



TÉCNICO
LISBOA

Simulação urbana em modelos abertos CityGML: aplicação à altura máxima de edifícios em Lisboa

Rafael Salero da Silva

Dissertação para obtenção do Grau de Mestre em

Engenharia Civil

Orientadores

Professor Doutor Francisco Afonso Severino Regateiro

Professora Doutora Maria Beatriz Marques Condessa

Júri

Presidente: Professora Doutora Maria do Rosário Maurício Ribeiro Macário

Orientador: Professora Doutora Maria Beatriz Marques Condessa

Vogal: Professora Doutora Ana dos Santos Morais de Sá

Outubro 2020

Declaração

Declaro que o presente documento é um trabalho original da minha autoria e que cumpre todos os requisitos do Código de Conduta e Boas Práticas da Universidade de Lisboa.

Resumo

As cidades estão cada vez mais a ser representadas por modelos 3D de cidade semânticos em diferentes áreas de aplicação. A partilha de elementos comuns entre aplicações incentiva a criação de um modelo global. O CityGML promove a interoperabilidade entre sistemas e é simultaneamente um modelo livre e aberto.

Este trabalho pretende contribuir para um melhor entendimento da criação e aplicação de modelos 3D de cidade semânticos com o intuito de responder a desafios práticos. Neste sentido, criou-se um modelo de cidade 3D semântico de uma parte da cidade de Lisboa para a simulação da evolução da altura dos edifícios a longo prazo tendo em conta a regulamentação urbanística em vigor. Foi desenvolvida uma ferramenta capaz de mapear a regulamentação sobre a máxima altura permitida num edifício e realizar simulações permitindo ao utilizador escolher um subconjunto das regras a simular.

A dissertação sistematiza uma forma para o tratamento de dados e fornece uma metodologia para a criação de modelos 3D de cidade semânticos, bem como a sua utilização no estudo do planeamento urbano. Os resultados obtidos, para o problema escolhido, permitem identificar quais os edifícios com maior potencial de crescimento e perceber a influência que as regras mapeadas provocam na ampliação ou reconstrução dos edifícios. Essa simulação pode ser útil tanto na perspetiva dos técnicos de planeamento da administração pública, como dos promotores privados.

Palavras Chave: Modelo 3D de cidade semântico; CityGML; Simulação; Altura máxima.

Abstract

Cities are increasingly being represented by 3D semantic city models in different application areas. Sharing common elements between applications encourages the creation of a global model. CityGML promotes interoperability between systems and is both a free and open model.

This work intends to contribute to a better understanding of the creation and application of 3D semantic city models in order to answer practical challenges. In this sense, a 3D semantic city model of part of Lisbon was created for the simulation of the evolution of the height of buildings in the long term, considering the urban regulations in force. A tool was developed to map the regulations on the maximum height allowed in a building and to carry out simulations allowing the user to choose a subset of the rules to be simulated.

The dissertation systematizes a way for data processing and provides a methodology for the creation of 3D semantic city models, as well as its use in the study of urban planning. The results obtained, for the chosen problem, allow to identify which buildings have the greatest growth potential and to perceive the influence that the mapped rules cause in the expansion or reconstruction of the buildings. This simulation can be useful both from the perspective of public administration planning technicians and private promoters.

Keywords: 3D Semantic city model; CityGML; Simulation; Maximum height

Índice

1. Introdução	1
1.1. Objetivo.....	2
2. Estado da Arte.....	4
2.1. Modelo 3D de cidade semântico	4
2.1.1. Formatos de representação abertos	5
2.1.2. Aplicações de modelos 3D de cidade semânticos	10
2.1.3. Visualização.....	12
2.2. Tecnologias de informação	13
2.3. Legislação e regulamentação urbanística	15
2.3.1. Regulamento geral das edificações urbanas	16
2.3.2. Regulamentação nos planos territoriais municipais	17
3. Desenvolvimento e Implementação	23
3.1. Metodologia	23
3.2. Formalização do problema	23
3.3. Esquema de dados.....	24
3.4. Carregamento e cálculo de dados	26
3.5. Simulação.....	29
3.6. Funções Desenvolvidas	30
3.6.1. Frente urbana	33
3.6.2. Pares de edifícios adjacentes.....	34
3.6.3. Restrição “R_Altura_Média”	36
3.6.4. Restrição “R_Altura_Vizinho”	37
3.6.5. Restrição “R_Distância_45”.....	37
3.6.6. Restrição “R_Distância_Tardoz”	43
3.6.7. Restrição “R_SVP”	45
3.6.8. Função principal	45
3.7. Visualização.....	48
4. Caso de Estudo.....	49

4.1. Execução da análise	49
4.2. Resultados.....	51
5. Conclusão e Trabalhos futuros	58
6. Bibliografia.....	61
7. Anexos	A

Índice de Figuras

Figura 2.1 - Exemplo de sintaxe CityGML (Fonte: OGC).....	6
Figura 2.2 – Módulos temáticos do CityGML (adaptado de Yao, et al. (2018)).....	6
Figura 2.3 - Níveis de detalhe, fonte (Biljecki, Level of Detail in 3D City Models, 2017)	7
Figura 2.4 - 3DCityDB, janela de configuração de base de dados	8
Figura 2.5 - Parte do esquema relacional do 3DCityDB (fonte: elaborado pelo autor)	9
Figura 2.6 - a) Estimativa de potencial energético nas coberturas de edifícios; b) Simulação de cheias (adaptado de Biljecki, et al., 2015)	11
Figura 2.7 - a) Análise de visibilidade de um edifício; b) Estimativa da poluição sonora (adaptado de Biljecki, et al., 2015).....	12
<i>Figura 2.8 - Subclasses de "geometry" (adaptado de ArcGis (2019))</i>	<i>14</i>
Figura 2.9 - Aplicação dos Artigos 59º-60º do RGEU (fonte: RMEU de Coimbra (2003)).....	17
Figura 2.10 - Classificação traçado urbano relativo a espaços consolidados - espaços centrais e residenciais (fonte: adaptado do Manual Técnico PDM Lisboa (2013))	19
Figura 2.11 - Exemplo de uma frente edificada do arruamento entre duas transversais	20
Figura 2.12 - Exemplo de edifício com os tipos de solo associados aos parâmetros A, B e C (fonte: Manual Técnico PDM Lisboa (2013)).....	21
Figura 2.13 - Classificação do património edificado e paisagístico - pormenor na Av. da República (fonte: adaptado da Planta de Qualificação do espaço urbano – Lisboa)	22
Figura 3.1 - Decomposição do método geral em 5 fases	23
Figura 3.2 - Modelo relacional "custom" (elaborado pelo autor).....	25
Figura 3.3 - Fases caracterização do modelo 3D de cidade semântico	28
Figura 3.4 - Funções diretamente executadas pelo utilizador (funções de 1º nível)	30

Figura 3.5 - Especificação da função custom.get_nearest_road_id	33
Figura 3.6 - Exemplo de classificação de quarteirões	34
Figura 3.7 - Especificação da função custom.insert_building_neighbour	35
Figura 3.8 - Especificação da função custom.rule_avg_height.....	36
Figura 3.9 - Especificação da função custom.rule_neighbour_height	37
Figura 3.10 - Especificação da função custom.get_point_front_or_back_wall	38
Figura 3.11 - Especificação da função custom.get_ortogonal_line_front_or_back_wall	40
Figura 3.12 - Especificação da função custom.rule_distance_45.....	41
Figura 3.13 - Exemplo de aplicação - distância livre na frente do edifício	42
Figura 3.14 -Exemplo de aplicação - distância livre na frente do edifício no caso de edifício de gaveto...	43
Figura 3.15 - Especificação da função custom.rule_back_wall_distance	44
Figura 3.16 - Exemplo de aplicação da Regra de Tardoz.....	45
Figura 3.17 - Especificação da função custom.main	48
Figura 4.1 - Área de estudo, Lisboa (fonte: Google Maps – Av. da República).....	49
Figura 4.2 - Exemplificação da execução da função principal	51
Figura 4.3 - Histograma de alterações efetuadas por rua.....	53
Figura 4.4 - Simulação S1, perspetiva por iteração	54
Figura 4.5 - S1 - Acréscimo de altura por edifício, versão ordenada.....	56
Figura 4.6 - S1 - Acréscimo de altura por edifício, versão aleatória	56
Figura 4.7 - (S1) Vista aérea sobre o caso de estudo, dividido em 4 fases	57
Figura 7.1 - Simulação S2, perspetiva por iteração	C
Figura 7.2 - Simulação S3, perspetiva por iteração	C
Figura 7.3 - Simulação S4, perspetiva por iteração	D

Índice de Quadros

Quadro 2.1 - Descrição de tabelas 3DCityDB.....	10
Quadro 3.1 - Restrições relativas à altura do edifício e dimensão dos logradouros.....	24
Quadro 3.2 - Descrição das tabelas "custom".....	25
Quadro 3.3 - Lista de funções auxiliares desenvolvidas.....	31
Quadro 4.1 - Simulações executadas	51
Quadro 4.2 - Resumo das alterações da simulação S1, valores médios por rua.....	52
Quadro 4.3 - Resumo alterações por simulação.....	55
Quadro 7.1 - Classificação de ruas quanto à morada.....	A
Quadro 7.2 - Lista de edifícios classificados na área de estudo.....	A

1. Introdução

Uma cidade é um “organismo vivo” com vários atores e entidades que convivem entre si, nem sempre sendo fácil a sua convivência e a sua compatibilidade, surgindo então a necessidade de um planeamento ativo capaz de combater os problemas, responder às ameaças e aproveitar as oportunidades. O urbanismo é reconhecido como uma disciplina que estuda a cidade, a regulação e procura harmonizar a cidade face aos problemas existentes, de forma a criar as condições necessárias à instalação de centros urbanos. Segundo o Glossário do Desenvolvimento Territorial, p. 31, (2011), o urbanismo é “uma disciplina do planeamento que lida com o desenvolvimento físico, social, económico e ambiental das regiões metropolitanas, dos municípios e dos bairros. O urbanismo concretiza-se através da elaboração de planos de uso do solo e também de regulamentos locais no domínio do ambiente e da edificação”.

Tradicionalmente os urbanistas recorrem a representações bidimensionais (2D) e a maquetes físicas (3D) para reproduções da realidade, mas os desenvolvimentos tecnológicos, quer ao nível da visualização, quer ao nível da manipulação da informação georreferenciada, têm alavancado o desenvolvimento de novas abordagens complementadas com representações tridimensionais virtuais interativas. Neste sentido, as cidades estão cada vez mais a ser representadas por modelos 3D de cidade semânticos (Kolbe, Gröger, & Plümer, 2005). Esta nova representação adiciona uma componente semântica aos elementos das representações tridimensionais, isto é, cada objeto pode ser classificado relativamente a determinadas características ou propriedades que sejam relevantes para o modelo. Assim, permite-se uma melhoria de tomada de decisões em aplicações como por exemplo, a propagação de ruído na zona envolvente de um eixo viário, tendo em conta propriedades acústicas do revestimento das fachadas dos edifícios, ou, a estimativa da exposição solar no telhado de um edifício, considerando a sombra criada pelos edifícios circundantes, com o objetivo de avaliar se é economicamente viável a instalação de um painel solar, ou ainda, a análise de visibilidade que permite identificar os locais estratégicos, por exemplo para colocar placards de publicidade ou planear a instalação de câmaras de segurança de modo a maximizar a área registada, entre outros (Biljecki, 2017).

À medida que são desenvolvidos mais modelos para diferentes áreas de aplicação, mais se justifica a partilha dos elementos que são comuns entre áreas de aplicação, assim, surge a necessidade de um modelo global capaz de mapear estes modelos 3D de cidade semânticos. O CityGML poderá ser a resposta a esta necessidade na medida em que é um modelo livre e aberto, permitindo que os seus utilizadores não fiquem dependentes de um fornecedor em particular e simultaneamente, este formato funciona como uma língua franca entre os utilizadores promovendo a interoperabilidade de sistemas (Tolmer, Castaing, Diab, & Morand, 2013).

Os estudos de caracterização territorial, elaborados no decurso do processo de revisão do PDM de Lisboa, (2012) p.27, “vieram confirmar a importância e o interesse na reabilitação urbana, enquanto instrumento

de captação de novos habitantes para a cidade, na medida em que mais de 50% dos novos residentes de Lisboa vieram residir para edifícios com mais de 30 anos”. Evidencia-se assim uma preferência por edifícios no centro de Lisboa em alternativa ao crescimento contínuo das periferias metropolitanas.

Segundo Biljecki (2017) em áreas urbanas densamente povoadas os promotores exploram os limites da regulamentação em vigor quanto à dimensão do edifício, assim, é razoável assumir que a altura de novas edificações ou no caso de ampliações, estas correspondam ao limite máximo permitido. Foi esta a motivação para o caso de aplicação escolhido.

1.1. Objetivo

O objetivo geral deste trabalho é contribuir para um melhor entendimento da criação e aplicação de modelos 3D de cidade semânticos com o intuito de responder a desafios práticos num contexto de urbanismo. Para a realização deste objetivo, privilegia-se a utilização de *software* livre e de código aberto, tanto na criação de uma base de dados como em todas as outras ferramentas de suporte à operação.

Assim, a conceção do modelo de base de dados está alinhada com o desafio prático, neste sentido será dirigida a análise com o intuito de criar um modelo com base em CityGML que possa garantir uma aplicabilidade a vários problemas na área do urbanismo, e ao mesmo tempo ser replicável em qualquer cidade.

Para tal, como aplicação específica do objetivo geral propõe-se a utilização de um modelo de cidade 3D semântico de uma parte da cidade de Lisboa para a simulação da evolução da altura dos edifícios a longo prazo tendo em conta a legislação e regulamentação urbanística em vigor.

Esta dissertação poderá beneficiar os produtores de regulamentação urbanística e responsáveis pelo planeamento urbano e gestão urbanística, no sentido de fornecer uma metodologia para a criação de um modelo que simule os resultados da regulamentação em vigor, bem como as ferramentas para as potenciais alterações. Adicionalmente, o setor de promoção e gestão imobiliária poderá servir-se da mesma simulação no sentido de avaliar quais os edifícios com um maior potencial de crescimento.

Estrutura da dissertação

A estrutura da dissertação assenta em cinco capítulos onde se pretende sistematizar o trabalho desenvolvido.

Este capítulo identifica uma breve antecipação do que irá ser desenvolvido ao longo da dissertação, iniciando-se com a apresentação da temática escolhida, seguido dos objetivos da dissertação, e das secções que integram a dissertação.

No capítulo 2, desenvolve-se a revisão bibliográfica sobre o modelo 3D de cidade semântico e as ferramentas de auxílio à criação do mesmo. Considerou-se ainda importante fazer uma contextualização complementar relativa à regulamentação urbanística aplicada à alteração de altura de edifícios.

O capítulo 3 aborda a metodologia usada para o cumprimento dos objetivos definidos, a formalização do problema e as restrições aplicáveis, assim como a implementação de um modelo 3D de cidade semântico e as funções desenvolvidas no âmbito da simulação. Por fim, é exposta a capacidade de visualização dos resultados da simulação.

No capítulo 4 é apresentado um caso de estudo (no município de Lisboa) e o seu enquadramento, através da aplicação direta da metodologia e ferramentas desenvolvidas na fase anterior. Neste capítulo descrevem-se as opções tomadas durante o processo e extraem-se os principais resultados da aplicação da ferramenta.

Por último, surge o capítulo 5, onde se expõem as conclusões relativas ao trabalho desenvolvido, assim como, possíveis desenvolvimentos futuros.

2. Estado da Arte

2.1. Modelo 3D de cidade semântico

Segundo o dicionário de Oxford, um modelo é “uma representação tridimensional de uma pessoa, coisa, estrutura proposta, geralmente numa escala menor que a original” (Oxford University Press, 2010). O termo “modelo” é uma das palavras mais usadas na comunidade científica. Um modelo universal que represente todos os aspetos do mundo real não é executável devido ao nível de complexidade. Assim, investigadores propõem e validam hipóteses, fazem previsões e trocam ideias, tudo isto baseados em modelos.

Um modelo de cidade é uma abstração de uma parte do mundo real que engloba entidades urbanas e o ambiente urbano global onde está localizada. Cada entidade do modelo corresponde a uma característica urbana do mundo real (Billen, et al., 2014). Estes modelos de cidades virtuais geralmente consistem em modelos digitais de terreno, edifícios, vegetação, ruas e outras infraestruturas, que podem ser representados através de sistemas de informação geográfica, SIG (Dollner, Bauman, & Buchholz, 2006). A visualização de um modelo 3D de cidade ajuda a criar uma melhor imagem mental dos edifícios propostos e a sua relação com outros objetos próximos, por exemplo as sombras que provocam noutros edifícios (Herbert & Chen, 2015), enquanto os modelos 2D dificultam a interpretação a pessoas sem formação técnica. Assim, a visualização 3D é uma forma mais inclusiva da população, podendo funcionar como uma ferramenta de comunicação. Os requisitos dos modelos 3D de cidade variam entre diferentes aplicações: em áreas como o turismo, entretenimento, é necessário um elevado grau de fotorrealismo; em aplicações com uma vertente analítica, a representação destes modelos serve como um meio para transmitir informações temáticas com um contexto geográfico (Dollner, Bauman, & Buchholz, 2006).

Aos modelos 3D de cidade é possível adicionar determinadas características ou propriedades de objetos, surgindo assim um modelo 3D de cidade semântico, este é uma representação tridimensional da cidade que também compreende características temáticas, como tipo de solo, material ou ano de construção. Deste modo, pode associar-se, ao nível dos objetos, propriedades que sejam relevantes para o modelo. Estes modelos permitem uma melhor compreensão dos dados, sendo particularmente úteis em contextos de simulação e tomada de decisão.

Através do mesmo modelo, é ainda possível realizar diferentes tipos de análises tendo em conta a perspetiva em que incide. Assim, ao associar uma área de estudo para estes modelos (nomeadamente consumo de energia, qualidade do ar, etc.), obtemos representações 3D que podem ser usadas para consultas temáticas, simulações e *data mining* (Dollner, Bauman, & Buchholz, 2006).

A modelação semântica de cidades requer uma fase de inserção e manutenção da informação no modelo, sendo o processo geralmente manual, podendo em alguns casos ser automatizado. Esta fase exige um consumo elevado de recursos pelo que, é vantajoso que os mesmos dados sejam utilizados em várias aplicações com área de estudo distintas. Deste modo, do ponto de vista do custo global do projeto é benéfica a utilização de um repositório central em detrimento de construir e validar vários modelos independentes, no entanto, esta opção exigiria encontrar um modelo abrangente que pudesse servir várias aplicações distintas, o que nem sempre pode ser alcançado (Kolbe T. , 2009).

2.1.1. Formatos de representação abertos

Os modelos 3D de cidade semântico devem ser representados num formato livre e aberto. Por livre entendemos que um utilizador pode utilizar sem custos monetários e aberto que tem acesso à estrutura sintática completa e ao seu significado. A mais-valia dos modelos livres e abertos é não provocar a dependência de um fornecedor em particular e ao mesmo tempo funcionar como uma língua franca entre os utilizadores, promovendo a interoperabilidade de sistemas.

Os formatos de modelos CityGML e IFC (*Industry Foundation Classes*) são considerados a referência para mapear modelos 3D semânticos (El-Mekawy, Östman, & Hijazi, 2012).

O IFC é um formato orientado a objetos desenvolvido pela *buildingSMART*, cujo objetivo é especificar uma linguagem comum para a tecnologia da indústria de construção que melhore a comunicação, produtividade, tempo de entrega, custo e qualidade ao longo do ciclo de vida do edifício (Donkers, Ledoux, Zhao, & Stoter, 2016). Através do IFC é possível modelar componentes tangíveis (portas, janelas, etc.) mas também conceitos abstratos como espaço, organização e processos (Tolmer, Castaing, Diab, & Morand, 2013). Uma das principais características é o facto de o nível de detalhe deste formato não estar normalizado, isto é, objetos de diferentes níveis de detalhe podem ser agrupados em categorias consoante a necessidade (ex: canalização, elementos estruturais, etc.).

O CityGML também é usado para armazenar e trocar modelos 3D de cidades semânticos e funciona tanto como um modelo de informação como um formato de dados, é um esquema aplicacional de GML, Geography Markup Language, ver Figura 2.1 (Open Geospatial Consortium, 2012). O CityGML tem o objetivo de chegar a uma definição comum dos conceitos básicos num modelo para que este possa ser um agregador central de informação (de forma relacional) de modo a que aplicações de diferentes áreas possam acoplar os seus atributos específicos à estrutura principal do CityGML.

Tanto o CityGML como o IFC são livres e abertos mas são limitados na medida em que não se tem uma solução para todas as aplicações, o IFC tem uma abordagem direcionada ao edifício, enquanto o CityGML procura ser uma solução a uma escala menor, tipicamente ao nível da cidade (Tolmer, Castaing, Diab, & Morand, 2013). Deste modo, privilegiou-se o uso do formato CityGML.

```

<CityModel xsi:schemaLocation="http://www.opengis.net/citygml/building/2.0
http://schemas.opengis.net/citygml/building/2.0/building.xsd
http://www.opengis.net/citygml/relief/2.0
http://schemas.opengis.net/citygml/relief/2.0/relief.xsd">
  <gml:name>Simple 3D city model LOD1 without Appearance</gml:name>
  <gml:boundedBy>
    <gml:Envelope srsDimension="3" srsName="urn:ogc:def:crs,crs:EPSG::25832,crs:EPSG::5783">
      <gml:lowerCorner>458868.0 5438343.0 112.0</gml:lowerCorner>
      <gml:upperCorner>458892.0 5438362.0 117.0</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <cityObjectMember>
    <bldg:Building gml:id="GM_7b1a5eff-ddad-4c3d-a507-3eb9ee0a9e68">
      <gml:name>Example Building LOD1 </gml:name>
      <bldg:function codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_function.xml">1000</bldg:function>
      <bldg:yearOfConstruction>1985</bldg:yearOfConstruction>
      <bldg:roofType codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_roofType.xml">1030</bldg:roofType>
      <bldg:measuredHeight uom="#m">5.0</bldg:measuredHeight>
      <bldg:storeysAboveGround>1</bldg:storeysAboveGround>
      <bldg:storeyHeightsAboveGround uom="#m">3.0</bldg:storeyHeightsAboveGround>
      <bldg:lod1Solid>
        <gml:Solid>
          <gml:exterior>
            <gml:CompositeSurface gml:id="lod1Surface">
              <!-- Face Side 1 -->
              <gml:surfaceMember/>
              <!-- Face Side 2 -->
              <gml:surfaceMember/>
              <!-- Face Side 3 -->
              <gml:surfaceMember/>
              <!-- Face Side 4 -->
              <gml:surfaceMember>
                <gml:Polygon>
                  <gml:exterior>
                    <gml:LinearRing>
                      <gml:posList>458875.0 5438355.0 112.0 458875.0 5438350.0 112.0 458875.0 5438350.0 116.0 458875.0 5438355.0
116.0 458875.0 5438355.0 112.0 </gml:posList>
                    </gml:LinearRing>
                  </gml:exterior>
                </gml:Polygon>
              </gml:surfaceMember>
            </gml:CompositeSurface>
          </gml:exterior>
        </gml:Solid>
      </bldg:lod1Solid>
    </bldg:Building>
  </cityObjectMember>
</CityModel>

```

Figura 2.1 - Exemplo de sintaxe CityGML (Fonte: OGC¹)

O CityGML define um esquema conceptual para as entidades mais relevantes num ambiente urbano, através de módulos temáticos como edifícios, estradas, túneis, pontes, vegetação (ver Figura 2.2), e específica como cada objeto deve ser decomposto e classificado nas suas componentes mais simples.

