



Universidade de Brasília

Faculdade de Tecnologia
Departamento de Engenharia Elétrica

Compressão do sinal de cor de uma nuvem de pontos com codificador aritmético

Miguel de Carvalho Pachá

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Engenharia Elétrica

Orientador

Prof. Dr. Eduardo Peixoto Fernandes da Silva

Brasília
2019



Universidade de Brasília

Faculdade de Tecnologia
Departamento de Engenharia Elétrica

Compressão do sinal de cor de uma nuvem de pontos com codificador aritmético

Miguel de Carvalho Pachá

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Engenharia Elétrica

Prof. Dr. Eduardo Peixoto Fernandes da Silva (Orientador)
Departamento de Engenharia Elétrica
Universidade de Brasília

Prof. Dr. Camilo Chang Dorea Prof. Dr. Renan Utida Ferreira
CIC/UnB FGA/UnB

Prof. Dr. João Paulo Leite
Coordenador do Bacharelado em Engenharia Elétrica

Brasília, 20 de novembro de 2019

1 Introdução

São muitas as aplicações de imagens tridimensionais: são utilizadas para criar ambientes imersivos de realidade virtual ou aumentada [1, 2, 3]; aliadas a técnicas de visão computacional, são cruciais para a representação do ambiente nos sistemas de navegação de veículos autônomos [4, 5, 6]; na indústria, também podem ser utilizados para controle de qualidade de peças [7]; na geologia e na geografia, são úteis para representar a cobertura do terreno [8, 9]; nas ciências forenses já existem estudos sobre o uso de *scanners* 3D na preservação de provas [10, 11].

No entanto, o custo de transmissão e armazenamento dos arquivos de imagens 3D pode ser um entrave para a disseminação destas aplicações. É portanto interessante estudar maneiras de diminuir este custo utilizando a compressão de dados. Este trabalho estuda a codificação de um tipo imagem tridimensional chamado *nuvem de pontos* (em inglês, *point clouds*). Como o próprio nome informa, nuvens de pontos representam objetos como um conjunto de pontos do espaço tridimensional. Este tipo de representação é muito utilizada em aplicações de visão computacional, com destaque para técnicas de navegação e de detecção de obstáculos. As nuvens de pontos aqui estudadas são imagens que associam a cada um dos pontos um valor de cor RGB. Algumas imagens de uma nuvem de pontos colorida podem ser vistas na figura 1.1.

Os trabalhos já realizados sobre a compressão deste tipo de imagem dividem, em geral,

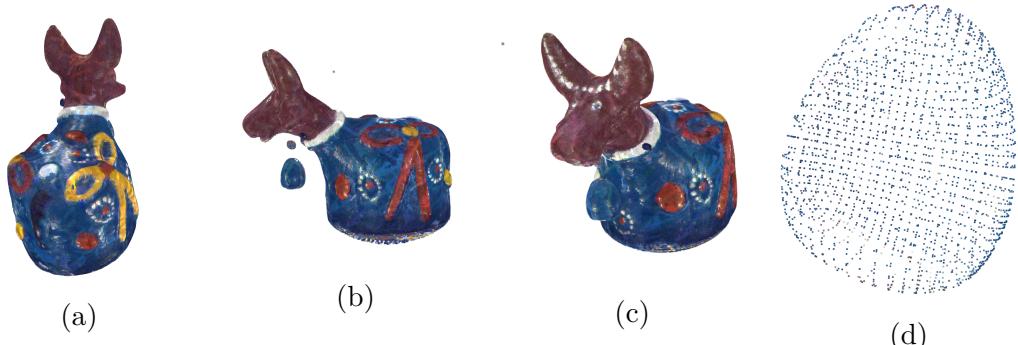


Figura 1.1: Exemplo de uma imagem em nuvem de pontos. Na subfigura (d), o detalhe do sino do boi. Fonte: [12]



(a) Point cloud 1. Nome do arquivo: ricardo9/frame0000.ply
(b) Point cloud 2. Nome do arquivo: ricardo9/frame0091.ply



(c) Point cloud 3. Nome do arquivo: phil9/frame0116.ply
(d) Point cloud 4. Nome do arquivo: andrew9/0263.ply



(e) Point cloud 5. Nome do arquivo: sarah9/frame0146.ply

Figura 1.2: Renderização frontal das nuvens de pontos utilizadas.

tridimensionais, onde cada ponto possui atributo de cor. Quando as coordenadas destes é quantizada, eles são chamados *voxels*, do inglês *volume picture element*. Um exemplo de nuvem de pontos pode ser visto na figura 2.1.

A representação em nuvens de pontos é uma representação esparsa, diferente da representação de uma imagem bidimensional (como uma foto, por exemplo). Enquanto uma imagem bidimensional geralmente é entendida como uma função que associa a todo e qualquer pixel um dado de cor, a representação de uma *point cloud* não é assim, porque nem todos os *voxels* são ocupados. Para quantificar a taxa de dados de uma representação de uma nuvem de pontos, mede-se quantos bits são gastos, em média, para representar um voxel ocupado. A unidade é abreviada *b pov* (*bits per occupied voxel*).

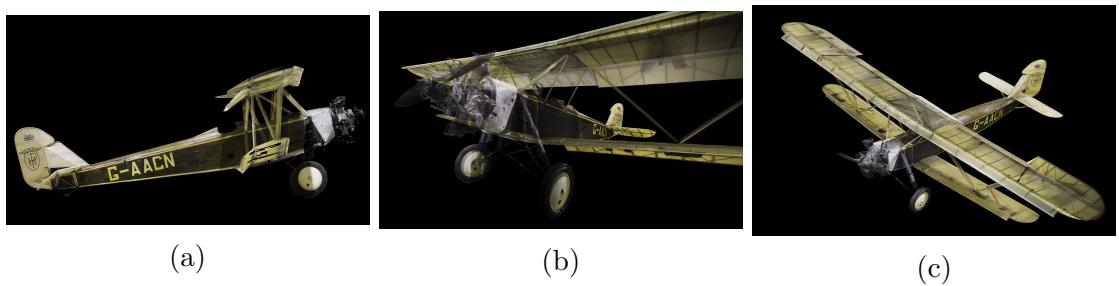


Figura 2.1: Uma nuvem de pontos representando o biplano Handley Page H.P.39. Fonte: [16]

2.1.2 Aplicações

A representação em nuvem de pontos vem tomado o lugar das representações poligonais em várias aplicações, especialmente em visão computacional e realidade virtual[2]. As *point clouds* podem ser usadas para criar ambientes imersivos, seja para entretenimento (como filmes e jogos) ou para comunicação pessoal (vídeo-conferências tridimensionais [1]). Também existem aplicações em sistemas de realidade aumentada [3], e já foram apresentados sistemas de interação homem-máquina baseados em nuvens de pontos[17]. Uma aplicação cada vez mais explorada são os sistemas de visão computacional para veículos autônomos, não só para navegação [4] e detecção de obstáculos [5] de veículos terrestres, e também para a coordenação de manobras complexas de veículos aéreos não-tripulados[6]. Aliadas a sistemas de informação geográfica, nuvens de pontos também podem ser utilizadas também para representar relevo e cobertura do terreno, tendo assim uma possível aplicação na agrimensura e no manejo de recursos naturais[8, 9]. Finalmente, nuvens de pontos podem ser usadas para fins acadêmicos (como o estudo de ambientes ou objetos de valor histórico [11]) e até forenses (preservação de cenas de crime e documentação de provas [18]).

