

INSTITUTO TECNOLÓGICO DE AERONÁUTICA



Peter Targino Klein

**AVALIAÇÃO DO FLUXO DE
PEDESTRES PARA EXTRAÇÃO
DE MÉTRICAS DE COMPORTAMENTO**

Trabalho de Graduação
2022

Curso de Engenharia Civil-Aeronáutica

Peter Targino Klein

**AVALIAÇÃO DO FLUXO DE
PEDESTRES PARA EXTRAÇÃO
DE MÉTRICAS DE COMPORTAMENTO**

Prof. Dr. Evandro José da Silva (ITA)

Orientador

Prof. Dr. Paulo Ivo Braga de Queiroz (ITA)

Coorientador

ENGENHARIA CIVIL-AERONÁUTICA

SÃO JOSÉ DOS CAMPOS, SP - BRASIL
INSTITUTO TECNOLÓGICO DE AERONÁUTICA

2022

Dados Internacionais de Catalogação-na-Publicação (CIP)

Divisão de Informação e Documentação

KLEIN, Peter Targino

Avaliação do fluxo de pedestres para extração de métricas de comportamento / Peter Targino Klein.
São José dos Campos, 2022.
48f.

Trabalho de Graduação – Engenharia Civil-Aeronáutica – Instituto Tecnológico de Aeronáutica,
2022. Orientador: Prof. Dr. Evandro José da Silva. Co-orientador: Prof. Dr. Paulo Ivo Braga de Queiroz.

1. Diagrama de Voronoi. 2. Triangulação de Delaunay. 3. Área de ocupação e influência. I. Instituto Tecnológico de Aeronáutica. II. Avaliação do fluxo de pedestres para extração de métricas de comportamento.

REFERÊNCIA BIBLIOGRÁFICA

KLEIN, Peter Targino. **Avaliação do fluxo de pedestres para extração de métricas de comportamento**. 2022. 48f. Trabalho de Conclusão de Curso. (Graduação em Engenharia Civil-Aeronáutica) – Instituto Tecnológico de Aeronáutica, São José dos Campos, 2022.

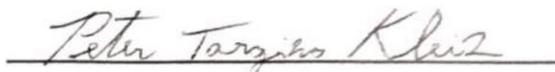
CESSÃO DE DIREITOS

NOME DO AUTOR: Peter Targino Klein

TÍTULO DO TRABALHO: Avaliação do fluxo de pedestres para extração de métricas de comportamento

TIPO DO TRABALHO/ANO: Graduação / 2022

É concedida ao Instituto Tecnológico de Aeronáutica permissão para reproduzir cópias deste trabalho de graduação e para emprestar ou vender cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte deste trabalho de graduação pode ser reproduzida sem a autorização do autor.



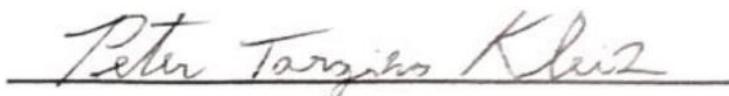
Peter Targino Klein

Rua Nelson Alves de Oliveira, 20

12225-180, São José dos Campos - SP

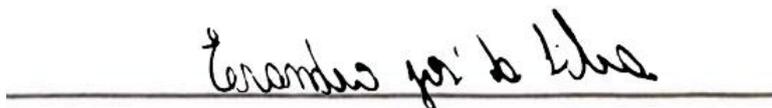
AVALIAÇÃO DO FLUXO DE PEDESTRES PARA EXTRAÇÃO DE MÉTRICAS DE COMPORTAMENTO

Essa publicação foi aceita como Relatório Final de Trabalho de Graduação



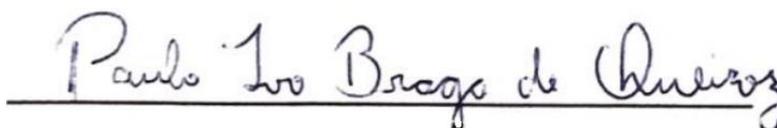
Peter Targino Klein

Autor



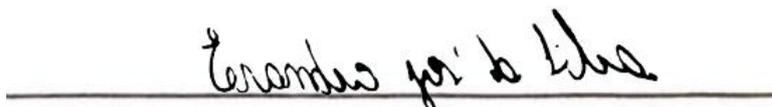
Prof. Dr. Evandro José da Silva (ITA)

Orientador



Prof. Dr. Paulo Ivo Braga de Queiroz (ITA)

Coorientador



Prof. Dr. Evandro José da Silva (ITA)

Coordenador do Curso de Engenharia Civil-Aeronáutica

São José dos Campos, 16 de novembro de 2022

Dedico este trabalho a todos os estudantes que almejam usar do conhecimento e da educação como transformadores do meio, colocando suas razões altruístas à frente de suas razões egoístas.

Agradecimentos

Primeiramente, quero agradecer a Deus, pelo dom da vida, e a todas as pessoas que dão significado à minha vida: Antonio Carlos, meu amado pai, Varenka, minha amada mãe, Victor, meu amado irmão, Antonia Antonieta (ou “Toeta”), que tanto cuidou de mim na infância e me incentivou como se filho dela eu fosse, Paulo Ivo (ou “Pi”), que foi o melhor professor conselheiro que eu poderia ter – principalmente nos momentos mais difíceis acadêmica e emocionalmente – e Natália, minha querida esposa e mãe dos meus futuros e numerosos filhos e filhas.

Agradeço também a todos os colegas que tive ao longo do FUND por aguentarem meu senso de humor bobo e me ajudarem no gagá das MATs, em especial o Arthur Covatti (ou “Pato Branco”), o Matheus Douglas de Matos (ou “Mato Grosso”) e o João Luís Freitas (ou “Provisório”).

Também agradeço aos professores que souberam me fazer ter mais gosto pelas disciplinas da grade curricular, em especial ao falecido Fábio Mokarzel (CES-11), que conseguiu fazer da sua disciplina algo que dava gosto de estudar ao mesmo tempo que evidenciava que eu não teria satisfação no curso de Engenharia da Computação (o qual eu cogitava fazer).

Claro, agradeço também aos colegas que tive ao longo das minhas três turmas no PROF, mas principalmente aos professores – com destaque para Evandro José, meu orientador, além do já citado professor Pi e dos demais professores, que se dedicaram à minha formação e foram pacientes com minhas pombices.

Por fim, agradeço a todos aqueles que irão ler este trabalho e usá-lo como base para estudos futuros – espero que não se frustrem demais com eventuais faltas de didática ou clareza de minha parte.

*"O louco não é o homem que perdeu a razão.
O louco é o homem que perdeu tudo, exceto a razão".*

Gilbert Keith Chesterton

Resumo

Com o desenvolvimento e a expansão do setor de serviço aeroportuário de transporte de passageiros, houve também o aumento do grau de exigências relacionadas a conforto e segurança. Uma destas exigências, impulsionada pela pandemia da COVID-19, foi em relação ao distanciamento físico entre determinado passageiro e seus vizinhos e à área efetivamente ocupada por este ao longo de sua travessia por dado trecho do Terminal de Passageiros. O presente trabalho utiliza conceitos e técnicas do campo da geometria computacional, em particular os Diagramas de Voronoi e as Triangulações de Delaunay, para avaliar a área de ocupação e influência de cada pessoa a cada instante de tempo durante suas passagens por diferentes cenários. Para tal, foi utilizada a Base de Dados de Fluxo de Pedestres disponibilizada pelo Centro de Pesquisa de Jülich (Alemanha) em seu sítio-web e a implementação dos Diagramas de Voronoi e das Triangulações de Delaunay correspondentes com três algoritmos distintos. É feita uma comparação entre os algoritmos selecionados em termos de eficiência computacional na Linguagem de Programação C++, a qual foi escolhida pela sua notória rapidez e possibilidade de aplicação em tempo real.

Abstract

With the development and the expansion of the airport passenger transportation service sector, also came the rise of the degree of demands related to ease and safeness. One of which, prompted up by the COVID-19 pandemic, was that of physical distancing between a given passenger and their neighbours and the actively occupied area by them along their passages across a given section of the passenger terminal. This current work makes use of concepts and techniques from the field of study of computational geometry, particularly that of Voronoi Diagrams and of Delaunay Triangulations, to evaluate the area of occupancy and influence of each person at each time instant during their crossings over different scenarios. For that end, it was made use of the Pedestrian Dynamics Data Archive database that is currently made available by the Research Center of Jülich (Germany) on their website and it was also made use of the implementation of the corresponding Voronoi Diagrams and Delaunay Triangulations with three distinct algorithms. A comparison is made between the selected algorithms in terms of computing efficiency in the C++ Programming Language, selected by its noteworthy speed and real-time application possibilities.

Sumário

1	INTRODUÇÃO	12
1.1	Objetivo geral	13
1.2	Objetivos específicos	13
1.3	Definição da abordagem e suas limitações	14
2	FUNDAMENTAÇÃO TEÓRICA.....	15
2.1	Diagramas de Voronoi.....	15
2.1.1	Definição de Diagrama de Voronoi.....	15
2.1.2	Triangulação e propriedades importantes.....	16
2.1.3	Definição de Triangulação de Delaunay	18
2.1.4	A dualidade entre Triangulação de Delaunay e Diagrama de Voronoi.....	18
2.1.5	Representação visual da métrica área.....	19
2.2	Algoritmos	20
2.2.1	Algoritmos DV-TD e TD-DV	20
2.2.2	Algoritmo de Varredura de Fortune – Diagrama de Voronoi	21
2.2.3	Algoritmo Dividir para Conquistar – Triangulação de Delaunay	21
2.2.4	Algoritmo Iterativo – Triangulação de Delaunay.....	22
2.3	Estrutura de Dados	23
3	BASE DE DADOS E CENÁRIOS DISTINTOS.....	26
3.1	Densidade e contrafluxo	27
3.2	Configurações dos corredores	28
3.3	Acerca dos experimentos.....	30

4	METODOLOGIA.....	32
4.1	Tratamento dos dados	32
4.2	Execução e comparação dos algoritmos.....	32
4.3	Particularidade na implementação quanto às configurações dos corredores.....	32
4.4	Dados de entrada e dados de saída	34
5	ANÁLISE DOS RESULTADOS	38
5.1	A métrica – área de ocupação e influência	38
5.2	Algoritmo e experimento – comparação entre tempos de execução	39
5.2.1	Algoritmo de Varredura de Fortune	40
5.2.2	Algoritmo Dividir para Conquistar.....	41
5.2.3	Algoritmo Iterativo	42
5.3	Comparação entre espaço de armazenamento da estrutura de dados	43
6	CONCLUSÃO E COMENTÁRIOS FINAIS.....	44
	REFERÊNCIAS	47

1 Introdução

O setor de transporte aéreo, enquanto serviço e, portanto, integrante do setor terciário da economia, só deve ser capaz de se sustentar mantendo algum grau de competitividade em relação às formas de transporte alternativas – e só deve ser capaz de se expandir caso o valor agregado do serviço se torne interessante ou vantajoso para uma parcela suficientemente grande do público que busca atender. Conforme apresentado em [1], viagens aéreas a negócios, até 2020, eram um fator importantíssimo para a difusão de conhecimento científico e práticas comerciais, crescendo a taxas superiores ao PIB global e movimentando 1,5 trilhão de dólares anuais (1,7% do PIB global).

Com a pandemia da COVID-19 e suas consequentes restrições sobre o setor de transporte aeroportuário, era de se esperar que fosse haver uma certa demora para que o tráfego aéreo como um todo retornasse aos padrões imediatamente anteriores às restrições quando estas fossem se abrandando – a realidade, porém, é de uma mudança mais abrupta: não apenas houve um retorno muito mais veloz que o imaginado inicialmente quanto ao crescimento do volume de tráfego aéreo, houve também uma tendência de etapas mais curtas, com maior número de etapas por passageiro. A Figura 1 a seguir evidencia a rápida retomada das atividades do setor de transporte aeroportuário.