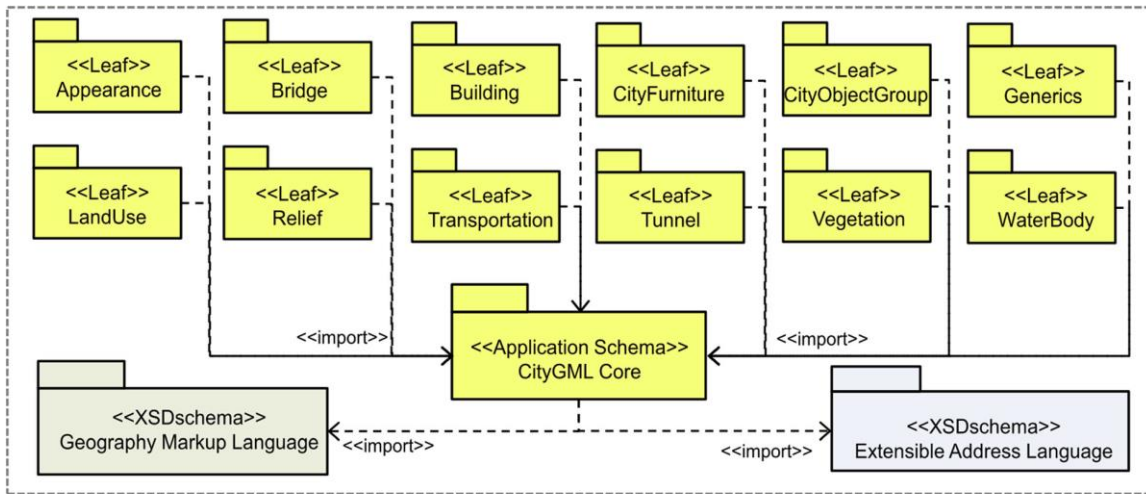


Figura 2.2 – Módulos temáticos do CityGML (adaptado de Yao, et al. (2018))

Os objetos no modelo CityGML podem ser representados em cinco níveis de detalhe (LODs) (ver Figura 2.3). Segundo Groger & Plumer (2012) a existência de LOD facilita tarefas de integração e

¹ Retirado de http://schemas.opengis.net/citygml/examples/2.0/building/Building_LOD1-EPG25832.gml

interoperabilidade entre ferramentas de importação e exportação. Os níveis mais elevados são mais precisos e têm uma maior complexidade estrutural que os níveis mais baixos. Os níveis são representados de 0 a 4, sendo o LOD0 o nível mais baixo. Neste nível um edifício é representado apenas pela sua área de implantação. O LOD1 é um modelo de blocos, onde os edifícios e as outras entidades são representados sem a estrutura do telhado. Neste nível a visualização dos objetos podem tornar-se mais próxima da realidade com a incorporação de texturas no modelo. A partir deste nível, os edifícios podem ser representados como a agregação de várias partes (outros blocos). Esta abordagem é benéfica porque permite que as diferentes partes do edifício estejam associadas a diferentes atributos, como a altura, ano de construção ou tipo de uso. Com o LOD2 é adicionado uma estrutura simples como o telhado ou varandas. De um ponto de vista geométrico uma diferença relevante é que as paredes exteriores de um edifício podem ser representadas através de múltiplas faces ou até faces curvas. O LOD3 acrescenta uma maior pormenorização do telhado e paredes exteriores através da incorporação de janelas, portas, chaminés, etc., essencialmente estruturas que possam ser observadas do exterior do edifício. O LOD4 é o nível mais pormenorizado, adiciona informação relativa ao interior dos edifícios (e.g. escadas, quartos e mobília). Para produzir resultados espaciais mais precisos pode ser necessário dados fonte em LOD mais elevados, no entanto, o custo de aquisição de dados é mais elevado para LODs de níveis superiores. Assim sendo, é essencial ajustar o nível de detalhe de acordo com os objetivos das simulações ou análises a realizar (Biljecki, 2017).



Figura 2.3 - Níveis de detalhe, fonte (Biljecki, *Level of Detail in 3D City Models*, 2017)

O CityGML permite que o mesmo objeto seja representado em diferentes LODs ao mesmo tempo, tornando possível executar análises em diferentes graus de resolução, esta capacidade permite adaptar a modelação aos dados e parametrização de regras (Nouvel, et al., 2015). Ainda assim, para a mesma análise, elementos no mesmo LOD são mais facilmente manipulados do que elementos em LOD diferentes.

O CityGML representa as relações geométricas e não espaciais entre as entidades com recurso ao esquema Geography Markup Language 3 (GML3) (Kolbe, Gröger, & Plümer, 2005). O CityGML é uma solução apropriada para ambientes exteriores e geralmente aplicado à escala da cidade.

Existem algumas ferramentas que permitem criar a estrutura de CityGML sob a forma de uma base de dados. No entanto, ainda não existe uma ferramenta regularmente seguida. Este não é um problema específico de Portugal, na Finlândia diferentes municípios deparam-se com o mesmo problema (Jokela,

2016). A disponibilização de vários formatos leva a problemas de interoperabilidade para as entidades responsáveis pelo desenvolvimento de modelos (Santos, 2016). Espera-se que a evolução de novas ferramentas torne o processo mais simples.

Um dos problemas associados à temática da modelação de cidades é a dificuldade de interoperabilidade entre os sistemas. Para além de CityGML, a utilização de sistemas de bases de dados, que suportam a interoperabilidade entre aplicações e pessoas, já foi proposta por outros autores (Preka & Doulamis, 2016). Seguindo esta orientação, neste trabalho são propostos os procedimentos desde a criação de um modelo em SketchUp, à sua representação numa base de dados CityGML/3DCityDB, implementada no sistema de bases de dados PostgreSQL/PostGIS.

2.1.1.1. CityGML em bases de dados

O 3DCityDB é uma ferramenta de código aberto, que permite construir um esquema de bases de dados que segue as regras de CityGML (Building Smart, 2019). O projeto começou em 2003 liderado por Thomas Kolbe e desde então os desenvolvimentos continuam a cargo de uma parceria entre várias organizações. Esta ferramenta incorpora ainda funcionalidades de importação, manutenção, análise, visualização e exportação de modelos 3D de cidade semânticos. O 3DCityDB (Figura 2.4) permite criar a estrutura de dados (*schema*) do modelo CityGML em dois sistemas de bases de dados espaciais e relacionais: *Oracle* com a licença espacial e PostgreSQL, na primeira existem versões com licenças de utilização livre, nomeadamente Oracle 10G R2, já no segundo caso é uma ferramenta de código aberto.

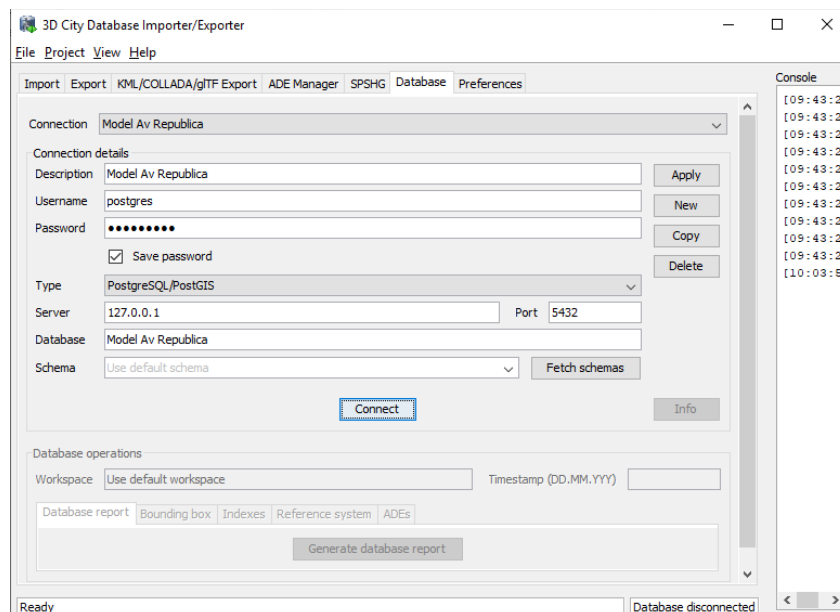


Figura 2.4 - 3DCityDB, janela de configuração de base de dados

Assim, o 3DCityDB facilita a importação de um ficheiro GML para a base de dados pretendida. Durante a importação ou exportação é possível fazer uma seleção dos objetos a importar, os critérios variam desde o módulo, a sua posição geográfica, ou pela sua referência única. A ferramenta tem a capacidade de processar modelos 3D de cidade semânticos com milhões de elementos em diferentes níveis de detalhe num curto espaço de tempo (Chair of Geoinformatics TUM, 2018). Relativamente ao processo de exportação dos objetos da base de dados, é possível exportar em formato GML ou noutros formatos como KML, COLLADA, glTF e SPSHG.

A Figura 2.5 apresenta algumas das tabelas que constituem o esquema relacional do 3DCityDB (*schema "citydb"*), são essencialmente tabelas referentes à classe edifícios (na Figura 2.2). O Quadro 2.1 apresenta algumas das características das tabelas referidas no esquema da Figura 2.5. Resumidamente, os dados espaciais que são importados ficam guardados na tabela *surface_geometry* e as restantes tabelas permitem associar a informação semântica do modelo.

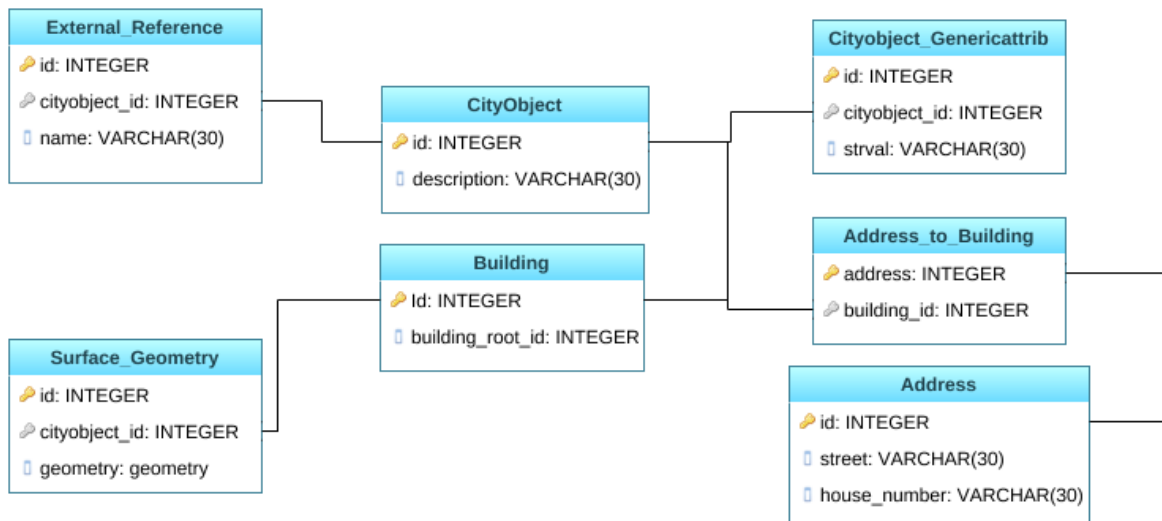


Figura 2.5 - Parte do esquema relacional do 3DCityDB (fonte: elaborado pelo autor)

Quadro 2.1 - Descrição de tabelas 3DCityDB

Tabela	Descrição
CityObject	Faz parte da classe base do <i>CityGML</i> , identifica o <i>LOD</i> ao qual o objeto se integra e é onde pode ser guardado informações quanto à criação e modificação do objeto.
External_Reference	É usado para guardar a referência externa do objeto no momento da sua importação. A partir deste campo pode-se identificar o <i>CityObject</i> no ficheiro original <i>GML</i> .
Building	Guarda a informação relativa ao conceito de edifício, permite que seja guardada informação específica do edifício como o ano de construção ou número de andares (abaixo e acima do solo). Permite relacionar agrupar superfícies através de um identificador comum <i>building_root_id</i> , por exemplo, no caso de duas estruturas, anexo e o edifício principal, estas podem possuir o mesmo identificador comum.
Surface_Geometry	Cada edifício é composto por várias superfícies (base, telhado e fachadas), esta tabela guarda as superfícies importadas para a base de dados.
CityObject _Genericattrib	É usado para representar o conceito de atributo genérico. Através desta tabela pode ser introduzida informação semântica que melhor classifique o modelo.
Address_to_Building	Tabela de ligação entre <i>Building</i> e <i>Address</i> .
Address	Guarda a morada do edifício.

2.1.2. Aplicações de modelos 3D de cidade semânticos

Os modelos 3D de cidade semânticos foram usados no passado principalmente para a visualização do edificado urbano. Atualmente, criam-se estes modelos com objetivos para além da visualização, isto é, são usados também para fazer análises e simulações (Ohori, Biljecki, Kumar, Ledoux, & Stoter, 2018). Os projetos realizados não se têm cingido apenas a uma área de estudo, de facto, pode-se verificar trabalhos realizados em várias cidades em áreas tão diversas como: planeamento urbano, ambiente, planeamento de telecomunicações, logística, imobiliário, entretenimento, educação, estimativa da exposição solar, cálculo de nível de ruído, segurança interna e navegação assistida (Grogger & Plumer, 2012). Todos estes modelos requerem informações adicionais sobre os objetos da cidade de uma forma padronizada. Nesta secção são apresentadas algumas aplicações, nas quais são identificadas as vantagens de usar estes modelos, (Biljecki, Stoter, Ledoux, Zlatanova, & Coltekin, 2015).

2.1.2.1. Estimativa da exposição solar

O cálculo de estimativa da exposição solar é um tema que sobressai entre os restantes pela quantidade de trabalhos desenvolvidos. Os modelos 3D de cidade são usados para calcular a viabilidade de instalação

de painéis fotovoltaicos nos telhados (Ver Figura 2.6-a) tendo em conta as sombras criadas pelos edifícios circundantes. Os principais parâmetros para esta estimativa são a orientação, área e inclinação do telhado; parâmetros estes que podem ser calculados através de operações espaciais no modelo. Outras abordagens têm em consideração também as fachadas dos edifícios assim como o material que as reveste.

2.1.2.2. Simulação de cheias

Simular o leito de cheia não é um conceito novo, foi principalmente realizado através de modelos digitais do terreno. No entanto, os modelos cidade 3D semânticos para além de permitirem analisar o risco de cheia provocado por precipitação intensa, também possibilitam o estudo de potenciais danos ao edifício (Ver Figura 2.6-b). Estes modelos podem ainda ser importantes para análise de evacuação e gestão de risco do ponto de vista das seguradoras e operadoras de gestão de águas.

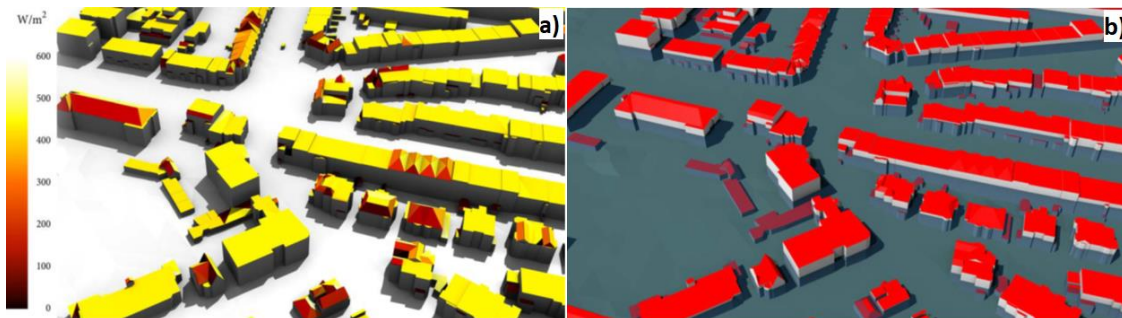


Figura 2.6 - a) Estimativa de potencial energético nas coberturas de edifícios; b) Simulação de cheias (adaptado de Biljecki, et al., 2015)

2.1.2.3. Detecção de alterações na cidade

Algumas instituições procuram realizar um inventário 3D sobre o registo de propriedade, incluindo casos mais complexos como cadastro vertical de todos os fogos ou construções subterrâneas como garagens e condutas. Estes dados são usados, por exemplo, para fazer uma visualização da cidade mais realista. Outro propósito é realizar uma análise temporal, verificando as alterações na morfologia da cidade ou a um nível de maior detalhe, extensões verticais ou horizontais que tenham sido acrescentadas aos edifícios.

2.1.2.4. Análise de visibilidade

Os modelos 3D de cidades são essenciais para determinar a linha de visão entre dois pontos num ambiente urbano ou o volume visível de um determinado ponto (Ver Figura 2.7-a). Estes estudos são realizados para estimar a visibilidade de pontos de referência da cidade (por exemplo, igrejas ou marcos históricos),

determinar pontos estratégicos para *placards* de publicidade, distribuição de câmaras de segurança de forma a maximizar a área registada, ou para avaliação imobiliária.

2.1.2.5. Nível de ruído

O planeamento do reforço de veículos de transporte pode ser complementado com uma estimativa dos efeitos da poluição sonora na zona envolvente ao eixo viário. Através desta análise é possível identificar os edifícios sujeitos a níveis de ruído demasiado elevados, ou como proceder para mitigar este problema através da colocação de barreiras (Ver Figura 2.7-b). Os modelos 2D são frequentemente usados para este propósito, todavia modelos 3D de cidade oferecem a vantagem de resultados mais precisos de fenómenos de refração. A particularidade semântica do modelo pode refinar a análise quando se conhece o revestimento das fachadas quando às suas propriedades acústicas.

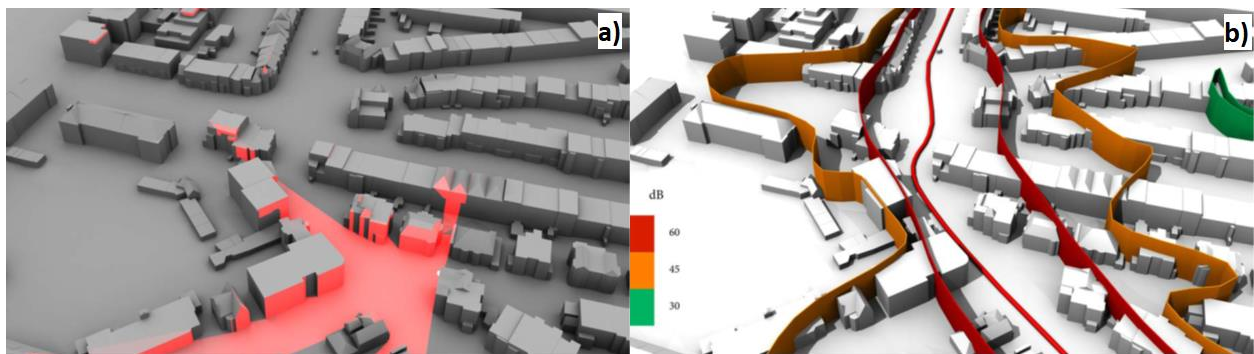


Figura 2.7 - a) Análise de visibilidade de um edifício; b) Estimativa da poluição sonora (adaptado de Biljecki, et al., 2015)

2.1.3. Visualização

No passado, os mapas em papel foram projetados com duas funcionalidades: ser o repositório de dados e um formato de apresentação. Com o desenvolvimento da cartografia digital e dos SIG, estas funcionalidades são cumpridas por plataformas diferentes (Moradi, 2017).

Ferramentas como globos virtuais permitem aos utilizadores uma experiência verdadeiramente interativa com o modelo 3D, através destas ferramentas pode-se fazer a navegação dentro do modelo e aceder a informação associada aos objetos do modelo. Existem algumas aplicações desktop destinadas à visualização de modelos *CityGML* como *FZK Viewer*, *FME Data Inspector* e *azul*. Alternativamente, existem aplicações *web* mas ainda apresentam alguns desafios como problemas de renderização em modelos de

maior dimensão ou dificuldades relacionadas com a instalação de *plugins* (Ohori, Biljecki, Kumar, Ledoux, & Stoter, 2018).

A visualização de CityGML pela Web requer a separação da informação geométricas da semântica. Vários formatos como X3D, KML / COLLADA, etc. podem ser usados, no entanto, quando os dados em CityGML são convertidos para esses formatos, alguma da informação semântica pode ser perdida. O Google Earth permite visualizar ficheiros KML / COLLADA sem que seja necessária parametrização adicional. Em alternativa, existem outras ferramentas web de código livre como Cesium JS, OpenWebGlobe ou WebGLEarth.

2.2. Tecnologias de informação

Para além dos sistemas de bases de dados referidos anteriormente, pretende-se agora complementar o conhecimento sobre ferramentas *open-source* e livres úteis para a implementação de processos de extração e transformação de dados geográficos, e em particular para o processo desenvolvido neste trabalho.

A aplicação Cadmapper (2013) disponibiliza ficheiros que incluem a topografia do terreno, edifícios, redes de infraestruturas de transporte e outras estruturas existentes. Esta permite extrair mapas de mais de 200 cidades ou seleccionar até 1km² de qualquer local do mundo sem qualquer custo. A fonte de dados utilizada é o OpenStreetMap Database, sendo que podem extrair-se mapas completos para serem editados em ferramentas de desenho como AutoCad, SketchUp, Rhinoceros e Illustrator.

O CityEditor (2016) é uma extensão do SketchUp que permite a importação e exportação de ficheiros CityGML. A sua utilização não é completamente livre, mas é possível adquirir uma licença gratuita por um período limitado. Esta extensão permite editar os atributos específicos da estrutura CityGML, assim sendo, pode enriquecer-se o desenho com descrições e relações entre os objetos inseridos na análise.

O PostgreSQL (1996) é um sistema aberto de gestão de bases de dados relacional de objetos. Este usa SQL (Structured Query Language), uma linguagem padrão para armazenar, manipular e seleccionar dados numa base de dados relacional. Uma característica das bases de dados é a possibilidade de interação em simultâneo entre vários utilizadores na mesma base de dados, isto porque, a utilização de uma transação em SQL garante sempre os princípios de atomicidade, consistência, isolamento e durabilidade (ACID). A atomicidade é a propriedade que garante que a transação é executada como um todo, se alguma parte da operação falhar então a transação será desfeita; a consistência é a propriedade que preserva o cumprimento das restrições impostas na base de dados; o isolamento garante que cada transação é independente das restantes (assim diferentes utilizadores não podem visualizar resultados parciais de outras transações); pela durabilidade impõe que, depois de terminada uma transação, esta apenas pode ser anulada por outra transação. O PostGis é uma extensão do PostgreSQL que permite acrescentar uma

dimensão espacial à base de dados. Esta extensão introduz três grupos de recursos: formatos de dados, índices e funções espaciais. Quanto aos formatos adicionados pelo PostGis, existem dois : “geography” e “geometry”. O primeiro é representado num referencial elipsoidal e deve ser usado para conjuntos de dados que abrangem diferentes continentes, o “geometry” é representado num referencial euclidiano e usualmente utilizado para conjuntos de dados mais locais, com a projeção de coordenadas adequado. Estes formatos permitem representar a informação através de três estruturas base (pontos, linhas, polígonos) ou como coleções, através de qualquer combinação das estruturas bases mencionadas anteriormente, ver Figura 2.8.

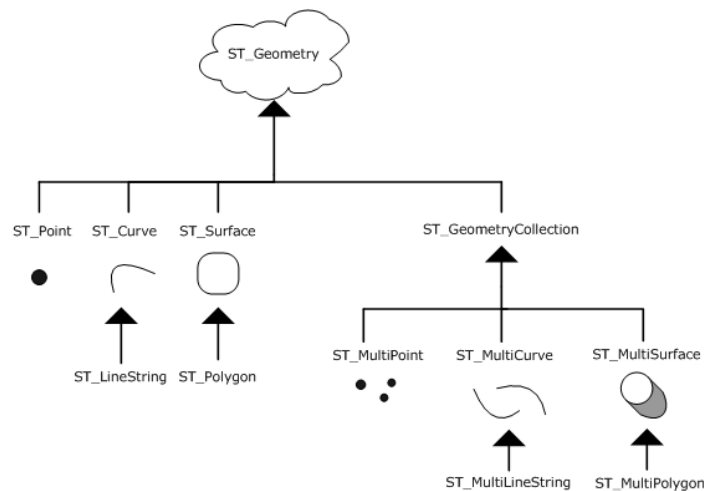


Figura 2.8 - Subclasses de "geometry" (adaptado de ArcGis (2019))

Quanto às funções espaciais, adicionados pela extensão PostGis, apresentam-se agora algumas das funções que permitem a transformação e análise de objetos geométricos:

- ST_Area, retorna a área de uma superfície / polígono;
- ST_ZMax, retorna a máxima cota de uma geometria, seja 2D ou 3D;
- ST_Translate, permite transladar uma geometria em qualquer direção;
- ST_Distance, retorna a distância mínima cartesiana entre duas geometrias em unidades projetadas;
- ST_Centroid, retorna o centro de massa de um conjunto, sejam estes pontos, linhas ou superfícies;
- ST_MakePoint, constrói um ponto;
- ST_MakeLine, constrói uma linha;
- ST_NPoints, retorna o número de pontos numa geometria;
- ST_PointN, retorna o ponto N numa geometria, sendo N o seu número de ordem;

- ST_Dump, permite desagregar geometrias nos seus componentes mais básicos;
- ST_Boundary, retorna a fronteira do objeto;
- ST_Intersection, retorna a interseção entre duas geometrias;
- ST_Azimuth, devolve o azimute entre dois pontos;
- ST_SetSRID, define o sistema de referência de coordenadas² para uma geometria.

2.3. Legislação e regulamentação urbanística

No que concerne à utilização de software, tem-se verificado por parte do Governo, a intenção de optar por uma estratégia com recurso a software de utilização livre. Salientam-se as diretrizes: Resolução da Assembleia da República nº 66 (2004) que recomenda ao Governo a tomada de medidas com vista ao desenvolvimento de software livre em Portugal, nomeadamente, a “obrigatoriedade de acesso ao código-fonte e especificações dos formatos de dados na aquisição de soluções informáticas destinadas à utilização pela Administração Pública”; a Resolução do Conselho de Ministro nº12 (2012) volta a salientar este ponto, “promover a utilização de software aberto nos sistemas de informação da Administração Pública sempre que a maturidade e o custo sejam favoráveis.”

De forma a aplicar o conceito de modelo 3D de cidade semântico no domínio da engenharia civil, elenca-se nos pontos seguintes a legislação e regulamentação urbanística em vigor.