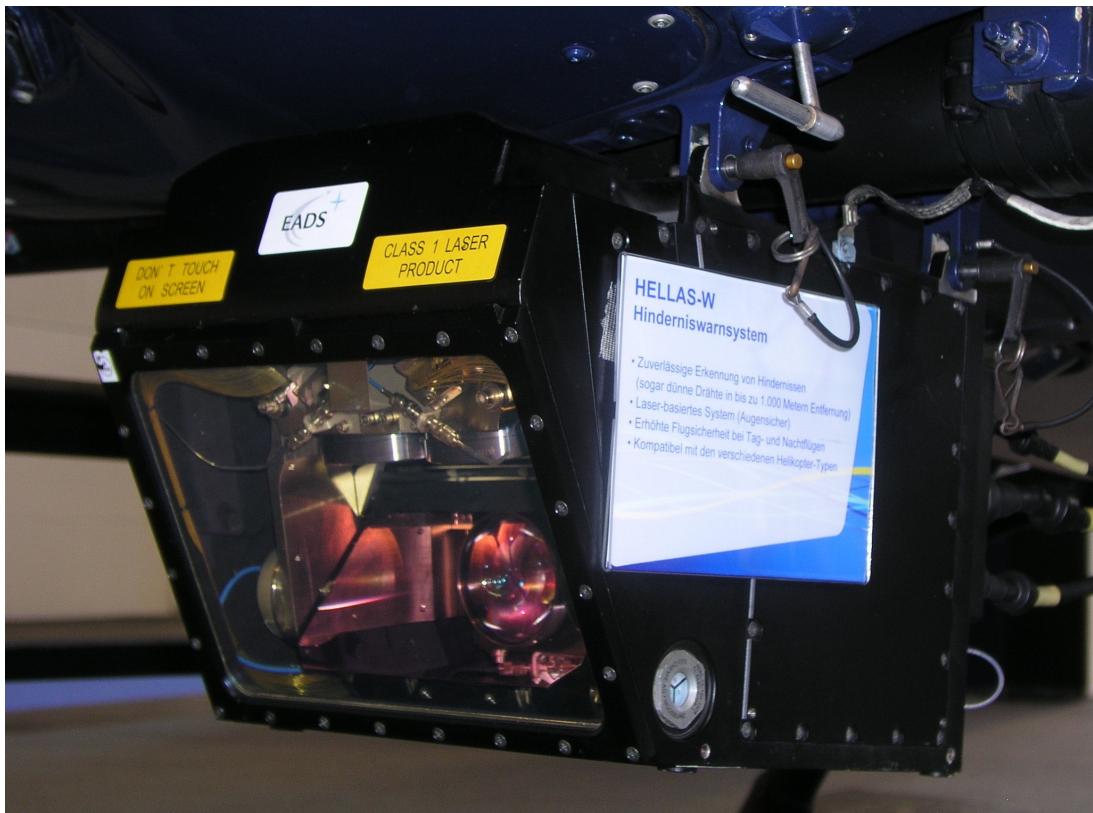


Figura 2.2: Um carro autônomo com sistema de *LiDAR* Fonte:Waymo

2.1.3 Obtenção e transmissão de nuvens de pontos

Nuvens de pontos podem ser obtidas seja a partir de dados brutos gerados por tecnologias especializadas, e também a partir do cruzamento de imagens convencionais em duas dimensões, usando o princípio da visão estereoscópica. A tecnologia mais usada para obtenção destes dados é o *LiDAR* (*Light Detection And Ranging*, detecção e telemetria por luz). Outra tecnologia que pode ser empregada são as câmeras RGB-D, que associam cada pixel a um valor de distância além da informação de cor.

Na figura 2.3 podemos ver exemplos de equipamentos de *LiDAR*. Este processo de obtenção de *point clouds* é denominado registro de pontos (*point set registration*). Um dos formatos de arquivo utilizado para salvar nuvens de pontos é o formato PLY (Polygon File Format [19]). Apesar de conter menos informação do que representações poligonais, este tipo de modelagem gera arquivos muito grandes, o que dificulta sua transmissão. De fato, o tamanho de cada quadro dos vídeos tridimensionais estudados neste trabalho é da ordem de alguns *megabytes*. Portanto, é interessante comprimir este tipo de informação para transmissão ou arquivamento.



(a) Um sistema HELLAS (*Helikopter Laser Radar*) montado em um helicóptero EC-125 da Polícia Federal da Alemanha. Fonte: User:Waerfelu/Wikimedia Commons (CC-BY-SA-3.0)



(b) Este *LiDAR* fixo é utilizado em um cruzamento em nível para detectar a presença de pessoas ou veículos na via do trem. Fonte: Yoji Isaka/Wikimedia Commons (CC-BY-2.1)

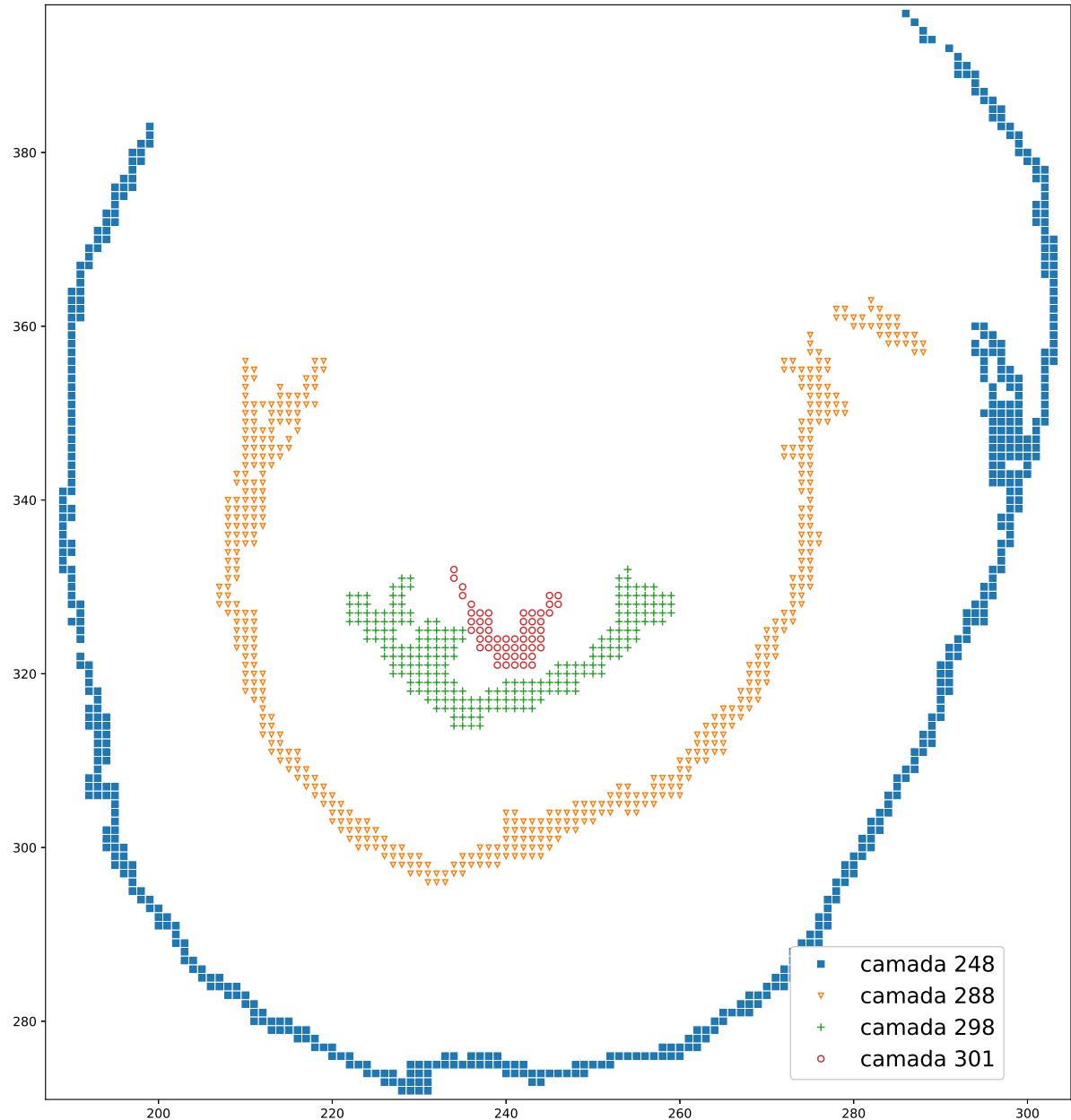
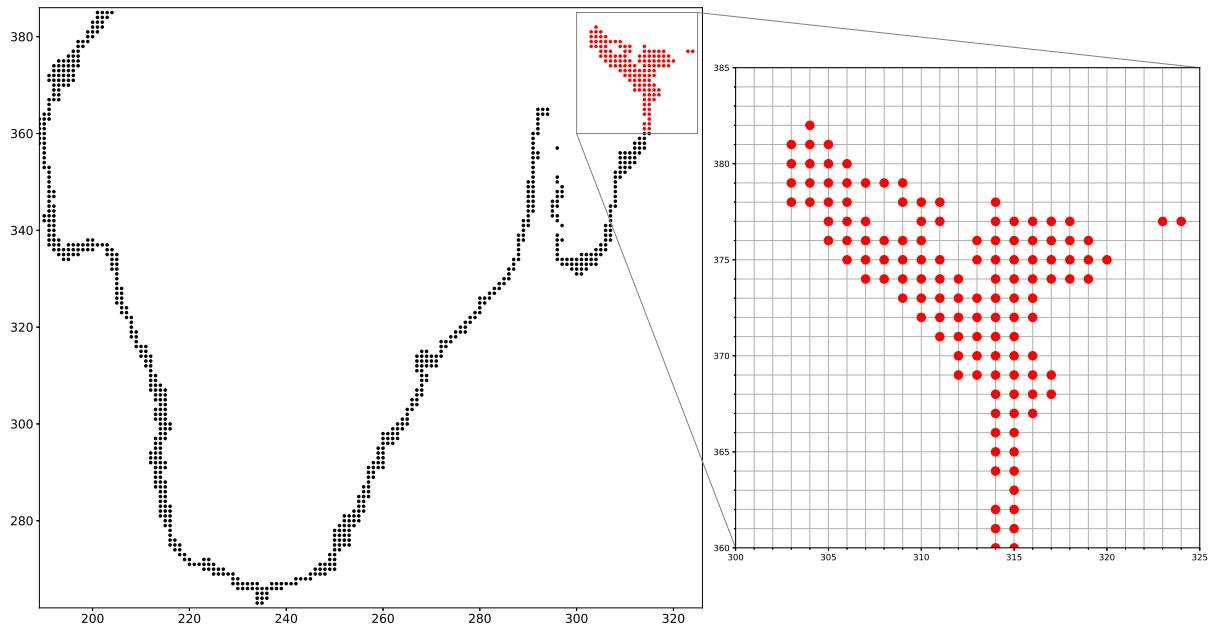
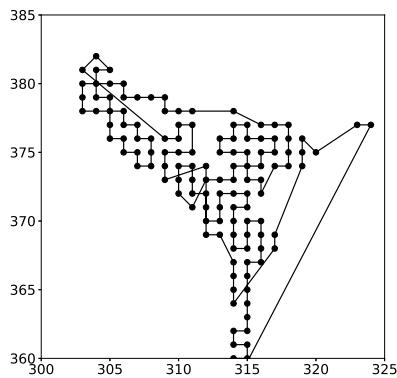