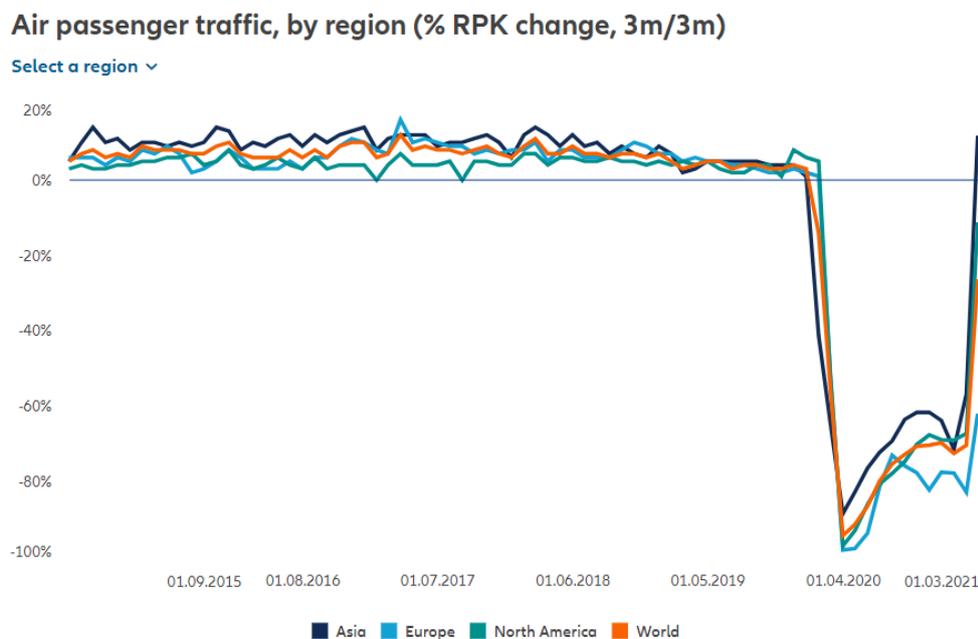


Figura 1: Variação trimestral do tráfego aéreo em RPK (*Revenue Passenger Kilometers*), com ênfase para a queda abrupta no primeiro trimestre de 2020 e a recuperação do setor no primeiro trimestre de 2021. **Fonte:** IATA, Allianz, Euler Hermes. 2022.

Com esta mudança repentina de cenário, certos problemas novos deverão surgir em um horizonte de tempo próximo – um dos quais é o de que os terminais aeroportuários irão corresponder a um percentual consideravelmente maior de tempo da experiência do passageiro médio, além de que irão operar sendo sujeitos a uma capacidade tal que, sob a ótica da indivisibilidade de oferta – princípio que, conforme descrito em [2], ocasionará sobrecargas mais frequentes sobre a infraestrutura aeroportuária atual, projetada segundo paradigmas antigos.

Desta forma, a fim de atender aos novos padrões de demanda dos passageiros, que deverão dar maior importância à qualidade do serviço prestado em um TPS (Terminal de Passageiros) onde a densidade de pessoas por área tenderá a crescer bastante rápido em corredores críticos nas horas-pico, é conveniente compreender bem a dinâmica do movimento de uma multidão e as implicações que um espaço cada vez mais restrito terá sobre isso.

1.1 Objetivo geral

O presente Trabalho de Graduação tem como objetivo avaliar a distribuição espacial de passageiros ao longo do tempo considerando diferentes cenários de fluxo em ambiente fechado. Para tanto, serão construídos Diagramas de Voronoi e Triangulações de Delaunay referentes a cada instante de tempo nos diferentes cenários e, sobre estes, serão feitas análises a fim de determinar em que situações há melhor performance de extração/visualização de dados do fluxo de pedestres.

1.2 Objetivos específicos

1. Discorrer sobre Diagramas de Voronoi e Triangulações de Delaunay, bem como suas aplicações no contexto da dinâmica de pedestres;
2. Explanar alguns algoritmos consolidados na literatura para a confecção dos Diagramas de Voronoi e Triangulações de Delaunay;
3. Implementar cada um dos algoritmos explanados em uma mesma linguagem, sendo escolhida a Linguagem de Programação C++;
4. Confeccionar os Diagramas de Voronoi e Triangulações de Delaunay em distintos cenários coletados de uma mesma base de dados, sendo escolhida a Base de Dados de Fluxo de Pedestres disponibilizada pelo Centro de Pesquisa de Jülich (Alemanha);

5. Comparar as performances dos algoritmos para cada cenário para determinar o algoritmo ideal para cada situação, identificando se algum deles pode ser considerado como o de melhor performance global.

1.3 Definição da abordagem e suas limitações

Dada a importância do tema e dada a extensão de possibilidades relacionadas às combinações de variáveis, fonte dos dados, linguagens de programação para implementação dos algoritmos, entre outros. É necessário definir até onde este Trabalho de Graduação deve adentrar o assunto para poder ao menos servir de fundamento para comparações de performance mais complexas em estudos futuros, mas não adentrar tanto que não seja possível cumprir com o prazo de conclusão e inviabilizar revisões minuciosas do texto deste Trabalho. Dito isto, a abordagem do presente Trabalho de Graduação foi restrita a:

- Cinco configurações de corredor (que podem ser vistas detalhadamente na seção **3.2 Configurações dos corredores**), variando-se as densidades e os sentidos de fluxo:
 1. Corredor em I (retilíneo);
 2. Corredor em L (quina ou *corner*);
 3. Corredor partido em T;
 4. Corredor partido em Y e
 5. Cruzamento em X.
- Três algoritmos para a geração dos Diagramas de Voronoi e Triangulações de Delaunay (que podem ser vistos detalhadamente na seção **2.2 Algoritmos**):
 1. Algoritmo de Varredura de Fortune;
 2. Algoritmo Dividir para Conquistar e
 3. Algoritmo Iterativo.

Conforme apontado em [3], há uma variedade muito grande de problemas e de algoritmos – o que complica ainda mais a questão comparativa quando se deseja a sobreposição de problemas para a extração de múltiplas métricas simultâneas. Para este Trabalho de Graduação, foi escolhida a métrica mais simples de ser extraída e visualizada: a área de ocupação e influência dos pedestres, representados como pontos no espaço \mathbb{R}^2 , ao longo de suas respectivas passagens.

2 Fundamentação Teórica

2.1 Diagramas de Voronoi

De acordo com [4], os Diagramas de Voronoi (também chamados de Tesselação de Dirichlet) são uma estrutura de dados geométricos muito bem estabelecidos atualmente devido à sua capacidade de representação de dados multidimensionais para problemas, por exemplo, de classificação ou clusterização, sendo o assunto de cerca de um em cada dezesseis artigos na Geometria Computacional desde sua introdução a este campo da ciência.

Conforme já mencionado no final da seção anterior, há uma variedade imensa de problemas e estratégias para solucioná-los. Limitando a abordagem ao espaço \mathbb{R}^2 (espaço de geometria euclidiana de duas dimensões), fica possível simplificar o problema proposto neste Trabalho de Graduação para a determinação da métrica de área de ocupação e influência.

2.1.1 Definição de Diagrama de Voronoi

Seja S um conjunto de pontos de uma região $\Omega \subseteq \mathbb{R}^2$. Ω é dividido em subconjuntos tal que, para cada ponto $p \in S$, está associado a si uma região V_p em que a distância de um ponto qualquer dentro desta região está mais próximo de p do que de qualquer outro ponto do conjunto S . Ou seja:

$$V_p = \{x \in \mathbb{R}^2 \mid \|x - p\| \leq \|x - q\|, \forall q \in S\} \quad (\text{eq. 1})$$

No caso de passageiros em um corredor, pode-se imaginar:

- Cada passageiro como um dos pontos de S ;
- $\partial\Omega$ (bordo da região Ω) corresponde às paredes ou outro obstáculo intransponível e
- O polígono que envolve cada ponto de S (como o que envolve o ponto p , o V_p descrito na **eq. 1**), também chamado de Célula de Voronoi, tem como área a própria métrica da área de ocupação e influência do pedestre.

A Figura 2 a seguir ilustra um Diagrama de Voronoi nestes conformes.

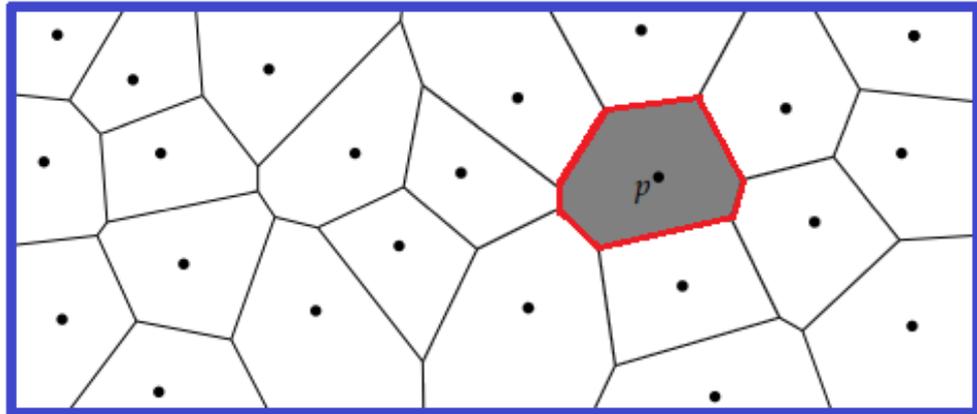


Figura 2: Exemplo de Diagrama de Voronoi, no qual se destaca o ponto p e sua célula. O polígono azul representa $\partial\Omega$ enquanto o polígono vermelho representa ∂V_p e as linhas pretas representam os bordos das demais células associadas aos pontos pertencentes a S . **Adaptado** de [5].

2.1.2 Triangulação e propriedades importantes

A triangulação de um conjunto de pontos S contidos em uma região $\Omega \subseteq \mathbb{R}^2$ consiste, segundo [6], em uma subdivisão de Ω determinada por um conjunto máximo de arestas que não se cruzam duas a duas e cujos vértices são os pontos de S .

Conforme evidenciado na definição, um mesmo conjunto de pontos pode passar por um procedimento de triangulação de vários modos distintos. A Figura 3 a seguir mostra diferentes triangulações de um mesmo conjunto de pontos, no caso, foi tomado o mesmo conjunto de pontos S da Figura 2:

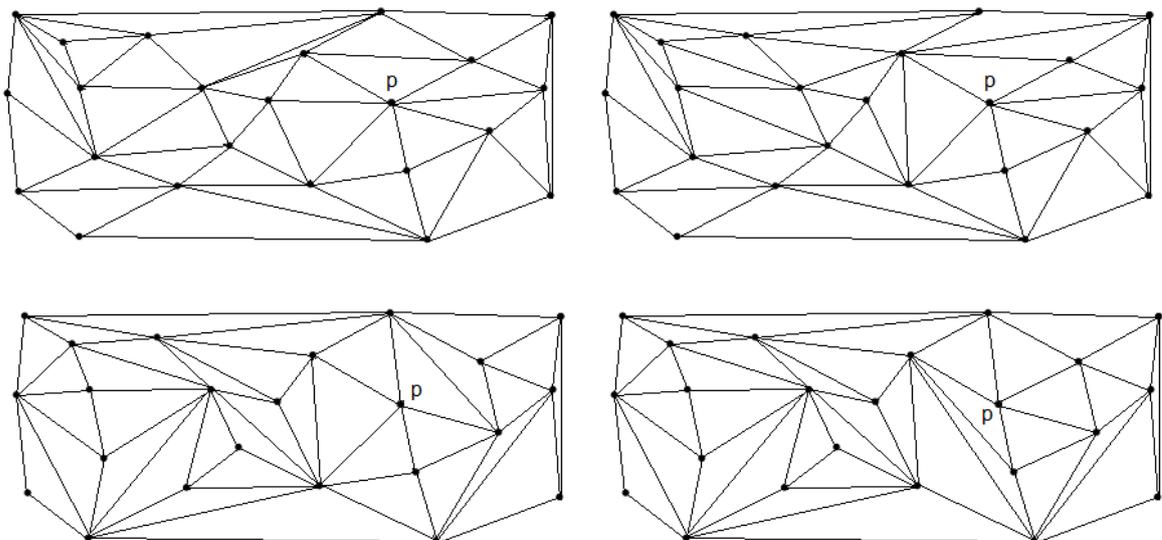


Figura 3: Exemplo de 4 triangulações possíveis para o mesmo conjunto de pontos S . **Fonte:** Próprio autor.

