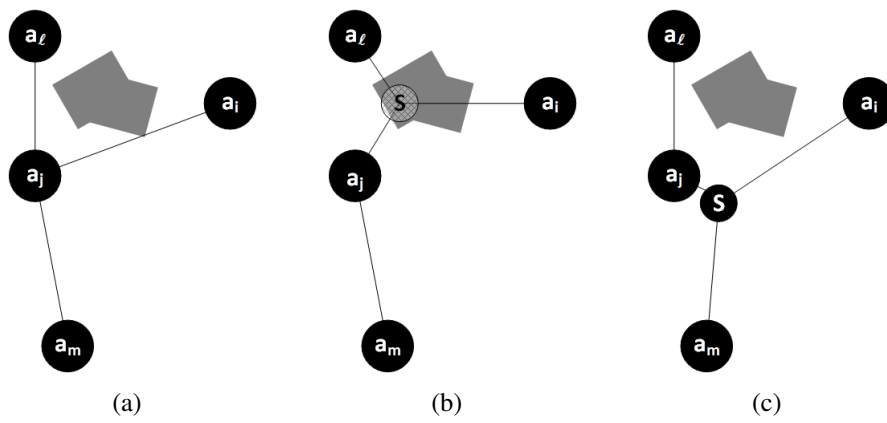


Figura 3.7: Escolha de localização do nó de Steiner.

# Capítulo 4

## Análise de desempenho dos algoritmos desenvolvidos

### 4.1 Introdução

O código desenvolvido foi escrito em linguagem C++ e o interface gráfico<sup>1</sup> foi desenvolvido através do programa Qt Creator<sup>2</sup>. Este interface foi fundamental na fase de desenvolvimento da heurística para a resolução do problema de Steiner com obstáculos.

Para avaliar o desempenho das heurísticas desenvolvidas, foi efetuado um conjunto de testes comparando os resultados obtidos pelo algoritmo exato, disponível em <http://www.cs.sunysb.edu/~algorithm/implement/geosteiner/distrib/>, com os obtidos pelas heurísticas desenvolvidas, obtidos num Pentium Dual-Core a 2GHz com 4GB de RAM. Para o problema sem obstáculos os resultados obtidos basearam-se em valores numéricos disponíveis. No problema com obstáculos, na ausência de valores numéricos foram utilizados resultados aproximados, obtidos através do interface gráfico (adaptado para o efeito), a partir de resultados gráficos disponíveis em [14].

### 4.2 Resultados obtidos para o problema de Steiner sem obstáculos

Na tabela 4.1 estão representados os resultados obtidos através do algoritmo exato [12] e da heurística implementada para o problema de Steiner sem obstáculos. Nestes resultados são apresentados os comprimentos das soluções obtidas para cada caso bem como o tempo de execução<sup>3</sup> de cada algoritmo. Os casos estudados estão disponíveis em [11] tendo sido considerados todos os casos, sendo que o número total de nós terminais varia de  $3 < n < 62$ . Existem casos que, apesar de terem poucos pontos, têm uma grande complexidade pois necessitam de vários nós de Steiner para obtenção da solução ótima. O último parâmetro da tabela, diferença relativa (em %), é dado pela seguinte expressão:

---

<sup>1</sup>Desenvolvido e gentilmente cedido por Luís Garrote.

<sup>2</sup>Programa gratuito e disponível em <http://qt-project.org/>.

<sup>3</sup>De notar que os tempos devolvidos pelo algoritmo exato se encontram em segundos.

$$Diferença\ relativa = \frac{SMT_{heur.} - SMT_{otimo}}{SMT_{otimo}} * 100 \quad (4.1)$$

Dos 46 casos apresentados, apenas 9 deles (aproximadamente 20%) têm uma diferença relativa superior a 1%, o que significa que a heurística implementada dá como solução uma boa aproximação à solução ótima. Contudo, esta diferença deve-se à construção da SMT através de uma MST, e só nos casos em que as ligações têm ângulos inferiores a 120° é que é gerado um ponto de Steiner. Quer isto dizer que, em alguns casos, o algoritmo exato encontra outras soluções que se afastam da MST.

Exemplo #	SOUKUP [11]	GeoSteiner 3.1		Heurística		Diferença relativa (%)
	Solução	Solução	Tempo(s)	Solução	Tempo(s)	
<b>1</b>	166,44	166,43993	0,00	166,44788	0,00	0,00477
<b>2</b>	150,05	150,04998	0,00	150,04999	0,00	0,00000
<b>2A</b>	207,77	207,76711	0,00	207,76711	0,00	0,00000
<b>2B</b>	213,88	213,87890	0,00	213,87890	0,00	0,00000
<b>2C</b>	204,41	204,40525	0,00	204,41646	0,00	0,00549
<b>2D</b>	218,42	218,42047	0,00	222,24438	0,00	1,75071
<b>2E</b>	220,53	220,52928	0,00	220,53720	0,00	0,00359
<b>2F</b>	217,78	217,77945	0,00	217,78466	0,00	0,00239
<b>2G</b>	160,18	156,42213	0,00	156,44183	0,00	0,01260
<b>3</b>	159,88	159,87517	0,00	164,73010	0,00	3,03670
<b>4</b>	127,41	127,41137	0,00	127,41137	0,00	0,00000
<b>5</b>	164,83	164,83376	0,00	164,83847	0,00	0,00286
<b>6</b>	128,62	127,33761	0,00	128,62118	0,00	1,00800
<b>7</b>	220,49	220,50942	0,00	220,50942	0,00	0,00000
<b>8</b>	123,04	123,04077	0,00	123,04077	0,00	0,00000
<b>9</b>	116,68	116,67809	0,00	116,67809	0,00	0,00000
<b>10</b>	164,28	164,27922	0,00	164,27922	0,00	0,00000
<b>11</b>	382,80	381,76188	1,20	386,23418	0,11	1,17149
<b>12</b>	172,22	170,64572	0,02	173,59118	0,01	1,72607
<b>13</b>	103,96	103,96152	0,00	103,96152	0,00	0,00000
<b>14</b>	181,82	181,81793	0,00	181,83502	0,00	0,00940
<b>15</b>	50,33	50,32862	0,00	50,32862	0,00	0,00000
<b>15A</b>	51,30	50,94613	0,00	50,94613	0,00	0,00000
<b>16</b>	25,28	25,28201	0,00	25,49464	0,00	0,84101
<b>16A</b>	19,90	19,89685	0,00	19,89685	0,00	0,00000
<b>16B</b>	12,44	12,43470	0,00	12,43470	0,00	0,00000
<b>16C</b>	117,82	117,81697	0,00	117,81697	0,00	0,00000
<b>16D</b>	20,44	20,44153	0,00	20,44153	0,00	0,00000
<b>17</b>	146,60	146,59774	0,00	146,59774	0,00	0,00000
<b>18</b>	104,21	101,98307	0,25	104,92622	0,17	2,88592
<b>19A</b>	233,55	233,21736	0,00	233,54963	0,01	0,14247
<b>19B</b>	284,08	281,42361	0,07	285,35959	0,06	1,39860
<b>20</b>	222,95	218,08475	0,06	218,64991	0,02	0,25914