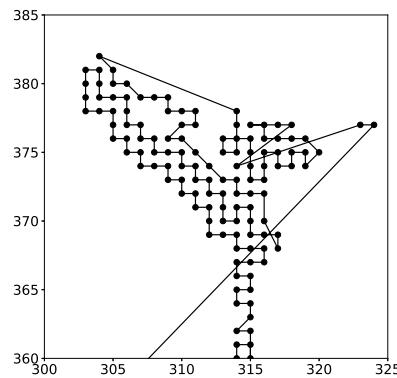
Figura 3.2: Alguns exemplos de camadas obtidas a partir do arquivo frame0000.ply do vídeo ricardo9



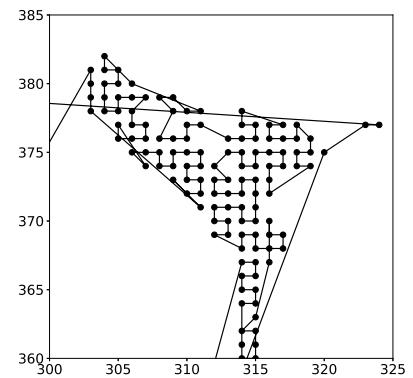
(a) Visão geral da camada, destacando o detalhe mostrado nas subfiguras abaixo



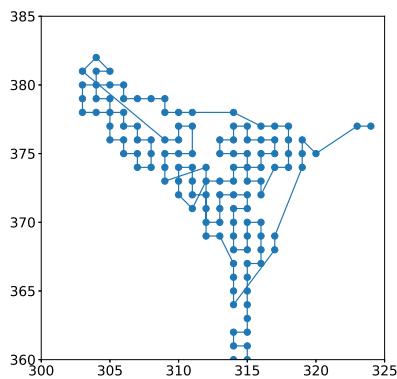
(b) Filamento gerado
(método “exato”)



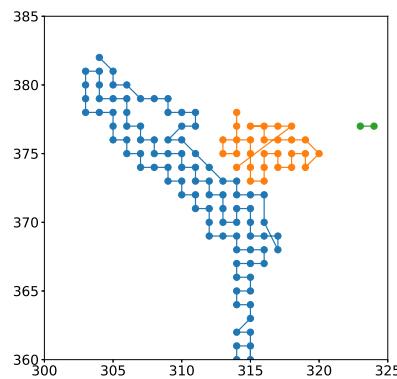
(c) Filamento gerado
(método “FLANN”)



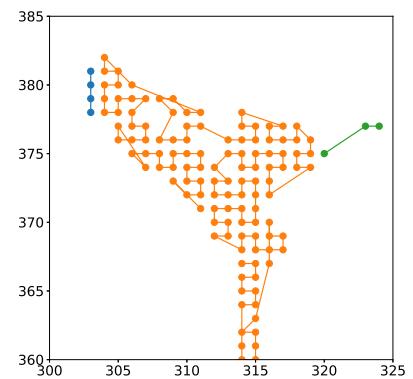
(d) Filamento gerado
(método “Hilbert”)



(e) Filamentos cortados
(limiar=8, “exato”)



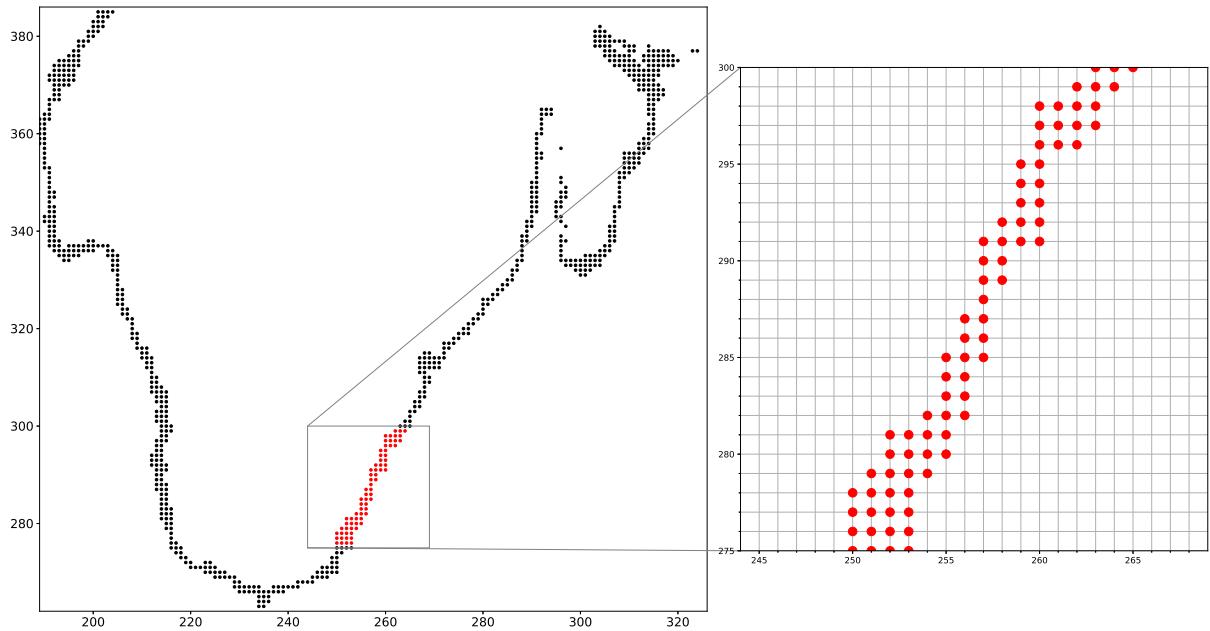
(f) Filamentos cortados
(limiar=8, “flann”)



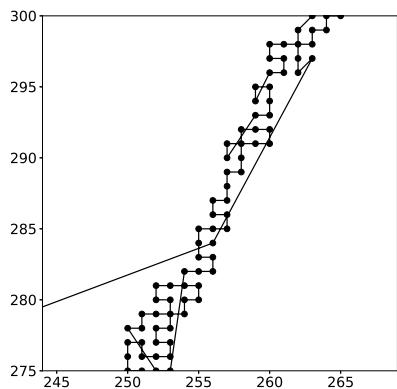
(g) Filamentos cortados
(limiar=8, “Hilbert”)

Figura 3.3: Ilustração da camada 150 da *point cloud* 1 em diferentes estágios do processamento, para os métodos “Exato”, “FLANN” e “Hilbert”.

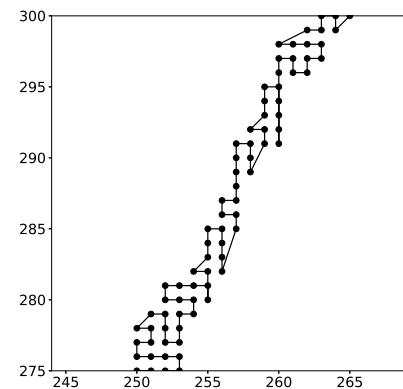
A janela do detalhe é $300 < x < 325, 360 < y < 385$



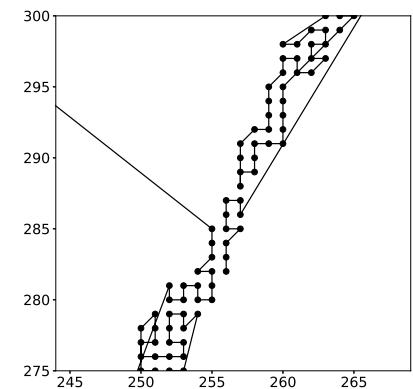
(a) Visão geral da camada, destacando o detalhe mostrado nas subfiguras abaixo



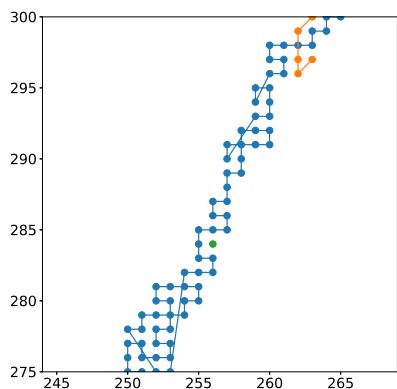
(b) Filamento gerado
(método “exato”)



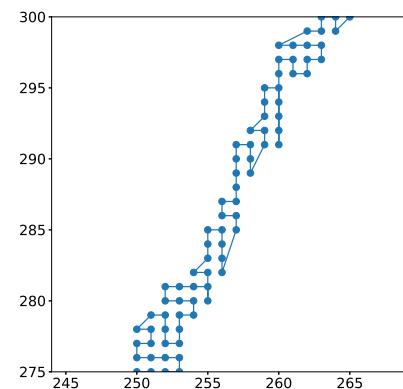
(c) Filamento gerado
(método “FLANN”)



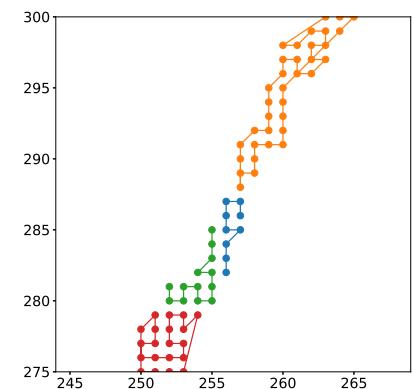
(d) Filamento gerado
(método “Hilbert”)



(e) Filamentos cortados
(limiar=8, “exato”)



(f) Filamentos cortados
(limiar=8, “flann”)



(g) Filamentos cortados
(limiar=8, “Hilbert”)

Figura 3.4: Ilustração da camada 150 da *point cloud* 1 em diferentes estágios do processamento, para os métodos “Exato”, “FLANN” e “Hilbert”.