É perceptível que, a fim de atender ao requisito de conter um máximo de arestas que se cruzam, a triangulação do conjunto de pontos necessariamente conterá um determinado conjunto de arestas externas chamado de envoltória convexa, definido em [7] como o menor conjunto convexo que contém S , podendo ser encarado como o polígono que delimita o que é considerado “o lado de dentro” e o que é considerado “o lado de fora”.

Ainda segundo [6], a envoltória convexa conduz a uma propriedade importante: se os n pontos de S não forem todos colineares entre si, com h pontos pertencentes à envoltória convexa e $k = n - h$ pontos pertencentes ao interior da envoltória convexa, então qualquer que seja a triangulação feita sobre S :

$$2*k + h - 2 = T \quad (\text{eq. 2})$$

$$2*k + 2*h - 3 = A \quad (\text{eq. 3})$$

Onde:

- T representa o número de triângulos e
- A representa o número de arestas.

Assim, retomando o exemplo do conjunto de pontos S utilizados nas Figuras 2 e 3, a envoltória convexa pode ser expressa na Figura 4:

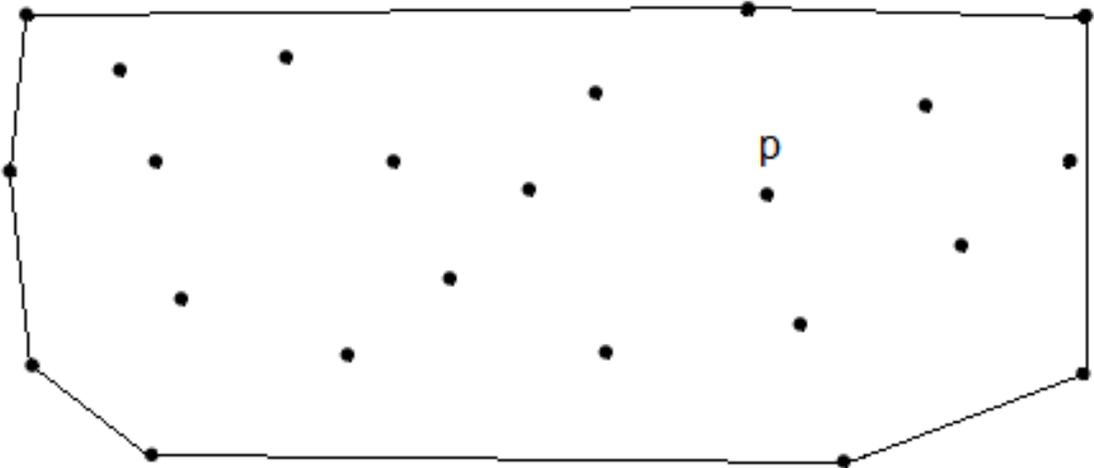


Figura 4: Envoltória convexa do conjunto de pontos S . **Fonte:** Próprio autor.

Como S não apresenta apenas pontos colineares neste exemplo, e como $h = 8$, $k = 15$, então necessariamente o número de triângulos e de arestas são, qualquer seja a triangulação feita:

- $T = 2*15 + 8 - 2 = 36$ e
- $A = 2*15 + 2*8 - 3 = 43$

Com relação ao número triangulações distintas que podem ser feitas a partir de um mesmo conjunto de pontos S , não é conhecida uma equação que determine o número de possibilidades para todo e qualquer S , de modo que é extremamente trabalhoso analisar várias combinações para avaliar qual delas serviria melhor a um dado propósito. Assim sendo, determinadas metodologias surgem para obter uma triangulação com alguma propriedade específica desejável, como a é a Triangulação de Delaunay.

2.1.3 Definição de Triangulação de Delaunay

A triangulação de um conjunto de pontos S não necessariamente segue uma regra para “priorizar” uma configuração em detrimento de outra, porém algumas triangulações fazem surgir padrões com mais utilidade prática do que outros. A Triangulação de Delaunay possui diversas aplicações, sendo a mais notória a modelagem de terrenos em 3D (conforme destacado em [5], em [6] e em [10]).

Existem várias propriedades equivalentes nesta triangulação, de modo que se pode definir a Triangulação de Delaunay de várias formas. A definição mais simples é aquela apresentada em [4]:

Seja S um conjunto de pontos de uma região $\Omega \subseteq \mathbb{R}^2$ e sejam os pontos $p, q, r \in S$. A triangulação feita é dita Triangulação de Delaunay caso, para todo trio de pontos $p, q, r \in S$, estes só sejam vértices de um mesmo triângulo se a circunferência definida por este trio de pontos não contenha outros pontos do conjunto S no seu interior.

2.1.4 A dualidade entre Triangulação de Delaunay e Diagrama de Voronoi

O Diagrama de Voronoi e a Triangulação de Delaunay são, como descrito por [3], basicamente equivalentes, haja vista que, ao se obter um deles, pode-se obter o outro com um algoritmo de complexidade $O(n)$.

O Diagrama de Voronoi apresentado na Figura 2, por exemplo, possui uma Triangulação de Delaunay associada a si que se obtém ligando os pontos do conjunto S cujas Células de Voronoi possuem arestas comuns, destacando que, a fim de concluir a envoltória convexa, as arestas das Células de Voronoi que tocam $\partial\Omega$ devem ser prolongadas a fim de identificar as arestas comuns que só são visíveis em $\mathbb{R}^2 - \Omega$. Na Figura 5 a seguir, é ilustrada a Triangulação de Delaunay descrita:

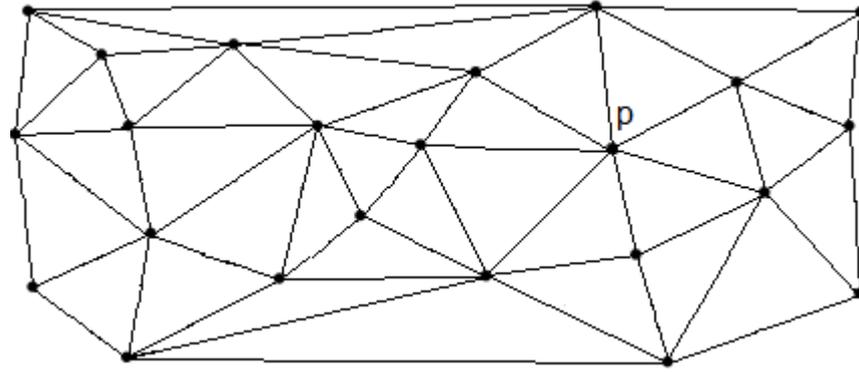


Figura 5: Triangulação de Delaunay associada ao Diagrama de Voronoi da Figura 2. **Fonte:** Próprio autor.

O algoritmo de complexidade $O(n)$ para a obtenção do Diagrama de Voronoi a partir da Triangulação de Delaunay é apresentado na seção **2.2. Algoritmos** mais adiante, bem como o algoritmo para o procedimento contrário de obtenção da Triangulação de Delaunay a partir do Diagrama de Voronoi.

2.1.5 Representação visual da métrica área

Geralmente, estudos clássicos como [11] dão tratamento ao fluxo de pedestres como uma análise em que é caracterizado o conjunto de pedestres como algo analiticamente indivisível e, sobre este conjunto, é realizado algum trabalho estatístico para caracterizar as variações entre os componentes do conjunto.

É claro e evidente que esta abordagem não é equivocada, mas não é a mais adequada passado meio século de crescimento exponencial de capacidade de processamento computacional, facilidade de coleta e capacidade de armazenamento de dados. O futuro de estudos deste ramo seguramente necessitará de dados mais específicos para cada elemento “passageiro” do conjunto “fluxo”.

Ou seja, ao passo que se costumava obter medidas e classificações como um avaliativo apenas na forma de média das características individuais dos elementos (tais como o nível de serviço, uma classificação única para todo o conjunto apesar de sua heterogeneidade), torna-se possível mensurar cada passageiro individualmente com nova metodologia (e, além da visão geral do conjunto fornecida pelas métricas tradicionais, dispõe-se da possibilidade da divisibilidade de dados).

Sob esta óptica, uma análise considerando dois tipos de medida de área de ocupação e influência surgem fazendo-se uso dos Diagramas de Voronoi e das Triangulações de Delaunay: uma delas caracterizada pela própria área da Célula de Voronoi associada a cada

passageiro-elemento e a outra caracterizada como um somatório de áreas dos Triângulos de Delaunay que dado passageiro-elemento compõe como vértice. Essas duas formas de se obter a métrica da área de ocupação e influência dos passageiros individualmente são apenas proposições – seguramente há outras formas de fazer e, a princípio, nenhuma representa a realidade melhor do que a outra. Tal comparação entre as formas de obtenção da métrica será retomada na seção **5.1 A métrica – área de ocupação e influência**, estando mais evidente até lá as diferenças entre as medidas e suas hipóteses.

2.2 Algoritmos

2.2.1 Algoritmos DV-TD e TD-DV

O algoritmo para a obtenção do Diagrama de Voronoi a partir da Triangulação de Delaunay é:

- 1) Para cada Triângulo de Delaunay, localize o seu circuncentro (ponto de encontro das mediatrizes de suas arestas);
- 2) Para cada circuncentro localizado, trace um segmento de reta unindo-o aos circuncentros dos Triângulos de Delaunay adjacentes (ou seja, os triângulos que possuem uma aresta em comum);
- 3) Os circuncentros dos Triângulos de Delaunay encontrados no passo 1 são os vértices das Células de Voronoi, ao passo que os segmentos de reta encontrados no passo 2 são as arestas das Células de Voronoi. O Diagrama de Voronoi está, então, montado.

O algoritmo para a obtenção da Triangulação de Delaunay a partir do Diagrama de Voronoi é:

- 1) Para cada Célula de Voronoi, trace um segmento de reta unindo o seu ponto interior pertencente ao conjunto S com os pontos de S contidos nas Células de Voronoi adjacentes (ou seja, as células que possuem uma aresta em comum);
- 2) Os segmentos de reta encontrados no passo 1 são as arestas dos Triângulos de Delaunay, ao passo que os pontos do conjunto S são os próprios vértices dos triângulos e que já estavam determinados antes da execução do passo 1. A Triangulação de Delaunay está, então, montada.