Atualmente, o sistema de gestão territorial em Portugal assenta na Lei de Bases Gerais da Política Pública de Solos, de Ordenamento do Território e de Urbanismo³ (LBPPSOTU), esta procura “valorizar as potencialidades do solo, salvaguardando a sua qualidade e a realização das suas funções ambientais, económicas, sociais e culturais, enquanto suporte físico e de enquadramento cultural para as pessoas e suas atividades”. Este sistema está estruturado em Instrumentos de Gestão Territorial (IGT) cujo regime jurídico (RJIGT) foi revisto pelo Decreto-Lei n.º 80/2015, de 14 de maio, onde são definidas as principais regras sobre o planeamento e ordenamento do território em Portugal. Os IGT são formalizados a diversos níveis territoriais: nacional, regional e municipal.

² A tabela public.spatial_ref_sys contém todos os sistemas predefinidos nos quais podem ser representados os dados espaciais, no entanto o PostGis permite a adicionar de novos sistemas customizados, de acordo com as regras em <http://spatialreference.org>.

³ Lei 31/2014, de 30 de maio, que revoga a Lei 48/98, de 11 de agosto.

O âmbito nacional é orientado pelo Programa Nacional da Política de Ordenamento do Território (PNPOT), que serve de referência e complementa os restantes programas e planos territoriais. Os restantes instrumentos neste nível são os Programas Sectoriais (PS) e os Programas Especiais (PE).

O âmbito regional é concretizado pelos Programas Regionais (PR) que têm como objetivo definir diretrizes para o uso, ocupação e transformação do território além de promover a integração dos instrumentos de nível superior e inferior.

O âmbito municipal, que define o regime do uso do solo e sua regulamentação em função das diretrizes de nível superior, é composto pelos Planos Directores Municipais (PDM), Planos de Urbanização (PU) e Planos de Pormenor (PP).

O desenvolvimento dos espaços urbanos é o resultado de um contínuo progresso da civilização. Segundo o estudo “*World Urbanization Prospects*” realizado pelas Nações Unidas (2014), nesse ano 54% da população mundial vivia em áreas urbanas, no entanto, projeta-se que em 2050, a proporção da população nestas condições atinja os 66%. Esta procura por espaço urbano leva à reabilitação ou construção de novos edifícios na cidade.

A edificação começou por ser uma atividade livre, sem necessidade de licenciamento ao nível do direito público. À medida que o processo de urbanismo evoluiu surgiu a necessidade de legislar e regulamentar esta operação. A aprovação do Regulamento Geral das Edificações Urbanas⁴ (RGEU) fixa as regras, num âmbito nacional, aplicáveis à execução de novas edificações ou de quaisquer obras de construção civil, reconstrução, ampliação, alteração, reparação ou demolição das edificações e obras existentes.

De acordo com o DL 136/2014, de 9 de setembro, tomam-se como obras de ampliação “as obras de que resulte o aumento da área de implantação, da área total de construção, da altura da fachada ou do volume de uma edificação existente”. No que concerne à ampliação de edifícios, o Regulamento geral das edificações urbanas (RGEU) e os planos municipais (PDM, PU e PP) são os documentos que fornecem índices e parâmetros que servem de base ao planeamento das edificações.

2.3.1. Regulamento geral das edificações urbanas

O RGEU aplica-se num contexto nacional às operações urbanísticas executadas dentro do perímetro urbano e das zonas rurais de proteção fixadas para as sedes de concelho, para que a sua aplicação seja considerada fora dessas zonas exige-se uma deliberação municipal. Este regulamento integra as construções num novo patamar de exigência, estabelecendo regras específicas relativamente a áreas, espessuras, secções, distâncias e pés-direitos, fixando limites para estes e outros parâmetros. O RGEU

⁴ DL 38 382, de 7 de agosto de 1951

centra-se também noutras preocupações como o parcelamento dos terrenos, a integração na Natureza das novas construções ou ainda questões ao nível do conforto ambiental e acústico.

De acordo com o RGEU, sinalizam-se os artigos 59º, 60º e 62º que dizem respeito a especificações relativas à altura de edifícios:

Artigo 59º - “A altura de qualquer edificação será fixada de forma que em todos os planos verticais perpendiculares à fachada nenhum dos seus elementos, com exceção de chaminés e acessórios decorativos, ultrapasse o limite definido pela linha reta a 45 graus, traçada em cada um desses planos a partir do alinhamento da edificação fronteira, definido pela intercessão do seu plano com o terreno exterior.”

Artigo 60º - “Independentemente do estabelecido no artigo anterior, a distância mínima entre fachadas de edificações nas quais existam vãos de compartimentos de habitação não poderá ser inferior a 10 metros” (Tal como mostra a Figura 2.9).

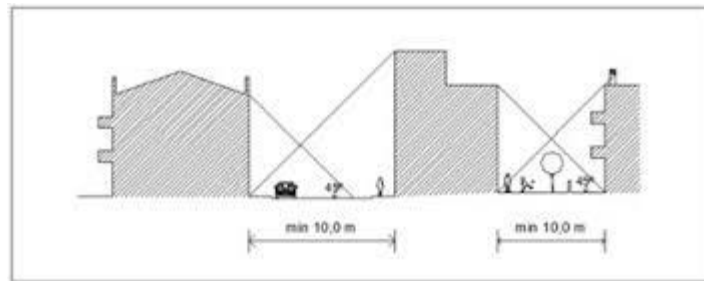


Figura 2.9 - Aplicação dos Artigos 59º-60º do RGEU (fonte: RMEU de Coimbra (2003))

Artigo 62º - “O logradouro (...) deverá ter em todos os seus pontos profundidade não inferior a metade da altura correspondente da fachada adjacente, medida na perpendicular a esta fachada no ponto mais desfavorável, com o mínimo de 6 metros e sem que a área livre e descoberta seja inferior a 40 metros quadrados. (...) Nos prédios de gaveto poderão dispensar-se as condições de largura e profundidade mínima de logradouro referidas no corpo deste artigo desde que fiquem satisfatoriamente asseguradas a iluminação, ventilação e insolação da própria edificação e das contíguas.”

2.3.2. Regulamentação nos planos territoriais municipais

O Plano Diretor Municipal (PDM) é um instrumento que estabelece o modelo de organização espacial, a estratégia de desenvolvimento do território municipal, a classificação do solo e as regras e parâmetros aplicáveis à ocupação, uso e transformação do solo. Na existência de Plano de Urbanização (PU) ou Planos de Pormenor (PP) o seguimento destas diretrizes deve ser prioritário face ao PDM.

Para a cidade de Lisboa, define-se no PDM que as especificações legais para a construção, ampliação e alteração de edifícios, diferenciam-se consoante a qualificação operativa e a qualificação funcional.

O PDM define que território do município corresponde na sua totalidade a solo urbanizado, este é classificado em duas categorias de acordo com a qualificação operativa:

- a) Espaços consolidados;
- b) Espaços a consolidar.

A qualificação funcional do solo distribui-se pelas seguintes categorias, em função da sua utilização dominante:

- a) Espaços centrais e residenciais;
- b) Espaços de atividades económicas;
- c) Espaços verdes;
- d) Espaços de uso especial de equipamentos;
- e) Espaços de uso especial de infraestruturas;
- f) Espaços de uso especial ribeirinho.

Os “espaços consolidados – espaços centrais e residenciais” (representados na Figura 2.10) podem ser classificados em quatro tipos de traçado (A, B, C, D). Estes espaços são classificados consoante características morfológicas, paisagísticas e ambientais, essencialmente os tipos de traçado têm uma correlação com a época histórica de construção (Vasques, 2017). “Traçados Urbanos A” – correspondem a traçados orgânicos (perfil espontâneo adaptado às condições topográficas do terreno com ruas estreitas e sinuosas) ou regulares (implantação de quarteirões retangulares relativamente torcidos de forma a se ajustarem à topografia e preexistências), por exemplo o Bairro Alto. “Traçados Urbanos B” – correspondem a traçados planeados dispostos em quarteirão pela aplicação de planos ortogonais, como resultado é característico ruas direitas e perpendiculares entre si, compreendem edifícios desde o século XVIII até ao presente (exemplos, avenidas novas e a baixa pombalina). “Traçados Urbanos C” – correspondem a traçados de implantação livre que abrangem tecidos urbanos essencialmente construídos na segunda metade do século XX, os edifícios são dispostos em forma de banda ou torre (e.g. Chelas e Telheiras). “Traçados Urbanos D” – geralmente correspondem a moradias isoladas, agrupadas em banda ou geminadas (e.g. Encarnação e Madre Deus).

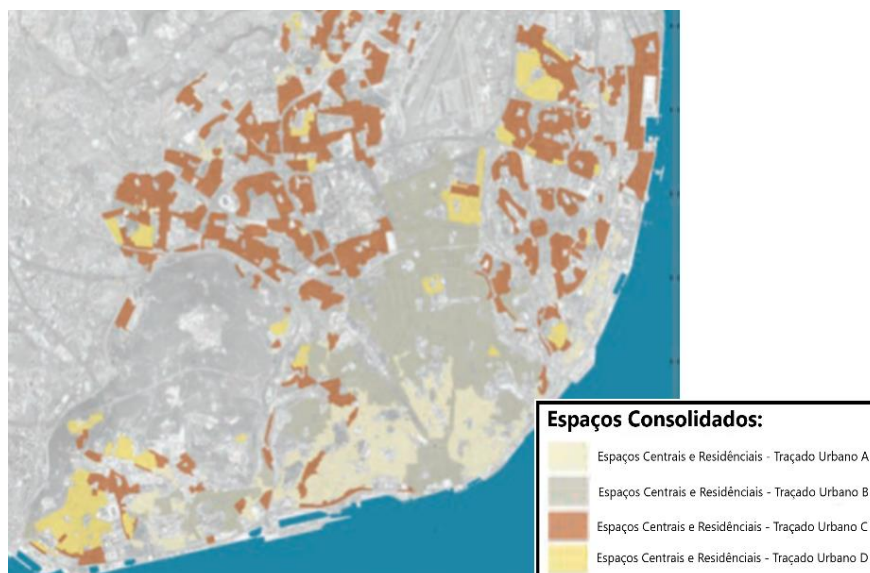


Figura 2.10 - Classificação traçado urbano relativo a espaços consolidados - espaços centrais e residenciais (fonte: adaptado do Manual Técnico PDM Lisboa (2013))

Como já referido, de acordo com a classificação em qualificação operativa e funcional, os requisitos legais para a construção, ampliação e alteração de edifícios são diferentes. A título de exemplo, para o solo classificado como “Espaços consolidados – Espaços centrais e residenciais – Traçado Urbano B”, sinalizam-se os artigos 42º, 44º e 45º.

O artigo 42º refere no número 4.a) que “altura máxima da edificação é a média das alturas dos edifícios da frente edificada do arruamento, entre duas transversais. Nesta operação, não é contabilizado o edifício mais alto, o mais baixo dessa frente e o prédio que é objeto da operação urbanística em causa”. O artigo 4º apresenta o conceito da “média da altura das fachadas”, este nota que “nas situações em que não seja possível recorrer à frente edificada entre duas transversais onde se localiza a operação, deve recorrer -se à frente edificada entre duas transversais mais próxima”. O número 4.b) indica exceções à alínea anterior. Entre elas a descrita na alínea i) “As obras em edifícios predominantemente de habitação, em parcela situada entre dois edifícios com uma altura da fachada superior à média das alturas das fachadas, em que pode ser adotada a altura do edifício confinante mais alto, desde que a superfície de pavimento acrescida se destine exclusivamente a habitação e 50% fique sujeita a valor máximo de renda ou preço de venda”.

A Figura 2.11 pretende ser um exemplo simples para demonstrar a aplicação do Artigo 42º do PDM de Lisboa. Considerando que o edifício alvo de operação urbanística é o edifício central, com uma altura $H_3 = 12m$, então a altura média dos edifícios da frente edificada do arruamento é $\frac{H_2+H_5}{2} = \frac{16+10}{2} = 13m$, H_3 é o edifício alvo de operação portanto é excluído do cálculo, assim como H_1 e H_4 porque são contabilizados como os edifícios mais baixo e mais alto. Assim tomando em consideração o número 4.a) do referido artigo,

o edifício poderia ser ampliado em 1m. No entanto, como existem dois edifícios adjacentes com uma altura de superior à média, então o edifício em análise pode tomar a altura do mais alto, 17m, tal como refere o número 4.b) do artigo.

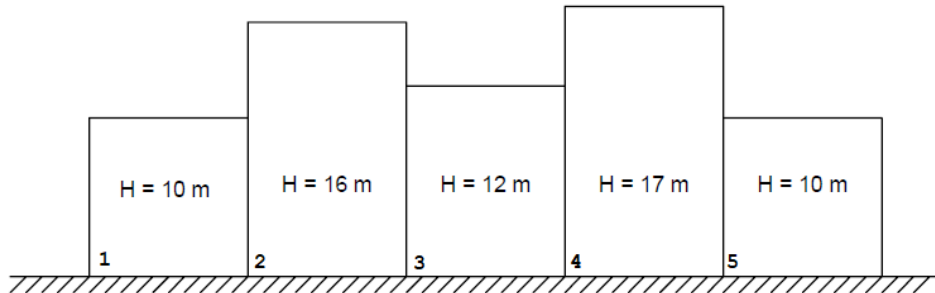


Figura 2.11 - Exemplo de uma frente edificada do arruamento entre duas transversais

O PDM de Lisboa tem o objetivo de requalificar os logradouros do ponto de vista ambiental, funcional e urbanístico. A introdução do parâmetro superfície vegetal ponderada (Svp) procura promover um equilíbrio na permeabilidade dos solos e a valorização de espaços intersticiais enquanto espaços verdes. Para este efeito a Svp é dada pela soma ponderada da área dos vários tipos de solo associado à regra Svp.

$$Svp = A + 0.6B + 0.3C \quad (1)$$

O tipo “A” é o terreno mais permeável em oposição à categoria “C”. A Figura 2.12 ilustra os diferentes tipos de solo e cobertos vegetais associados aos parâmetros A, B e C. Em que “A” corresponde ao valor unitário mínimo em m² de solo orgânico sem construção abaixo ou acima do solo, aplicável à área do logradouro ou à área líquida do loteamento acrescida das áreas cedidas para espaços verdes e de utilização coletiva; “B” é o valor unitário em m² de superfície vegetal sobre laje com um mínimo de 1 metro de terra viva/substrato, não incluindo a camada de drenagem; “C” é o valor unitário em m² de superfície vegetal sobre laje com um mínimo de 0,3 metros de terra viva/substrato, não incluindo a camada de drenagem, acrescido do valor unitário de poço ou trincheira de infiltração ou de cisterna de armazenamento de água, obtido a partir da equivalência do seu volume em área, em que 1 m³ corresponde para efeitos de cálculo a 1m²;

No seguimento do exemplo sobre “Espaços consolidados – Espaços centrais e residenciais – Traçado Urbano B”, seguem-se as seguintes restrições segundo o artigo 44º do PDM de Lisboa:

$$SVP \geq 0.5A_{log} \quad (2)$$

$$A \geq 0.2 A_{log} \quad (3)$$

Em que A_{log} é a área total do logradouro.

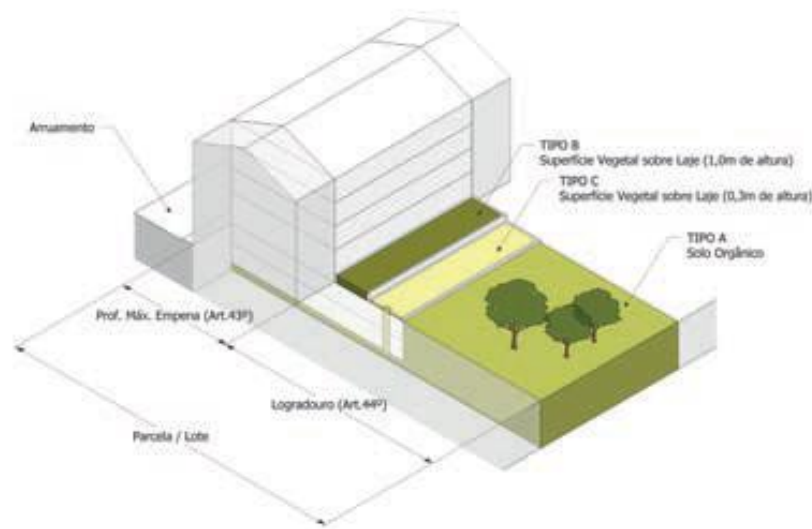


Figura 2.12 - Exemplo de edifício com os tipos de solo associados aos parâmetros A, B e C (fonte: Manual Técnico PDM Lisboa (2013))

A Carta Municipal do Património Edificado e Paisagístico refere imóveis, classificados ou em vias de classificação⁵, que tenham interesse urbanístico, arquitetónico ou cultural, tanto individualmente, como para o conjunto em que se integra. O Artigo 45º do PDM de Lisboa refere que, as intervenções devem privilegiar a sua conservação e valorização, não podendo realizar-se qualquer obra sem autorização e acompanhamento do órgão competente da administração central.

Por exemplo, o “Edifício na Av. de Berna, n.º 1 a 1A, e Av. da República” (referência 3251) está classificado como “imóvel classificado” na carta municipal anteriormente referida.

⁵ Imóveis que podem vir a ser classificados posteriormente à entrada em vigor do presente plano diretor municipal.

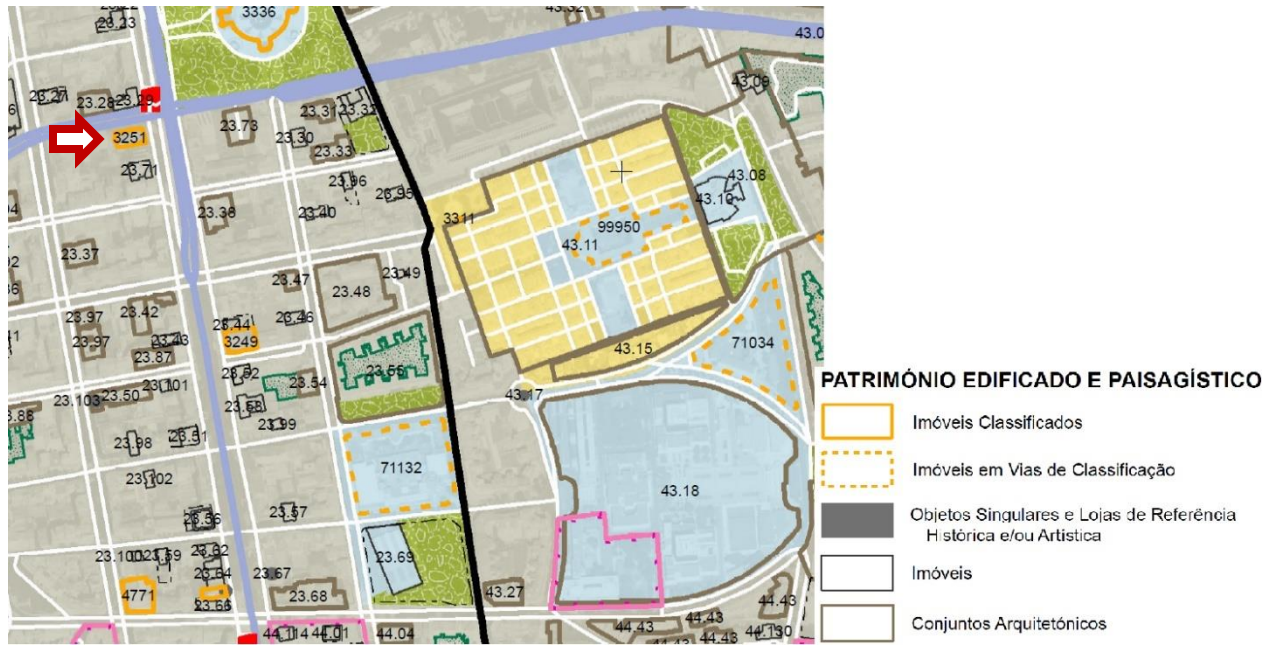


Figura 2.13 - Classificação do património edificado e paisagístico - pormenor na Av. da República (fonte: adaptado da Planta de Qualificação do espaço urbano – Lisboa ⁶)

⁶ Retirado de

https://www.lisboa.pt/fileadmin/cidade_temas/urbanismo/pdm/Planta_Qualificacao_do_Espaco_Urbano.pdf

3. Desenvolvimento e Implementação

3.1. Metodologia

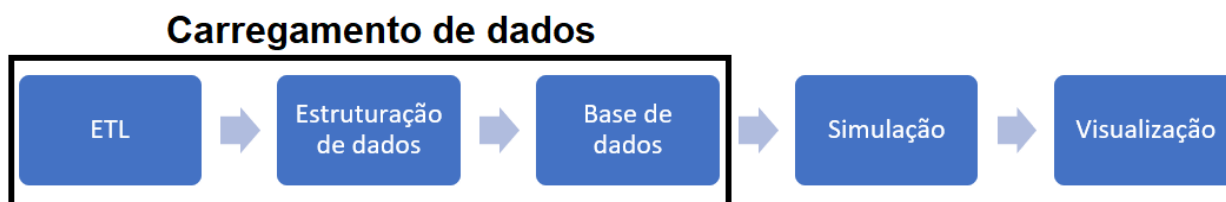


Figura 3.1 - Decomposição do método geral em 5 fases

Esta dissertação procura propor a aplicação de tecnologias de informação na área de urbanismo. A Figura 3.1 identifica um método generalizado das fases que podem ser seguidas para clarificar este processo.

A primeira fase do processo compreende a extração, transformação e carregamento de dados (ETL⁷), esta fase foca-se na obtenção e transferência de dados (possivelmente de diversas fontes) para um repositório comum, de forma a agrupar, validar e, se necessário, fazer conversões de formato tendo em vista a sua adequação ao próximo passo. A fase seguinte consiste em incluir os dados numa estrutura formal que permita armazenar e trocar dados com outras entidades. A fase “base de dados” consiste na importação dos registos para uma base de dados já com a estrutura e relações determinadas nas fases anteriores. A etapa “Simulação” é a mais genérica do processo, tem como objeto o desenvolvimento de uma ferramenta customizada de forma a responder ao âmbito particular da análise proposta. Por último, a visualização dos resultados da simulação.

3.2. Formalização do problema

De forma a particularizar a utilização de tecnologias de informação, propõe-se a aplicação num problema específico na área de urbanismo de modo a exemplificar o método anteriormente descrito. Assim, o problema proposto nesta dissertação consiste em simular a evolução da altura dos edifícios a longo prazo, através de um modelo 3D de cidade semântico.

⁷ ETL: “*Extract, Transform and Load*”, acrónimo comum usado em sistemas de bases de dados.

Segundo Biljecki (2017), para zonas densamente povoadas o espaço urbanizado é escasso, como tal, há a tendência de explorar os limites da regulamentação em vigor quanto à dimensão dos edifícios. Assim, a altura de novas edificações tende a corresponder ao limite máximo permitido.

Tendo em conta a legislação em vigor analisada no ponto 2.3, podemos aferir a seguinte variável objetivo:

$$H_i^t = \max(H_i^0; \min(R_{ij}^t)) \quad (4)$$

O resultado principal desta simulação é a variável H_i^t , a altura de um edifício i para um instante t . Para aferir esta variável será necessário comparar a altura inicial do edifício i , H_i^0 , com a máxima altura permitida para o edifício i , tendo em conta j restrições ao longo do tempo t , R_{ij}^t . Por exemplo, no caso de uma simulação com apenas ativas a R_{i1}^t e R_{i2}^t , se estas retornarem respetivamente os valores $15m$ e $20m$, será o valor de $15m$ o que é comparado com a altura atual do edifício.

Tendo em conta os artigos do regulamento do PDM de Lisboa e no RGEU apresentados no ponto 2.3, o quadro seguinte enumera as restrições que concorrem para a definição da altura do edifício. As especificações de implementação serão apresentadas no ponto 3.6.

Quadro 3.1 - Restrições relativas à altura do edifício e dimensão dos logradouros

R_{ij}^t	Restrição	Artigo
R_{i1}^t	R_Alaltura_Média	PDM de Lisboa – 42º, 4.a)
R_{i2}^t	R_Alaltura_Vizinho	PDM de Lisboa – 42º, 4.b)
R_{i3}^t	R_Distância_45	RGEU – 59º
R_{i4}^t	R_Distância_Tardoz	RGEU – 62º
R_{i5}^t	R_SVP	PDM de Lisboa – 44º
R_{i6}^t	R_Imóvel_Classificado	PDM de Lisboa – 45º

As restrições R_{i5}^t e R_{i6}^t são diferentes das restantes na medida em que estas não resultam num valor máximo de altura permitida por edifício, são validações que são necessário garantir durante o processo iterativo. A R_{i5}^t exige que o logradouro do edifício possua uma área mínima de solo tipo A (mais permeável), enquanto que a R_{i6}^t limita a lista de edifícios sujeita a ampliação.