<b>21</b>	213,93	213,81261	0,05	213,93520	0,04	0,05733
<b>21A</b>	135,57	129,89804	0,06	129,95860	0,04	0,04662
<b>22</b>	87,89	87,89125	0,00	87,89244	0,00	0,00136
<b>23</b>	76,60	76,60261	0,00	76,60482	0,00	0,00288
<b>24</b>	142,48	142,48159	0,00	143,50082	0,00	0,71535
<b>24A</b>	143,12	143,12456	0,00	143,12456	0,00	0,00000
<b>25</b>	141,80	141,79883	0,02	141,84796	0,01	0,03465
<b>26</b>	227,70	197,67196	0,02	197,67609	0,02	0,00209
<b>27</b>	131,54	131,52909	0,00	132,55081	0,00	0,77680
<b>28</b>	234,46	233,07646	0,08	236,72248	0,02	1,56430
<b>29</b>	219,74	218,69241	0,04	219,76216	0,03	0,48916
<b>30</b>	193,95	193,09954	0,08	193,58743	0,03	0,25266
<b>31</b>	139,99	136,60254	0,32	145,44600	0,17	6,47386

Tabela 4.1: Comparação de resultados para os 46 casos disponíveis em [11], relativos ao problema de Steiner sem obstáculos.

Apesar dos resultados da heurística não serem tão bons como os obtidos pelo algoritmo exato, o tempo de execução da heurística é menor para todos os casos, como seria de esperar (e era desejável).

Como exemplo ilustrativo das árvores obtidas pela heurística desenvolvida e pelo algoritmo exato, nas figuras 4.1 e 4.2 estão representados os casos 11 e 30 da tabela 4.1.

### 4.3 Resultados obtidos para o problema de Steiner com obstáculos

Dado que na literatura existe pouca informação sobre o problema de Steiner Euclidiano com obstáculos, ao contrário do que acontece com o problema de Steiner retilíneo com obstáculos para o qual existem várias propostas recentes, também não existem disponíveis resultados que possam ser usados para comparação com a heurística proposta nesta dissertação. Recorreu-se então ao interface gráfico usado neste trabalho que, depois de alterado, permitiu "copiar" as figuras existentes em [14], permitindo assim fazer a análise de resultados que se segue. De salientar que a reprodução das árvores obtidas em [14] foi feita com o maior rigor possível, embora exista algum erro associado.

Na figura 4.3 estão alguns exemplos de árvores de Steiner com obstáculos obtidas pelo algoritmo exato [14] e na tabela 4.2 estão apresentadas as respectivas soluções numéricas obtidas a partir da reprodução dessas árvores através do interface gráfico. As soluções e os tempos de execução obtidos pela heurística, para a construção da MSTO e da SMTO, são também apresentados nessa tabela. As árvores obtidas pela heurística desenvolvida para o mesmo conjunto de nós terminais e para os mesmos obstáculos estão representadas nas figuras 4.4 e 4.5.

Nos resultados apresentados na tabela 4.2, os tempos de execução da heurística são muito mais baixos dos que os apresentados em [14], como era desejado, embora não seja possível uma