A janela do detalhe é $244 < x < 269, 275 < y < 300$

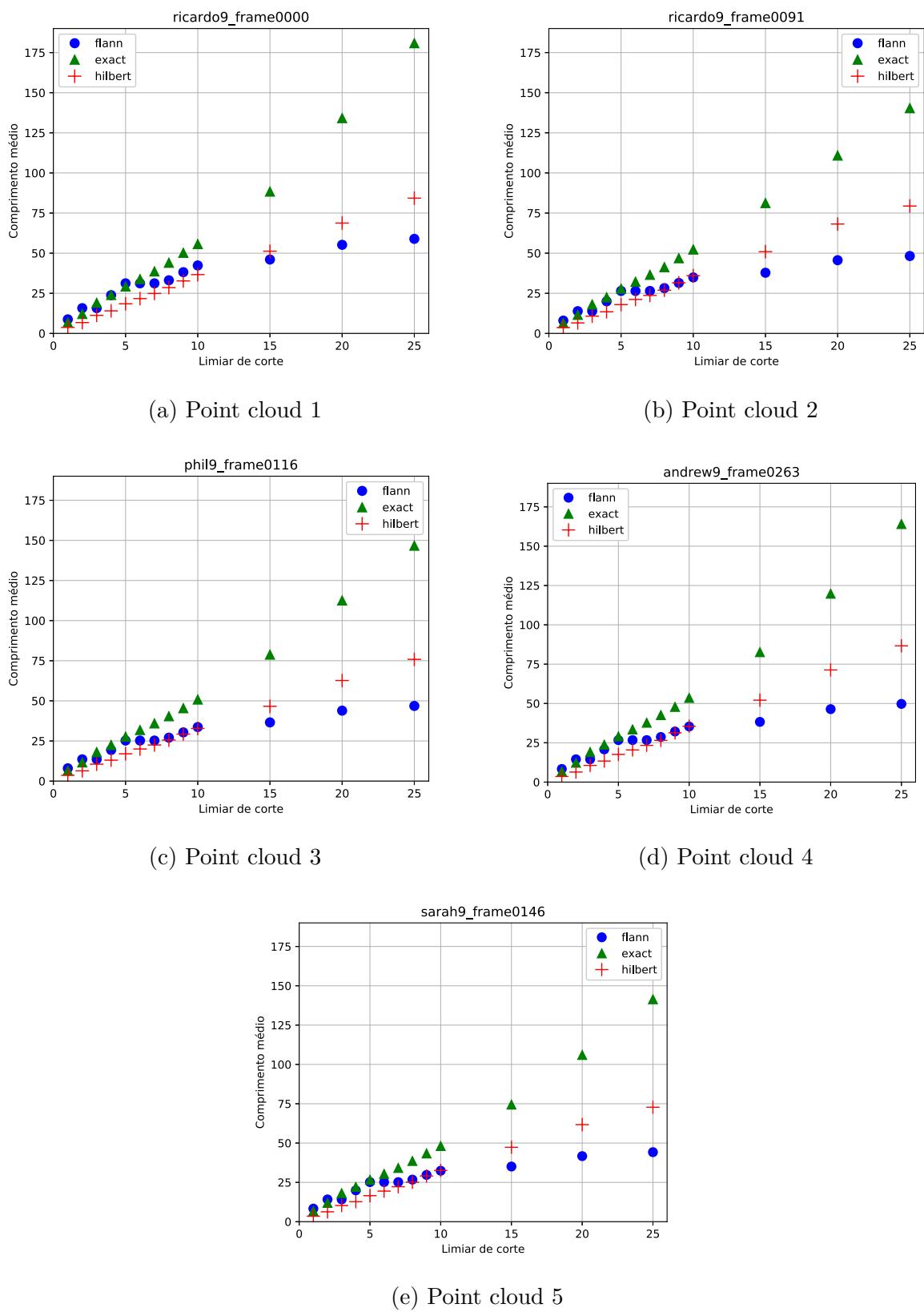


Figura 4.1: Comprimento médio dos filamentos em função do limiar de distância de corte, para diferentes métodos de geração de filamento

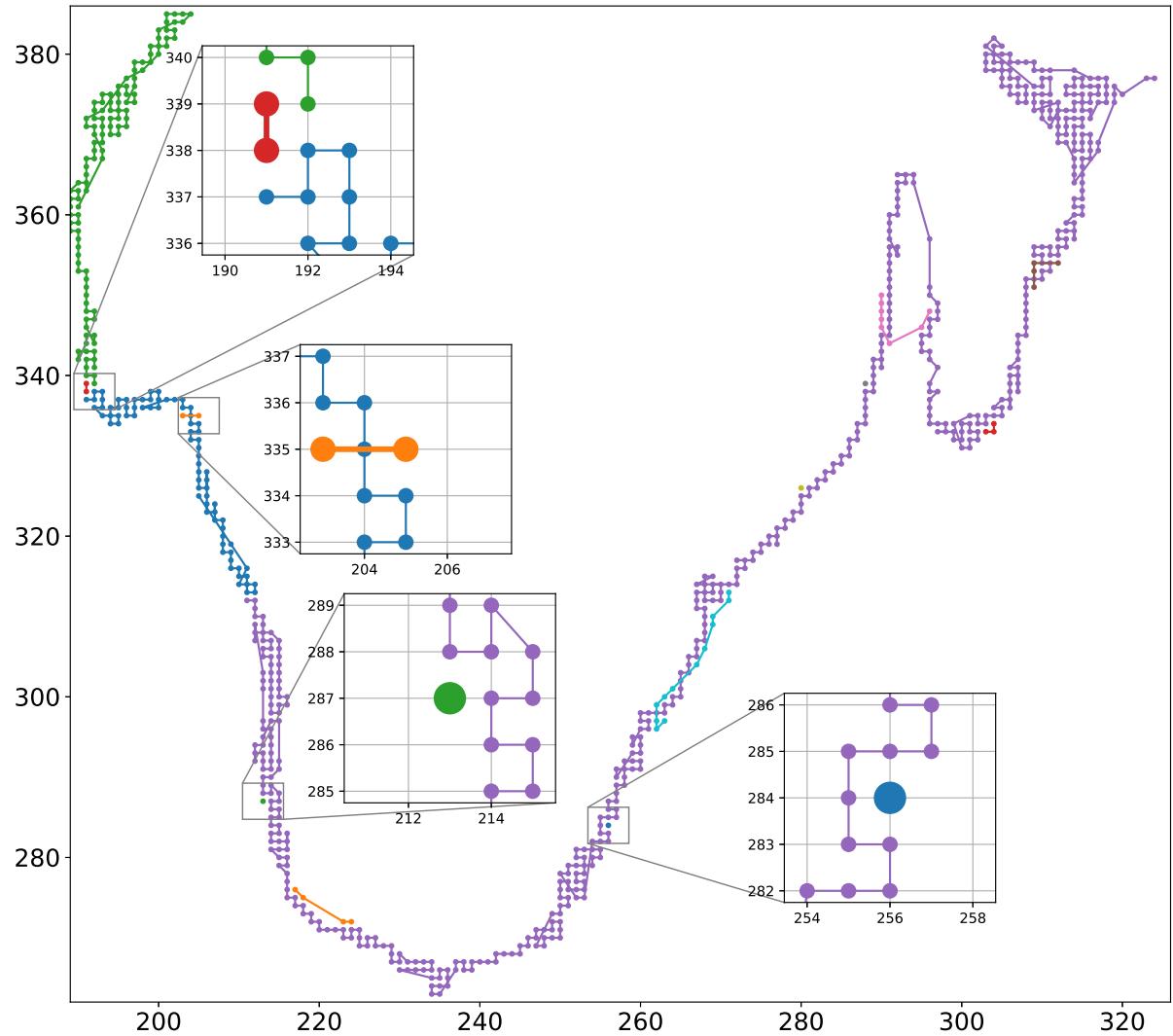


Figura 4.2: Exemplos de filamentos muito curtos compostos por pontos “esquecidos” pelo algoritmo “exato”.