3.3. Esquema de dados

Para a realização da simulação da altura dos edifícios, desenvolveu-se uma ferramenta em SQL totalmente personalizável com recurso às capacidades espaciais da base de dados PostGresSQL/PostGis. A solução desenvolvida é uma aplicação em código aberto na qual se privilegiou o desenvolvimento de funções

genéricas que permitam a sua utilização em tarefas semelhantes (por exemplo, extração de pontos médios de segmentos, criação de vetores ortogonais, entre outros).

A ferramenta foi desenvolvida dentro de um novo *schema* com o nome “*custom*” para que seja clara a separação das entidades da ferramenta 3DCityDB, das realizadas no desenvolvimento deste projeto. A Figura 3.2 ilustra o esquema relacional “*custom*” (sinalizado a verde) e a forma como este se integra com o modelo 3DCityDB (sinalizado a azul) já apresentado no ponto 2.1.1.1. O Quadro 3.2 descreve as características das tabelas do *schema* “*custom*”.

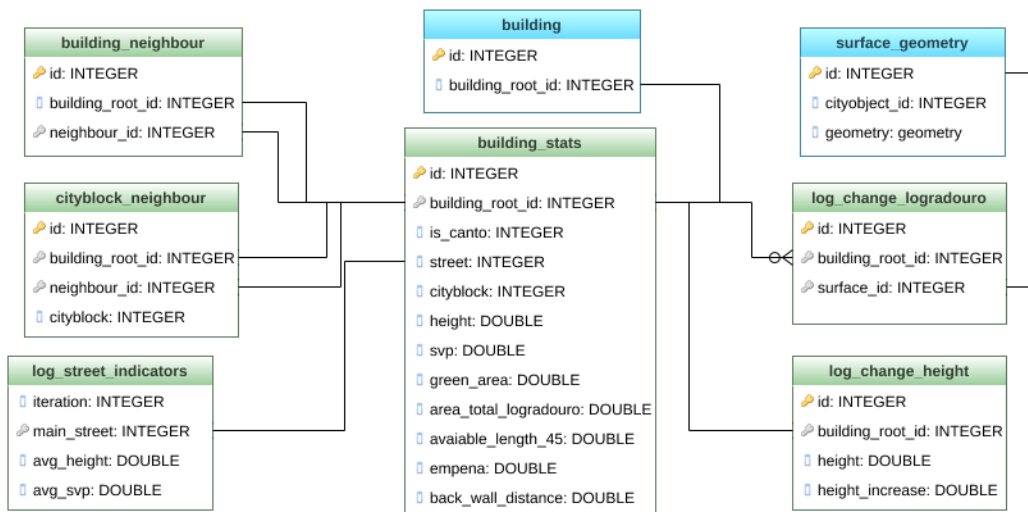


Figura 3.2 - Modelo relacional "custom" (elaborado pelo autor)

Quadro 3.2 - Descrição das tabelas "custom"

Tabela	Descrição
building_stats	Esta é a principal tabela da aplicação, serve para guardar a informação estática relativamente ao edifício (rua; quarteirão; altura; superfície vegetal ponderada, associado ao Artigo 44º do PDM de Lisboa; área Tipo A, associado à permeabilidade do solo; área do logradouro; distância livre segundo a regra dos 45º, associado ao Artigo 59º do RGEU; empena; distância livre em profundidade do logradouro, associado ao Artigo 62 do RGEU).
building_neighbour	Guarda os pares de edifícios que são adjacentes.
cityblock_neighbour	Guarda os pares de edifícios com lotes adjacentes.
log_change_height	Regista as alterações de altura dos edifícios ao longo da simulação.

Tabela	Descrição
log_change _logradouro	Regista partes de logradouros que mudam de solo tipo C para tipo A, ao longo da simulação.
log_street_indicators	Regista as alterações de altura (média) e superfície vegetal ponderada (média) ao longo da simulação. A informação é disponibilizada sobre a perspetiva da rua.

3.4. Carregamento e cálculo de dados

Para criar um modelo 3D de cidade semântico que permita simular a altura dos edifícios, é necessário definir a morfologia dos edifícios e o logradouro associado a cada edifício. A Figura 3.3 representa as várias fases caracterização do modelo.

A versão V0 corresponde à extração original de um ficheiro *SketchUp* da plataforma *Cadmapper*. Esta versão inclui o contorno da área de implantação dos edifícios, estradas e anexos aos edifícios. Os anexos foram removidos do modelo de forma manual.

É necessário complementar o modelo com informação quanto ao lote, uma vez que no ponto 2.3, foi apurada regulamentação na qual faz referência a propriedade como a dimensão do logradouro ou a sua natureza. Assim sendo, a versão V1 complementa a anterior introduzindo o conceito de lote, na qual é associado o logradouro ao edifício, desta forma, tanto o edifício principal como as parcelas do logradouro têm o mesmo identificador (*building_root_id*). Uma vez que a informação de cadastro correspondente a cada edifício não é pública, foi estimada a possível dimensão de cada logradouro associado a cada edifício. Sempre que possível definiu-se o logradouro de forma a ser um prolongamento do edifício em causa. Também foram criadas divisões do logradouro em pequenas parcelas e classificadas, através de um atributo genérico, em solo do tipo A ou C, conforme apresentado no ponto 2.3.2. Sendo que o tipo A corresponde a parcelas com solo com um elevado índice de permeabilidade enquanto que o solo do tipo C corresponde a um baixo índice. Esta classificação foi realizada tendo por base a imagem satélite da área de estudo (V0.1 e V0.2), sendo necessário a utilização da extensão *CityEditor*.

A versão V2 adiciona à prévia versão a altura de cada edifício. Para realizar esta tarefa é recolhido através do *Google Earth*, a elevação do edifício. Todavia, esta informação não está disponível para a totalidade dos edifícios. Assim sendo, foi estimada a altura de um edifício comparando com outro edifício vizinho no qual a informação esteja disponível. Os casos que não podem ser estimados por este método, foram estimados com base no número de pisos, em que cada piso corresponde a 3 metros de altura

A extensão *CityEditor* permite fazer uma caracterização semântica do modelo, para tal, fez-se o levantamento dos edifícios que correspondem a prédios de gaveto e dos edifícios presentes na Carta Municipal do Património Edificado e Paisagístico. Para os prédios de gaveto preencheu-se o atributo

“descrição” com o valor “canto”, no caso dos edifícios classificados no PDM Lisboa preencheu-se o atributo “class” com o código de referência identificado no PDM. Distinguiu-se ainda os edifícios pertencentes à área de estudo e aqueles que não pertencem, para tal, foi classificado cada edifício através de um atributo genérico com o valor “*buildings*” e “*buildings_out*” para o primeiro e segundo caso, respetivamente.

Por último, fez-se a caracterização semântica quanto à morada, uma vez que este é um processo manual, optou-se por atribuir uma morada apenas às ruas, para os edifícios este campo não foi preenchido.

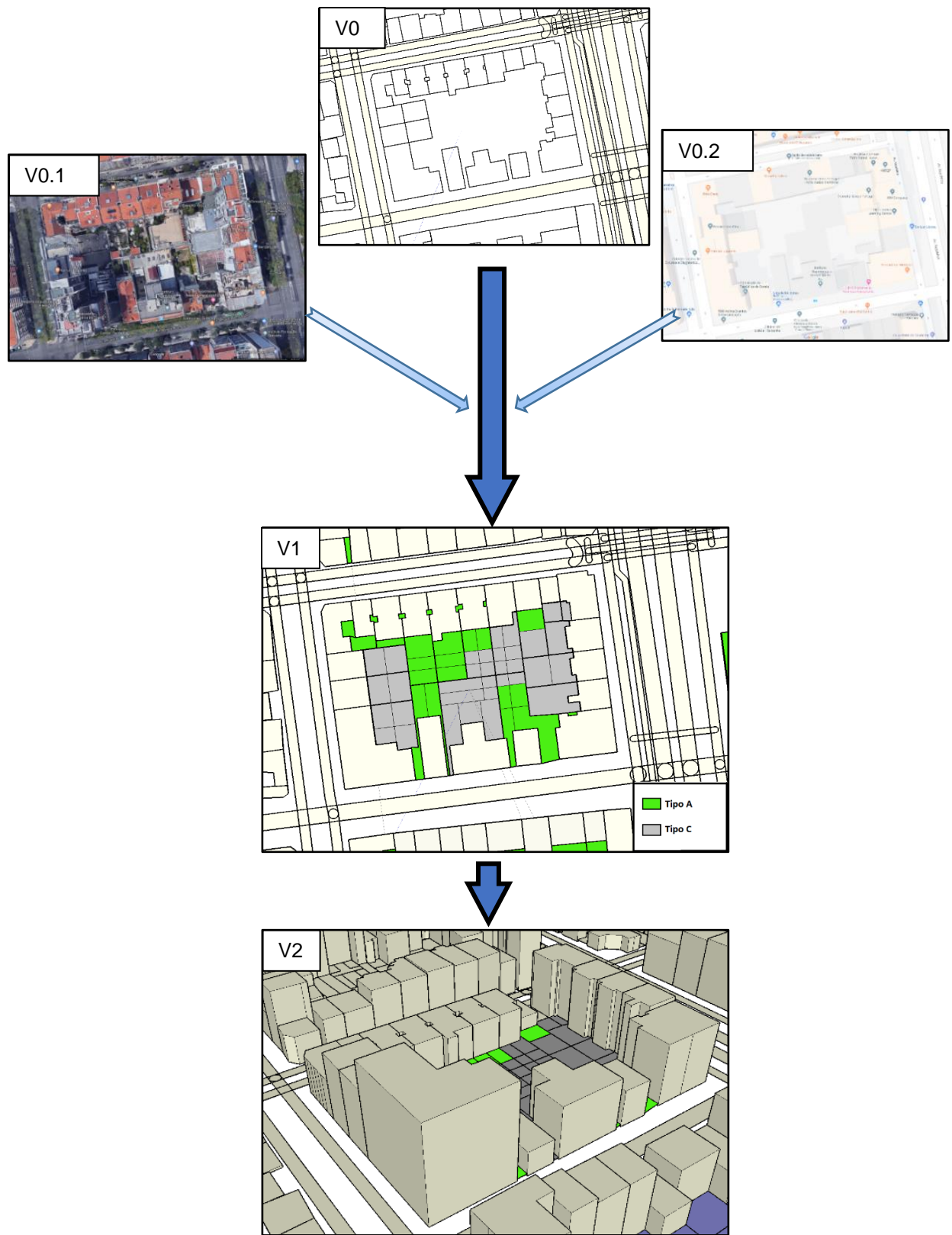


Figura 3.3 - Fases caracterização do modelo 3D de cidade semântico

Para importar este modelo para a base de dados, o próximo passo corresponde à segunda fase do método geral e consiste na exportação do modelo (em *SketchUp*) para um ficheiro GML, tal pode ser realizado a partir da extensão *CityEditor*. Na terceira fase do método geral realiza-se a importação do ficheiro GML para a base de dados através do módulo “*Import*” do 3DCityDB (Figura 2.4). Neste módulo é necessário escolher o ficheiro de origem e através do botão é realizada a importação. O resultado da importação do 3DCityDB, é o carregamento na base de dados da informação das tabelas apresentadas na Figura 2.5.

A partir deste momento, é possível fazer o cálculo de indicadores relativos à informação inicial dos edifícios, nomeadamente:

- a altura do edifício;
- a área do logradouro associada a cada edifício (classificada no tipo A ou C conforme a permeabilidade);
- a frente urbana em que o edifício se insere (detalhe no ponto 3.6.1);
- os pares de edifícios que são adjacentes ao próprio edifício (detalhe no ponto 3.6.2).

Este cálculo é executado através da função *custom.insert_main_startup*, e resulta no preenchimento das seguintes tabelas (ver Figura 3.4):

- *custom.building_stats*;
- *custom.building_neighbour*;
- *custom.cityblock_neighbour*

3.5. Simulação

O quarto passo do método geral consiste em executar uma simulação tendo em conta o problema analisado. Neste caso criou-se a função “*custom.main*” (apresentada em detalhe no ponto 3.6.8) de forma a aferir a altura do edifício (*custom.building_stats.height*), dada pelo resultado variável H_i^t na equação (4). Além deste campo são calculadas alterações relativas à natureza do logradouro e criado um registo de alterações quanto à altura do edifício e composição do seu logradouro. As tabelas e atributos que guardam informação dinâmica são as seguintes:

- | | |
|------------------------------------|--------------------------------|
| • <i>building_stats.height</i> | • <i>log_change_height</i> |
| • <i>building_stats.green_area</i> | • <i>log_change_logradouro</i> |
| • <i>building_stats.svp</i> | • <i>temporal_analysis</i> |

3.6. Funções Desenvolvidas

Nos subpontos seguintes formaliza-se a implementação das restrições indicadas anteriormente no Quadro 3.1. As quatro funções apresentadas são o ponto de contato entre o utilizador e a ferramenta desenvolvida, em termos de utilização da ferramenta, a Figura 3.4 clarifica o processo.

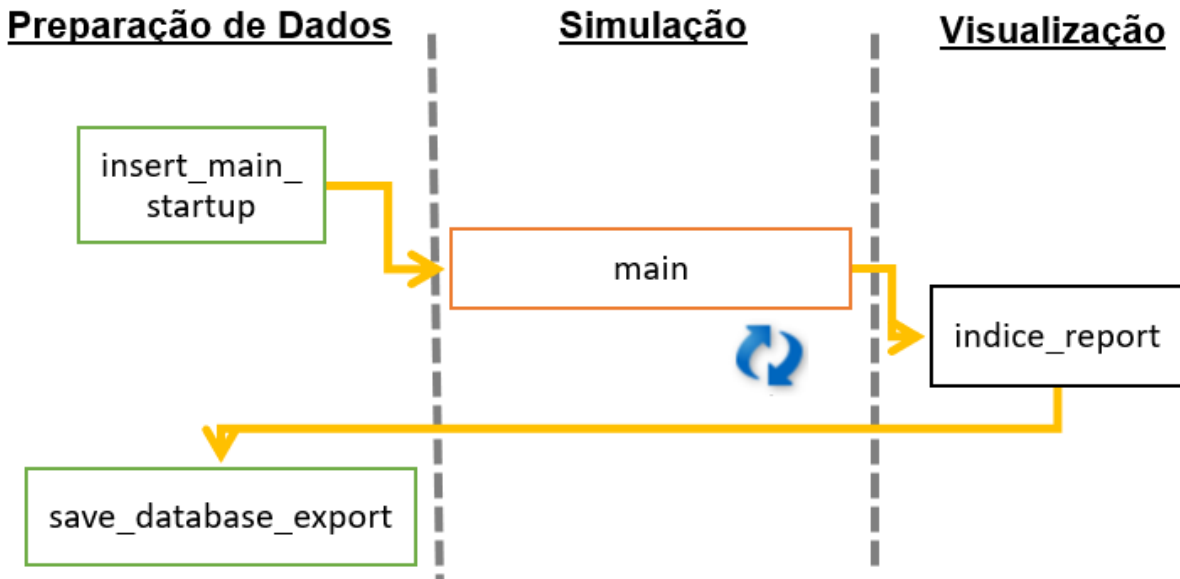


Figura 3.4 - Funções diretamente executadas pelo utilizador (funções de 1º nível)

O primeiro passo é executar a função `custom.insert_main_startup`, esta irá realizar as tarefas de preparação de dados, isto é, a partir da informação presente no *schema* “*citydb*” são realizadas operações que validam a qualidade dos dados e procede a correções caso seja necessário⁸. Adicionalmente são calculados indicadores que traduzem a situação inicial dos edifícios. A etapa seguinte é realizar a simulação, através da função `custom.main` (detalhada no ponto 3.6.8). Através da *view* `custom.indice_report` obtêm-se um quadro resumo com os resultados da simulação (a estrutura da *view* é apresentada posteriormente no Quadro 4.2). Nesta fase, é possível o utilizador alterar os parâmetros de entrada para realizar outra simulação. Uma vez terminada a simulação, a função `custom.prepare_database_export` materializa os resultados no *schema* “*citydb*”, isto é, altera a dimensão dos edifícios da sua altura inicial para a altura final calculada durante a simulação.

⁸ Por exemplo, durante a exportação do desenho (em *SketchUp*) para o formato *CityGML* são adicionados polígonos que compõem a palavra “*unlicensed*”, a função `custom.insert_main_startup` retira estes polígonos da base de dados. Este problema surge no caso da licença da extensão *CityEditor* ser experimental.

Em suma, foram desenvolvidas 39 funções que podem ser categorizadas em três níveis quanto ao grau de interação com o utilizador. O primeiro nível, contém 4 funções com as quais o utilizador interage diretamente. O segundo nível integra as funções que diretamente implementam as restrições do PDM de Lisboa e do RGEU (apresentadas anteriormente no Quadro 3.1), estas funções distinguem-se das restantes por terem o prefixo “rule_” e são explicadas de uma forma mais detalhada nos pontos seguintes. O último nível compreende as 32 funções que auxiliam a execução das restantes funções apresentadas, tanto na preparação de dados como na simulação, Quadro 3.3.

Quadro 3.3 - Lista de funções auxiliares desenvolvidas

#	Nome	Descrição
1	make_extended_line	A partir de dois pontos iniciais cria um segmento de reta prolongado num determinado em X unidades para cada um dos lados.
2	fix_height_specific_reference	Identifica edifícios com mais do que uma face à mesma cota e corrige o mesmo.
3	fix_surface_floating	Garante a qualidade dos dados: translada edifícios "flutuantes" para a cota zero; elimina superfícies inferiores a 1m ² ; remove polígonos que perfazem a palavra “unlicensed”.
4	insert_building_neighbour	Insera os pares de relações entre cada edifício e o seu vizinho (edifício confinante), isto é, procura outros edifícios que façam fronteira com o edifício em análise.
5	insert_building_block	Insera os pares de relações entre cada lote e os seus vizinhos.
6	insert_building_neighbour_all	Classifica todos os edifícios quanto aos vizinhos do edifício e do lote.
7	get_cityblock	Retorna o <i>id</i> do quarteirão do edifício.
8	get_main_surface_id	Retorna o <i>id</i> da superfície do edifício ao qual corresponde a área de implantação.
9	get_nearest_road_id	Retorna o <i>id</i> da rua mais próxima de uma determinada superfície.
10	get_nearest_road_name	Retorna o nome da rua mais próxima de uma determinada superfície.
11	get_second_nearest_road_id	Retorna o <i>id</i> da segunda rua mais próxima de uma determinada superfície.
12	get_area_type	Retorna a área de logradouro de um determinado tipo.
13	get_svp	Retorna o valor de Superfície Vegetal Ponderada por edifício.
14	get_empena_length	Retorna a empena do edifício.
15	get_total_area_garden	Retorna a área total do logradouro.

#	Nome	Descrição
16	get_is_canto	Identifica se um edifício é um prédio de gaveto.
17	get_avg_height_by_cityblock	Retorna a média de alturas da frente urbana em que o edifício está inserido.
18	get_back_wall	Retorna o centro da linha que delimita o polígono da superfície principal mais afastada da rua.
19	insert_backup	Faz o backup das tabelas <i>citydb</i> para que se possa guardar a situação inicial em cada simulação.
20	replace_backup_image	Repõe a imagem inicial das características da área de estudo.
21	simulation_commit_height_changes	Materializa o resultado da simulação. Altera a dimensão de cada objeto associado à altura dos edifícios.
22	prepare_database_export	Materializa o resultado da simulação. Elimina todas as superfícies da base de dados que não sejam alvo de análise (Ex: estradas e edifícios fora da área de estudo).
23	get_point_front_or_back_wall	Retorna um <i>array</i> com os dois pontos do segmento de reta que definem a frente ou tardoz do edifício. Corresponde ao segmento do edifício mais perto ou mais afastado da rua selecionada.
24	get_ortogonal_vector	Retorna um vetor ortogonal à fachada principal ou tardoz do edifício, que aponta para o exterior do edifício.
25	get_ortogonal_line_front_or_back_wall	Retorna uma linha ortogonal à frente ou tardoz do edifício.
26	insert_building_stats	Calcula os parâmetros da tabela <i>custom.building_stats</i> para um edifício.
27	insert_building_stats_all	Calcula os parâmetros da tabela <i>custom.building_stats</i> para todos os edifícios.
28	c_log_verde	Retorna o identificador do logradouro do Tipo A.
29	c_log_cinza	Retorna o identificador do logradouro do Tipo C.
30	c_canto	Retorna o identificador do prédio de gaveto.
31	c_building	Retorna o identificador do edifício que pertence à área de estudo.
32	c_building_out	Retorna o identificador do edifício que não pertence à área de estudo.

Os pontos seguintes explicam o modo de implementação dos conceitos mais relevantes para o utilizador da ferramenta. Por motivos de simplificação, nem todos os processos são extensamente detalhados na dissertação, no entanto, o leitor pode compreender a forma como as restantes tarefas foram implementadas através do código fonte.

3.6.1. Frente urbana

Para o cálculo da altura média dos edifícios da frente edificada do arruamento entre duas transversais, doravante, frente urbana, em primeiro lugar é necessário identificar a frente urbana em que cada edifício se insere. Para tal, uma possível forma de classificação é diferenciar através de duas características: a rua mais próxima e o quarteirão a que este pertence. Deste modo, sabe-se que dois edifícios pertencem à mesma frente urbana quando o par, “rua mais próxima” – “quarteirão”, é o mesmo. Caso apenas uma das características seja comum a dois edifícios, estes pertencem a frentes urbanas diferentes.

De forma a identificar a rua mais próxima foi desenvolvida a função *custom.get_nearest_road_id* (Figura 3.5). Primeiro são identificadas todas as superfícies referentes a vias de comunicação, ou seja, as superfícies que têm a morada preenchida (linhas 8 a 15 da Figura 3.5). Na linha 8 é identificada a superfície em análise, posteriormente, é criada uma lista de superfícies referentes a vias de comunicação ordenada pela distância à superfície de referência (linha 16). Apenas é escolhido o primeiro elemento dessa lista (linha 17) e retornado esse elemento (linha 19).

```
1 CREATE OR REPLACE FUNCTION custom.get_nearest_road_id
2 (p_surface_id IN citydb.surface_geometry.id%TYPE)
3 RETURNS citydb.surface_geometry.id%TYPE AS $$
4 DECLARE
5 v_closest_surface citydb.surface_geometry.id%TYPE;
6 BEGIN
7 SELECT v2.id INTO STRICT v_closest_surface
8 FROM ( SELECT id, geometry FROM citydb.surface_geometry WHERE id = p_surface_id ) v1
9 , ( SELECT s.id, s.geometry
10 FROM citydb.surface_geometry s
11 , citydb.building b
12 , citydb.address_to_building ab
13 WHERE s.cityobject_id = b.id
14 AND b.building_root_id = ab.building_id
15 ) v2
16 ORDER BY ST_Distance(v1.geometry, v2.geometry)
17 LIMIT 1;
18
19 RETURN v_closest_surface;
20 END;
21 $$ LANGUAGE plpgsql;
```

Figura 3.5 - Especificação da função *custom.get_nearest_road_id*

De forma a identificar o quarteirão, poderiam ser usadas várias abordagens. A primeira seria, na fase de extração de dados, classificar manualmente cada edifício com um atributo que possa distinguir os edifícios quanto a esta característica. No entanto, esta abordagem é muito tediosa e é complexo escalar a mesma solução para conjuntos com uma maior dimensão de dados. Outra alternativa é criar *clusters* de edifícios que correspondem a quarteirões, em que a particularidade de cada quarteirão é que todos os lotes de edifícios estão conectados, por outras palavras, a distância entre dois lotes deve ser zero. Para implementar esta solução, a primeira fase é criar uma lista de pares de edifícios nestas condições. Sempre que é introduzido um novo par na lista é atribuído também um identificador de quarteirão, para tal, verifica-

se a existência de alguma ligação entre o par a analisar e os registos já inseridos; quando existe ligação atribui-se o mesmo identificador de quarteirão, caso contrário é atribuído um novo.

A Figura 3.6 ilustra um exemplo simples de 7 lotes agrupados em 2 quarteirões e o processo de atribuição de quarteirões aos pares adjacentes. O par “1-2” é o primeiro registo, este é classificado com o quarteirão “A”. Quando o par “1-4” é inserido verifica-se que o lote 1 já está classificado como “A”, então o registo também é classificado como “A”. No terceiro registo, “3-5”, ainda nenhum dos lotes “3” ou “5” está classificado, portanto é atribuído o valor “B” e assim sucessivamente.