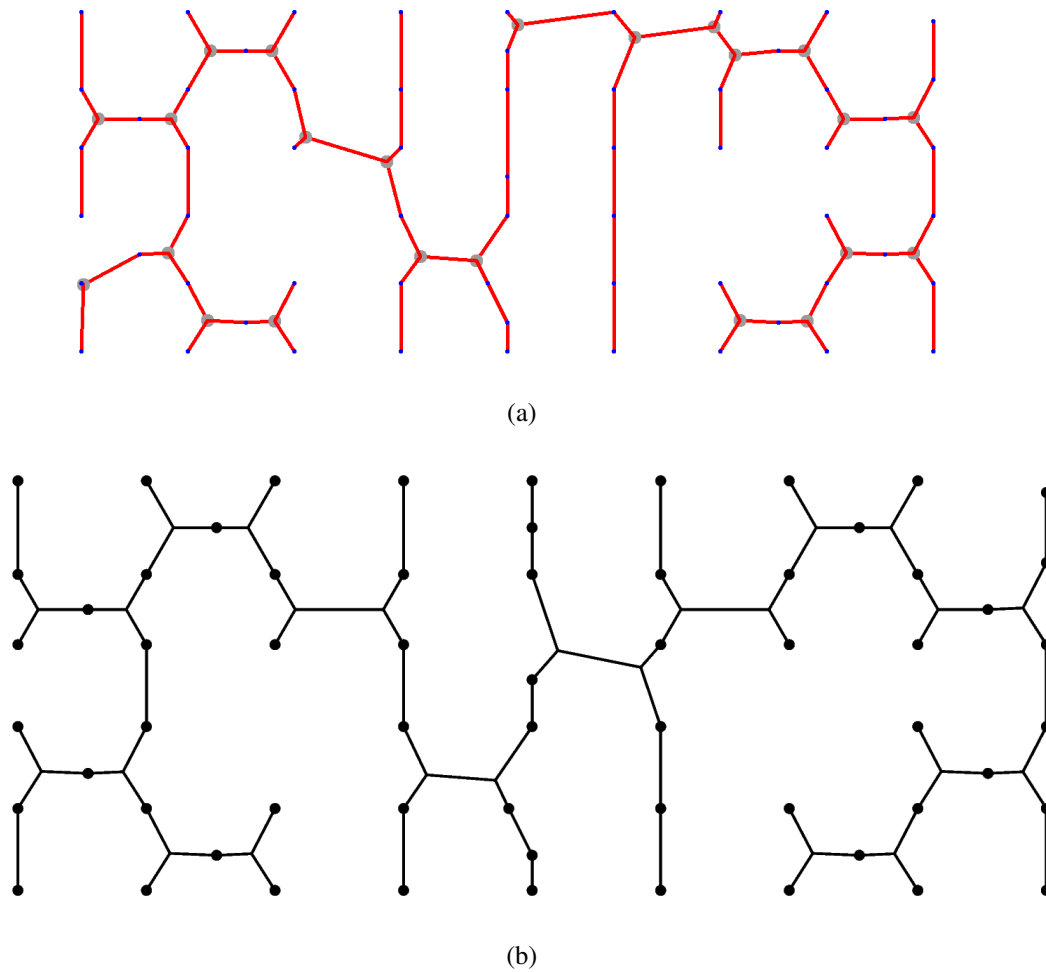


Figura 4.1: SMT construída através da heurística e do algoritmo exato para o caso 11 de [11].

comparação rigorosa destes valores, dado que não foram obtidos nas mesmas circunstâncias, ao contrário dos resultados anteriores obtidos para o problema de Steiner Euclidiano sem obstáculos.

Contudo, relativamente às soluções obtidas, conclui-se que os resultados são parecidos, nomeadamente em termos da topologia da árvore obtida, sendo a diferença relativa máxima das soluções inferior à obtida para o caso do problema de Steiner sem obstáculos. No entanto é de referir que o número de casos testados para este problema é bastante inferior ao número de casos considerado para o problema de Steiner sem obstáculos.

Na figura 4.6 apresenta-se uma carta topográfica onde está representado um traçado aéreo real referente a uma parte de uma rede de telecomunicações. A árvore de Steiner sobreposta sobre a carta foi obtida através da heurística para o problema de Steiner sem obstáculos. Comparando as duas distâncias (obtidas a partir do interface gráfico) existe uma redução de aproximadamente 5% (distância real= 779 e distância proposta= 742) do traçado real face à proposta de traçado obtida através da heurística desenvolvida. Partindo do princípio que não existia no terreno restrições que impedissem ou dificultassem a implementação do novo traçado este permitiria poupar 5 % dos custos dos cabos envolvidos.

Na figura 4.7 está representado um outro traçado real de cabos telefónicos para ligação de

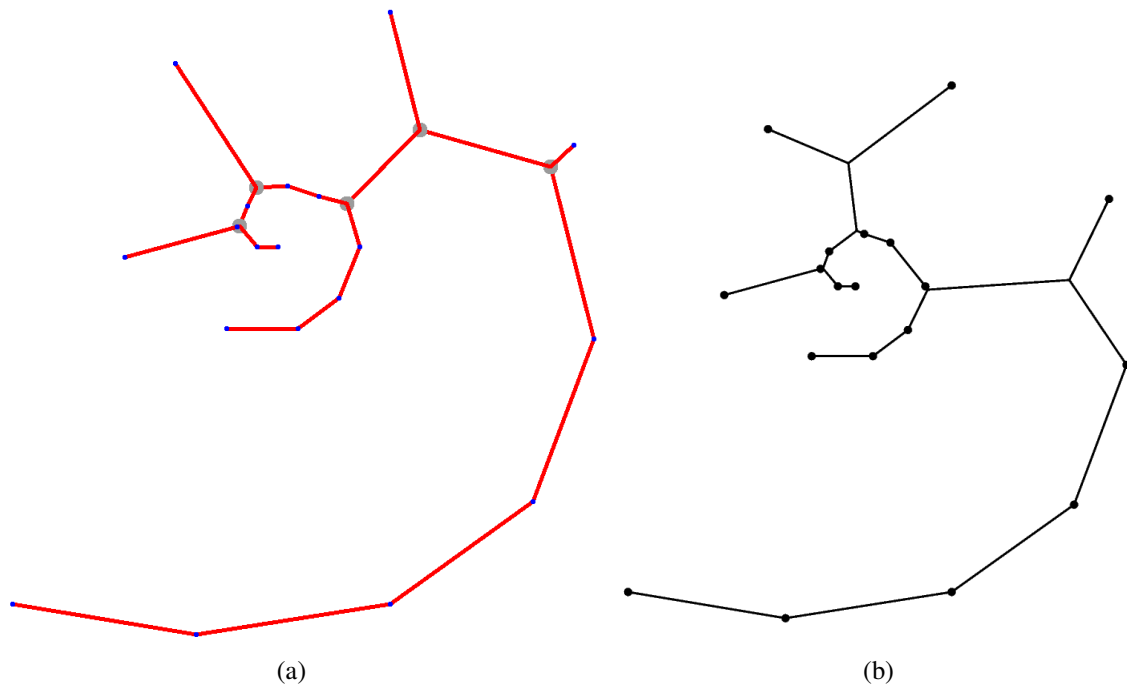


Figura 4.2: SMT construída através da heurística e do algoritmo exato para o caso 30 de [11].

clientes associados ao agregado habitacional representado. Como se pode ver na figura, o traçado passa por cima das habitações, o que não é possível na realidade. De fato estas habitações vão ter que ser contornadas por este traçado. Ainda na figura 4.8 estão ilustradas duas propostas de traçados, um correspondente à MST (representada com traço mais grosso a vermelho) e o outro obtido pela heurística (sem considerar obstáculos), representada com traço mais fino. Ambas as propostas permitem reduzir o tamanho total do traçado poupando cerca de 10% dos cabos a utilizar (no caso da MST) e poupando ainda mais se se considerar o traçado correspondente à SMT (aproximadamente 16%). Se se tiver em consideração os obstáculos existentes (neste caso as habitações)(ver figura 4.9), o comprimento da árvore aumenta, mas ainda assim é menor que o traçado "real" sem considerar os obstáculos.