Estes algoritmos, porém, dependem da obtenção prévia da Triangulação de Delaunay ou do Diagrama de Voronoi, sendo incapazes de produzi-los apenas com o conjunto de pontos S na região $\Omega \subseteq \mathbb{R}^2$. Para tal, serão utilizados os 3 algoritmos descritos nas subseções a seguir:

2.2.2 Algoritmo de Varredura de Fortune – Diagrama de Voronoi

Este algoritmo, obtido de [9], para a obtenção do Diagrama de Voronoi a partir do conjunto de pontos S é:

- 1) É tomada uma Linha de Varredura e uma direção a qual esta linha irá varrer (por exemplo, direção norte-sul);
- 2) Ao longo da passagem da linha sobre o plano que contém os pontos de S , são feitas parábolas caracterizadas por terem a Linha de Varredura como reta diretiz e por ter um ponto de S que já foi varrido pela linha como foco (por exemplo, se a linha varre na direção norte-sul, então as parábolas vão sendo formadas primeiramente nos pontos ao norte do plano);
- 3) Quando três parábolas (associadas aos pontos a, b, c do conjunto S) se cruzam em um mesmo ponto α não pertencente a S , naquele ponto está localizado um vértice de uma Célula de Voronoi, cujas arestas são formadas ligando α aos pontos onde as parábolas associadas a dois dos pontos a, b, c do conjunto S se cruzam com a parábola associada a outro ponto do conjunto S .

2.2.3 Algoritmo Dividir para Conquistar – Triangulação de Delaunay

Este algoritmo, obtido de [10], para a obtenção da Triangulação de Delaunay a partir do conjunto de pontos S é:

- 1) Sobre o conjunto de pontos S , é traçada sua envoltória convexa. As arestas da envoltória convexa são todas arestas de Triângulos de Delaunay;
- 2) Identifica-se qual vértice da envoltória convexa está mais ao norte no plano para então identificar suas arestas, as quais são chamadas de tangente superior comum;
- 3) Identifica-se qual vértice da envoltória convexa está mais ao sul no plano para então identificar suas arestas, as quais são chamadas de tangente inferior comum;
- 4) Com uma tangente superior comum identificada, é identificado o ponto de S não pertencente à envoltória convexa que, quando ligado aos vértices dessa tangente superior comum, forma o triângulo cujo círculo circunscrito não contém outros pontos de S . Isso é feito para ambas as arestas identificadas no passo 2;

- 5) Com uma tangente inferior comum identificada, é identificado o ponto de S não pertencente à envoltória convexa que, quando ligado aos vértices da tangente inferior comum, forma o triângulo cujo círculo circunscrito não contém outros pontos de S . Isso é feito para ambas as arestas identificadas no passo 3;
- 6) Os dois triângulos formados no passo 4 são Triângulos de Delaunay, para cada uma das arestas formadas no passo 4, é identificado o ponto de S não pertencente à envoltória convexa que, quando ligado aos vértices dessa aresta, forma o triângulo cujo círculo circunscrito não contém outros pontos de S ;
- 7) Os dois triângulos formados no passo 5 são Triângulos de Delaunay, para cada uma das arestas formadas no passo 5, é identificado o ponto de S não pertencente à envoltória convexa que, quando ligado aos vértices dessa aresta, forma o triângulo cujo círculo circunscrito não contém outros pontos de S ;
- 8) Repete-se os passos 6 e 7 até que o número de arestas formadas se iguale ao valor determinado na **eq. 3** da subseção **2.1.2 Triangulação e propriedades importantes**.

2.2.4 Algoritmo Iterativo – Triangulação de Delaunay

Este algoritmo, obtido de [8], para a obtenção da Triangulação de Delaunay a partir do conjunto de pontos S é:

- 1) Cria-se um Triângulo Muito Grande (TMG) que contenha todos os n pontos do conjunto S em seu interior;
- 2) Adiciona-se o primeiro ponto do conjunto S e realiza-se a única triangulação possível envolvendo este ponto do conjunto S e os vértices do TMG;
- 3) Adiciona-se o próximo ponto do conjunto S e verifica-se dentro de qual(is) círculo(s) circunscrito(s) da atual triangulação ele está contido. Ele pode estar contido dentro de, no mínimo, apenas um ou, no máximo, dois círculos circunscritos, e isso define qual dos passos 4 ou 5 será cumprido;
- 4) Caso o ponto esteja contido dentro de um único círculo circunscrito, então efetua-se a única triangulação possível envolvendo este ponto do conjunto S e as três arestas do triângulo que o continha quando o ponto foi adicionado no passo 3;
- 5) Caso o ponto esteja contido dentro de dois círculos circunscritos, a aresta em comum dos triângulos associados a esses círculos circunscritos é apagada e o

ponto adicionado no passo 3 é ligado a cada um dos vértices do TMG e/ou pontos do conjunto S que são vértices do quadrilátero resultante da aresta apagada;

- 6) Repete-se os passos 3 a 5 até que todos os n pontos tenham sido incluídos;
- 7) O TMG tem seus três vértices apagados, bem como todas as arestas que se ligavam a estes, sobra apenas os n pontos do conjunto S com as correspondentes arestas dos Triângulos de Delaunay.

2.3 Estrutura de Dados

É mais simples para um ser humano visualizar o Diagrama de Voronoi ou a Triangulação de Delaunay desenhada ou “plotada” numa tela do que ler a descrição geométrica de cada ponto, aresta ou polígono. Para um computador, porém, é mais eficiente armazenar apenas as descrições geométricas e desenhar ou “plotar” na tela apenas para a visualização de um usuário caso este solicite, de modo que é necessário compreender que tipo de dados se deseja armazenar para cada situação para o uso eficiente da memória do computador.

Para armazenar uma Triangulação de Delaunay, é necessário:

- As coordenadas dos pontos do conjunto S , que coincidem com os vértices dos Triângulos de Delaunay;
- Os três vértices que compõem cada um dos Triângulos de Delaunay, sendo cada aresta caracterizada pelos dois vértices os quais ela une.

Para armazenar um Diagrama de Voronoi, é necessário:

- As coordenadas dos pontos do conjunto S , que são os núcleos das Células de Voronoi;
- Os vértices das Células de Voronoi;
- As arestas das Células de Voronoi, sendo cada aresta caracterizada pelos dois vértices os quais ela une.

Entretanto, todo o propósito de obtenção das Triangulações de Delaunay e dos Diagramas de Voronoi para o problema proposto remete à determinação da área de ocupação e influência de cada ponto do conjunto S e do seu passageiro correspondente, de modo que é

de grande valia armazenar também o valor desta métrica associado a cada ponto de S conforme as proposições postas nas Tabelas 1 e 2 a seguir.

Tabela 1: Estrutura de Dados necessários para o armazenamento da Triangulação de Delaunay associada a S e da métrica área de ocupação e influência de cada ponto de S .

Informação Necessária	Forma de Armazenamento
Ponto p do conjunto S	Coordenadas no \mathbb{R}^2 do ponto; Indexação do ponto; Índices de cada um dos Triângulos de Delaunay que possuem p como vértice; Área de ocupação e influência do ponto, calculada como sendo a terça parte do somatório das áreas de cada Triângulo de Delaunay que possui o ponto p como um de seus vértices.
Triângulos de Delaunay	Índices de cada um dos três pontos de S correspondentes aos vértices de cada Triângulo de Delaunay; Área do triângulo, calculada a partir das coordenadas no \mathbb{R}^2 dos seus três vértices; Indexação de cada Triângulo de Delaunay.

Destaca-se que a Triangulação de Delaunay é consideravelmente mais eficiente em termos de armazenamento devido à simplicidade da divisão geométrica do plano em triângulos (uma vez que conhecer seus três vértices é suficiente para determiná-lo), sendo esta preferível em relação à divisão em polígonos convexos com mais de três lados no Diagrama de Voronoi (que necessita do armazenamento das arestas para garantir que as células serão convexas).

Tabela 2: Estrutura de Dados necessários para o armazenamento do Diagrama de Voronoi associado a S e da métrica área de ocupação e influência de cada ponto de S .

Informação Necessária	Forma de Armazenamento
Vértices das Células de Voronoi	<p>Coordenadas no \mathbb{R}^2 de cada vértice;</p> <p>Indexação de cada vértice.</p>
Arestas das Células de Voronoi	<p>Índices dos dois vértices que compõem a aresta;</p> <p>Indexação de cada aresta.</p>
Ponto p do conjunto S e sua Célula de Voronoi associada	<p>Coordenadas no \mathbb{R}^2 do ponto p;</p> <p>Índices das arestas que compõem a célula associada ao ponto p;</p> <p>Área da célula, calculada a partir das coordenadas no \mathbb{R}^2 dos vértices. Esta também é a área de ocupação e influência do ponto p;</p> <p>Indexação do conjunto ponto p + Célula de Voronoi associada a p.</p>

3 Base de dados e cenários distintos

A base de dados considerada, disponível para acesso em <https://ped.fz-juelich.de/da/doku.php>, conta com os resultados de diversos experimentos controlados realizados pelo Centro de Pesquisa de Jülich com o intuito de analisar a dinâmica, o comportamento de uma multidão em certas circunstâncias específicas impostas.

As variações entre os experimentos considerados para este Trabalho de Graduação envolveram, principalmente:

- O número de voluntários em cada passagem;
- As distâncias entre as paredes (larguras dos trechos retilíneos);
- As configurações (os formatos) dos corredores;
- De qual extremidade do corredor os voluntários entram no experimento e para qual extremidade estes se dirigem e
- Se há ou não há outros voluntários vindo na direção contrária.

Os dados para cada experimento são fornecidos por meio das variáveis apresentadas na Tabela 3, obtidos com detecção automática pela câmera dos organizadores dos experimentos com os devidos ajustes – manuais ou computados por transformações geométricas – para diminuir erros de medida ocasionados por efeito paralaxe:

Tabela 3: Formatação dos dados fornecidos pelo Centro de Pesquisa de Jülich.

# PersID	Frame	X	Y	Z	Rot
Número de identificação do voluntário	Em qual quadro da filmagem foi registrada a posição do voluntário	Coordenada X do voluntário	Coordenada Y do voluntário	Coordenada Z do voluntário	Direção para onde o rosto do voluntário está apontando

As duas últimas colunas, entretanto, não são de serventia para o tipo de análise proposta:

- A variável **Z**, que na prática consiste simplesmente na estatura do voluntário, é descartada e
- A variável **Rot**, que não faz sentido na representação do voluntário como um ponto no \mathbb{R}^2 , é descartada.