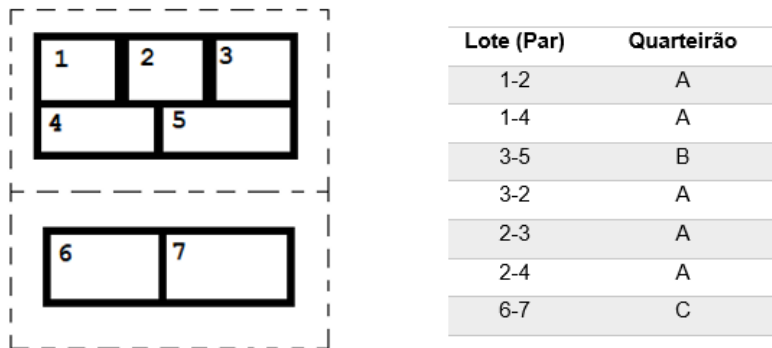


Figura 3.6 - Exemplo de classificação de quarteirões

Após este processo podem existir edifícios que pertencem a diferentes quarteirões, portanto, a segunda fase consiste em reclassificar os lotes com mais do que um quarteirão associado. No exemplo, o lote 3 tem dois quarteirões associados, sendo assim, deve-se alterar o cluster do registo “3-5” para “A”. Em suma, neste exemplo, os lotes ficam classificados em apenas dois quarteirões. No código fonte⁹, as funções *custom.insert_building_block* (linha 343) e *custom.insert_building_neighbour_all* (linha 412) detalham, respetivamente, a implementação da primeira e segunda fase do processo descrito.

3.6.2. Pares de edifícios adjacentes

A identificação dos edifícios adjacentes advém da necessidade de identificar se um edifício é confinado por dois edifícios mais altos (Artigo 42º, 4.b do PDM de Lisboa). A tabela *custom.building_neighbour* guarda os pares de edifícios adjacentes, sendo que esta tabela é preenchida através da função *custom.insert_building_neighbour* (Figura 3.7). Para tal, são identificados os casos nos quais a distância

⁹ Disponível em

<https://github.com/UrbanSimulCityGML/LisbonBuildingsMaxHeight/blob/master/custom%20functions.sql>

entre a base do edifício em análise e a base de outro edifício seja zero, ou seja, a distância entre os polígonos (da base do edifício) é nula. Assim, o primeiro passo, é identificar a superfície principal¹⁰ do edifício (linha 12 da Figura 3.7), que é comparada em termos de proximidade com as superfícies principais dos restantes edifícios (linhas 13 a 20); de seguida filtra-se pelo atributo genérico de forma a garantir que apenas são selecionadas superfícies que pertencem a edifícios na área de estudo (linha 18). Na linha 21, são filtradas as superfícies que são adjacentes à superfície principal do edifício. Posteriormente são iterados todos os registos da lista de edifícios adjacentes, e para cada um dos pares, verifica-se se já foram anteriormente inseridos na tabela *custom.building_neighbour* (linhas 23 a 26), caso não tenha sido, é inserido o par do edifício em análise e edifício adjacente (linhas 28 a 31).

```

1 CREATE OR REPLACE FUNCTION custom.insert_building_neighbour ( p_building_root_id IN
  citydb.building.building_root_id%TYPE)
2 RETURNS void AS $$
3 DECLARE
4     v_surface_id    citydb.surface_geometry.id%TYPE;
5     v_exist_record  smallint;
6     r_neighbour     record;
7 BEGIN
8     v_surface_id := custom.get_main_surface_id(p_building_root_id);
9
10    FOR r_neighbour IN (
11        SELECT v2.building_root_id id
12        FROM ( SELECT id, geometry FROM citydb.surface_geometry WHERE id = v_surface_id) v1
13             , ( SELECT s.id, s.geometry, b.building_root_id
14                FROM surface_geometry s, building b, cityobject_genericattrib g
15                WHERE s.cityobject_id = b.id
16                  AND b.id = g.cityobject_id
17                  AND s.id != v_surface_id
18                  AND g.strval = custom.c_building()
19                  AND ST_Area(s.geometry) != 0
20                  AND ST_ZMax(s.geometry) = 0 ) v2
21        WHERE ST_Distance(v1.geometry, v2.geometry) = 0
22    ) LOOP
23        SELECT count(1) INTO v_exist_record
24        FROM custom.building_neighbour
25        WHERE building_root_id = p_building_root_id
26              AND neighbour_id = r_neighbour.id;
27
28        IF v_exist_record = 0 THEN
29            INSERT INTO custom.building_neighbour (building_root_id, neighbour_id)
30            VALUES (p_building_root_id, r_neighbour.id);
31        END IF;
32    END LOOP;
33 END;
34 $$ LANGUAGE plpgsql;

```

Figura 3.7 - Especificação da função *custom.insert_building_neighbour*

¹⁰ A superfície principal do edifício é o polígono que representa a base do edifício. Para identificar esta superfície são filtrados os polígonos do edifício que não estejam à cota zero e que possuam uma área em planta diferente de zero.

3.6.3. Restrição “R_Altura_Média”

Este ponto e os seguintes apresentam funções que diretamente implementam as restrições do PDM de Lisboa e do RGEU. A função `custom.get_avg_height_by_cityblock` (linha 11 da Figura 3.8) calcula a altura média da frente urbana em que o edifício se insere, sendo que são excluídos da análise o edifício mais alto e mais baixo da frente urbana, assim como o próprio edifício. Nos casos em que a frente urbana contabiliza três ou menos edifícios, não é aplicável este método, assim, deve recorrer-se à frente urbana mais próxima. Para implementar este requisito identifica-se as restantes frentes urbanas e seleciona-se um edifício representativo dessa frente (linhas 22 a 28). Posteriormente compara-se a distância entre as superfícies principais o edifício em análise e os edifícios selecionados em cada frente urbana (linha 29). Uma vez criada uma lista ordenada pela distância entre o edifício em análise e os edifícios selecionados de outras frentes urbanas, são iterados os registos até que seja encontrada uma frente urbana com mais de três edifícios (linhas 32 e 33).

```
1 CREATE OR REPLACE FUNCTION custom.rule_avg_height (
2     p_building_root_id IN custom.building_stats.building_root_id%TYPE
3     , p_street          IN custom.building_stats.street%TYPE
4     , p_cityblock      IN custom.building_stats.cityblock%TYPE )
5 RETURNS custom.building_stats.height%TYPE AS $$
6 DECLARE
7     v_avg_height        custom.building_stats.height%TYPE;
8     v_main_surface      citydb.surface_geometry.id%TYPE;
9     i                   RECORD;
10 BEGIN
11     v_avg_height := custom.get_avg_height_by_cityblock(p_building_root_id, p_street, p_cityblock);
12
13     IF v_avg_height IS NULL THEN
14         v_main_surface := custom.get_main_surface_id(p_building_root_id);
15
16         -- Cria uma lista de edificios representativos de uma frente urbana ordenado pela distância de
17         -- cada edificio e o edificio em análise
18         FOR i IN (
19             SELECT v2.street, v2.cityblock, ST_Distance(v0.geometry, v2.geometry)
20             FROM ( SELECT id, geometry FROM citydb.surface_geometry WHERE id = v_main_surface ) v0
21                 , ( SELECT vl.*, s.geometry
22                     FROM surface_geometry s
23                     , ( SELECT custom.get_main_surface_id(min(building_root_id)) main_surface_id
24                         , street, cityblock
25                         FROM custom.building_stats
26                         WHERE street != p_street
27                           AND cityblock != p_cityblock
28                           GROUP BY street, cityblock ) v1
29                     WHERE s.id = v1.main_surface_id ) v2
30             ORDER BY ST_Distance(v0.geometry, v2.geometry)
31         ) LOOP
32             -- volta-se a calcular a média da altura dos edificios da nova frente urbana seleccionada
33             v_avg_height := custom.get_avg_height_by_cityblock(p_building_root_id, i.street,
34                 i.cityblock);
35             EXIT WHEN v_avg_height IS NOT NULL;
36         END LOOP;
37     END IF;
38     RETURN COALESCE(v_avg_height, 0);
39 END;
40 $$ LANGUAGE plpgsql;
```

Figura 3.8 - Especificação da função `custom.rule_avg_height`

3.6.4. Restrição “R_Altura_Vizinho”

Segundo esta regra, a altura máxima de um edifício é dada pela altura do maior edifício adjacente. A implementação desta regra (Figura 3.9) tira partido do preenchimento da tabela *custom.building_neighbour* durante a fase de preparação de dados. Para obter a altura do maior edifício adjacente, compara-se o edifício em análise com os edifícios vizinhos (linhas 14 a 19 da Figura 3.9), e verifica-se se o edifício em análise tem dois edifícios vizinhos com uma altura superior ao próprio (linhas 9 e 10) e à média de alturas da frente urbana (linha 20). O resultado será a altura do maior edifício adjacente desde que se verifiquem as condições acima descritas. Tendo em conta a condicionante do artigo 42º do PDM, assume-se que o acréscimo de área de construção fique destinado exclusivamente à habitação e que 50% desta área esteja sujeita a um preço de venda.

```
1 CREATE OR REPLACE FUNCTION custom.rule_neighbour_height (
2   p_building_root_id IN custom.building_stats.building_root_id%TYPE )
3 RETURNS custom.building_stats.height%TYPE AS $$
4 DECLARE
5   v_neighbour_max_height custom.building_stats.height%TYPE;
6 BEGIN
7   SELECT CASE
8     -- Tem que ter dois edifícios vizinhos com a altura superior ao próprio edifício
9     WHEN SUM(CASE WHEN bs.height >= h.height THEN 1 ELSE 0 END ) = 2
10    THEN MAX(bs.height)
11    ELSE 0
12    END max_height
13 INTO STRICT v_neighbour_max_height
14 FROM custom.building_stats bs
15   , custom.building_neighbour bn
16   , ( SELECT height FROM custom.building_stats WHERE building_root_id =
17     p_building_root_id ) h
17 WHERE 1=1
18    AND bs.building_root_id = bn.neighbour_id
19    AND bn.building_root_id = p_building_root_id
20    AND custom.rule_avg_height(bs.building_root_id, bs.street, bs.cityblock) <
21    bs.height;
22 -- Apenas são considerados edifícios vizinhos com a altura superior à altura média da frente
23 urbana
24 RETURN v_neighbour_max_height;
END;
```

Figura 3.9 - Especificação da função *custom.rule_neighbour_height*

3.6.5. Restrição “R_Distância_45”

Pretende-se agora determinar a distância livre na frente do edifício (Artigo 59º do RGEU), este é um processo mais complexo, no qual a Figura 3.13 sinaliza pontos auxiliares. O primeiro passo é determinar o ponto médio sobre a fachada frontal do edifício (ponto A), este será o ponto médio do segmento de reta entre pontos F e G. Os pontos F e G, pertencem ao segmento de reta do perímetro do edifício que está

mais próximo da rua em frente ao edifício, a função `custom.get_point_front_or_back_wall` (Figura 3.10) retorna esses mesmos pontos. Para obter estes pontos, primeiro são extraídos os segmentos de reta do polígono principal (linhas 18 a 20 da Figura 3.10), e posteriormente os dois pontos extremos de cada segmento de reta (linhas 15 e 16). A próxima fase consiste em criar uma lista ordenada pela distância entre os pontos de cada segmento e a superfície pertencente à rua mais próxima, o resultado da função será apenas o primeiro segmento da lista (linhas 23 e 24).

```

1 CREATE OR REPLACE FUNCTION custom.get_point_front_or_back_wall(
2     p_main_surface IN citydb.surface_geometry.id%TYPE
3     , p_side_wall  IN VARCHAR(30)
4     , p_road_id    IN citydb.surface_geometry.id%TYPE )
5 RETURNS geometry[] AS $$
6 DECLARE
7     v_start_point geometry;
8     v_end_point   geometry;
9     points        geometry[];
10 BEGIN
11     IF p_side_wall = 'front' THEN
12         SELECT aux.sp start_point, aux.ep end_point
13         INTO v_start_point, v_end_point
14         FROM (SELECT -- extrai os 2 pontos do segmento de reta
15                 ST_PointN(geom, generate_series(1, ST_NPoints(geom)-1)) as sp
16                 , ST_PointN(geom, generate_series(2, ST_NPoints(geom) )) as ep
17             FROM ( -- extrai os segmentos de reta do polígono principal
18                 SELECT (ST_Dump(ST_Boundary(geometry))).geom
19                 FROM citydb.surface_geometry
20                 WHERE id = p_main_surface
21                 ) AS linestrings ) AS aux
22             , ( SELECT geometry FROM citydb.surface_geometry WHERE id = p_road_id ) road
23         ORDER BY ST_Distance(road.geometry, aux.sp) + ST_Distance(road.geometry, aux.ep)
24         -- Ordena pela soma das distâncias entre cada um dos pontos do segmento até à rua a analisar
25         LIMIT 1;
26     ELSIF p_side_wall = 'back' THEN
27         SELECT aux.sp start_point, aux.ep end_point
28         INTO v_start_point, v_end_point
29         FROM ( SELECT ST_PointN(geom, generate_series(1, ST_NPoints(geom)-1)) as sp,
30                 ST_PointN(geom, generate_series(2, ST_NPoints(geom) )) as ep
31             FROM ( SELECT (ST_Dump(ST_Boundary(geometry))).geom
32                 FROM citydb.surface_geometry
33                 WHERE id = p_main_surface
34                 ) AS linestrings ) AS aux
35             , ( SELECT geometry FROM citydb.surface_geometry WHERE id = p_road_id ) road
36         ORDER BY ST_Distance(road.geometry, aux.sp) + ST_Distance(road.geometry, aux.ep) DESC
37         -- Neste caso a ordem é inversa
38         LIMIT 1;
39     ELSE
40         RAISE EXCEPTION 'custom.get_point_front_or_back_wall(%, %): Side from main_surface
not enabled', p_main_surface, p_side_wall USING HINT = 'Choose between ''front'' or ''back''';
41     END IF;
42
43     points:= ARRAY( SELECT v_start_point UNION ALL SELECT v_end_point );
44     RETURN points;
45 END;

```

Figura 3.10 - Especificação da função `custom.get_point_front_or_back_wall`

Uma vez identificado o ponto médio sobre a fachada frontal do edifício (ponto A), o segundo passo é calcular um vetor perpendicular ao segmento \overline{FG} com norma unitária. Genericamente um vetor \vec{u} em R^2 é

representado pelas coordenadas $(x_1; y_1)$. Assim um vetor perpendicular \vec{v} $(x_2; y_2)$, tem de verificar a seguinte condição:

$$\vec{v} \cdot \vec{u} = 0$$

Considerando R^2 , existem duas soluções com norma unitária.

O conjunto de soluções do vetor $\vec{v} = \begin{cases} \frac{(-y_1; x_1)}{\|\vec{u}\|} \\ \frac{(y_1; -x_1)}{\|\vec{u}\|} \end{cases}$

Em que a norma de um vetor é: $\|\vec{u}\| = \sqrt{x_1^2 + y_1^2}$

As duas soluções têm a mesma direção e diferem no sentido do vetor. Uma delas é no sentido da frente do edifício enquanto que a outra para o tardoz. Quando ao ponto A é aplicada cada uma das soluções anteriores, são obtidos dois pontos, ponto B e H. A solução pretendida é aquela a que a distância entre estes dois pontos e a rua mais próxima é menor (ponto B), a função *custom.get_ortogonal_vector*¹¹ implementa esta lógica.

O terceiro passo é prolongar o segmento de reta \overline{AB} de forma a que intersecte com os edifícios nesta direção, dando origem à reta C. A função *custom.get_ortogonal_line_front_or_back_wall* permite extrair a reta C (Figura 3.11). Para obter um vetor ortogonal à fachada principal do edifício é executada a função *custom.get_ortogonal_vector* (linha 12 da Figura 3.11). De seguida, calcula-se o ponto médio do da fachada principal (linha 18), posteriormente, a partir desse ponto é projetada uma linha com a direção do vetor obtido (linha 14) com um comprimento de 100 metros¹².

```
1 CREATE OR REPLACE FUNCTION custom.get_ortogonal_line_front_or_back_wall (
2     p_main_surface IN citydb.surface_geometry.id%TYPE
3     , p_side_wall   IN VARCHAR(30)
4     , p_road_id    IN citydb.surface_geometry.id%TYPE
5 ) RETURNS geometry AS $$
6 DECLARE
7     v_result      geometry;
8     v_vetor_x     numeric;
9     v_vetor_y     numeric;
10 BEGIN
11     SELECT a.arr[1], a.arr[2] INTO STRICT v_vetor_x, v_vetor_y
12     FROM (SELECT custom.get_ortogonal_vector(p_main_surface, p_side_wall, p_road_id) arr) a;
13
14     SELECT ST_MakeLine(c.midpoint_wall, c.end_point_ortogonal) INTO v_result
15     FROM ( -- ponto ortogonal(à frente do edifício) a uma distância de 100 unidades
```

¹¹ A especificação da função *custom.get_ortogonal_vector* está disponível no código fonte na linha 1294.

¹² Assume-se que o edifício na frente do edifício em análise deve estar a uma distância inferior a 100m.


```

16      SELECT      ST_Translate(b.midpoint_wall,    v_vetor_x*100,    v_vetor_y*100,    0)
end_point_ortogonal
17      , b.midpoint_wall
18      FROM (      SELECT ST_Centroid(ST_MakeLine( a.arr[1], a.arr[2])) midpoint_wall
19      FROM (      SELECT custom.get_point_front_or_back_wall(p_main_surface,
p_side_wall, p_road_id) arr
20      ) a
21      ) b
22      ) c;
23
24      RETURN v_result;
25  END;

```

Figura 3.11 - Especificação da função *custom.get_ortogonal_line_front_or_back_wall*

Posteriormente são contabilizados os edifícios que intersectam a reta C. O resultado do exercício, é a distância entre o ponto A e o ponto E, sendo o ponto E o ponto de outro edifício sobre a reta C mais próximo do edifício em análise. A função *custom.rule_distance_45* (Figura 3.12), descreve a implementação completa desta regra usando as funções anteriormente descritas. Assim, identifica-se a superfície principal do edifício mais perto, segundo um vetor ortogonal à frente do edifício em análise (linhas 53 a 67 da Figura 3.12). Posteriormente, calcula-se a distância entre o edifício em análise e o edifício anteriormente identificado segundo o vetor ortogonal (linhas 70 a 75).

```

1  CREATE OR REPLACE FUNCTION custom.rule_distance_45 (
2      p_building_root_id  IN citydb.building.building_root_id%TYPE
3      , p_is_canto        IN custom.building_stats.is_canto%TYPE )
4  RETURNS custom.building_stats.avaiable_length_45%TYPE AS $$
5  DECLARE
6      v_main_surface_id    citydb.surface_geometry.id%TYPE;
7      v_closest_road      citydb.surface_geometry.id%TYPE;
8      v_2nd_closest_road  citydb.surface_geometry.id%TYPE;
9      v_geom_orto_line    geometry;
10     v_geom_mid_point_wall geometry;
11     v_x_ortogonal_vector NUMERIC;
12     v_y_ortogonal_vector NUMERIC;
13     v_surface_front_building citydb.surface_geometry.id%TYPE;
14     v_side_wall          VARCHAR(30) := 'front';
15     v_extend_lenght      NUMERIC := 30; -- prolongar os segmentos de reta (que correspondem
à fachada) em 30m para cada um dos lados
16     v_result            custom.building_stats.avaiable_length_45%TYPE;
17  BEGIN
18     v_main_surface_id := custom.get_main_surface_id(p_building_root_id);
19     v_closest_road := custom.get_nearest_road_id(v_main_surface_id);
20     v_2nd_closest_road := custom.get_second_nearest_road_id(v_main_surface_id);
21
22     IF p_is_canto = 0 THEN
23         SELECT custom.get_ortogonal_line_front_or_back_wall(v_main_surface_id, v_side_wall,
v_closest_road) INTO v_geom_orto_line;
24
25         -- obtém o ponto central da fachada principal
26         SELECT ST_Centroid(ST_MakeLine( a.arr[1], a.arr[2] ) ) INTO v_geom_mid_point_wall
27         FROM ( SELECT custom.get_point_front_or_back_wall(v_main_surface_id, v_side_wall,
v_closest_road) arr ) a;
28     ELSE
29         -- obtém o ponto de interseção no prolongamento das duas fachadas do edifício -- Ponto P
30         SELECT ST_Intersection(custom.make_extended_line(a.arr[1], a.arr[2],
v_extend_lenght), custom.make_extended_line(b.arr[1], b.arr[2], v_extend_lenght))
31         INTO v_geom_mid_point_wall

```

```

32     FROM ( SELECT custom.get_point_front_or_back_wall(v_main_surface_id, v_side_wall,
v_closest_road) arr ) a
33     , ( SELECT custom.get_point_front_or_back_wall(v_main_surface_id, v_side_wall,
v_2nd_closest_road) arr ) b;
34
35     -- obtém a soma dos vetores ortogonais a cada uma das fachadas principais
36     SELECT a.arr[1] + b.arr[1] x_final, a.arr[2] + b.arr[2] y_final
37     INTO v_x_ortogonal_vector, v_y_ortogonal_vector
38     FROM ( SELECT custom.get_ortogonal_vector(v_main_surface_id, v_side_wall,
v_closest_road) arr ) a
39     , ( SELECT custom.get_ortogonal_vector(v_main_surface_id, v_side_wall,
v_2nd_closest_road) arr ) b;
40
41     -- cria uma linha desde o ponto P com a direção da soma dos vetores ortogonais a cada
uma das fachadas principais
42     SELECT ST_MakeLine(d.midpoint_canto, d.end_point_ortogonal) INTO v_geom_orto_line
43     FROM ( SELECT ST_Translate(c.midpoint_canto, v_x_ortogonal_vector*100,
v_y_ortogonal_vector*100, 0) end_point_ortogonal
44     , c.midpoint_canto
45     FROM ( SELECT ST_Centroid (ST_Collect( ST_Centroid(ST_MakeLine( a.arr[1],
a.arr[2])) , ST_Centroid(ST_MakeLine( b.arr[1], b.arr[2])) )) midpoint_canto
46     FROM ( SELECT custom.get_point_front_or_back_wall(
v_main_surface_id, v_side_wall, v_closest_road) arr ) a
47     , ( SELECT custom.get_point_front_or_back_wall(
v_main_surface_id, v_side_wall, v_2nd_closest_road) arr ) b
48     ) c
49     ) d;
50     END IF;
51
52     --Identifica a superfície principal do edifício mais perto, segundo um vetor ortogonal à
frente do edifício em análise.
53     SELECT s.id INTO v_surface_front_building
54     FROM citydb.surface_geometry s
55     , citydb.building b
56     , cityobject_genericattrib g
57     , ( SELECT v_geom_orto_line orto_line ) c
58     , ( SELECT v_geom_mid_point_wall midpoint_wall ) d
59     WHERE s.id != v_main_surface_id
60     AND s.cityobject_id = b.id
61     AND b.id = g.cityobject_id
62     AND g.strval IN ( custom.c_building(), custom.c_building_out() )
63     AND ST_Area(s.geometry) != 0
64     AND ST_ZMax(s.geometry) = 0
65     AND ST_DWithin(c.orto_line, s.geometry, 0) IS TRUE
66     ORDER BY ST_Distance(d.midpoint_wall, s.geometry)
67     LIMIT 1;
68
69     -- Calcula a distância até ao edifício anteriormente identificado segundo o vetor ortogonal
70     SELECT ST_Distance(ST_Intersection(s.geometry, c.orto_line), d.midpoint_wall)
71     INTO v_result
72     FROM citydb.surface_geometry s
73     , ( SELECT v_geom_orto_line orto_line ) c
74     , ( SELECT v_geom_mid_point_wall midpoint_wall ) d
75     WHERE s.id = v_surface_front_building;
76
77     -- Caso não tenha encontrado um edifício no mesmo enfiamento o valor omissso será 50 metros
78     RETURN COALESCE(v_result, 50);
79     END;
80     $$ LANGUAGE plpgsql;

```

Figura 3.12 - Especificação da função custom.rule_distance_45

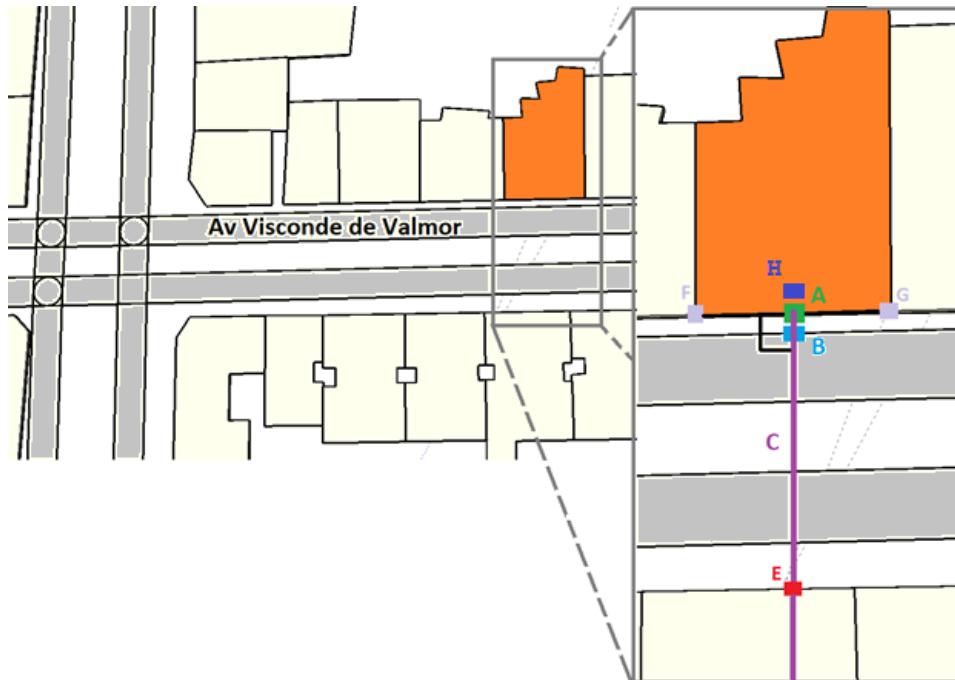


Figura 3.13 - Exemplo de aplicação - distância livre na frente do edifício

A fórmula de cálculo da distância livre na frente do edifício tem uma componente diferente no caso de um edifício de gaveto, a Figura 3.14 ilustra este caso. À semelhança do exemplo anterior é identificado o vetor ortogonal à fachada do edifício sendo que nos edifícios de canto são consideradas duas fachadas (vetor $\overrightarrow{AA'}$ e $\overrightarrow{BB'}$), neste caso o vetor ortogonal usado será a soma dos vetores ortogonais a cada uma das fachadas (linhas 36 a 39 da Figura 3.12). Este exemplo tem a particularidade de o canto do edifício ter um remate, nestes casos é identificado o ponto C através do prolongamento das faces da fachada até se interceptarem num ponto (linha 30 a 33), os edifícios com o canto arredondado são tratados de forma semelhante. A partir de dois pontos, a função *custom.make_extended_line* (linha 30) cria um segmento de reta e faz o prolongamento do mesmo para cada um dos lados. A reta D é a referência para calcular a distância entre os dois edifícios, esta reta é obtida através da soma dos vetores $\overrightarrow{AA'}$ e $\overrightarrow{BB'}$ com origem no ponto C e prolongada até coincidir com os dois edifícios sobre esta direção (linhas 42 a 49), tal como representado na figura. Assim, o resultado do exercício é a distância entre os pontos C' e E.