<b>Caso</b>	<b>Exato Solução</b>	<b>Heurística - MSTO</b>		<b>Heurística - SMTO</b>		<b>Diferença relativa (%)</b>
		Solução	Tempo(s)	Solução	Tempo(s)	
<b>1</b>	205,162	233,147	0,002	205,165	0,007	0,00146226
<b>2</b>	276,756	292,69	1,801	279,823	1,935	1,10819639
<b>3</b>	598,713	626,957	1,831	598,675	2,365	-0,00634695
<b>4</b>	426,239	442,043	0,372	426,69211	0,383	0,10630421
<b>5</b>	945,612	966,75	8,230	949,6	8,322	0,42173746
<b>6</b>	3522,24	3619,33	78,166	3514,07	90,930	-0,23195467
<b>7</b>	1807,59	1867,8	0,415	1822,02	0,457	0,7983005
<b>8</b>	3661,85	3712,74	9,754	3665,9587	10,148	0,11220285
<b>9</b>	5865,41	6016,66	95,512	5894,06	99,453	0,4884569
<b>10</b>	282,801	285,977	0,004	282,793	0,006	-0,00282884
<b>11</b>	319,08	323,853	0,035	319,023	0,041	-0,01786386

Tabela 4.2: Resultados numéricos relativos às figuras 4.3, 4.4 e 4.5.



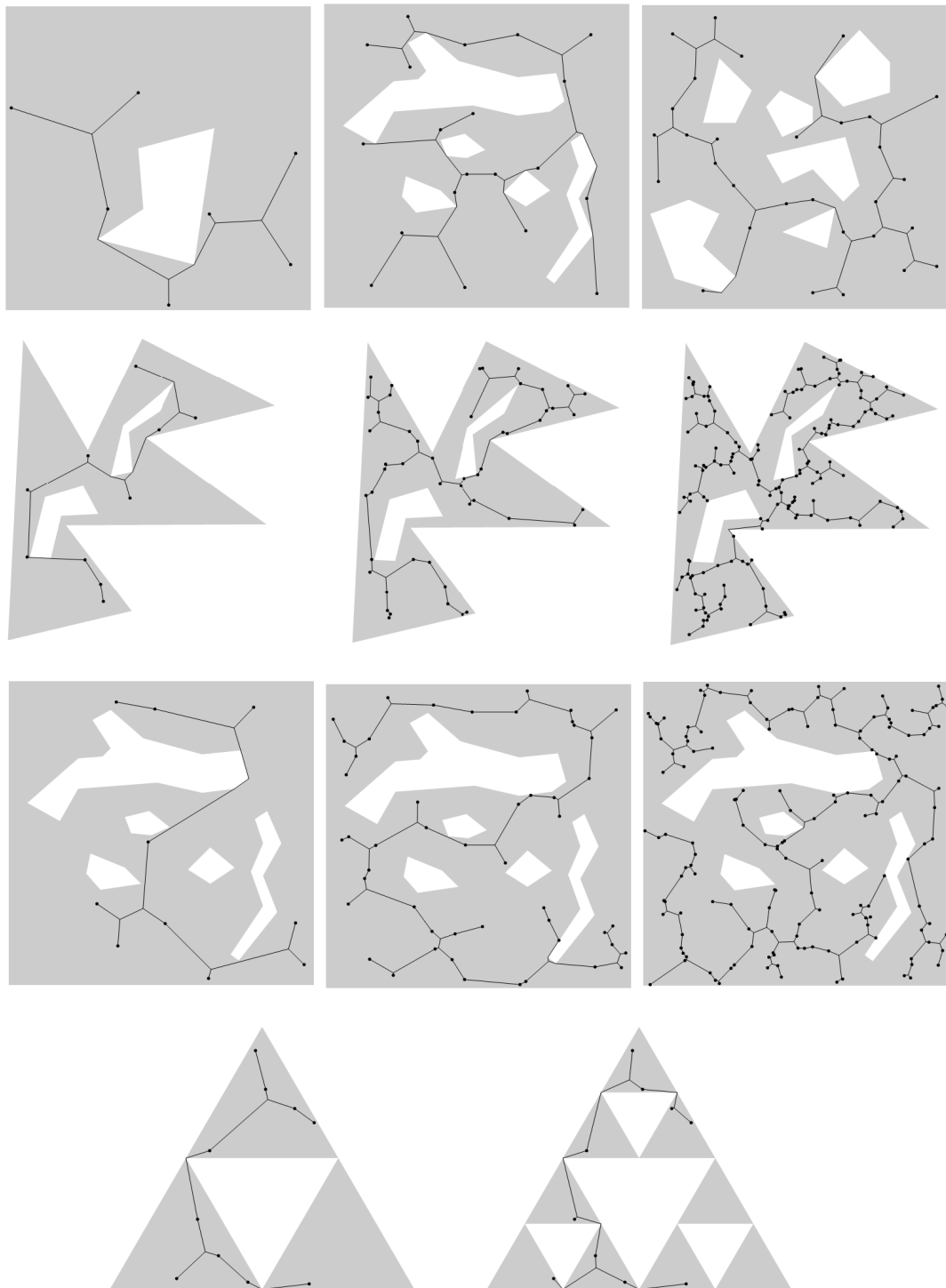


Figura 4.3: Árvores de Steiner com obstáculos [14].