3.1 Densidade e contrafluxo

A densidade do corredor, conforme previamente abordado na subseção **2.1.5 Representação visual da métrica área**, em estudos clássicos desconsidera a heterogeneidade da distribuição dos passageiros. Na fase de tratamento de dados, seção do próximo capítulo, é possível contar a quantidade de passageiros no corredor em dado momento, inclusive subdividindo-o a gosto, e determinar a média do espaço ocupado no corredor inteiro ou em uma de suas partes para determinar o nível de serviço, como realizado em [12] e evidenciado na Figura 6 e nas Tabelas 4 e 5 a seguir:

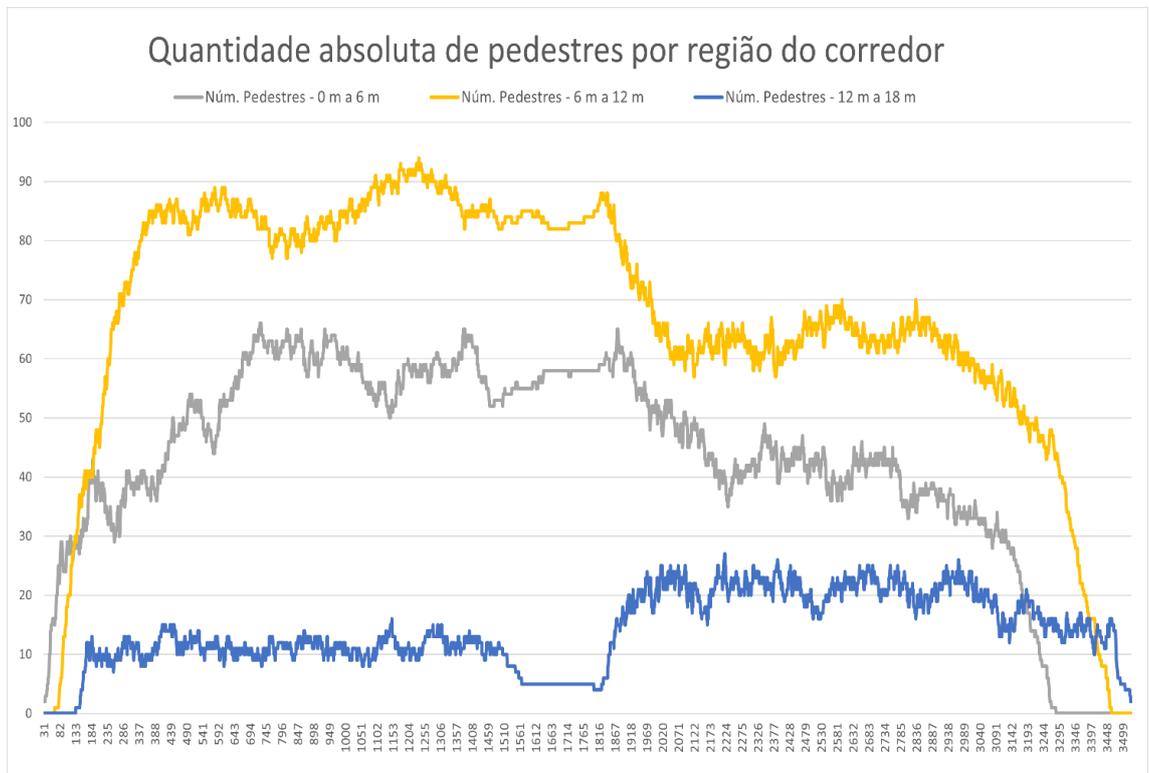


Figura 6: Exemplo de divisão de corredor retangular de 5 metros de largura por 18 metros de comprimento em três regiões para análise em separado de cada trecho. **Fonte:** [12].

Tabela 4: Estatísticas do fluxo representado na Figura 6. **Adaptado** de [12].

Número total de pedestres na travessia	Média de velocidade dos pedestres neste cenário (m/s)	Número máximo de pedestres simultâneos	Área por pedestre no pico do fluxo (m ²)	Nível de serviço no pico do fluxo
917	0,609	164	0,549	E

Tabela 5: Nível de serviço conforme descrito por [11] convertido para unidades do Sistema Internacional. **Fonte:** [12].

Nível de serviço	Área por pedestre
A	$S_p \geq 3,256 \text{ m}^2$
B	$3,256 \text{ m}^2 > S_p \geq 2,326 \text{ m}^2$
C	$2,326 \text{ m}^2 > S_p \geq 1,395 \text{ m}^2$
D	$1,395 \text{ m}^2 > S_p \geq 0,930 \text{ m}^2$
E	$0,930 \text{ m}^2 > S_p \geq 0,465 \text{ m}^2$
F	$0,465 \text{ m}^2 > S_p$

Com relação à presença de contrafluxo, convém definir o sentido de fluxo: diz que o fluxo é unidirecional quando todos os seus elementos possuem a mesma origem e o mesmo destino. A condição em que a relação Origem-Destino não é necessariamente a mesma entre dois elementos de um dado fluxo caracteriza o chamado contrafluxo.

3.2 Configurações dos corredores

Conforme dito previamente na seção **1.3 Definição da abordagem e suas limitações**, foram trabalhadas cinco configurações de corredor, variando-se as densidades e os sentidos de fluxo, as quais são representadas nas Figuras 7 a 11 a seguir:



Figura 7: Exemplo de Corredor em I (retilíneo). **Fonte:** C. P. de Jülich.



Figura 8: Exemplo de Corredor em L (quina ou *corner*). **Fonte:** C. P. de Jülich.



Figura 9: Exemplo de Corredor partido em T. **Fonte:** C. P. de Jülich.



Figura 10: Exemplo de Corredor partido em Y. **Fonte:** C. P. de Jülich.



Figura 11: Exemplo de Cruzamento em X. **Fonte:** C. P. de Jülich.

3.3 Acerca dos experimentos

A Base de Dados de Fluxo de Pedestres disponibilizada pelo Centro de Pesquisa de Jülich conta com diversos outros experimentos, alguns envolvendo filas ou gargalos e outros que classificam os passageiros com base em algum conjunto de características. A Figura 12 a seguir ilustra bem esse tipo de cenário com um exemplo contendo cinco categorias de passageiros:



Figura 12: Exemplo de experimento envolvendo a contabilização de características individuais dos passageiros. **Fonte:** C. P. de Jülich.

O presente Trabalho de Graduação desconsidera a existência de diferenças significativas entre os passageiros, no que se refere às suas características individuais, e é tomado o pressuposto de que a única diferença de comportamento entre dois passageiros advém da disponibilidade de espaço nas suas imediações.

Conforme explicitado na seção **1.3 Definição da abordagem e suas limitações**, os passageiros são, para os propósitos deste Trabalho de Graduação, pontos no espaço \mathbb{R}^2 e deseja-se avaliar a melhor forma de extrair e representar os dados referentes à área de ocupação e influência.

4 Metodologia

4.1 Tratamento dos dados

Os dados obtidos do Centro de Pesquisa de Jülich forneceram anomalias (coordenada fora do escopo do experimento ou inexistente) na proporção aproximada de 1 em cada 100 mil linhas, de modo que o tratamento da base de dados consistiu, essencialmente, em organizá-los em uma ordem mais conveniente de leitura: As linhas de dados foram ordenadas conforme a ordem crescente dos quadros (*Frames*), a fim de representar a passagem do tempo, e as coordenadas dos voluntários presentes em cada quadro foram armazenadas em arquivos separados para cada quadro de cada experimento.

Com esta separação de dados, em arquivos identificados por experimento e por quadro do experimento, foi dado início à execução dos algoritmos em C++.

4.2 Execução e comparação dos algoritmos

Normalmente, ao se comparar a performance de algoritmos, o principal fator de análise está em sua complexidade, que costuma ser representada na notação *Big O*, não necessariamente mensurando o tempo absoluto de execução, apenas a tendência de crescimento do tempo de execução de modo relativo.

Para este Trabalho de Graduação, os algoritmos são executados uma vez para cada experimento e é medida a rapidez com que o algoritmo é executado, bem como o espaço de armazenamento requerido dos arquivos de saída contendo a Estrutura de Dados para cada quadro das gravações dos experimentos.

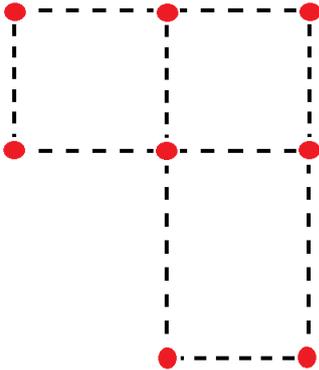
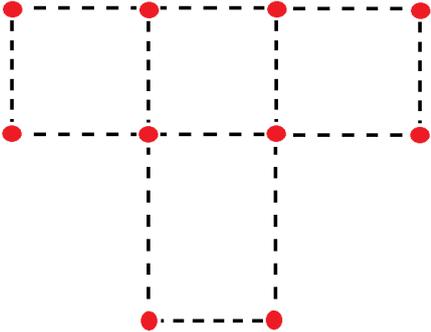
4.3 Particularidade na implementação quanto às configurações dos corredores

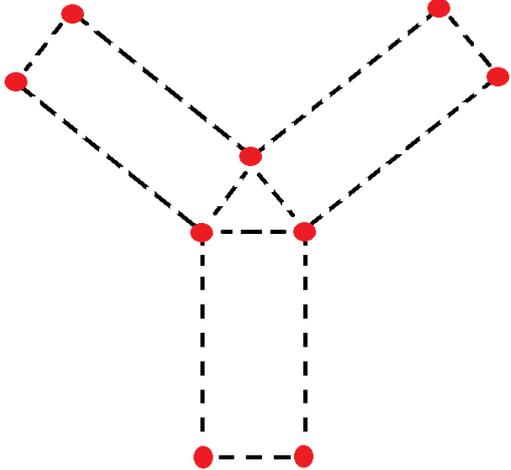
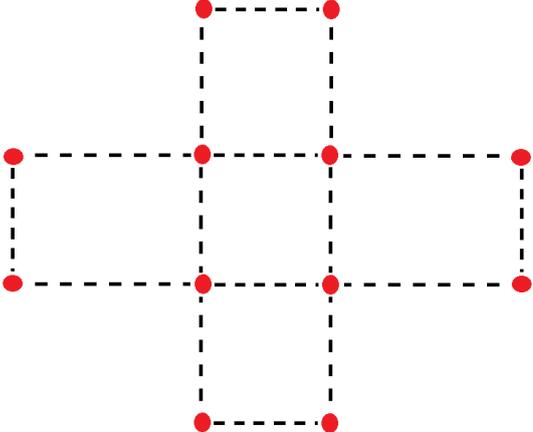
Na construção dos Diagramas de Voronoi e das Triangulações de Delaunay, foi observado que se fazia necessário subdividir os corredores em regiões convexas (como no exemplo mostrado na Figura 6 na seção **3.1 Densidade e contrafluxo**) para que não fossem

construídas arestas de Triângulos de Delaunay ou de Células de Voronoi que ultrapassassem as paredes dos corredores – intransponíveis, pela condição de contorno do problema.

A fim de assim fazê-lo, cada configuração de corredor é então dividida em uma quantidade mínima de regiões convexas – explicitadas na Tabela 6 a seguir – e em cada uma das regiões convexas é feita uma Triangulação de Delaunay ou um Diagrama de Voronoi inteiramente contidos nesta região, considerando os pontos que delimitam cada região como sendo os pontos que constituem a envoltória convexa do subconjunto de passageiros naquela região.

Tabela 6: Divisão das diferentes configurações de corredor em regiões convexas. **Fonte:** Próprio autor.

Configuração de corredor	Número mínimo de regiões convexas	Número de pontos para divisão em um mínimo de regiões convexas	Divisão em regiões convexas
I	1	4	
L	3	8	
T	4	10	

Configuração de corredor	Número mínimo de regiões convexas	Número de pontos para divisão em um mínimo de regiões convexas	Divisão em regiões convexas
Y	4	9	 <p>The diagram shows a Y-shaped corridor formed by dashed lines. It has 9 red dots representing points. The points are located at the four outer vertices of the Y, the two vertices where the arms meet the stem, and the two vertices of the stem's base. Dashed lines connect these points to divide the corridor into four convex regions: two triangles at the top and two rectangles at the bottom.</p>
X	5	12	 <p>The diagram shows a cross-shaped corridor formed by dashed lines. It has 12 red dots representing points. The points are located at the four corners of the horizontal bar, the four corners of the vertical bar, and the four points where the bars intersect. Dashed lines connect these points to divide the corridor into five convex regions: two rectangles on the left and right, and two rectangles at the top and bottom.</p>

4.4 Dados de entrada e dados de saída

A Tabela 3 do Capítulo 3 **Base de dados e cenários distintos** evidencia a forma como os dados são obtidos e na seção **4.1 Tratamento dos dados** é descrita a forma como estes são preparados para a execução dos algoritmos. O ato de importação e tratamento de dados é feito completamente pelo *software* gerenciador de banco de dados *Microsoft Access*, onde é realizado o procedimento visível nas Figuras 13 e 14 adiante, que ilustram um exemplo de dados de entrada antes e após o tratamento, respectivamente, lidos pelo *software*.

```

# <number> <frame> <x> [in m] <y> [in m] <z> [in m] <rot> [in rad] <id> <flag>
#
# The numbers are consecutive, for bookkeeping.
# The IDs are the IDs of each marker head.
# The flags mark the source of each data point.
#
# Flags
# - Detection
#   1 : detection
# - Tracking
#   11 : optical flow while tracking
#   12 : linear interpolation while tracking
#   13 : linear extrapolation while tracking
# - Transformation
#   21 : interpolation while transforming
# - Stitching
#   31 : interpolation while stitching
# - Transformation
#   41 : interpolation while transforming
# - Editing
#   51 : changed manually
#   52 : extrapolation, linear or optical flow
1   924   -4.098000   6.328000   1.750000   1.574000   1999   52
1   925   -4.082000   6.333000   1.750000   1.574000   1999   52
1   926   -4.076000   6.336000   1.750000   1.574000   1999   52
1   927   -4.068000   6.336000   1.750000   1.574000   1999   52
1   928   -4.061000   6.334000   1.750000   1.574000   1999   52
1   929   -4.047000   6.324000   1.750000   1.574000   1999   52
1   930   -4.040000   6.317000   1.750000   1.574000   1999   52

```

Figura 13: Exemplo de arquivo de entrada antes do tratamento de dados. Este arquivo em específico possui dados para 3501 frames de um fluxo em um corredor partido em Y. **Fonte:** C. P. de Jülich.