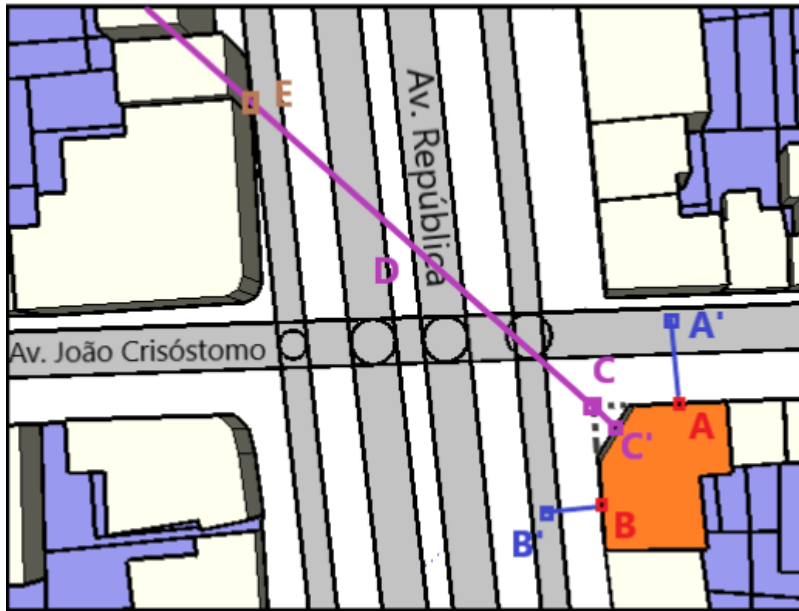


Figura 3.14 -Exemplo de aplicação - distância livre na frente do edifício no caso de edifício de gaveto

3.6.6. Restrição “R_Distância_Tardoz”

A determinação da distância livre no tardoz do edifício é um processo com algumas semelhanças ao processo anterior. A função *custom.rule_back_wall_distance* (Figura 3.15) propõe uma implementação desta regra. Na Figura 3.16 são destacados o edifício em análise (a laranja), o logradouro associado ao edifício (a azul) e os pontos necessários para analisar o problema. Começa-se por localizar o ponto médio da fachada de tardoz (ponto C – linhas 33 e 34 da Figura 3.15), este é o ponto médio dos pontos A e B, que são os dois pontos que estão mais afastados da rua em frente ao edifício (pertencentes aos segmentos de reta do perímetro do edifício). Posteriormente é calculado um vetor ortogonal ao segmento \overline{AB} , com origem em C que aponta para o exterior do edifício (reta D – linha 32). O passo seguinte é extrair todos os segmentos que representam o perímetro dos polígonos classificados como logradouro (linhas 15 a 31), para serem interseçados com a reta D. De seguida é seleccionado o segmento mais afastado do edifício, (segmento \overline{EF} – linhas 36 e 37) e é calculada a distância mínima entre o segmento e o edifício (linha 12). Tendo em conta esta restrição, a altura máxima do edifício é o dobro da distância de tardoz apurada.

```

1 CREATE OR REPLACE FUNCTION custom.rule_back_wall_distance ( p_building_root_id IN
  custom.building_stats.building_root_id%TYPE )
2 RETURNS custom.building_stats.back_wall_distance%TYPE AS $$
3 DECLARE
4     v_main_surface citydb.surface_geometry.id%TYPE;
5     v_side_wall    VARCHAR(30) := 'back';
6     v_closest_road citydb.surface_geometry.id%TYPE;
7     v_result       custom.building_stats.back_wall_distance%TYPE := 0;
8 BEGIN

```

```

9     v_main_surface := custom.get_main_surface_id(p_building_root_id);
10    v_closest_road := custom.get_nearest_road_id(v_main_surface);
11
12    SELECT ST_Distance(main_building.geometry, e.line ) -- calcula a distância entre o
13    INTO v_result -- segmento mais afastado e o edifício principal
14    FROM ( SELECT ST_MakeLine(sp,ep) line -- O segmento de reta do perímetro do logradouro
15    -- que intersesta com a reta ortogonal ao tardoz e que está mais afastada do tardoz
16    FROM ( SELECT ST_PointN(geom, generate_series(1, ST_NPoints(geom)-1)) as sp
17    , ST_PointN(geom, generate_series(2, ST_NPoints(geom) )) as ep
18    FROM ( SELECT (ST_Dump(ST_Boundary(geometry))).geom -- extrai os
19    segmentos de reta dos polígonos
20    FROM citydb.surface_geometry
21    WHERE id IN ( -- Lista todos os polígonos pertencentes ao
22    edifício em análise classificados como logradouro
23    SELECT s.id
24    FROM citydb.surface_geometry s
25    , citydb.building b
26    , cityobject_genericattrib g
27    WHERE 1=1
28    AND s.cityobject_id = b.id
29    AND b.id = g.cityobject_id
30    AND g.strval IN ( custom.c_log_verde(),
31    custom.c_log_cinza() )
32    AND b.building_root_id = p_building_root_id
33    AND ST_Area(s.geometry) != 0 )
34    ) AS linestrings
35    , ( SELECT custom.get_ortogonal_line_front_or_back_wall(v_main_surface,
36    v_side_wall, v_closest_road) orto_line ) c
37    , ( SELECT ST_Centroid(ST_MakeLine( a.arr[1], a.arr[2])) midpoint_wall
38    FROM ( SELECT custom.get_point_front_or_back_wall(v_main_surface,
39    v_side_wall, v_closest_road) arr ) a ) d
40    -- Calcula a distância segundo a direção da linha ortogonal à fachada de tardoz
41    do edifício
42    ORDER BY ST_Distance(ST_Intersection(ST_MakeLine(sp,ep), c.orto_line),
43    d.midpoint_wall) DESC NULLS LAST
44    LIMIT 1 ) e
45    , (SELECT geometry FROM citydb.surface_geometry WHERE id = v_main_surface)
46    main_building ;
47
48    RETURN COALESCE(v_result, 0);
49 END;
50 $$ LANGUAGE plpgsql;

```

Figura 3.15 - Especificação da função custom.rule_back_wall_distance

Em suma, no exemplo apresentado na Figura 3.16, a função calcula a distância entre o edifício e o segmento \overline{EF} . Contudo o segmento \overline{EH} também está no limite posterior do logradouro, e quando analisada a distância entre o edifício e o segmento \overline{EH} verifica-se que esta é ligeiramente menor do que a distância apurada pela função. Assim, a abordagem tomada apresenta uma aproximação ao resultado pretendido. Note-se que o caso descrito foi selecionado propositalmente de forma a expor a limitação desta abordagem, e este não reproduz a norma dos casos em estudo.

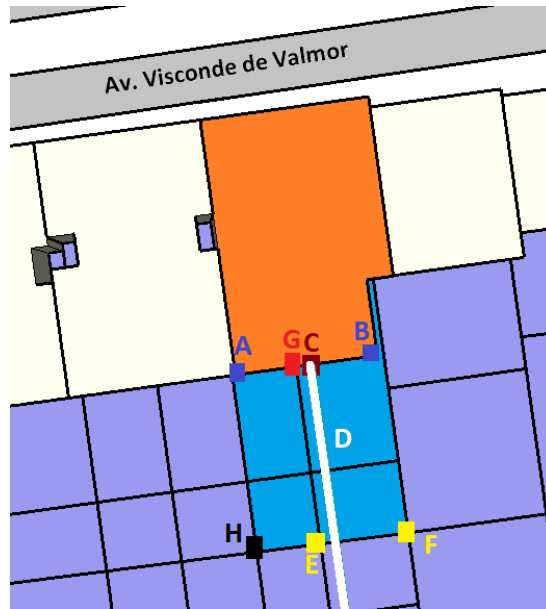


Figura 3.16 - Exemplo de aplicação da Regra de Tardoz

3.6.7. Restrição “R_SVP”

A regra “SVP” pretende formalizar o artigo 42º do PDM Lisboa. Esta regra difere das restantes, na medida em que não devolve um valor de altura a ser comparado para a altura final do edifício. O intuito da mesma é, ao longo da simulação, alterar a natureza do logradouro quanto à sua permeabilidade de forma a cumprir os requisitos legais apresentados no ponto 2.3.2. Assim, é calculada a superfície vegetal ponderada e, para os edifícios que não são de gaveto, altera-se a classificação das parcelas de solo do logradouro de tipo “C” para tipo “A”, de forma iterativa até que a regra seja cumprida.

3.6.8. Função principal

O objetivo principal desta ferramenta é possibilitar uma simulação sobre a altura dos edifícios, pela avaliação do resultado da variável H_i^t na equação (4), tendo em conta as restrições anteriormente analisadas ou apenas um subconjunto das mesmas, isto é, o utilizador poderá analisar os efeitos na altura dos edifícios de, por exemplo, apenas a restrição “R_Altura_Média”, ou verificar a influência nos resultados em adicionar uma nova restrição à análise (por exemplo, “R_Altura_Média” e “R_Altura_Vizinho”).

Este processo de simulação é iterativo, visto que a alteração da altura de um edifício no conjunto de dados pode influenciar a altura máxima permitida noutro edifício, tal como se verifica nas restrições “R_Altura_Média” e “R_Altura_Vizinho”. O resultado desta simulação será um conjunto de listas que

registam as alterações dos edifícios em cada iteração do processo (tabelas: *custom.log_change_height*; *custom.log_change_logradouro*; *custom.temporal_analysis*).

Para a implementação do processo desenvolveu-se a função *custom.main* (Figura 3.17), onde, para cada iteração, é calculada a altura máxima de cada edifício tendo em conta o limite de cada restrição

Para a implementação do processo desenvolveu-se a função *custom.main* (Figura 3.17), onde, para cada iteração, é calculada a altura máxima de cada edifício tendo em conta o limite de cada restrição (linhas 46 a 49 da Figura 3.17). Identifica-se a regra mais condicionante e por consequência a altura máxima que o edifício pode assumir (linhas 29 a 41). O passo seguinte, é selecionar o edifício com maior potencial de crescimento, ou seja, edifícios com uma maior diferença entre a altura calculada e a altura atual do edifício (linhas 27 a 43). Quando o potencial de crescimento é superior a 3 metros é registada a alteração e repete-se todo o processo com outro edifício (linha 58) até que não exista nenhum edifício com potencial de crescimento. Com recurso a este processo iterativo permite-se que as alterações que aconteçam num determinado edifício possam influenciar os restantes edifícios nas iterações posteriores. Definiu-se a marca de 3 metros (pé-direito típico do piso de um edifício) como um limiar ao qual para valores inferiores, o proprietário já não teria um incentivo suficiente a realizar uma ampliação ao edifício.

Tendo em conta o artigo 45º do PDM Lisboa, existem edifícios aos quais não é permitida a sua ampliação (edifícios identificados na Carta Municipal do Património Edificado e Paisagístico, ver ponto 2.3.2), por este motivo, foi incorporado um mecanismo que exclui da simulação edifícios assinalados neste contexto (linhas 51 a 56).

Resumidamente, esta função implementa diretamente a equação (4), primeiro é identificada a altura máxima mais condicionante tendo em conta as restrições ativas e posteriormente compara-se com altura atual do edifício adicionada de 3 metros. No caso de o valor calculado ser superior, regista-se a alteração do edifício e continua-se o processo até que não existam mais edifícios a alterar.

```
1 CREATE OR REPLACE FUNCTION custom.main(  
2     p_maximum_iterations          INTEGER DEFAULT 1000  
3     , p_rule_avg_height           CHAR DEFAULT 'Y'  
4     , p_rule_neighbour_height     CHAR DEFAULT 'Y'  
5     , p_rule_available_distance_45 CHAR DEFAULT 'Y'  
6     , p_rule_back_wall_distance  CHAR DEFAULT 'Y'  
7     , p_rule_svp                  CHAR DEFAULT 'Y'  
8     , p_rule_classified_building CHAR DEFAULT 'Y' )  
9 RETURNS void AS $$  
10 DECLARE  
11     c_maxium_height              custom.building_stats.height%TYPE := 50;  
12     k                             RECORD;  
13     v_count_iteration            INTEGER := 0;  
14     v_count_building_changes     INTEGER := 0;  
15     v_building_root_id          custom.building_stats.building_root_id%TYPE;  
16     v_calc_diference_height     custom.building_stats.height%TYPE := 50;  
17     v_building_height           custom.log_change_height.height%TYPE;  
18 BEGIN
```

```

19     PERFORM      custom.main_preparation(p_rule_avg_height,      p_rule_neighbour_height,
p_rule_avaiable_distance_45,      p_rule_back_wall_distance,      p_rule_svp,
p_rule_classified_building);

20
21     --Início da Simulação
22     WHILE v_count_iteration <= p_maximum_iterations
23     LOOP
24         BEGIN
25             SELECT aux.* --Lista de edifícios com potencial de realizar alterações
26             FROM (
27                 SELECT j.building_root_id
28                     , j.height
29                     , COALESCE( LEAST(
30                         CASE WHEN p_rule_avg_height = 'Y' AND p_rule_neighbour_height = 'Y'
31                             THEN GREATEST (j.avg_height, j.neighbour_height)
32                             WHEN p_rule_avg_height = 'Y' THEN j.avg_height
33                             ELSE NULL END
34                         , CASE WHEN p_rule_avaiable_distance_45 = 'Y'
35                             THEN j.avaiable_length_45
36                             ELSE NULL END
37                         , CASE WHEN p_rule_back_wall_distance = 'Y'
38                             THEN CASE WHEN j.is_canto = 0
39                                 THEN j.back_wall_distance
40                                 ELSE 50 END
41                             ELSE NULL END
42                     ) , j.height ) - j.height calc_diference_height
43                 INTO STRICT v_building_root_id, v_building_height, v_calc_diference_height
44                 FROM (
45                     SELECT i.building_root_id, street, is_canto, height, c.classified_building
46                     , custom.rule_avg_height(i.building_root_id, i.street, i.cityblock)
47                     , custom.rule_neighbour_height(i.building_root_id) neighbour_height
48                     , FLOOR(i.avaiable_length_45) avaiable_length_45
49                     , FLOOR(2*i.back_wall_distance) back_wall_distance
50                     FROM custom.building_stats i
51                     LEFT JOIN (     SELECT building_root_id
52                     , CASE WHEN p_rule_classified_building = 'Y' THEN class ELSE
53                     NULL END AS classified_building
54                     FROM citydb.building
55                     WHERE class IS NOT NULL ) c ON i.building_root_id =
56                     c.building_root_id
57                     ) j
58                     WHERE j.classified_building IS NULL
59                     ) aux
60                 WHERE aux.calc_diference_height >= 3 -- Apenas é registada uma alteração quando a
61                 diferença de alturas é pelo menos 3m
62                 ORDER BY calc_diference_height DESC, building_root_id
63                 LIMIT 1;
64             EXCEPTION
65                 WHEN no_data_found THEN
66                     EXIT;
67             END;
68
69             UPDATE custom.building_stats
70             SET height = v_building_height + v_calc_diference_height
71             WHERE building_root_id = v_building_root_id;
72
73             v_count_iteration := v_count_iteration + 1;
74
75             INSERT INTO custom.log_change_height (building_root_id, height,height_increase)
76             VALUES (v_building_root_id, v_building_height + v_calc_diference_height,
77             v_calc_diference_height);
78
79             IF p_rule_svp = 'Y' THEN
80                 PERFORM custom.building_log_rule_svp_one(v_building_root_id);
81             END IF;
82
83             INSERT INTO custom.temporal_analysis (iteration, main_street, avg_height,
84             percent_green_area, avg_svp)

```



```

80         SELECT v_count_iteration, main_street, avg_height, percent_green_area, avg_svp
81         FROM custom.indice_report;
82     END LOOP;
83
84     SELECT count(DISTINCT building_root_id), count(1)
85     INTO v_count_building_changes, v_count_iteration
86     FROM custom.log_change_height;
87
88     RAISE NOTICE 'Foram inseridas % alterações num total de % edificios', v_count_iteration,
89     v_count_building_changes;
90 END;
```

Figura 3.17 - Especificação da função custom.main

3.7. Visualização

A visualização do modelo 3D num ambiente interativo pode ter o impacto positivo de aproximar os destinatários da simulação ao resultado da mesma. O 3DCityDB para além de permitir fazer a importação do modelo GML para a base de dados, dispõe também de um módulo que permite extrair os elementos da base de dados para o formato KML. Este é um processo simples que pode ser realizado através do separador “KML/COLLADA/gITF Export” (apresentado anteriormente na Figura 2.4), sendo apenas necessário configurar o nome do ficheiro e fazer a exportação através do botão “Export”.

O link¹³ contém o documento extraído por este processo. Para visualizar e interagir com o modelo pode ser usado o programa Google Earth, obtendo-se uma visualização semelhante à Figura 4.7, apresentada no capítulo seguinte.

13

<https://github.com/UrbanSimulCityGML/LisbonBuildingsMaxHeight/blob/master/Situa%C3%A7%C3%A3o%20Inicial.kml>

4. Caso de Estudo

4.1. Execução da análise

Pretende-se agora exemplificar o processo o descrito no ponto 3.1, para o objetivo formalizado nesta dissertação: a simulação da evolução da altura dos edifícios a longo prazo, através de um modelo 3D de cidade semântico. Em primeiro lugar foi definida a área de estudo representada na Figura 4.1.

A área de estudo situa-se em Lisboa, na zona do Saldanha e é delimitada a Norte pela Av. João XXI / Av. de Berna, a Sul pela Av. Duque de Ávila, a Oeste pela Av. 5 de Outubro e a Este pela Av. Defensores de Chaves. Esta área corresponde a uma zona consolidada, inserida numa malha urbana regular ortogonal. A maioria dos edifícios é destinada à habitação, sendo o piso térreo por vezes usado em atividades de comércio ou serviços. Pontualmente, existem edifícios destinados exclusivamente à prestação de serviços. Relativamente à mobilidade, a zona dispõe de uma forte rede viária caracterizada por uma presença multimodal, a Av. da República destaca-se das restantes vias da área de estudo pela sua superior largura de via.



Figura 4.1 - Área de estudo, Lisboa (fonte: Google Maps – Av. da República)

Para construir um modelo que cumpra os objetivos da simulação é necessário partir de uma situação inicial que represente o estado atual da área de estudo. Para tal, através da plataforma *Cadmapper*, foi extraído um ficheiro *SketchUp* com as superfícies de implantação dos edifícios da área de estudo e da zona adjacente. O desenho retirado está referenciado no sistema de coordenadas “WGS 84 / UTM zone 29N”¹⁴.

De forma a preparar um modelo 3D de cidade semântico realizou-se a caracterização de algumas entidades da forma apresentada no ponto 3.4, dos 1007 edifícios retirados do *capmapper*, 241 pertencem à área de estudo e foram classificados em *layers* como “*buildings*”, os restantes como “*buildings_out*”; aos 13 objetos referentes a vias de comunicação adicionou-se a morada, sendo que, nos de maior dimensão especificou-se número de porta como “Par” ou “Ímpar” (apresentado posteriormente no Quadro 7.1, em anexo); relativamente aos logradouros, foram detalhadas parcelas associadas a cada um dos lotes da área de estudo, no total foram classificadas 313 parcelas do tipo A e 320 parcelas do tipo C, de acordo com a permeabilidade do solo; classificaram-se ainda os edifícios de gaveto (71 casos) e identificados 54 edifícios classificados no PDM de Lisboa com interesse patrimonial (apresentado posteriormente no Quadro 7.2, em anexo). Para terminar a preparação do desenho, estimou-se a altura conforme o método anteriormente no ponto 3.4 e foram alterados apenas os edifícios pertencentes à área de estudo. A partir deste ponto pode gerar-se um ficheiro GML através da extensão CityEditor com o nível de detalhe LOD1.

Paralelamente foi instalada uma base de dados PostGresSQL e criada a extensão PostGis¹⁵. O programa 3DCityDB incorpora scripts para a criação da estrutura que represente o CityGML na base de dados escolhida (este processo irá criar os schemas *citydb* e *citydb_pkg*¹⁶). A partir deste ponto pode importar-se o ficheiro GML criado para a base de dados com recurso ao 3DCityDB.

Para instalar a ferramenta apresentada nos pontos anteriores deve-se executar código no ficheiro “*custom functions.sql*” presente no repositório <https://github.com/UrbanSimulCityGML/LisbonBuildingsMaxHeight>. Esta ação irá criar todas tabelas e funções desenvolvidas nesta dissertação. O próximo passo é fazer o pré-processamento dos dados através da função “*custom.insert_main_startup*”. Em termos de desempenho, o tempo de execução da função referida foi de 67 seg, tendo em conta que o teste foi conduzido num computador pessoal DELL Latitude E7470 com 8 GB de RAM e um processador Intel Core i7-6600U.

¹⁴ A configuração da base de dados requer um sistema de coordenadas. O link (EPSG, 2020) indica a instrução necessária para esta parametrização em PostgreSQL/PostGis.

¹⁵ Para criar a extensão deve-se executar o seguinte comando: “CREATE EXTENSION postgis;”

¹⁶ Para a instalação da estrutura base CityGML numa base de dados através da ferramenta 3DCityDB pode seguir-se como exemplo o documento de Preka & Doulamis (2016).

A função principal é o ponto de contacto entre o utilizador e a ferramenta, a Figura 4.2 exemplifica como pode ser executada. Neste exemplo, a simulação tem em conta todas as restrições desenvolvidas, o tempo de execução foi de 11 seg. Tal como referido, a ferramenta também permite analisar apenas a interação de um subconjunto de regras, nesse caso altera-se o parâmetro de entrada da regra que não se pretenda incluir de “Y” para “N”.

```

1 DO $$
2 DECLARE
3     v_maximum_iterations          INTEGER := 1000;
4     v_rule_avg_height             CHAR := 'Y';
5     v_rule_neighbour_height       CHAR := 'Y';
6     v_rule_avaiable_distance_45  CHAR := 'Y';
7     v_rule_back_wall_distance    CHAR := 'Y';
8     v_rule_svp                   CHAR := 'Y';
9     v_p_rule_classified_building CHAR := 'Y';
10 BEGIN
11     PERFORM custom.main(v_maximum_iterations,v_rule_avg_height, v_rule_neighbour_height,
12         v_rule_avaiable_distance_45, v_rule_back_wall_distance, v_rule_svp,
13         v_p_rule_classified_building);
14 END; $$

```

Figura 4.2 - Exemplificação da execução da função principal

4.2. Resultados

Este capítulo procura expor alguns exemplos de análises que se podem realizar utilizando a ferramenta desenvolvida. O Quadro 4.1 enumera um conjunto de simulações nas quais são especificadas as restrições em análise.

Quadro 4.1 - Simulações executadas

Simulação	Descrição	Nº Alterações
S1	Todas as restrições estão ativas.	34
S2	Todas as restrições estão ativas exceto “Edifícios Classificados”.	55
S3	Ativa a restrição “Altura Média”.	92
S4	Ativas as restrições “Altura média” e “Altura Vizinho”.	106

As simulações escolhidas permitem validar qualitativamente a ferramenta através da comparação entre simulações. Para tal são analisadas as alterações de uma simulação base e comparada com os resultados de outra simulação, desta forma verifica-se a influência que acrescentar ou retirar regras provoca nos resultados. Segue-se agora uma análise às 34 alterações da simulação S1 por rua.