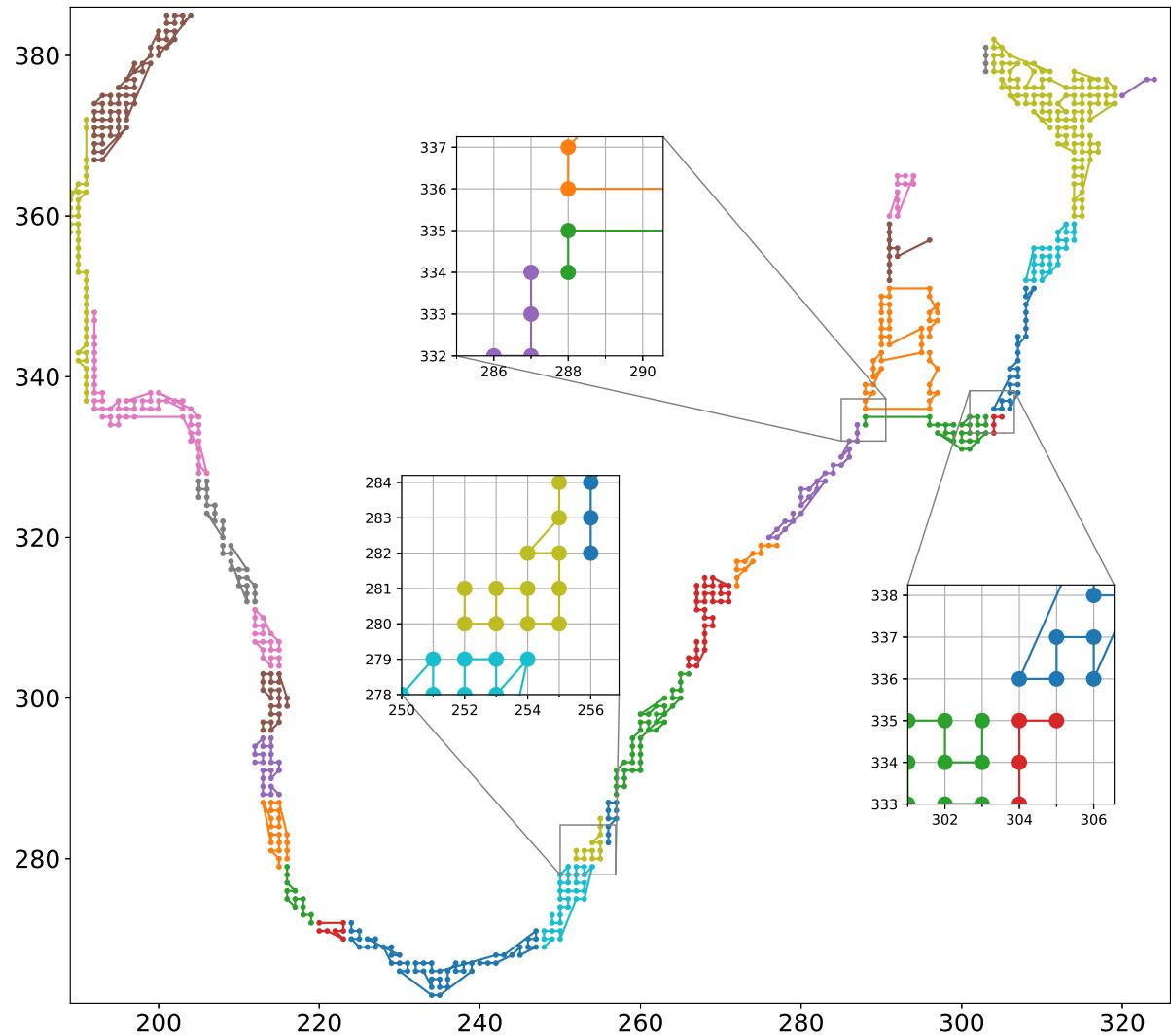
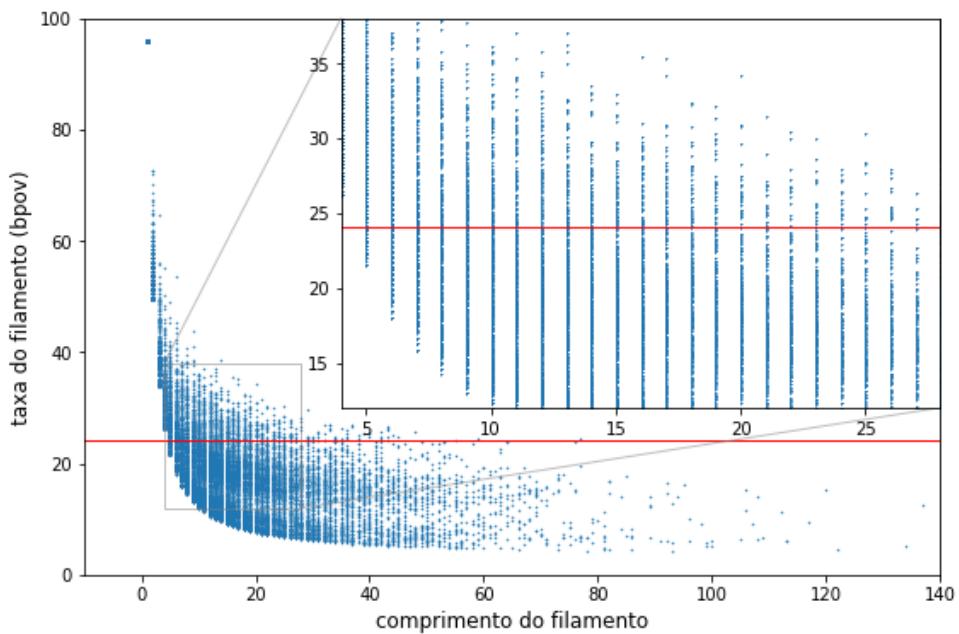
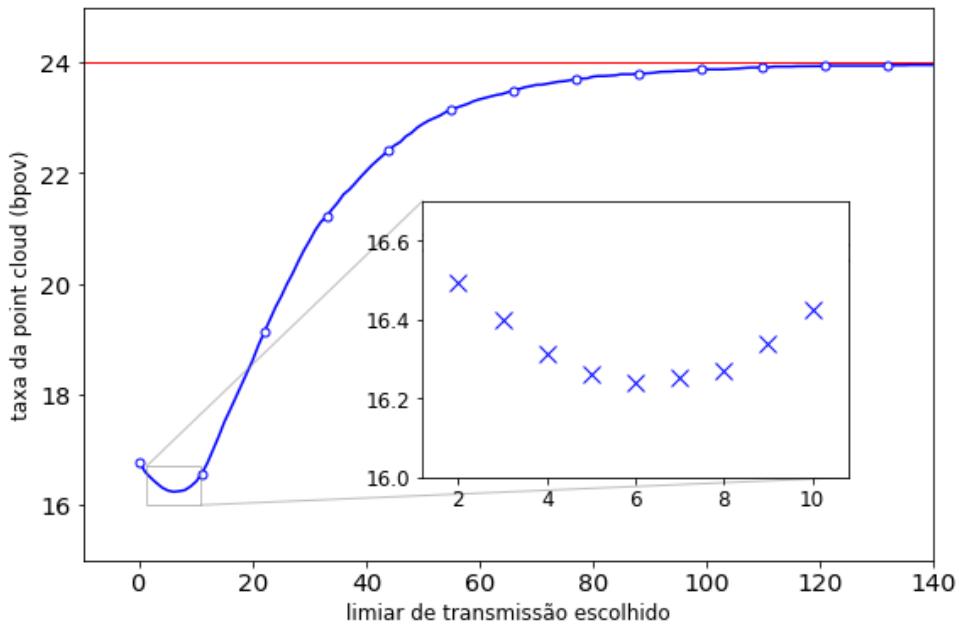


Figura 4.3: Regiões de fratura do método “Hilbert”



(a) Cada ponto representa um filamento



(b) Minimizando a taxa da *point cloud*

Figura 4.4: Taxa dos filamentos (a) e taxa de toda a nuvem (b) em função do limiar de transmissão. Dados obtidos a partir da *Point cloud*: sarah9/frame0116, com algoritmo “Hilbert” e limiar de corte = 5.