Linha	Person ID	Frame	X	Y
200	83	2022	-0.703000	1.260000
201	80	2022	1.571000	1.741000
202	91	2022	-1.576000	-2.984000
203	97	2022	-0.561000	-1.292000
204	82	2022	-2.325000	2.120000
205	100	2022	-1.111000	0.808000
206	84	2022	3.771000	-0.786000
207	94	2022	0.622000	-0.313000
208	99	2022	-1.338000	-2.646000
209	18	2022	-0.260000	2.983000
210	22	2022	-1.194000	4.047000
211	19	2022	-2.374000	5.234000
212	2	2022	-1.154000	2.251000
213	3	2022	-0.325000	-0.961000
214	10	2022	4.085000	-0.577000
215	36	2022	-2.145000	-3.440000
216	38	2022	0.590000	2.735000
217	37	2022	4.663000	0.825000
218	43	2022	-2.778000	3.034000
219	50	2022	4.574000	1.556000
220	39	2022	-1.143000	-0.989000
221	31	2022	0.629000	3.125000
222	28	2022	4.397000	0.072000

Figura 14: Exemplo de arquivo de entrada após o tratamento de dados. Estes dados dizem respeito ao mesmo arquivo ilustrado na Figura 13, onde são exibidos alguns dos 227 pontos correspondentes aos voluntários visíveis no quadro de número 2022. **Fonte:** Próprio autor.

Os arquivos de saída consistem no que foi descrito nas Tabelas 1 e 2 na seção **2.3 Estrutura de Dados**, de modo que se obtém o Diagrama de Voronoi ou a Triangulação de Delaunay descrita matematicamente, sem representação visual como é o caso da Figura 2 e da Figura 5. As Figuras 15 e 16 a seguir ilustram dois exemplos de Estrutura de Dados, um na forma de Diagrama de Voronoi e o outro na forma de Triangulação de Delaunay, respectivamente:

Índice do vértice	X do vértice	Y do vértice	Índice da aresta	Índice do vértice 1	Índice do vértice 2
16	-2.876	0.020	94	73	90
17	0.617	-2.233	95	63	66
18	-1.303	-0.183	96	70	72
19	1.039	-0.096	97	63	60
20	-0.894	-1.383	98	66	98
21	2.270	-2.019	99	79	99
22	-0.566	-0.015	100	90	72
23	0.114	-1.311	101	63	93
24	3.600	2.597	102	66	70
25	-1.477	1.950	103	61	89
26	-2.904	0.174	104	99	84
27	0.734	-0.492	105	96	90

Índice de p	X de p	Y de p	Área de p	Núm. Arestas	Índice Aresta 1	Índice Aresta 2	Índice Aresta 3	Índice Aresta 4	Índice Aresta 5	Índice Aresta 6	Índice Aresta 7
142	0.032	1.752	1.257	4	123	59	163	218	-	-	-
143	0.340	-0.346	1.927	7	176	156	220	212	233	89	191
144	-0.349	-0.255	0.510	6	86	98	141	62	25	188	-
145	-0.369	0.035	1.732	5	129	195	165	174	128	-	-
146	-0.017	-0.845	0.963	5	23	126	184	140	166	-	-
147	-0.358	-2.560	1.221	4	137	37	205	231	-	-	-
148	1.336	0.022	1.492	6	226	68	92	176	168	220	-
149	0.715	-0.029	1.873	6	210	115	176	219	11	133	-
150	-0.755	-0.264	2.904	5	186	129	48	239	150	-	-
151	0.030	-0.013	0.280	5	95	177	90	138	24	-	-
152	0.540	-0.041	2.708	4	134	231	221	44	-	-	-
153	-1.389	-1.186	0.723	4	15	37	164	166	-	-	-
154	1.297	-0.241	0.862	5	128	200	14	42	57	-	-
155	1.955	-0.642	2.751	4	229	136	228	92	-	-	-
156	1.609	0.274	2.914	4	34	60	119	196	-	-	-
157	0.425	2.911	3.012	6	21	26	133	147	51	200	-
158	-0.253	0.991	1.456	7	67	60	175	53	98	216	181
159	1.136	-1.436	0.861	6	238	192	103	219	132	130	-

Figura 15: Exemplo de Estrutura de Dados descrevendo matematicamente um Diagrama de Voronoi referente aos dados tratados ilustrados na Figura 14, onde são exibidos alguns vértices de célula à esquerda no topo, algumas arestas de célula à direita no topo e as células em si na parte de baixo. **Fonte:** Próprio autor.

Índice do Triângulo	Índice vértice 1	Índice vértice 2	Índice vértice 3	Área
57	154	177	203	1.625
58	95	114	140	0.634
59	127	142	169	0.517
60	156	158	191	1.650
61	55	78	100	1.749
62	74	94	117	1.807
63	136	139	160	0.874
64	123	186	206	1.722
65	140	167	193	1.796
66	185	200	210	1.236
67	135	205	211	1.213
68	135	148	163	0.546
69	168	205	208	0.980

Índice de p	X de p	Y de p	Área de p	Núm. Triângulos	Índice triângulo 1	Índice triângulo 2	Índice triângulo 3	Índice triângulo 4	Índice triângulo 5	Índice triângulo 6	Índice triângulo 7
142	0.032	1.752	1.188	4	59	77	102	126	-	-	-
143	0.340	-0.346	1.904	7	89	156	186	196	198	204	210
144	-0.349	-0.255	0.455	6	46	62	88	92	132	188	-
145	-0.369	0.035	1.668	5	128	129	165	173	174	-	-
146	-0.017	-0.845	0.928	5	23	126	184	190	196	-	-
147	-0.358	-2.560	1.197	4	137	138	205	207	-	-	-
148	1.336	0.022	1.358	6	68	74	92	149	168	172	-
149	0.715	-0.029	1.649	6	110	115	176	179	191	193	-
150	-0.755	-0.264	2.606	5	86	129	138	139	150	-	-
151	0.030	-0.013	0.258	5	95	100	110	118	124	-	-
152	0.540	-0.041	2.626	4	134	135	155	156	-	-	-
153	-1.389	-1.186	0.619	4	76	77	80	81	-	-	-
154	1.297	-0.241	0.771	5	57	73	91	110	-	-	-
155	1.955	-0.642	2.737	4	92	136	168	169	-	-	-
156	1.609	0.274	2.747	4	34	60	119	120	-	-	-
157	0.425	2.911	2.569	6	21	26	51	54	72	78	-
158	-0.253	0.991	1.440	7	53	60	67	98	101	116	128
159	1.136	-1.436	0.815	6	103	130	132	144	149	173	-

Figura 16: Exemplo de Estrutura de Dados descrevendo matematicamente uma Triangulação de Delaunay referente aos dados tratados ilustrados na Figura 14, onde são exibidos alguns Triângulos de Delaunay no topo e os pontos p do conjunto S na parte de baixo. **Fonte:** Próprio autor.

As Figuras 15 e 16 são imagens obtidas de trechos de arquivos exportados do *Microsoft Access* para o *Microsoft Excel*, *software* onde as Estruturas de dados podem ser armazenadas em separado por quadro de cada experimento para facilitar a leitura dos dados em arquivos menores – haja vista que o arquivo do *Microsoft Access* contendo toda a base de dados de cada experimento pode ser difícil de manusear por conter a base toda.

5 Análise dos resultados

5.1 A métrica – área de ocupação e influência

Na subseção **2.1.5 Representação visual da métrica área**, foi discutido que a representação desta métrica poderia ser feita de duas maneiras e que, a princípio, as duas seriam válidas.

O que se observou foi que ambas as medidas de área de ocupação e influência, tanto pela área da Célula de Voronoi do ponto quanto pela terça parte do somatório de áreas dos Triângulos de Delaunay, apontaram para uma distribuição bastante desigual de área de ocupação e influência entre os pontos.

Os valores de área de ocupação e influência dos pontos foram expressos com três casas decimais de precisão e a distribuição variou conforme a configuração dos corredores. Tal distribuição foi estimada fazendo a divisão em quartis das áreas de ocupação e influência em dez quadros de cada experimento e tirando a média de cada estatística, conforme expresso na Tabela 7 adiante. A média (calculada como a área total do corredor dividida pelo número de voluntários presentes de cada quadro em cada experimento) é parametrizada em 100 % a fim de facilitar as comparações entre os quartis em cada configuração de corredor, enquanto o *software* utilizado para tal foi o *Microsoft Excel*, que não é tão adequado para este propósito quanto *softwares* tais como o *RStudio*, mas que é fácil de manusear e oferece estimativas suficientemente boas dado o espaço amostral propositalmente pequeno que se tomou.

Tabela 7: Distribuição aproximada das áreas de ocupação e influência em relação à média para as configurações de corredor testadas. A seleção de outros quadros para fazer a estimativa percentual pode conduzir a resultados consideravelmente diferentes. **Fonte:** Próprio autor.

Configuração de corredor	Média	1º quartil	2º quartil	3º quartil
I	100 %	73 %	106 %	180 %
L	100 %	60 %	96 %	177 %
T	100 %	53 %	100 %	143 %
Y	100 %	51 %	88 %	150 %
X	100 %	47 %	99 %	139 %

Além desta distribuição não-uniforme das áreas de ocupação e influência, foi observado que, frequentemente, os valores da métrica apontados pelo Diagrama de Voronoi eram ligeiramente diferentes dos valores da mesma métrica apontados pela Triangulação de Delaunay referentes ao mesmo voluntário no mesmo frame de um experimento. Costumeiramente, os valores da métrica no Diagrama de Voronoi eram cerca de 10 % maiores para voluntários próximos ao centro do corredor, e eram até 25 % menores para voluntários próximos aos pontos que delimitavam pontos de entrada ou saída da região de análise do experimento.

Fica evidente, desta forma, que a obtenção da métrica área de ocupação e influência ponto a ponto pode se dar de qualquer modo que se considere mais conveniente, seja pela obtenção do Diagrama de Voronoi, seja pela obtenção da Triangulação de Delaunay, seja pela média simples utilizada para obtenção do nível de serviço no pico do fluxo como mostrado na Tabela 4. Porém, com uma distinção acentuada entre os resultados dos métodos, convém uma análise mais aprofundada em estudos futuros sobre o significado prático e a utilidade de cada resultado.

5.2 Algoritmo e experimento – comparação entre tempos de execução

Conforme proposto na seção **1.1 Objetivo geral**, deseja-se saber qual combinação de algoritmo e cenário é a mais rápida computacionalmente. Para tal, foi tomada a média de tempo de execução dos algoritmos em dado quadro com uma quantidade de pontos pré-determinada e foi construído um gráfico para cada configuração de corredor com cada algoritmo, apresentados nas subseções seguintes.