Quadro 4.2 - Resumo das alterações da simulação S1, valores médios por rua¹⁷

Rua	Altura (m)	Altura Inicial (m)	Svp (m2)	Svp Inicial (m2)	Nº edifícios alterados	Nº edifícios
Global	24.9	23.8	163.4	159.2	33	241
Av. 5 de Outubro	30.9	30.3	129.6	124.9	2	23
Av. Barbosa du Bocage	24.2	23.3	131.2	131.2	3	29
Av. João XXI / Av. de Berna	25.3	20.6	193.1	169.2	3	12
Av. da República	29.3	27.9	204.4	196.5	8	40
Av. Defensores de Chaves	21.8	21.6	134.6	134.6	1	26
Av. Duque de Avila	24.6	23.9	243.1	243.1	2	10
Av. Elias Garcia	25.3	24.5	108.9	101.7	6	31
Av. João Crisóstomo	20.2	19.5	170.4	170.4	2	22
Av. Miguel Bombarda	25.6	24.2	238.6	238.6	3	18
Av. Visconde de Valmor	20.3	19.3	158.1	156.1	3	30

Verifica-se, como esperado, que a altura média da área de estudo aumenta, de 23.8m para 24.9m (Quadro 4.2). Esta simulação provocou 34 alterações distribuídas por 33 edifícios, ou seja, existe um edifício que foi aumentado em dois momentos distintos. A Figura 4.3 permite compreender a distribuição das alterações efetuadas por edifício. A alteração mais recorrente é de 3m (ocorreu 6 vezes) e é na Av. João XXI / Av. de Berna que se encontra o edifício com mais potencial de crescimento (28m). Acrescenta-se ainda que todas as ruas têm pelo menos uma alteração ao longo do total do número de iterações, em média cerca de 14% dos edifícios de cada rua são alterados. Relativamente à superfície vegetal ponderada, esta aumenta, ou seja, são convertidos logradouros do Tipo C em Tipo A. Esta alteração no tipo de logradouro verifica-se nos edifícios pertencentes a quatro avenidas (sinalizado a verde no Quadro 4.2). Segundo os resultados da simulação, é na Av. João XXI / Av. de Berna que se esperam alterações mais marcantes, três dos doze edifícios são alterados, sendo um deles o que apresenta um maior potencial de crescimento e que, ao nível da rua espera-se um aumento de 23% na altura média dos edifícios (uma alteração de 20.6m para 25.3m).

¹⁷ No final de cada simulação, a *view custom.indice_report* permite extrair a informação presente neste quadro.

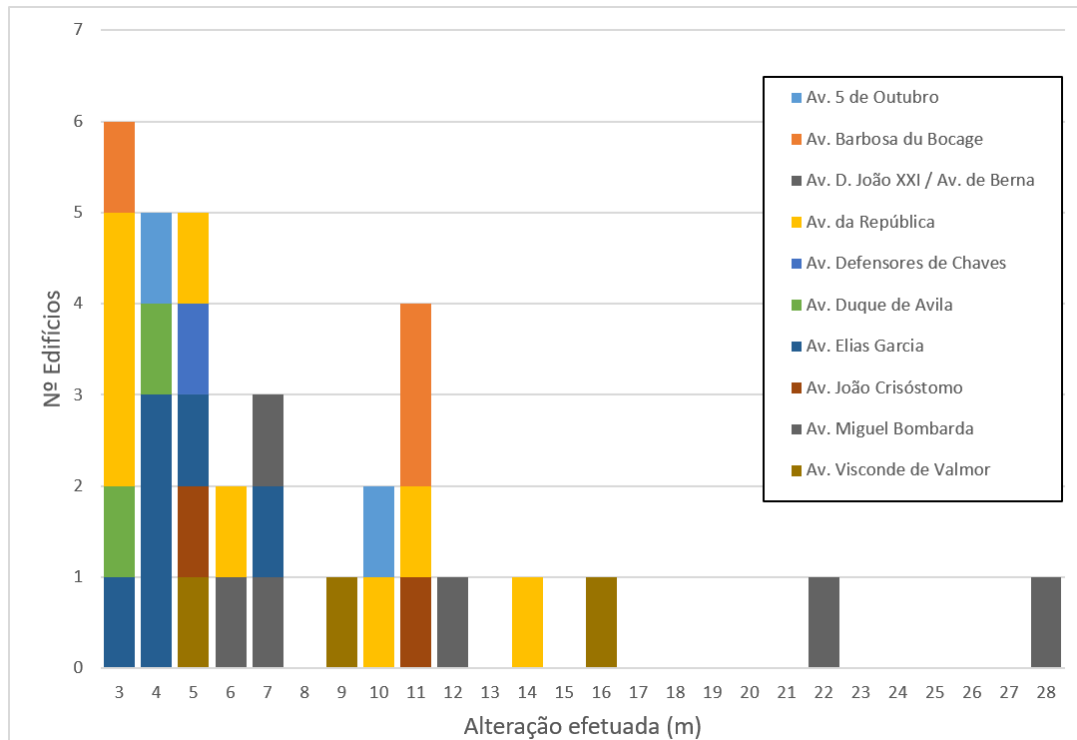


Figura 4.3 - Histograma de alterações efetuadas por rua

A Figura 4.4 permite ter uma perspetiva da evolução da altura média dos edifícios ao longo da simulação. Cada iteração da simulação corresponde a uma alteração num determinado edifício, não se pode tomar este como um conceito temporal na medida em que o tempo decorrido entre cada alteração não será certamente o mesmo para todas as iterações. Em anexo é disponibilizada a análise iterativa para as restantes simulações.

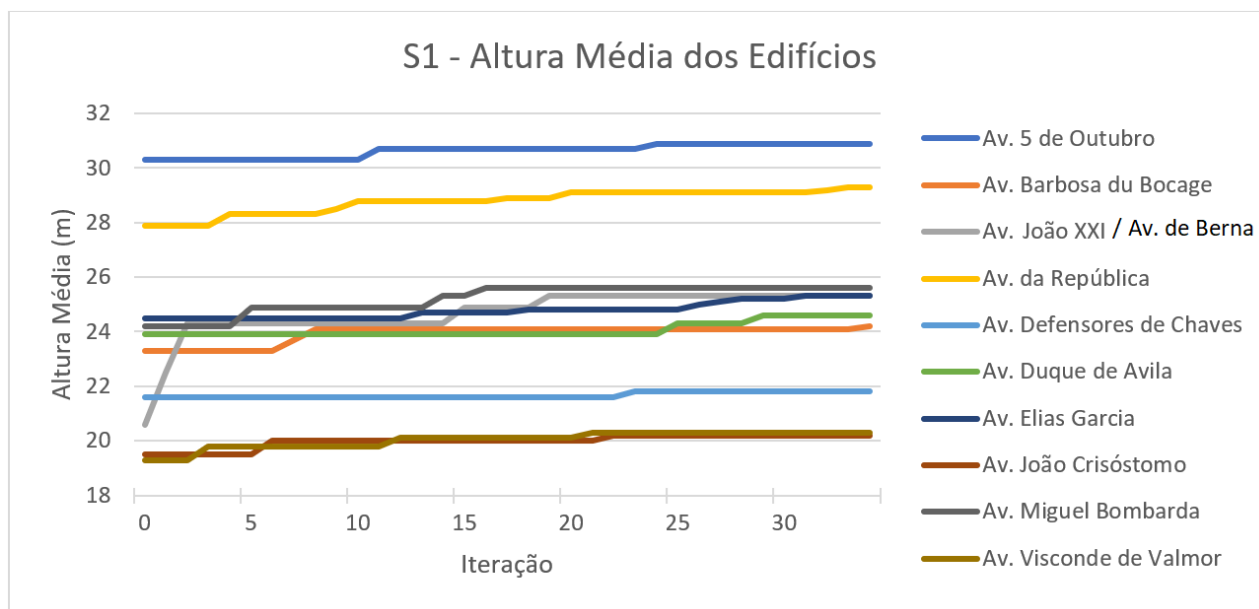


Figura 4.4 - Simulação S1, perspectiva por iteração

O Quadro 4.3 permite comparar as alterações de altura média dos edifícios e o número de edifícios alterados em cada simulação. Ao tomar a S1 como um ponto base verificamos que na S2 em todas as ruas a altura dos edifícios mantém-se ou aumentou. Este é o comportamento esperado uma vez que na S2 estamos a retirar uma restrição, aumentando o número de edifícios disponíveis para realizar alterações em 33 (número de edifícios classificados - Quadro 7.2). Ao comparar o número de edifícios alterados em S2 face a S1 conclui-se que os 20 edifícios adicionalmente alterados ($53 - 33 = 20$) são edifícios classificados, ou seja, segundo as regras implementadas no modelo, 20 dos 33 edifícios classificados (67%) seriam ampliados caso esta limitação não o prevenisse.

Ao analisar as simulações S3 e S4, valida-se que em todas ruas a altura média é superior a S1 uma vez que estas simulações têm menos regras aplicadas do que a S1. Ao comparar S3 com S4 pode-se quantificar o impacto que acrescentar a regra “Altura Vizinho” tem sobre os edifícios em análise. No total, mais 10 edifícios têm uma extensão de altura, sendo que a Av. João Crisóstomo é o local com mais impacto, uma extensão de 4 edifícios leva a que a altura média aumente 3.8m (de 21.5m para 25.3m).

Quadro 4.3 - Resumo alterações por simulação

Avenida	Altura					Nº Edifícios Alterados				
	Inicial	S1	S2	S3	S4	Inicial	S1	S2	S3	S4
Global	23.8	24.9	25.8	26.8	27.4	241	33	53	83	93
Av. 5 de Outubro	30.3	30.9	31.7	31.6	32.1	23	2	5	5	6
Av. Barbosa du Bocage	23.3	24.2	24.2	27.0	27.0	29	3	3	14	14
Av. João XXI / Av. de Berna	20.6	25.3	28.3	28.8	28.8	12	3	5	6	6
Av. da República	27.9	29.3	30.9	31.5	32.0	40	8	13	12	15
Av. Defensores de Chaves	21.6	21.8	22.0	23.1	23.3	26	1	2	4	5
Av. Duque de Avila	23.9	24.6	26.2	26.4	26.4	10	2	3	3	3
Av. Elias Garcia	24.5	25.3	25.7	27.1	27.3	31	6	9	15	15
Av. João Crisóstomo	19.5	20.2	20.2	21.5	25.3	22	2	2	7	11
Av. Miguel Bombarda	24.2	25.6	27.4	28.6	29.7	18	3	6	7	8
Av. Visconde de Valmor	19.3	20.3	21.0	21.9	21.9	30	3	5	10	10

Um ponto relevante da simulação é a forma como, em cada iteração, é escolhido qual o edifício a fazer a alteração. A simulação é um procedimento faseado, em cada iteração é analisada a lista de edifícios com potencial de realizar alterações e escolhido o edifício com o maior potencial de crescimento¹⁸. Esta abordagem leva a que o número de iterações necessárias para que a simulação termine seja menor, isto porque, a escolha do edifício com maior potencial leva a que numa iteração, o novo limiar das funções “Altura média” e “Altura Vizinho” seja superior a qualquer alternativa de escolher outro edifício. Na linha 59 da Figura 3.17 pode ver-se a especificação desta abordagem. Outra abordagem seria, em cada iteração, escolher de forma aleatória um edifício disponível na lista de edifícios com potencial de realizar alterações. Para implementar esta alternativa, a linha 59 da Figura 3.17 deve ser “ORDER BY Random()”.

Os gráficos seguintes, Figura 4.5 e Figura 4.6, mostram o acréscimo de altura no edifício em cada iteração conforme as abordagens anteriormente sugeridas, esta representação permite distinguir a evolução dos edifícios ao longo do tempo. As duas abordagens produzem resultados semelhantes, na primeira são

¹⁸ Em cada instante é possível avaliar o potencial de crescimento de um edifício, este é a altura máxima permitida subtraída da altura antes da alteração.

acrescentados 261m à globalidade dos edifícios no caso de estudo, pela segunda abordagem, 274m. Nota-se ainda que a abordagem aleatória terminou com 37 iterações, mais três que a abordagem anterior.

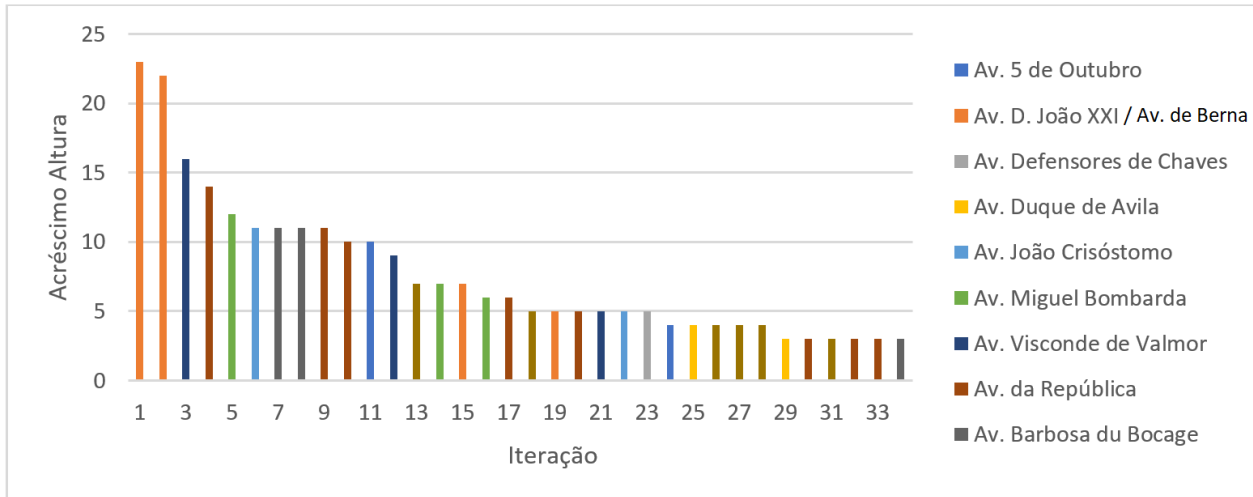


Figura 4.5 - S1 - Acréscimo de altura por edifício, versão ordenada

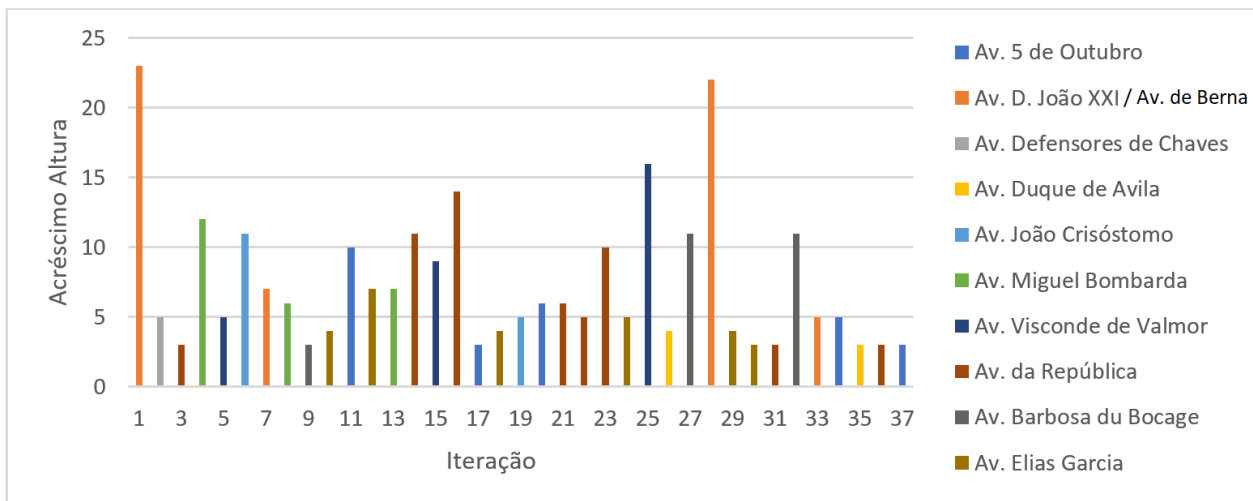


Figura 4.6 - S1 - Acréscimo de altura por edifício, versão aleatória

A Figura 4.7 é uma vista aérea sobre o resultado da simulação S1 com o sentido Sul-Norte dividida em quatro fases de acordo com a ordem em que os edifícios foram alterados, esta imagem inclui edifícios amarelos que representam aqueles que não tiveram qualquer alteração durante o processo, e em alternativa, os edifícios a azul (fase 1), verde (fase 2), rosa (fase 3) e vermelho (fase 4) de acordo com a fase em que ocorreu a alteração.

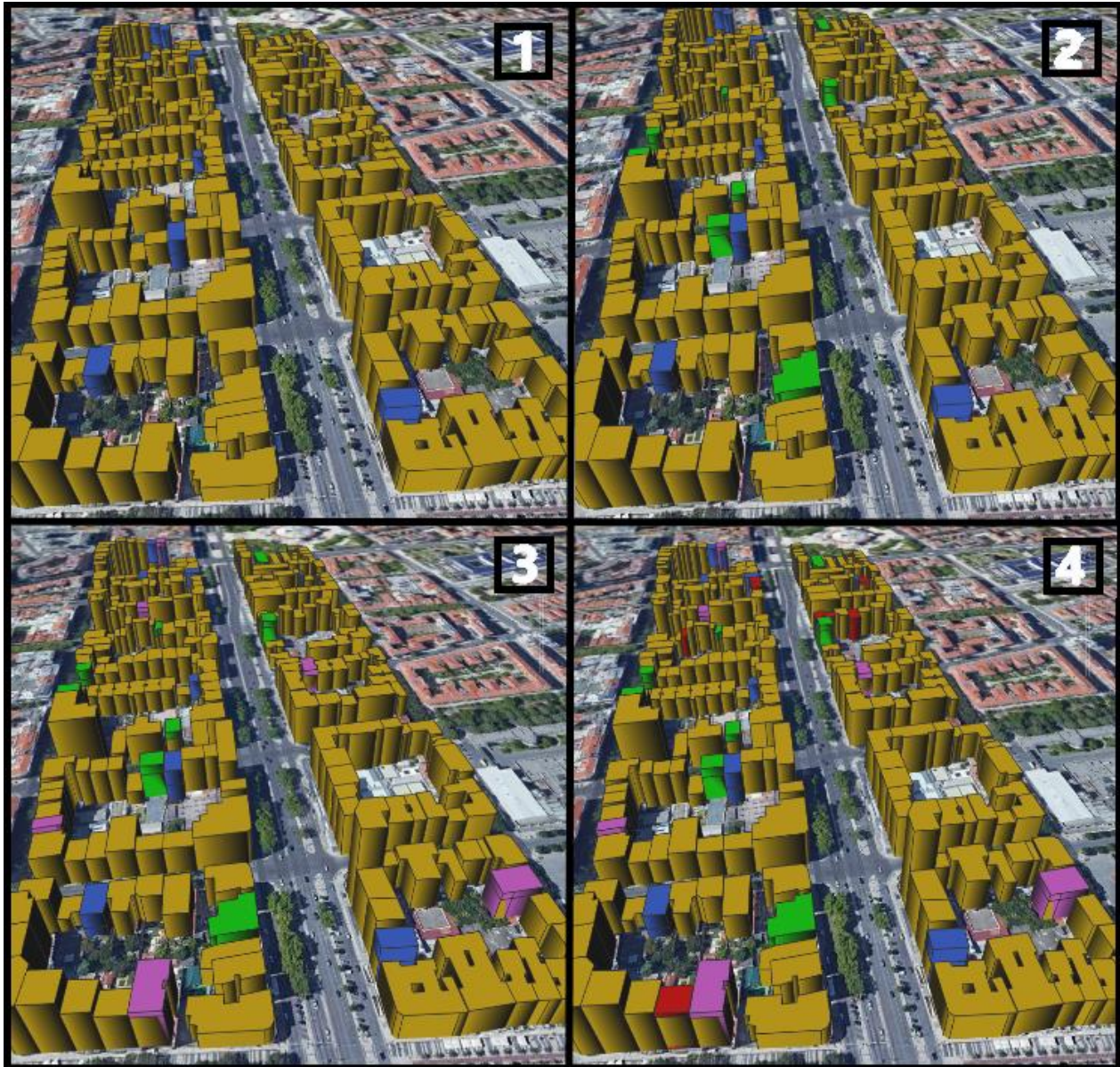


Figura 4.7 - (S1) Vista aérea sobre o caso de estudo, dividido em 4 fases

5. Conclusão e Trabalhos futuros

Esta dissertação apresenta um estudo sobre a formulação de um modelo 3D de cidade semântico, aplicado ao problema da simulação da altura máxima de edifícios usando CityGML. Corrobora-se assim que os modelos 3D de cidade semântico quando associados a um formato padronizado como o CityGML, são capazes de arquivar, transferir e organizar informação de forma a que possam servir de base a análises e simulações com aplicações em diversas áreas como: planeamento urbano, logística, imobiliário, exposição solar, exposição ao ruído, entre outras.

A metodologia seguida procura sistematizar a forma como se podem endereçar problemas derivados do tratamento de dados, independentemente da tecnologia de informação usada. A metodologia consiste em extrair os dados (de uma ou várias fontes) e tratá-los de forma a integrarem uma estrutura normalizada, sendo depois transferidos para um sistema de base de dados. Posteriormente, procura-se realizar uma simulação consoante o objetivo do estudo. A última fase consiste em visualizar e analisar os resultados obtidos pela simulação.

Esta dissertação pretende também ser um guia de como desenvolver este tipo de modelo, pelo que houve uma particular atenção à possibilidade de ser replicável noutros contextos fornecendo uma descrição passo a passo dos vários procedimentos de preparação do modelo e de introdução de dados.

Uma importante componente desta dissertação foi desenvolver uma ferramenta¹⁹ que seja capaz de mapear os pontos relevantes em termos de legislação e regulamentação urbanística ao nível nacional e municipal sobre o tema da máxima altura permitida num edifício. Para este mapeamento foram implementadas as regras nos artigos 42º, ponto 4, alíneas a) e b) (“altura média” e “altura do vizinho”), artigo 44º (superfície vegetal ponderada) e 45º (“imóvel classificado”) do PDM de Lisboa e ainda os artigos 59º e 62º (“regra 45º” e “distância de tardoz”) do RGEU.

Esta ferramenta pretende simplificar o processo de realização de uma simulação, no sentido, em que as regras referidas foram implementadas como funcionalidades na base de dados, eliminando o requisito do utilizador possuir conhecimentos de programação. Assim, a ferramenta deverá ser uma interface entre o utilizador e a simulação, na qual podem ser analisadas simultaneamente todas as restrições sinalizadas ou apenas um subconjunto das mesmas. O modelo construído é aplicável à cidade de Lisboa e tem por base o pressuposto que continuará a existir uma tendência de procura de habitação na cidade. Como tal, existe a tendência em explorar tanto quanto possível o limite permitido de altura do edifício.

¹⁹ A ferramenta desenvolvida está disponível no link:

<https://github.com/UrbanSimulCityGML/LisbonBuildingsMaxHeight>.

Como forma de aplicação e validação da ferramenta desenvolvida, procedeu-se à análise de uma área de estudo consolidada situada na zona do Saldanha em Lisboa. A área de estudo contém 241 edifícios e através da simulação S1 (em que todas as regras estão ativas), verifica-se que existem 33 edifícios com um potencial de serem aumentados, sendo na Av. João XXI / Av. de Berna que se preveem mais alterações significativas. Nesta rua a altura média dos edifícios cresce de 20.6 m para 25.3 m, ainda assim verifica-se que todas as ruas da área de estudo preveem pelo menos a alteração a um edifício ao longo da simulação. Quando ao índice de Superfície Vegetal Ponderada também se espera um aumento nesta variável em quatro das dez ruas analisadas.

A partir de uma segunda simulação (S2) em que se permite que os edifícios classificados no PDM Lisboa também estejam sujeitos a alterações, o modelo prevê o aumento da altura em 53 edifícios (mais 20 edifícios que na simulação anterior, S1). Verifica-se que dos 54 edifícios classificados, 20 seriam aumentados caso não existisse esta imposição legal.

Com o intuito de quantificar o impacto que um subconjunto de regras tem sobre os edifícios em análise, exemplificou-se a influência que a alínea b) tem sobre a alínea a) do ponto 4 do artigo 42º do PDM Lisboa. Ao comparar a simulação S3 (na qual está ativa apenas a regra “altura média”) com a simulação S4 (estão ativas as regras “altura média” e “altura vizinho”), conclui-se que mais 10 edifícios têm uma extensão de altura face à situação base S3, sendo que a Av. João Crisóstomo é a rua com maior impacto, uma alteração em 4 edifícios conduz a um aumento da altura média da rua em 3.8m (de 21.5m para 25.3m).