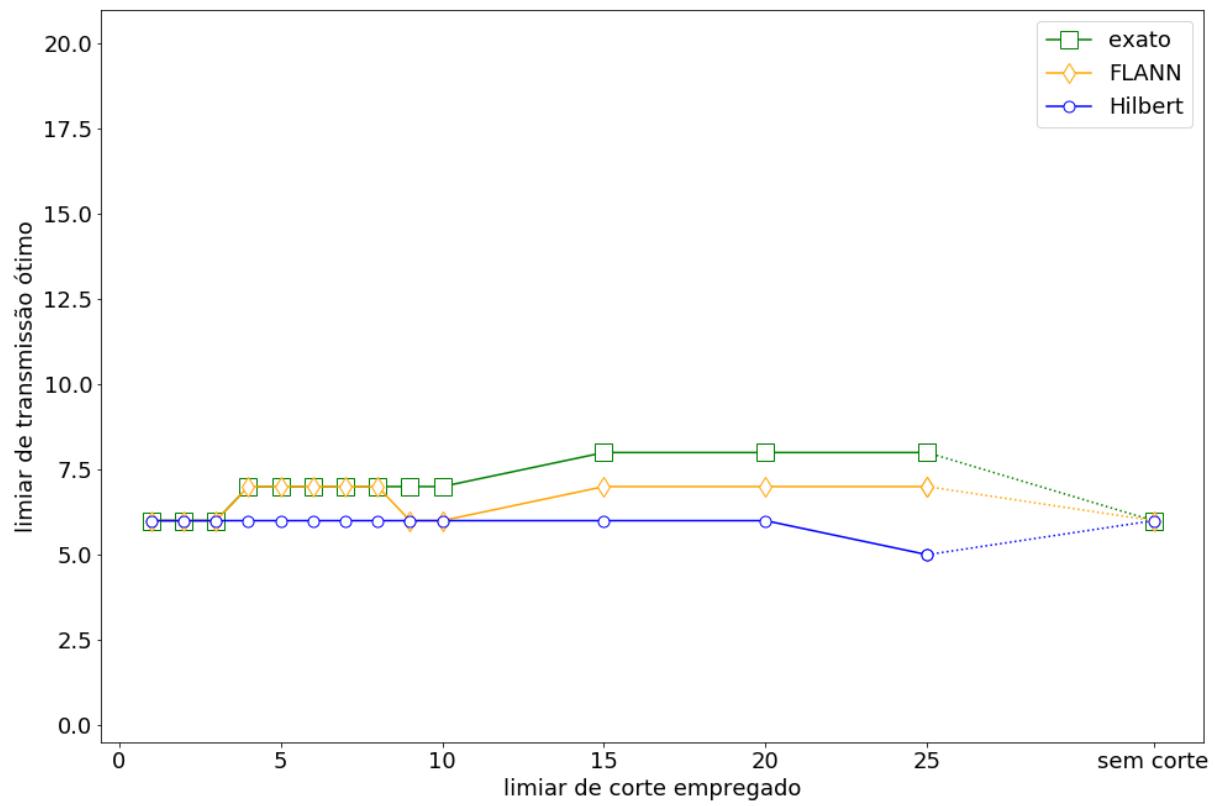
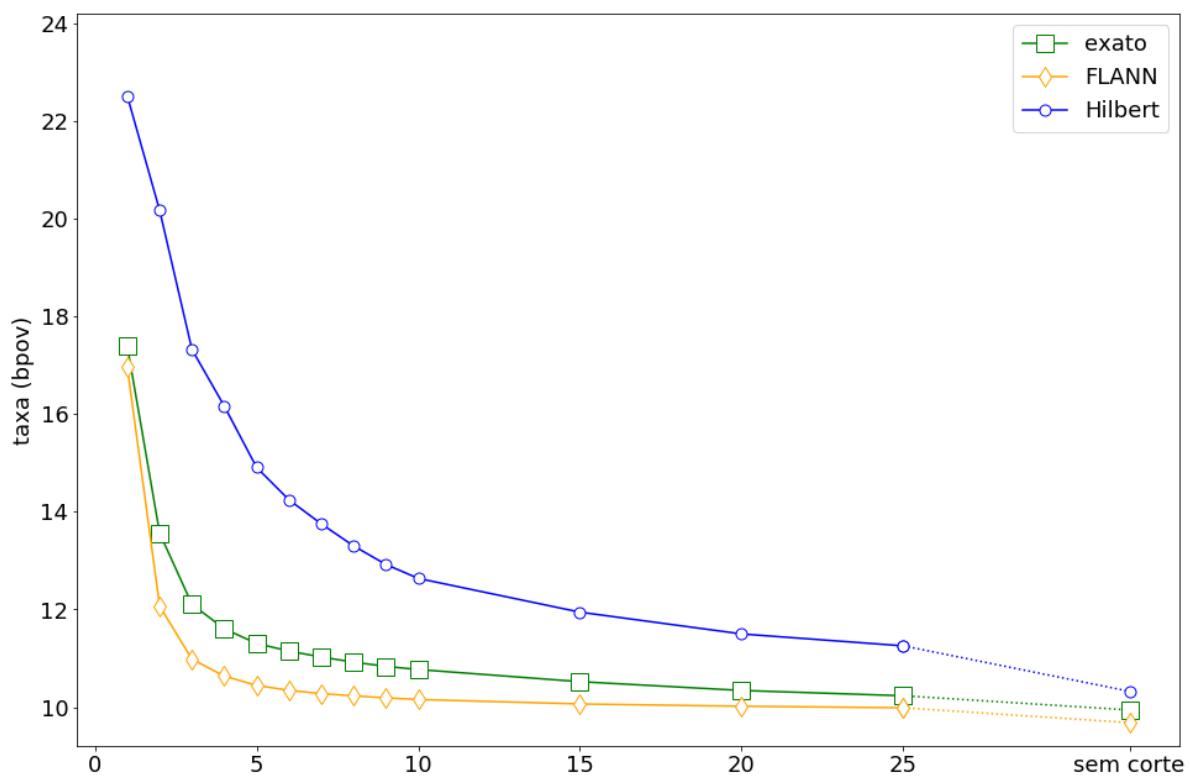


Figura 4.5: Resultados para o arquivo ricardo9/frame0000

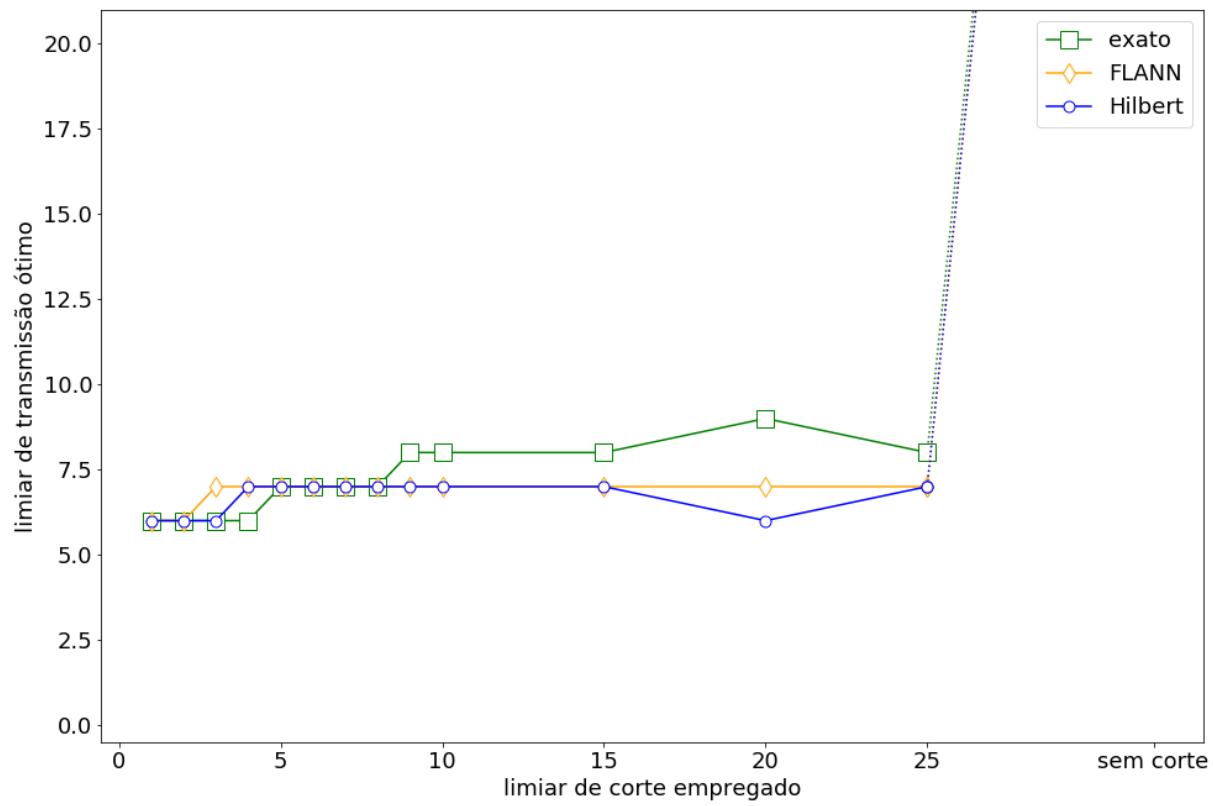
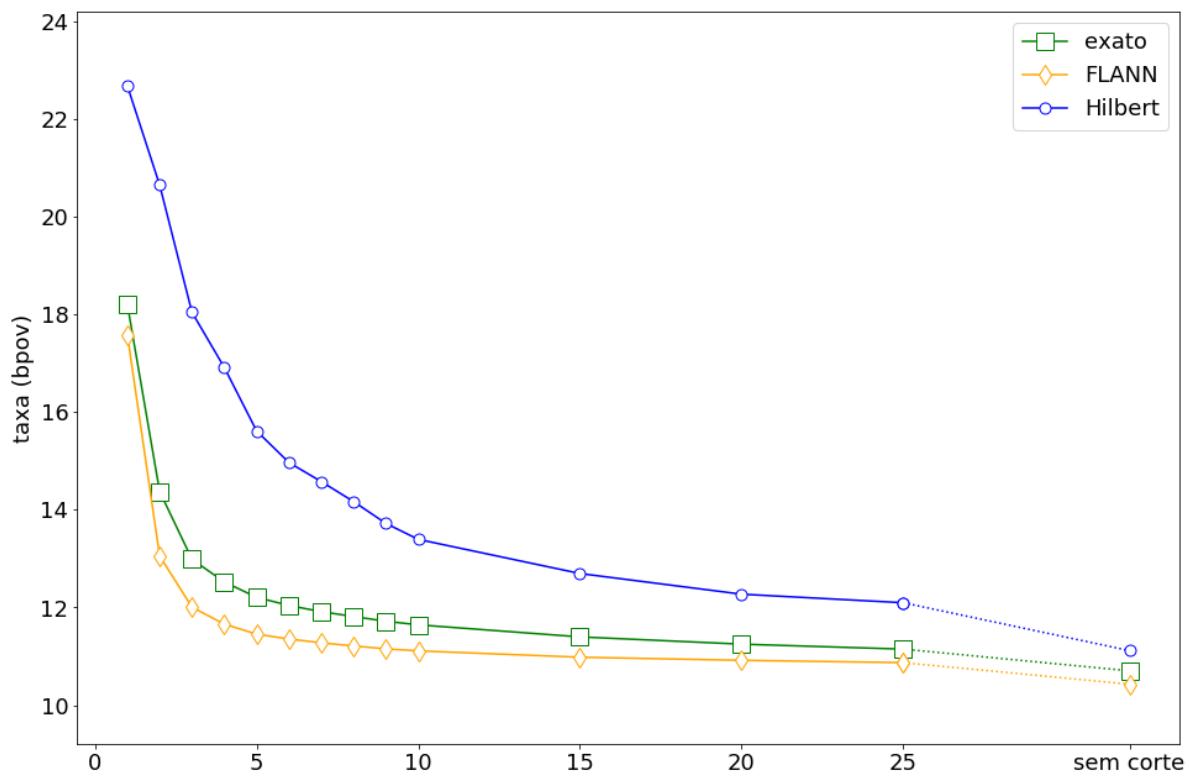