As quantidades de quadros com o correspondente número de pontos por quadro foram as explicitadas na Tabela 8 a seguir, válida para todos os algoritmos nas subseções seguintes:

Tabela 8: Quantidades de pontos em cada quadro para as configurações de corredor testadas. A seleção de outras quantidades de pontos para fazer a estimativa gráfica pode conduzir a resultados ligeiramente diferentes.

Configuração de corredor	10 pontos por quadro	35 pontos por quadro	75 pontos por quadro	120 pontos por quadro	180 pontos por quadro
I / L	60 quadros	45 quadros	35 quadros	25 quadros	10 quadros
T / Y	40 quadros	30 quadros	25 quadros	20 quadros	10 quadros
X	20 quadros	40 quadros	50 quadros	30 quadros	10 quadros

Os tempos de execução dos algoritmos são obtidos com o uso da biblioteca `std::chrono` para C++ para serem, então, manualmente inseridos em um arquivo `.xlsx` (referentes ao *software Microsoft Excel*) onde são tabelados (e representados neste Trabalho pelas Tabelas 9, 10 e 11) e comparados graficamente nas Figuras 17, 18 e 19, respectivamente, das subseções seguintes.

5.2.1 Algoritmo de Varredura de Fortune

Tirando a média de tempo de execução conforme quantidades indicadas na Tabela 8, o Algoritmo de Varredura de Fortune para a construção de Diagramas de Voronoi resultou no que está indicado na Figura 17 e na Tabela 9:

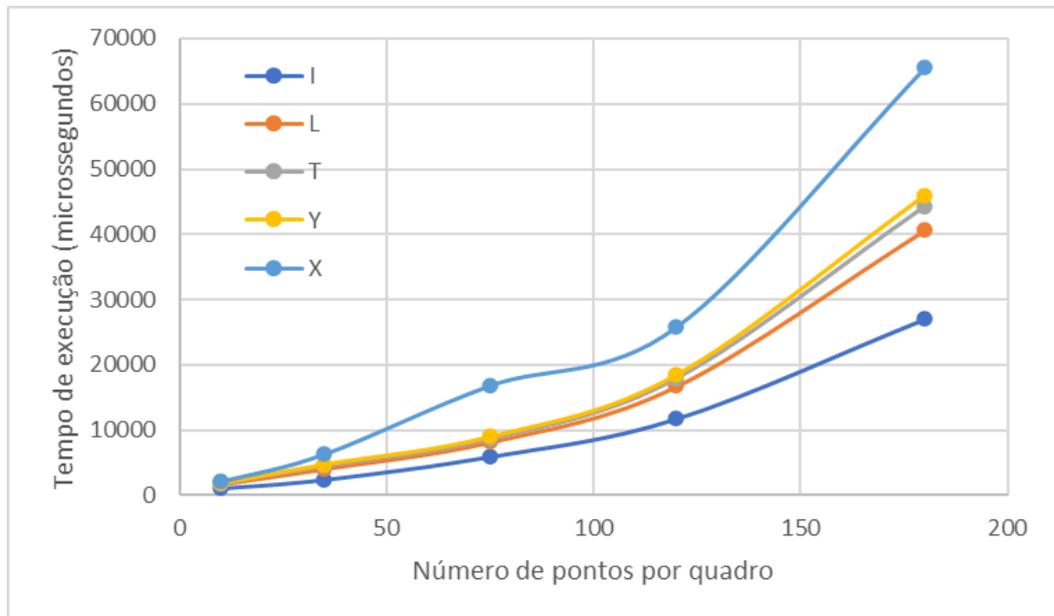


Figura 17: Tempos de execução do Algoritmo de Varredura de Fortune, conforme configuração do corredor e número de pontos em C++. **Fonte:** Próprio autor.

Tabela 9: Pontos presentes na Figura 17.

	I (μs)	L (μs)	T (μs)	Y (μs)	X (μs)
10 pontos	1074	1742	1837	1904	2092
35 pontos	2429	4106	4490	4782	6328
75 pontos	5933	8194	8622	9030	16848
120 pontos	11752	16749	17935	18509	25824
180 pontos	27040	40683	44286	46025	65582

5.2.2 Algoritmo Dividir para Conquistar

Tirando a média de tempo de execução conforme quantidades indicadas na Tabela 8, o Algoritmo Dividir para Conquistar para a construção de Triangulações de Delaunay resultou no que está indicado na Figura 18 e na Tabela 10:

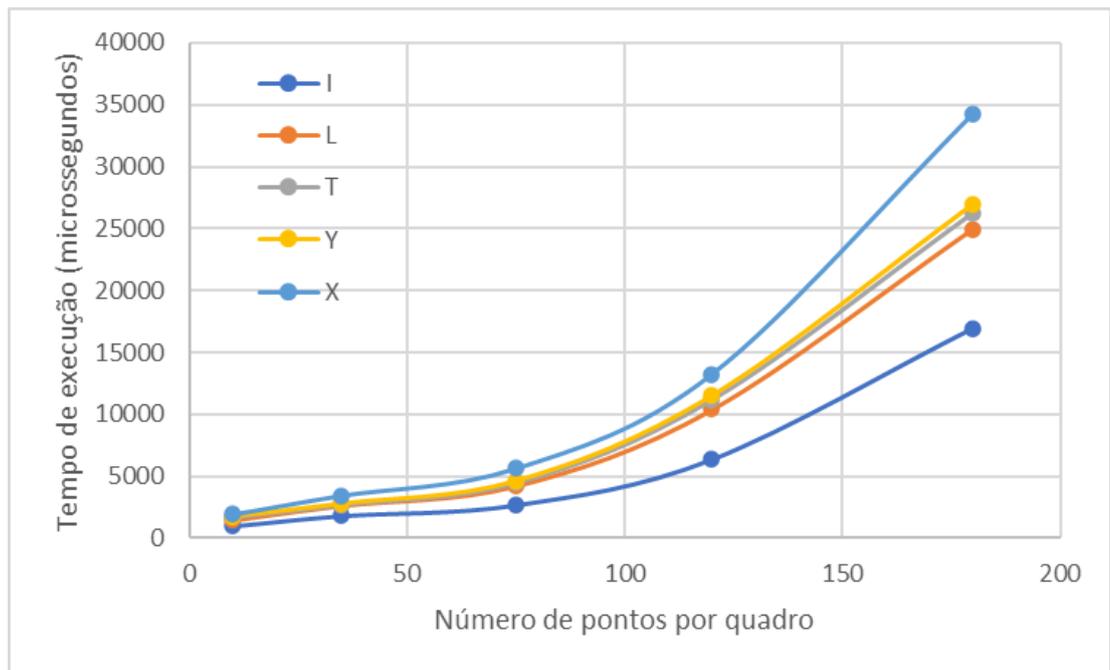


Figura 18: Tempos de execução do Algoritmo Dividir para Conquistar, conforme configuração do corredor e número de pontos em C++. **Fonte:** Próprio autor.

Tabela 10: Pontos presentes na Figura 18.

	I (μs)	L (μs)	T (μs)	Y (μs)	X (μs)
10 pontos	997	1492	1717	1754	1932
35 pontos	1821	2660	2723	2782	3450
75 pontos	2689	4246	4502	4611	5649
120 pontos	6379	10402	11207	11509	13266
180 pontos	16960	24973	26281	27005	34306

5.2.3 Algoritmo Iterativo

Tirando a média de tempo de execução conforme quantidades indicadas na Tabela 8, o Algoritmo Iterativo para a construção de Triangulações de Delaunay resultou no que está indicado na Figura 19 e na Tabela 11:

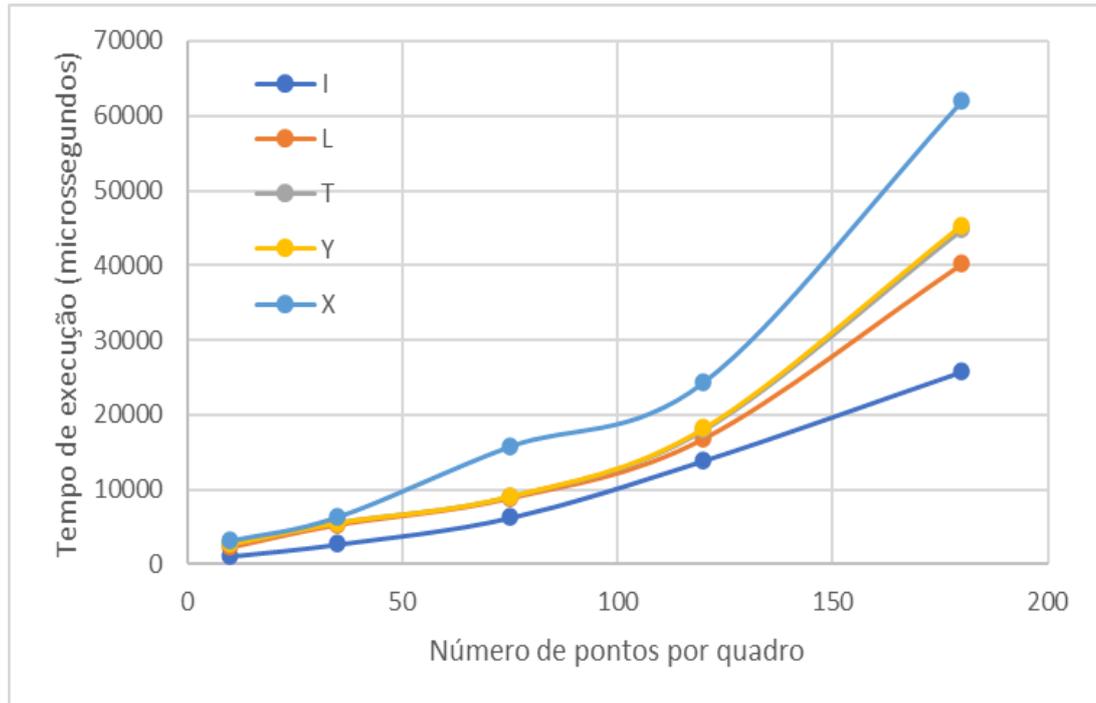


Figura 19: Tempos de execução do Algoritmo Iterativo, conforme configuração do corredor e número de pontos em C++. **Fonte:** Próprio autor.

Tabela 11: Pontos presentes na Figura 19.

	I (μs)	L (μs)	T (μs)	Y (μs)	X (μs)
10 pontos	1139	2378	2802	2732	3193
35 pontos	2742	5381	5683	5572	6407
75 pontos	6276	8935	9124	9089	15822
120 pontos	13893	16891	18030	18201	24400
180 pontos	25738	40226	44792	45382	61962

5.3 Comparação entre espaço de armazenamento da estrutura de dados

Conforme já discutido na seção 2.3 **Estrutura de Dados**, a Triangulação de Delaunay requer menor espaço de armazenamento. A fim de mensurar qual o ganho de espaço de armazenamento da Triangulação de Delaunay em relação ao Diagrama de Voronoi em dado quadro com mesma quantidade de pontos, foi tomado a média de tamanho dos arquivos de saída com quantidades fixas de pontos nos conforme do que foi feito ao longo da seção 5.2 **Algoritmo e experimento – comparação entre tempos de execução**.

As Tabelas 12 e 13 a seguir mostram o tamanho médio de um arquivo .xlsx (referentes ao *software Microsoft Excel*) contendo as estruturas de dados para Diagramas de Voronoi (como na Figura 15) e para Triangulações de Delaunay (como na Figura 16) respectivamente:

Tabela 12: Armazenamento médio requerido na estrutura de dados em Diagrama de Voronoi para as configurações de corredor testadas. A seleção de outros arquivos de saída pode conduzir a resultados ligeiramente diferentes.