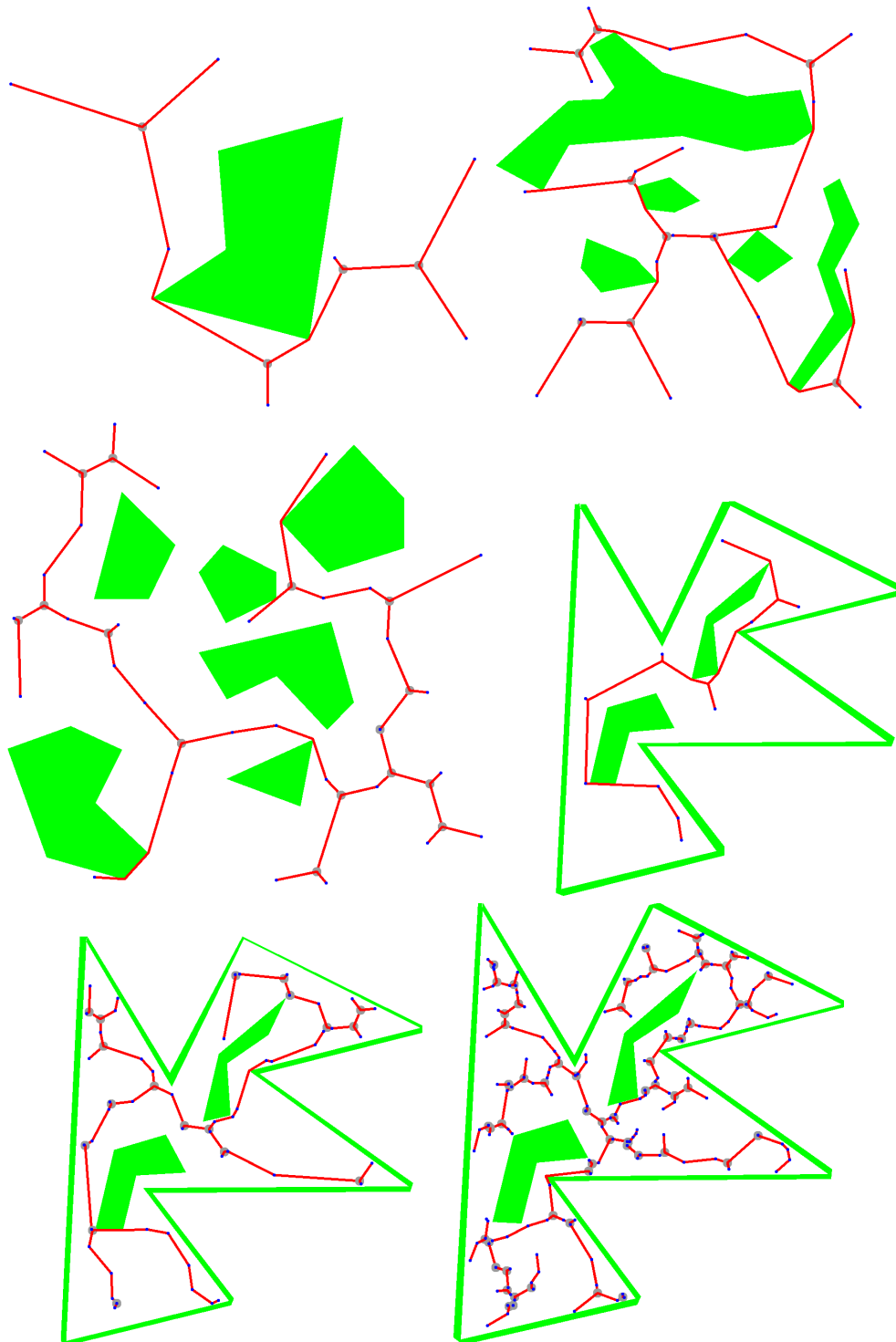


Figura 4.4: Árvores obtidas através da heurística para os primeiros seis exemplos da figura 4.3.