A simulação é um processo iterativo, em que em cada fase é analisado o potencial de crescimento de todos os edifícios e selecionado apenas um para alterar. O potencial de crescimento não é constante, porque a alteração de altura de um determinado edifício pode influenciar o potencial de outro edifício. Para a seleção do edifício a alterar, analisaram-se os resultados de duas diferentes abordagens, a primeira consiste em selecionar sempre o edifício com maior potencial, enquanto a segunda abordagem seleciona aleatoriamente um edifício com potencial de crescimento. A primeira abordagem conduz a um menor número de iterações necessárias para que a simulação termine, isto porque, a escolha do edifício com maior potencial leva a que na próxima iteração, o novo limiar das funções “Altura média” e “Altura Vizinho” seja superior a qualquer alternativa escolhida. Numa análise a longo prazo, *a priori* não existe evidência que a primeira abordagem resulte em edifício mais altos, contudo os resultados são semelhantes.

Uma das limitações desta dissertação advém da necessidade de obtenção de dados para que se possa realizar simulações numa área de estudo de maior dimensão, isto porque, apesar do modelo desenvolvido conseguir responder a simulações com maior quantidade de dados, alguns destes são recolhidos e tratados de forma manual o que requer bastante tempo. Ao nível da diversidade, seria interessante a obtenção de outros indicadores urbanísticos como o ano de construção do edifício, dimensão do lote, tipologia do edifício, classificação quanto à utilização predominante, etc, que não são públicos ou são de difícil acesso.

Nesta dissertação privilegiou-se a incorporação de regulamentação urbanística proveniente de fontes oficiais, uma alternativa seria propor e simular novas regras (sejam elas à imagem de outro país ou da autoria do utilizador) a fim de se verificar o seu impacto no conjunto de dados. Do ponto de vista dos responsáveis pelo planeamento urbano, permitiria perceber e quantificar o impacto na alteração ou na introdução de uma nova medida. Nesta dissertação apenas foi simulada a alteração de altura do edifício, seria também interessante, que uma nova área de implantação fosse proposta à luz das regras definidas em vigor. Neste trabalho não foi tomado em consideração a morfologia do terreno, no entanto, seria um ponto a incluir de forma a estudar a sua influência nas regras apresentadas. Por último, as funções utilizadas foram desenvolvidas em SQL, sendo esta uma linguagem bastante adotada e em constante evolução, seria importante otimizar os componentes desenvolvidos com o fim de melhorar o seu desempenho para conjuntos de dados de maior dimensão.

6. Bibliografia

- ArcGis. (2019). *PostgreSQL data types supported in ArcGIS*. Obtido em Março de 2019, de <https://desktop.arcgis.com/en/arcmap/10.3/manage-data/gdbs-in-postgresql/data-types-postgresql.htm>
- Biljecki, F. (2017). *Level of Detail in 3D City Models*. Technische Universiteit Delft, Zagreb, Croatia. Obtido de <https://filipbiljecki.com/phd/dissertationFilipBiljecki-lowresolution.pdf>
- Biljecki, F., Ledoux, H., & Stoter, J. (2017). Generating 3D city models without elevation data. *Computers, Environment and Urban Systems*, 64, 1-18.
- Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., & Coltekin, A. (2015). Applications of 3D City Models: State of the Art Review. (W. Kainz, Ed.) *ISPRS International Journal of Geo-Information*, 4(4), 2842-2889.
- Billen, R., Cutting-Decelle, A.-F., Marina, O., Almeida, J.-P., Caglioni, M., Falquet, G., . . . Zlatanova, S. (2014). *3D City Models and urban information: Current issues and perspectives*. Cedex, França. Obtido de https://www.researchgate.net/publication/260172519_3D_City_Models_and_urban_information_Current_issues_and_perspectives/download
- Building Smart. (2019). *Building Smart International*. Obtido em Abril de 2019, de https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/
- Cadmapper LLC. (2013). Obtido em Janeiro de 2019, de Cadmapper: <https://cadmapper.com/>
- Câmara Municipal Coimbra. (2003). *Regulamento municipal de urbanização e edificação (RMEU) de Coimbra*. Coimbra. Obtido em Setembro de 2018, de http://www.oasrn.org/pdf_upload/RMUE_Coimbra_I.pdf
- Câmara Municipal de Lisboa. (2012). Regulamento Plano Director Municipal de Lisboa. Lisboa. Obtido em Fevereiro de 2020, de https://informacoeseservicos.lisboa.pt/fileadmin/download_center/normativas/regulamentos/urbanismo/Regulamento_PDM.pdf
- Câmara Municipal de Lisboa. (2013). *Manual Técnico do Plano Diretor Municipal de Lisboa*. Lisboa. Obtido em Janeiro de 2020, de https://www.lisboa.pt/fileadmin/cidade_temas/urbanismo/documentos/MT_PDM_adenda.pdf
- Chair of Geoinformatics TUM. (2018). *3D City Database for CityGML Version 4.0.0*. Technical University of Munich, Geoinformatics, Germany.
- Döllner, J., & Buchholz, H. (2005). *Continuous Level-of-Detail Modeling of Buildings in 3D City Models*. Bremen. Obtido de [http://delivery.acm.org/10.1145/1100000/1097089/p173-](http://delivery.acm.org/10.1145/1100000/1097089/p173-61)

dollner.pdf?ip=193.136.132.10&id=1097089&acc=ACTIVE%20SERVICE&key=2E5699D25B4FE09E%2E454625C777251F56%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&__acm__=1553598200_15112a9d03f6aa8c36121323a62d4e65

- Dollner, J., Bauman, K., & Buchholz, H. (2006). Virtual 3D City Models as Foundation of Complex Urban Information Spaces. *Sustainable Solutions for the Information Society - Proceedings of 11th international conference on Urban Planning and Spatial Development in the Information Society* (pp. 107-112). Wien, Austria: Manfred Schrenk. Obtido de https://www.corp.at/archive/CORP2006_DOELLNER.pdf
- Donkers, S., Ledoux, H., Zhao, J., & Stoter, J. (2016). Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. Obtido de <https://onlinelibrary.wiley.com/doi/pdf/10.1111/tgis.12162>
- El-Mekawy, M. (2010). *Integrating BIM and GIS for 3D City Modelling - The Case of IFC and CityGML*. Obtido de <http://www.diva-portal.org/smash/get/diva2:390653/FULLTEXT02.pdf>
- El-Mekawy, M., Östman, A., & Hijazi, I. (2012). *A Unified Building Model for 3D Urban GIS*. ISPRS International Journal of Geo-Information. Obtido de <https://www.mdpi.com/2220-9964/1/2/120/pdf>
- El-Mekawy, M., Östman, A., & Shahzad, K. (2011). *Towards Interoperating CityGML and IFC Building Models: A Unified Model Based Approach*.
- EPSG. (2020). *WGS 84 / UTM zone 29N*. Obtido em Janeiro de 2020, de <https://epsg.io/32629>
- Figueiredo, F. (2007). *Planeamento de Rotas de Veículos com Entregas: Aplicação a um caso de estudo real*. I.S.T.
- Galego, A. (2014). *Exploração do potencial dos City Information Models em Instrumentos de Gestão Territorial à escala Urbana*.
- Groger, G., & Plumer, L. (2012). CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, 12-33.
- Herbert, G., & Chen, X. (2015). A comparison of usefulness of 2D and 3D representations of urban planning. *Cartography and Geographic Information Science*, 22-32. Obtido de <https://www.tandfonline.com/doi/pdf/10.1080/15230406.2014.987694?needAccess=true>
- Hunter, J., Brooking, C., Reading, L., & Vink, S. (2015). *A Web-based system enabling the integration, analysis, and 3D sub-surface visualization of groundwater monitoring data and geological models*.
- Jokela, J. (2016). *CityGML building model production from airborne laser scanning*. Aalto University, Espoo.

- Kolbe, T. (2009). Representing and Exchanging 3D City Models with CityGML. Em *3rd International Workshop on 3D Geo-Information*. Seoul, Korea: Springer Verlag. Obtido de <https://mediatum.ub.tum.de/doc/1145752/file.pdf>
- Kolbe, T. H., Gröger, G., & Plümer, L. (2005). CityGML – Interoperable Access to 3D City Models. *Proceedings of the Int. Symposium on Geo-information for Disaster Management*. Delft: Springer Verlag. Obtido de <https://mediatum.ub.tum.de/doc/1145762/file.pdf>
- Long, N., Fleming, K., & Brackney, L. (14 de 11 de 2011). *An Object-Oriented Database for Managing Building Modeling Components and Metadata*. Obtido de <https://www.nrel.gov/docs/fy12osti/51835.pdf>
- Longley, P., Goodchild, M., Maguire, D., & Rhind, D. (2011). *Geographic Information Systems & Science*. Toronto: John Wiley and Sons.
- Moradi, D. (2017). *3D Thematic Mapping and Visualization in CesiumJS Virtual Globe*. Technische Universität München.
- Nações Unidas. (2014). *World Urbanization Prospects*. Obtido de <https://esa.un.org/unpd/wup/publications/files/wup2014-highlights.pdf>
- Nouvel, R., Kaden, R., Bahu, J., Kaempf, J., Cipriano, P., Lauster, M., . . . Casper, E. (2015). *Genesis of the CityGML Energy ADE*. Obtido de <https://mediatum.ub.tum.de/doc/1429761/file.pdf>
- Ohuri, K. A., Biljecki, F., Kumar, K., Ledoux, H., & Stoter, J. (2018). Modelling cities and landscapes in 3D with CityGML. Em Springer, *Building Information Modeling* (pp. pp. 199–215). Cham: Springer International Publishing. doi:10.1007/978-3-319-92862-3_11
- Open Geospatial Consortium. (2012). *OGC City Geography Markup Language (CityGML) Encoding Standard*. Obtido de <https://www.opengeospatial.org/standards/citygml>
- Ordenamento do Território do Conselho da Europa. (2011). *Glossário do Desenvolvimento Territorial*. Títulos Gerais. Obtido em Março de 2020, de <https://rm.coe.int/16806f7d5d>
- Oxford University Press. (2010). Oxford Dictionary of English. Em A. Stevenson (Ed.).
- Postgis. (s.d.). Obtido em 16 de Fevereiro de 2019, de https://postgis.net/docs/using_postgis_dbmanagement.html
- Preka, D., & Doulamis, A. (2016). 3D Building Modeling In LOD2 using the CityGML Standard. *11th 3D Geoinfo Conference. XLII-2/W2*, pp. 20-21. Athens, Greece: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Obtido em 2019, de <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W2/11/2016/isprs-archives-XLII-2-W2-11-2016.pdf>

- Resolução da Assembleia da República nr.º 66/2004. (15 de Outubro de 2004). Diário da República. 1ª Série, nº 243, p. 6306. Obtido em Fevereiro de 2019, de <https://dre.pt/application/conteudo/575739>
- Resolução do Conselho de Ministros nº 12. (07 de Fevereiro de 2012). Diário da República. Nº27, p. 604. Obtido em Fevereiro de 2019, de <https://dre.pt/application/conteudo/543701>
- Santos, M. (2016). *Modelação tridimensional em SIG para avaliação do potencial solar fotovoltaico em meio urbano*. Universidade de Lisboa, Lisboa.
- Saran, S., Oberai, K., Wate, P., Konte, A., Dutta, A., Kumar, K., & Kumar, S. (2018). *Utilities of Virtual 3D City Models Based on CityGML: Various Use Cases*.
- Stadler, A., Nagel, C., König, G., & Kolbe, T. (2009). *Making interoperability persistent: A 3D geo database based on CityGML*.
- The PostgreSQL Global Development Group. (1996). *PostgreSQL*. Obtido em Abril de 2019, de <https://www.postgresql.org/>
- Tolmer, C., Castaing, C., Diab, Y., & Morand, D. (2013). CityGML and IFC: going further than LOD. *International Congress Digital Heritage*. Marseille France. Obtido de https://www.researchgate.net/publication/259397110_CityGML_and_IFC_going_further_than_LOD
- Trimble Inc. (2016). *CityEditor*. Obtido em Fevereiro de 2019, de <https://extensions.sketchup.com/sv/content/cityeditor-2>
- Vasques, A. M. (2017). *A multifuncionalidade como instrumento de valorização das áreas urbanas centrais*.
- Yao, Z., Nagel, C., Kunde, F., Hudra, G., Wilkomm, P., Donaubauer, A., . . . Kolbe, T. (2018). *3DCityDB - a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML*.

7. Anexos

Quadro 7.1 - Classificação de ruas quanto à morada

Nome	Número da Porta
Avenida 5 de Outubro	
Avenida Barbosa du Bocage	Ímpar
Avenida Barbosa du Bocage	Par
Avenida João XXI / Avenida de Berna	
Avenida da República	Ímpar
Avenida da República	Par
Avenida Defensores de Chaves	
Avenida Duque de Avila	
Avenida Elias Garcia	Ímpar
Avenida Elias Garcia	Par
Avenida João Crisóstomo	
Avenida Miguel Bombarda	
Avenida Visconde de Valmor	Ímpar
Avenida Visconde de Valmor	Par

Quadro 7.2 - Lista de edifícios classificados na área de estudo

Código	Designação
23.100	Edifício de Habitação unifamiliar/Avenida João Crisóstomo,29
23.101	Edifício de Habitação unifamiliar/Avenida Visconde Valmor,31
23.102	Edifício de Habitação unifamiliar/Avenida Miguel Bombarda,35
23.103	(Antiga) mercearia com fachada de azulejo/Av. 5 de Outubro, 84 -92; Av. Visconde de Valmor, 45 -45C
23.37	Conjunto de dois edifícios de habitação plurifamiliar/Av. Cinco de Outubro, 128 -138 e 140 -144
23.38	Conjunto de três edifícios de habitação plurifamiliar/Av. da República, 48 e 48B e Av. Barbosa du Bocage, 47 -47A
23.42	Conjunto de quatro edifícios de habitação plurifamiliar/Av. Elias Garcia, 63 - 67A, 69 -75, 77 e 79
23.43	Edifício de habitação plurifamiliar/ Av. da República, 49 -49D (Prémio Valmor 1923)
23.44	Edifício de habitação unifamiliar/Av. da República, 40
23.46	Edifício de habitação plurifamiliar/Av. Defensores de Chaves, 69
23.50	Conjunto arquitetónico/Av. Visconde de Valmor, 35, 37, 39, 41, 43 e 45 -45C; Av. Cinco de Outubro, 84 a 92
23.51	Edifício de habitação plurifamiliar/Av. da República, 37 -37F; Av. Miguel Bombarda, 12 -14E

23.52	Edifício de habitação plurifamiliar/Av. da República, 36 -36F; Av. Visconde de Valmor, 27 -27F
23.53	Edifício de habitação plurifamiliar/Av. da República, 32 -32B
23.54	Conjunto de dois edifícios de habitação plurifamiliar/Av. Defensores de Chaves, 61 -61F e 63 -63A
23.56	Edifício de habitação plurifamiliar (fachada)/Av. da República, 25 -25A; Av. João Crisóstomo, 24
23.57	Edifício de habitação plurifamiliar/Av. João Crisóstomo, 18 -18D
23.59	Edifício de habitação plurifamiliar/Av. João Crisóstomo, 25 -25A
23.62	Edifício de habitação plurifamiliar/Av. da República, 23; Av. João Crisóstomo, 19 (IIM e Prémio Valmor 1913)
23.64	Edifício de habitação plurifamiliar/Av. da República, 19
23.66	Colégio Académico/Av. da República, 13; Av. Duque d'Ávila, 38 -38K
23.67	Restaurante e snack bar Galeto/Av. da República, 14
23.68	Conjunto arquitetónico/Av. Duque d'Ávila, 18 -18B, 20 -22A, 24 -24B, 26 -26B, 28 -30A e 32 -32F e Av. da República, 10 -10G
23.71	Edifício de habitação plurifamiliar/Av. da República, 71 -73
23.73	Vila Santos/Acesso: Campo Pequeno, 74
23.87	Conjunto de quatro edifícios de habitação plurifamiliar/Av. Visconde Valmor, 26, 28, 30 e 32
23.97	Conjunto de quatro edifícios de habitação plurifamiliar/Avenida Cinco de Outubro, 100 a 114; Avenida Elias Garcia, 95
23.98	Edifício de Habitação unifamiliar/Avenida Miguel Bombarda, 24 -34
23.99	Edifício de Habitação plurifamiliar/Avenida Miguel Bombarda, 6
3249	Edifício na Avenida da República, n.º 38/ 38 -A e na Avenida Visconde Valmor, n.º 22.
3251	Edifício na Avenida de Berna, n.º 1 a 1A, e Avenida da República.
4771	Edifício na Avenida de 5 de Outubro, esquina com a Avenida do Duque de Ávila.

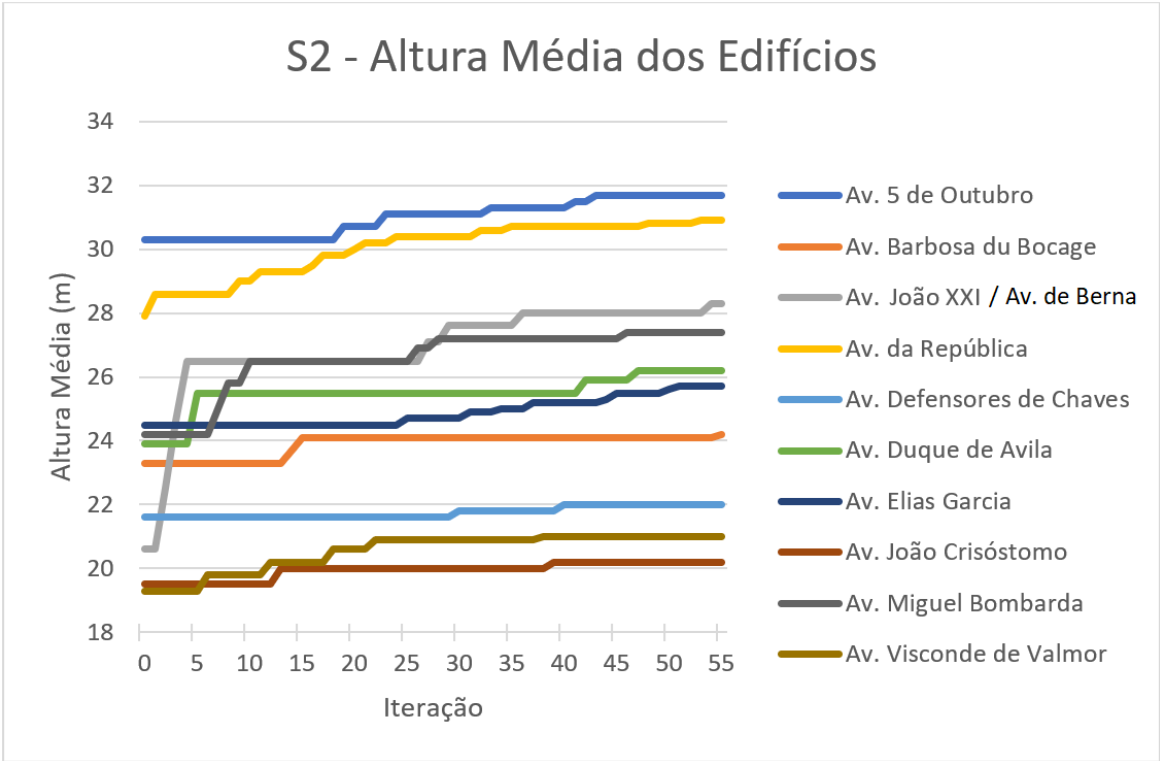


Figura 7.1 - Simulação S2, perspectiva por iteração

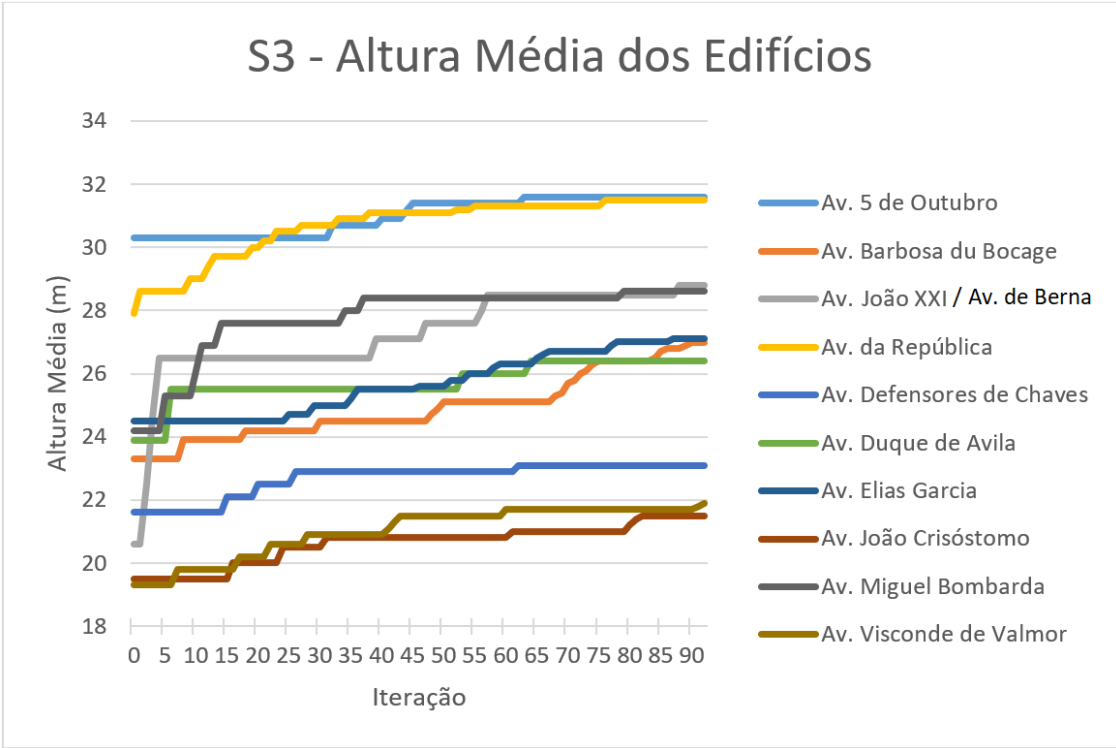


Figura 7.2 - Simulação S3, perspectiva por iteração

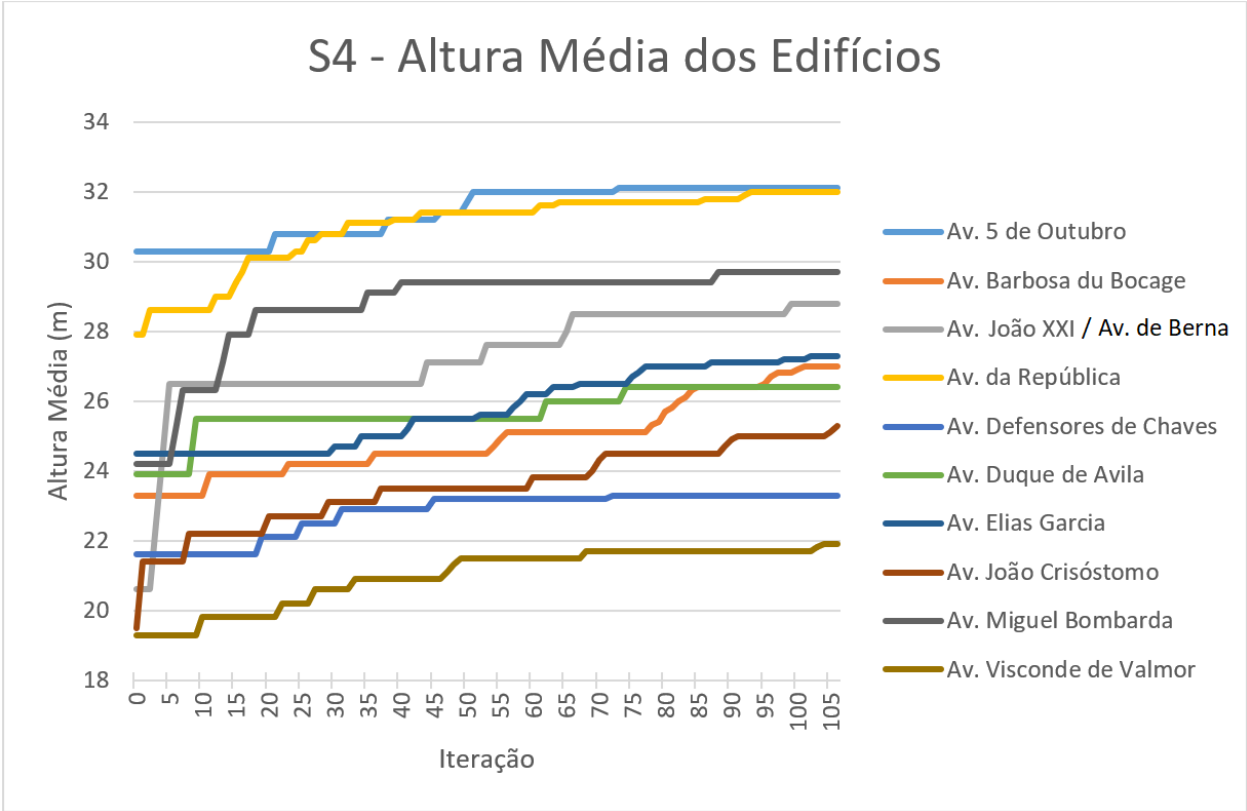


Figura 7.3 - Simulação S4, perspectiva por iteração