	I (KB)	L (KB)	T (KB)	Y (KB)	X (KB)
10 pontos	90	91	91	92	92
35 pontos	173	175	176	177	179
75 pontos	289	296	298	299	300
120 pontos	361	362	362	363	363
180 pontos	520	522	522	521	522

Tabela 13: Armazenamento médio requerido na estrutura de dados em Triangulação de Delaunay para as configurações de corredor testadas. A seleção de outros arquivos de saída pode conduzir a resultados ligeiramente diferentes.

	I (KB)	L (KB)	T (KB)	Y (KB)	X (KB)
10 pontos	74	75	75	74	76
35 pontos	146	148	148	149	150
75 pontos	250	251	252	253	253
120 pontos	321	322	325	324	325
180 pontos	462	463	464	463	463

Como se antecipava, a estrutura de dados em Triangulação de Delaunay ocupa menos espaço na memória do computador (até 20 % por arquivo, aproximadamente).

6 Conclusão e comentários finais

Com relação ao desempenho geral dos algoritmos, não se percebeu uma diferença de performance associada à variável “contrafluxo” pois, no ato de leitura dos dados, as coordenadas dos pontos em um quadro da gravação não influenciavam em como os algoritmos interpretariam os pontos no quadro seguinte. Por mais que o contrafluxo condicione os passageiros a se deslocarem mais lentamente, isso não interfere no tempo de execução de cada quadro. É evidente que um fluxo mais lento requer mais quadros de gravação e isso requer, conseqüentemente, mais espaço de armazenamento, mas no que se refere ao tempo de execução quadro a quadro, não é percebida diferença.

A variável “configuração dos corredores” levou a diferenças relevantes pois, na prática, requisitou a subdivisão dos corredores em trechos convexos, conforme detalhado na Tabela 6 na seção **4.3 Particularidade na implementação quanto às configurações dos corredores**. A execução relativamente rápida dos algoritmos em cada trecho convexo em separado levou a tempos bastante semelhantes de execução dentro de um mesmo algoritmo quando a quantidade de passageiros alcançava valores suficientemente elevados (algo próximo a setenta e cinco passageiros em um quadro). As configurações de corredor L, T e Y tiveram desempenho praticamente idêntico em termos de tempo de execução e, devido a falta de repetições de execução dos códigos implementados, é possível que as configurações I e X não se distanciem tanto das outras três quando for utilizado um computador mais potente.

Por fim, ficou evidente que a variável que impactou o tempo de performance mais consideravelmente foi a quantidade de passageiros em um quadro, com alguma variação de tempo entre as distintas configurações de corredor com mesmo número de passageiros em determinado *Frame* e nenhuma variação quanto ao número de configurações Origem-Destino presentes no fluxo.

O Algoritmo Dividir para Conquistar teve melhor desempenho em todos os experimentos na condição em que estes foram realizados e na forma como foi feita a implementação de cada um dos algoritmos selecionados. É possível que a forma como se deu a implementação dos algoritmos tenha influenciado no resultado, haja vista que:

- O tamanho do passo da linha de varredura no Algoritmo de Fortune pode ser dinamicamente alterado: em vez de se fazer uso do Método de Passo Fixo, isso é, em vez de deslocar a linha de varredura a uma taxa constante (que pode ser

adequada para casos de maior densidade local, mas que pode ser pequena demais para maior espaçamento entre os pontos de S), pode ser adotado o Método de Passo Variável – alterar a taxa de deslocamento da linha de varredura de acordo com o distanciamento entre os pontos de S mais próximos à linha;

- No Algoritmo de Fortune, da forma como este foi implementado para este Trabalho de Graduação, foi feita uma varredura para cada região convexa do corredor. Possivelmente há uma forma viável e otimizada de usar uma linha de varredura única simultânea para todas as regiões convexas do corredor;
- O Algoritmo Iterativo requer que um Triângulo Muito Grande (TMG) seja criado e contenha todos os pontos do conjunto S . Ao se proceder como foi descrito na seção **4.3 Particularidade na implementação quanto às configurações dos corredores**, determinando a envoltória convexa de cada região convexa que compõe o corredor, cria-se um TMG com base no “Número de pontos para divisão em um mínimo de regiões convexas” (terceira coluna da Tabela 6) . Isso reduz consideravelmente a quantidade de vezes em que é preciso verificar se um ponto do conjunto S está dentro do TMG – por exemplo, um Cruzamento em X só precisaria de 12 pontos que são sempre os mesmos a cada *Frame* de um experimento, não precisando verificar nenhum dentre centenas de passageiros que um único *Frame* pode conter para garantir um TMG. Porém, a cada *Frame* no Algoritmo Iterativo, se faz necessário montar e apagar um TMG um número de vezes correspondente a “Número mínimo de regiões convexas” (segunda coluna da Tabela 6);
- Mas também, cada um dos três algoritmos escolhidos inicia com a divisão do corredor em regiões convexas conforme indicado na Tabela 6, ou seja, todos já iniciam com a etapa inicial do Algoritmo Dividir para Conquistar, que acaba por necessariamente pular um passo em relação aos demais dentro das condições de contorno dos experimentos;
- O Algoritmo Iterativo pode ser otimizado caso não seja considerado que há restrições devido a paredes, isso é, que todos os passageiros já estão contidos em uma única região convexa, como um pátio cujas bordas se estendem além do observável pela obtenção dos dados, descartando a necessidade de criar um TMG a cada análise, já incluindo na inicialização do Algoritmo Iterativo as coordenadas fixas dos vértices de um TMG ignorando a etapa de verificação –

para cada ponto de S ou de sua envoltória convexa – se este está contido no interior do TMG.

Dessa forma, A Triangulação de Delaunay tem um algoritmo que é o melhor dentro do escopo de análise escolhido – corredores divididos em regiões convexas – e possui uma Estrutura de Dados mais compacta. Isso é indicativo de que talvez a melhor forma de se trabalhar seja apenas com os Triângulos de Delaunay e deixar que os Diagramas de Voronoi sejam aplicados apenas em análises menos simplórias, por exemplo, considerando as dimensões do passageiro que, agora, não pode ser considerado um ponto no espaço \mathbb{R}^2 e que, por consequência, não poderia servir como vértice para vários triângulos.

Haja vista que cada passageiro passaria a ser um conjunto de pontos nestas análises menos simplórias, isso implicaria algoritmos diferentes para as triangulações, ou mesmo o uso de Inteligência Artificial (I.A.) e Aprendizado de Máquina (A.M.) para identificar “classes” de passageiros e permitir aplicação de modelos distintos de corpo extenso, entre pessoas adultas, crianças, malas, pessoas com dificuldade de locomoção e afins, tal como é feito em áreas como engenharia de tráfego, em que os veículos são modelados conforme o seu porte (motocicleta, veículo comum de médio porte ou caminhão/ônibus), o número de faixas disponíveis, curvas ou rotatórias, dentre outros.

Referências

- [1] COSCIA, M.; NEFFKE, F. M. H. & HAUSMANN, R. Knowledge diffusion in the network of international business travel. **Nature Human Behavior**, v. 4, p. 1011–1020, 2020.
- [2] LOPES, D. R.; RODRIGUES FILHO, O. S. **Aeroportos: tópicos em planejamento e projeto**. 1. ed. Curitiba: Appris, 2021.
- [3] REZENDE, P. J. de; STOLFI, J. **Fundamentos de Geometria Computacional**. Campinas: IX Escola de Computação, 1994.
- [4] AURENHAMMER, F.; KLEIN, R.; LEE, D. T. **Voronoi Diagrams and Delaunay Triangulations**. Singapura: World Scientific Publishing Co., 2013.
- [5] LEDOUX, H.; OHORI, K. A.; PETERS, R. **Computational Modelling of Terrains**. v 0.8. Delft: Delft University of Technology, 2021. Cap. 3, p. 29-40. Notas de aula.
- [6] DEVADOSS, S. L.; O’ROURKE, J. **Discrete and Computational Geometry**. Princeton: Princeton University Press, 2011. Cap. 3-4, p. 59-117.
- [7] VANDERBEI, R. J. **Linear Programming: Foundations and Extensions**. 4. ed. Princeton: Springer, 2014. Cap. 10, p. 141-150.
- [8] BOWYER, A. Computing Dirichlet tessellations. **The Computer Journal**, v. 24, n. 2, p. 162–166. Jan. 1981.
- [9] FORTUNE, Steven. A Sweepline Algorithm for Voronoi Diagrams. **Algorithmica**, v. 2, n. 4, p. 153–174. Out. 1986.
- [10] LEE, D. T.; SCHACHTER, B. J. Two Algorithms for Constructing a Delaunay Triangulation. **International Journal of Computer and Information Sciences**. v. 9, n. 3, P. 219-227, Fev. 1980.
- [11] FRUIN, J. J. Designing for pedestrians: a level-of-service concept. **Highway Research Record**. v. 355, p. 89-103. 1971.
- [12] KLEIN, P. Influência dos níveis de serviço nas velocidades de fluxo em corredores. In: APRESENTAÇÃO DE EXAME DA DISCIPLINA IT-200, 2022, São José dos Campos. **Anais [...]**. São José dos Campos.

FOLHA DE REGISTRO DO DOCUMENTO			
1. CLASSIFICAÇÃO/TIPO TC	2. DATA 25 de novembro de 2022	3. REGISTRO N° DCTA/ITA/TC-113/2022	4. N° DE PÁGINAS 48
5. TÍTULO E SUBTÍTULO: Avaliação do fluxo de pedestres para extração de métricas de comportamento			
6. AUTOR(ES): Peter Targino Klein			
7. INSTITUIÇÃO(ÕES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÕES): Instituto Tecnológico de Aeronáutica – ITA / Centro de Operações Espaciais – COMAE			
8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR: 1. Diagrama de Voronoi. 2. Triangulação de Delaunay. 3. Área de ocupação e influência.			
9. PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO: 1. Diagramas de Voronoi. 2. Geração de grades (matemática). 3. Terminais de passageiros. 4. Geometria computacional. 5. Transportes.			
10. APRESENTAÇÃO: X Nacional Internacional ITA, São José dos Campos. Curso de Graduação em Engenharia Civil-Aeronáutica. Orientador: Prof. Dr. Evandro José da Silva; coorientador: Prof. Dr. Paulo Ivo Braga de Queiroz. Publicado em 2022.			
11. RESUMO: <p>Com o desenvolvimento e a expansão do setor de serviço aeroportuário de transporte de passageiros, houve também o aumento do grau de exigências relacionadas a conforto e segurança. Uma destas exigências, impulsionada pela pandemia da COVID-19, foi em relação ao distanciamento físico entre determinado passageiro e seus vizinhos e à área efetivamente ocupada por este ao longo de sua travessia por dado trecho do Terminal de Passageiros. O presente trabalho utiliza conceitos e técnicas do campo da geometria computacional, em particular os Diagramas de Voronoi e as Triangulações de Delaunay, para avaliar a área de ocupação e influência de cada pessoa a cada instante de tempo durante suas passagens por diferentes cenários. Para tal, foi utilizada a Base de Dados de Fluxo de Pedestres disponibilizada pelo Centro de Pesquisa de Jülich (Alemanha) em seu sítio-web e a implementação dos Diagramas de Voronoi e das Triangulações de Delaunay correspondentes com três algoritmos distintos. É feita uma comparação entre os algoritmos selecionados em termos de eficiência computacional na Linguagem de Programação C++, a qual foi escolhida pela sua notória rapidez e possibilidade de aplicação em tempo real.</p>			
12. GRAU DE SIGILO: (X) OSTENSIVO () RESERVADO () SECRETO			