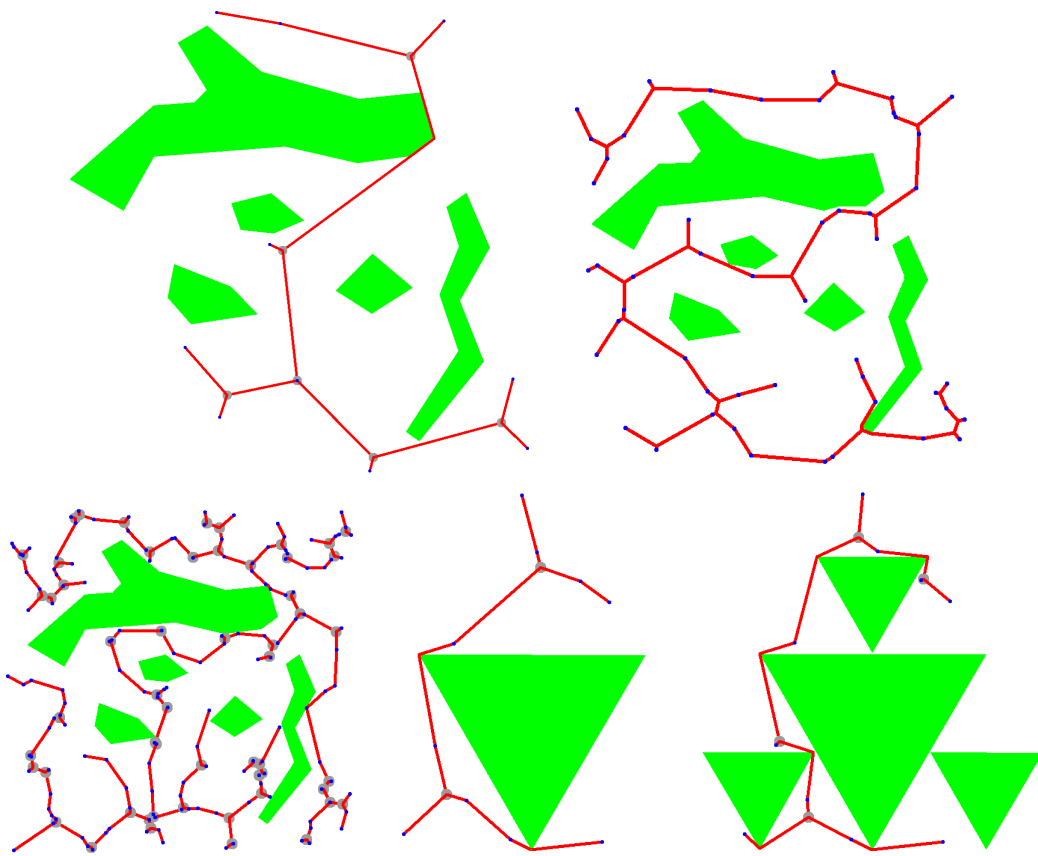


Figura 4.5: Árvores obtidas através da heurística para os últimos cinco exemplos da figura 4.3.

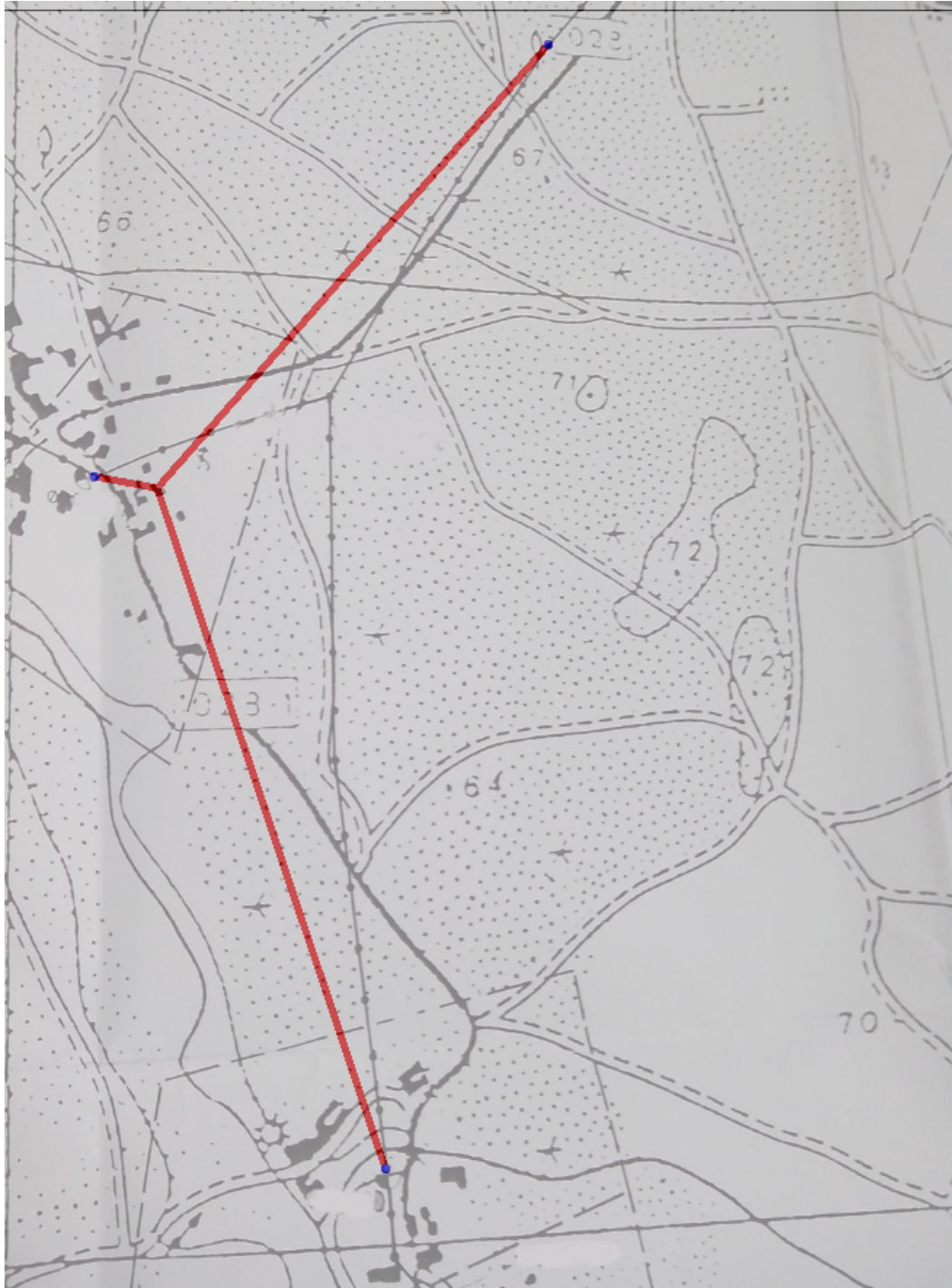


Figura 4.6: Carta topográfica com sobreposição da SMT.