Figura 4.6: Resultados para o arquivo ricardo9/frame0091

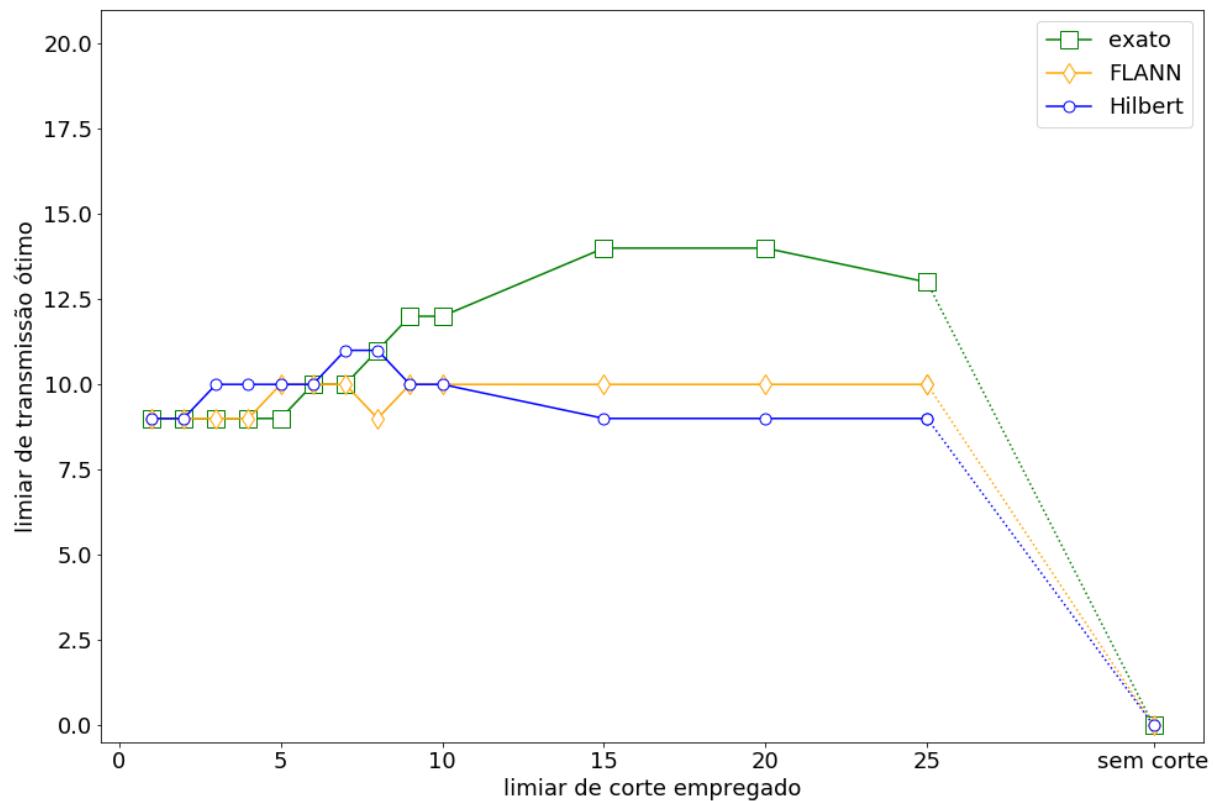
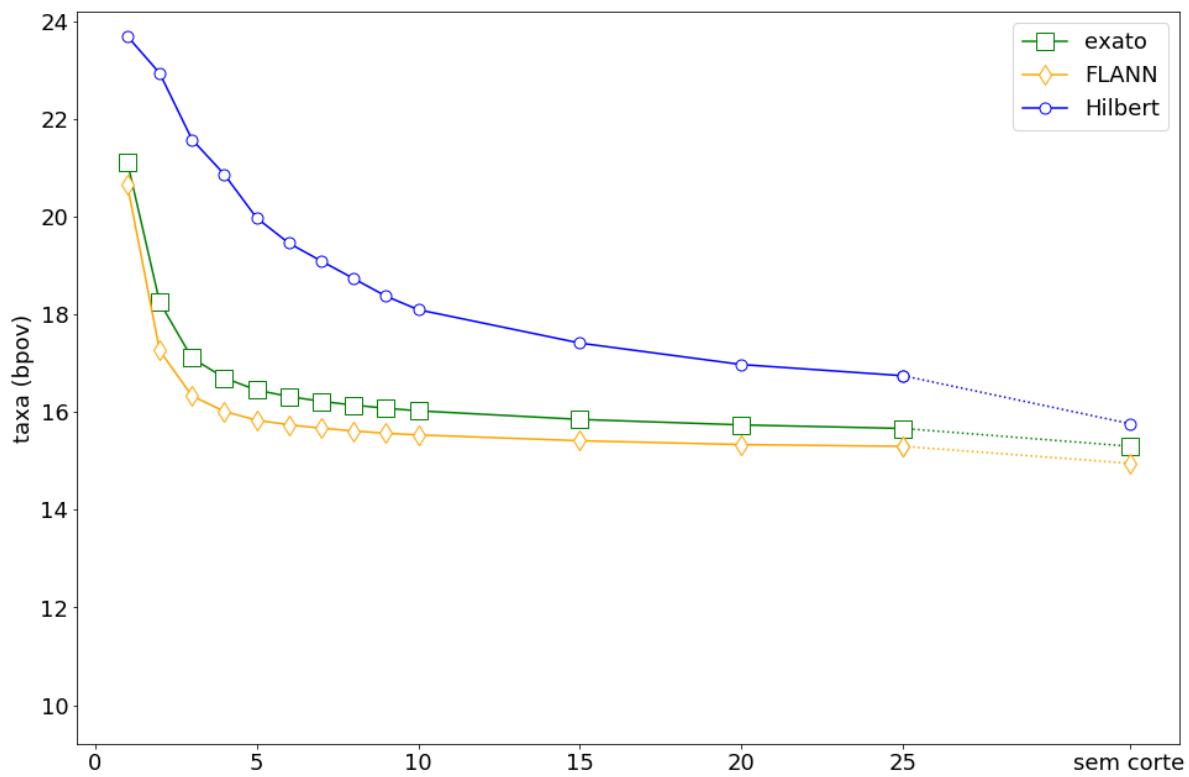