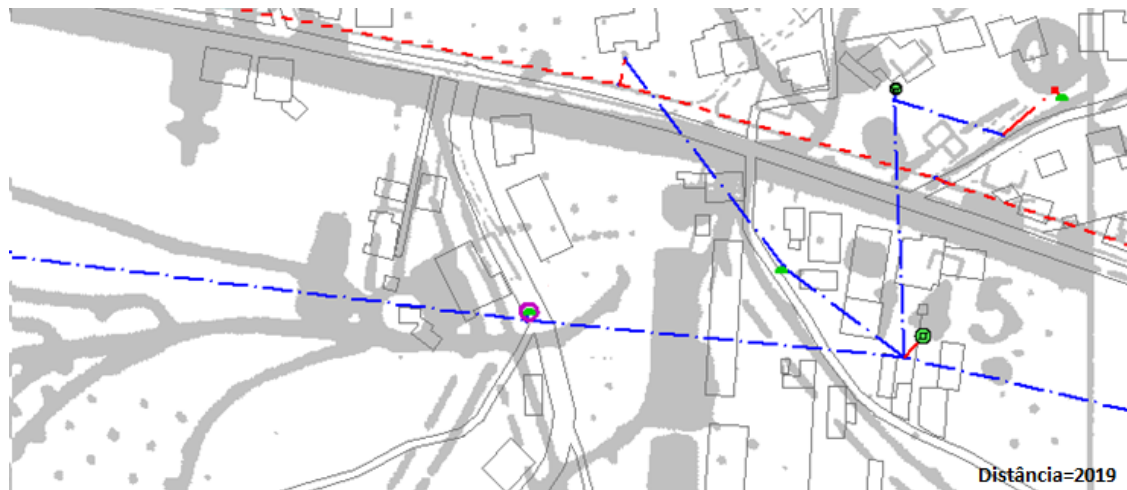


Figura 4.7: Carta topográfica com proposta inicial de um traçado.

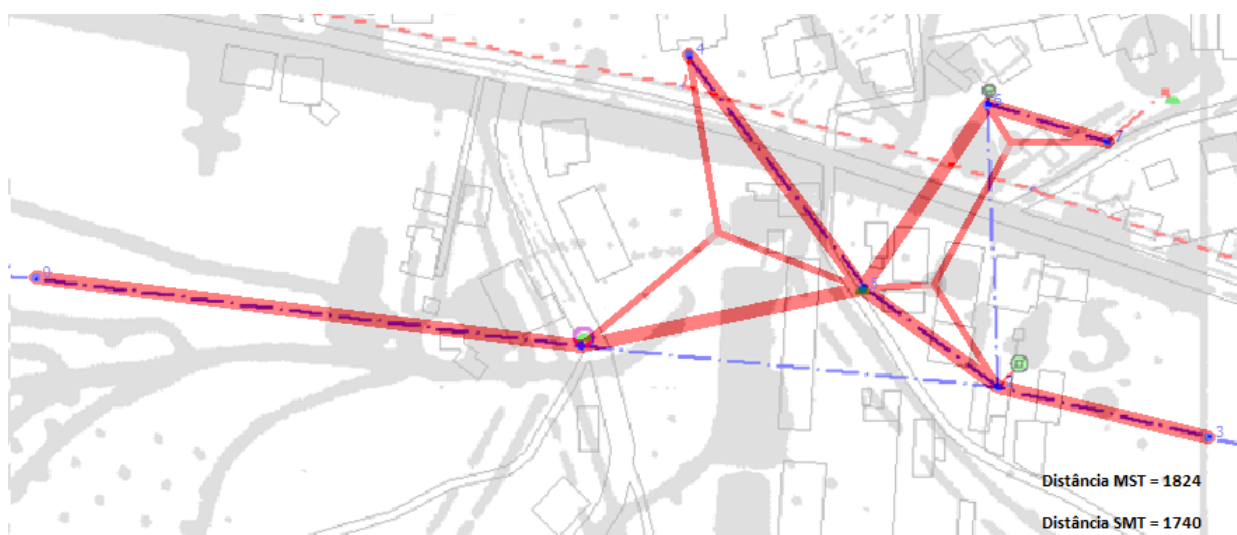


Figura 4.8: Carta topográfica com proposta de um novo traçado através da MST e SMT.

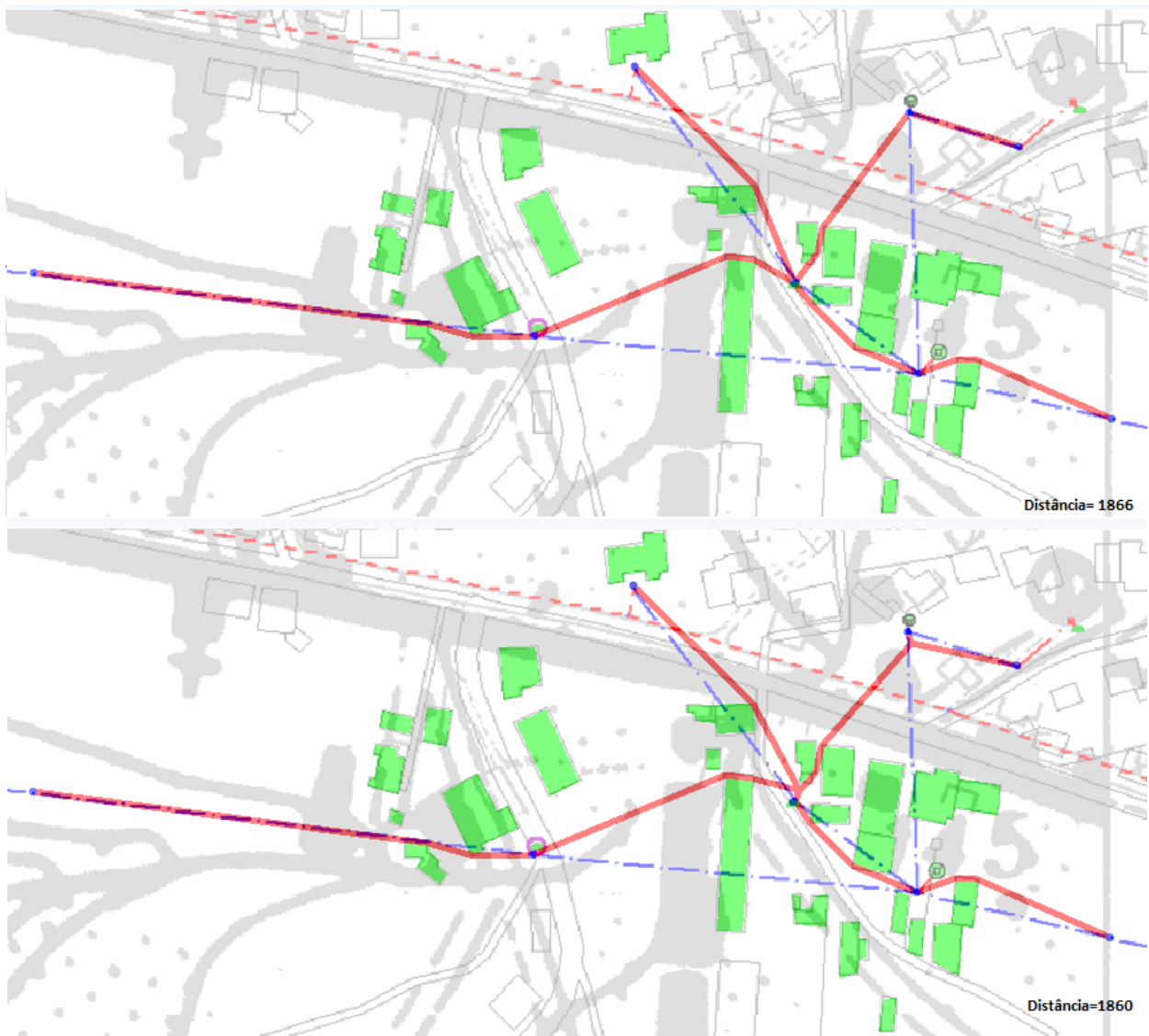


Figura 4.9: Carta topográfica com proposta de um novo traçado através da MSTO e SMTO.

# Capítulo 5

## Conclusões

Nesta dissertação de mestrado foi estudado o problema de Steiner Euclidiano e o problema de Steiner Euclidiano com obstáculos, tendo sido analisadas algumas heurísticas e algoritmos exatos propostos para a resolução destes problemas. É ainda apresentada uma nova heurística para a resolução do problema de Steiner com obstáculos. Não se conhecendo nenhum trabalho recente que aplique estes conceitos à resolução de um problema real, ao contrário do problema de Steiner retilíneo, onde existem várias propostas recentes, e sendo a proposta mais recente um algoritmo exato, com complexidade elevada, o desenvolvimento desta nova heurística, com resultados bastante próximos dos exatos, permite a continuação deste trabalho para o desenvolvimento de um sistema de apoio ao projeto de novos traçados de redes de telecomunicações.

### 5.1 Trabalho futuro

A aplicação dos algoritmos desenvolvidos num sistema de apoio ao desenvolvimento de projetos de telecomunicações obriga à consideração dos seguintes problemas:

- Análise de custos de uma nova ligação. Em determinadas situações a inclusão de um nó de Steiner poderá não ser vantajosa. Se considerarmos traçados aéreos, por exemplo, a inclusão de um nó de Steiner implica a adição de um novo poste e a consideração de mais uma ligação. O custo desta solução poderá ser superior ao que se poupa com a redução do comprimento total das ligações consideradas e o nó de Steiner poderá não ser vantajoso.
- Diferentes áreas geográficas poderão ter associados custos diferentes que poderão ter implicação na topologia a escolher. A consideração destes custos é também muito importante para a escolha de novos traçados. Assim, pode não compensar a passagem de um traçado numa zona de valor mais elevado podendo uma árvore de maior comprimento conduzir a uma solução mais barata. A geografia do terreno pode condicionar de forma semelhante o desenho de traçados de redes.
- Tendo em conta que na construção de um traçado, obrigatoriamente são necessárias novas ligações (por exemplo, de 100 em 100 metros ou de 1 em 1 quilómetro) a inclusão de uma nova derivação (nó de Steiner) poderá ser feita num espaço demasiado curto, entre as ligações referidas, podendo, mais uma vez, não ser desejável.





# Bibliografia

- [1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.
- [2] R. Courant and H. Robbins. *What is Mathematics?: An Elementary Approach to Ideas and Methods*. Oxford University Press Inc, New York, 1941.
- [3] D. R. Frank Hwang and P. Winter. *The Steiner tree problem*. Elsevier Science Publishers B.V., 1992.
- [4] M. R. Garey, R. L. Graham, and D. S. Johnson. The complexity of computing steiner minimal trees. *SIAM Journal on Applied Mathematics*, 32(4):835–859, 1977.
- [5] J.-M. Ho, G. Vijayan, and C. Wong. A new approach to the rectilinear steiner tree problem. In *Design Automation, 1989. 26th Conference on*, pages 161–166, 1989.
- [6] J.-M. Ho, G. Vijayan, and C.-K. Wong. New algorithms for the rectilinear steiner tree problem. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 9(2):185–193, 1990.
- [7] F. Hwang. A linear time algorithm for full steiner trees. *Operations Research Letters*, 4(5):235 – 237, 1986.
- [8] G. M. L. Kou and L. Berman. A fast algorithm for steiner trees. *Acta Informatica*, 15(2):141–145, June 1981.
- [9] Z. A. Melzak. On the problem of steiner. *Canad. Math. Bull.*, 4(2):143–148, May 1961.
- [10] T. Simpson. *The Doctrine and Application of Fluxions*. 1750.
- [11] J. Soukup and W. F. Chow. Set of test problems for the minimum length connection networks. *SIGMAP Bull.*, (15):48–51, Nov. 1973.
- [12] P. Winter and M. Zachariasen. Euclidean steiner minimum trees: An improved exact algorithm. *John Wiley & Sons, Inc.*, pages 149–166, 1997.
- [13] M. Zachariasen. Rectilinear full steiner tree generation. *NETWORKS*, 33:125–143, 1997.
- [14] M. Zachariasen and P. Winter. Obstacle-avoiding euclidean steiner trees in the plane: An exact algorithm. In *In Lecture Notes in Computer Science*, volume 1619, pages 282–295. Springer, 1999.

- [15] M. Zachariasen, M. Zachariasen, P. Winter, and P. Winter. Concatenation-based greedy heuristics for the euclidean steiner tree problem. Technical report, Algorithmica, 1997.