Figura 4.7: Resultados para o arquivo phil9/frame0116

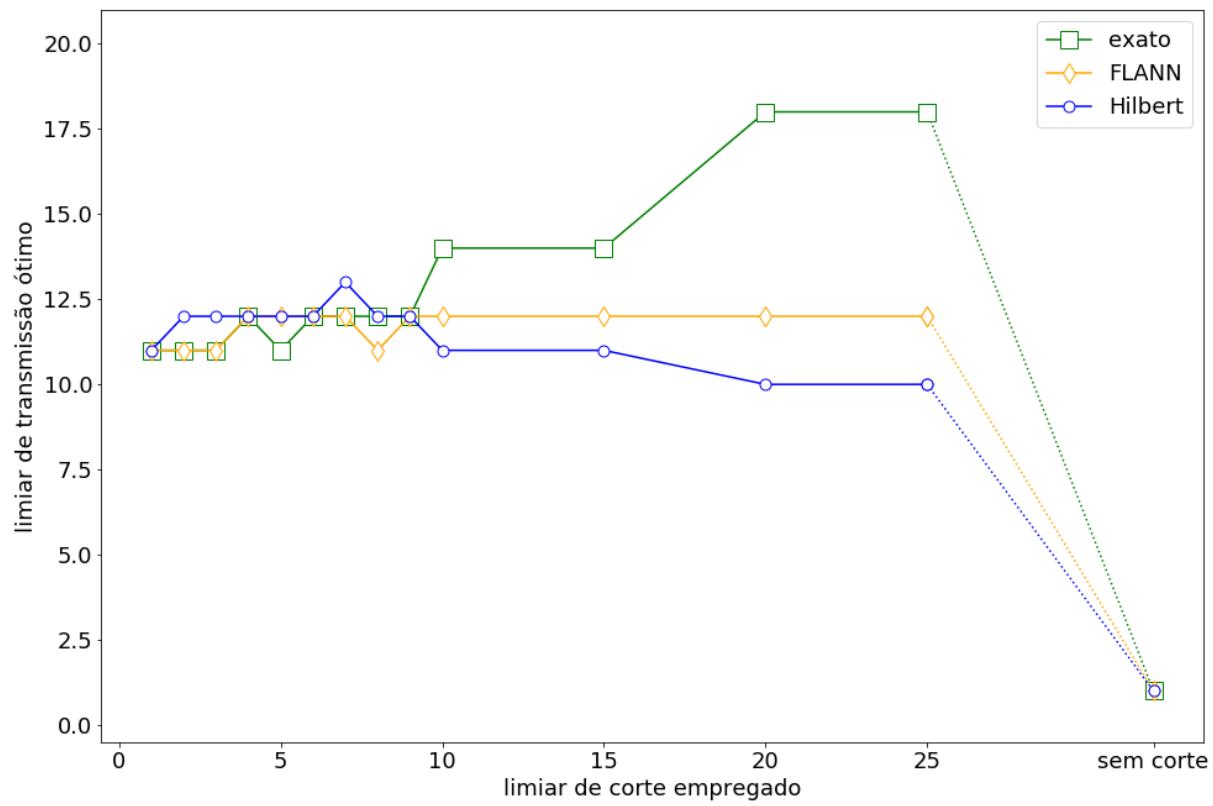
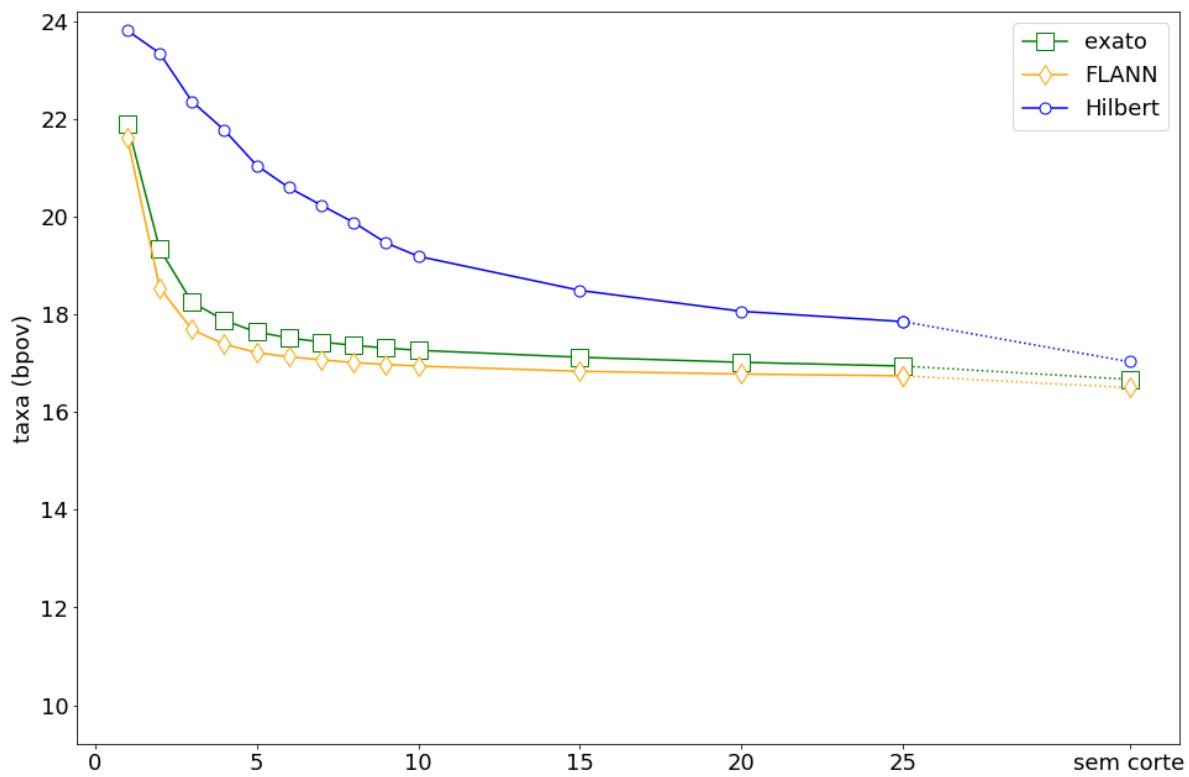


Figura 4.8: Resultados para o arquivo andrew9/frame0263

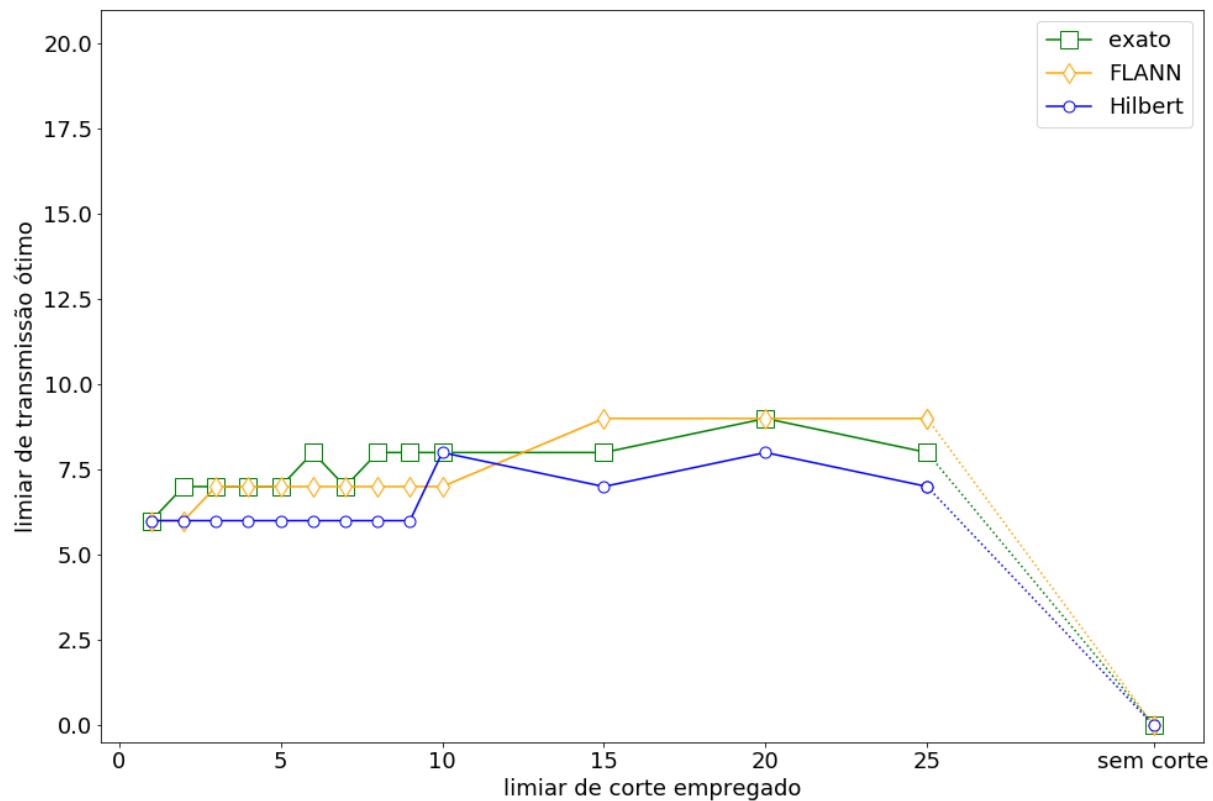
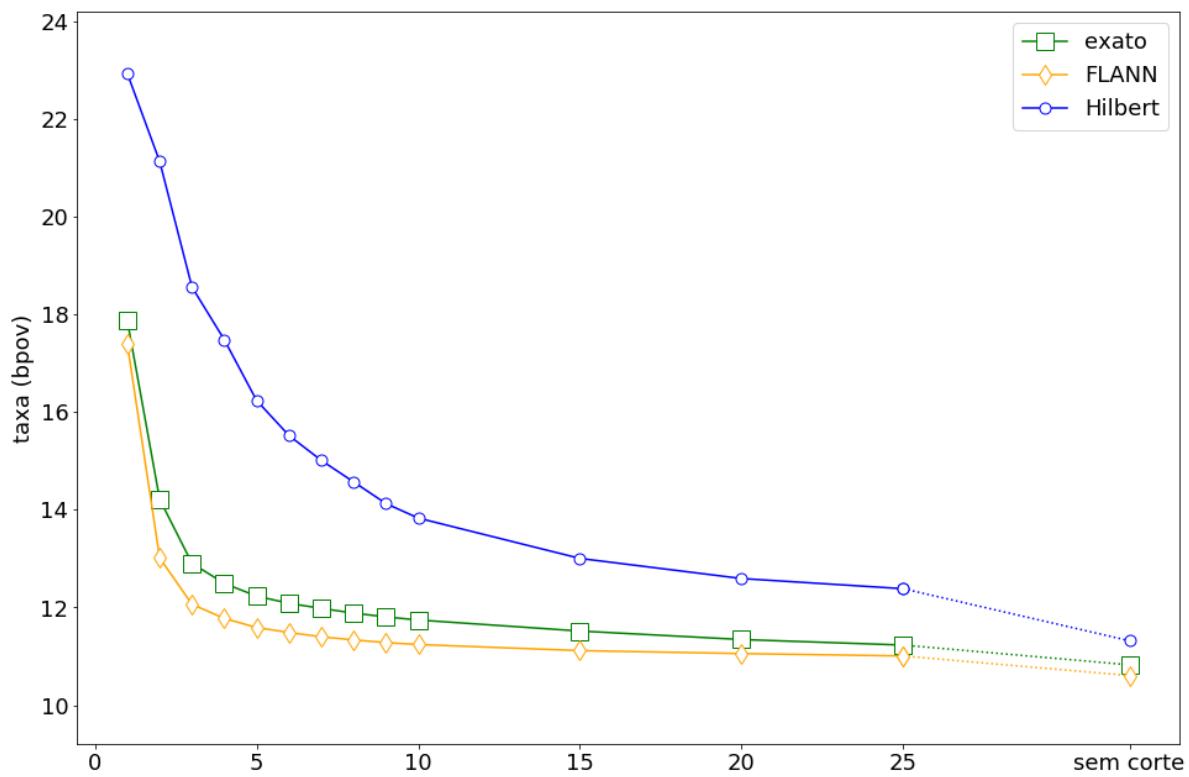


Figura 4.9: Resultados para o arquivo sarah9/frame0146