



Universidade Federal do Rio de Janeiro
Escola Politécnica
Programa de Engenharia Urbana

Halan Bastos Oliveira

APLICAÇÃO DA METODOLOGIA BIM E DE FERRAMENTAS GIS PARA
AUTOMAÇÃO DE MODELAGEM DE REDES DE DRENAGEM PLUVIAL
URBANA

Rio de Janeiro

2025

Halan Bastos Oliveira

APLICAÇÃO DA METODOLOGIA BIM E DE FERRAMENTAS GIS PARA
AUTOMAÇÃO DE MODELAGEM DE REDES DE DRENAGEM PLUVIAL
URBANA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Urbana, Escola Politécnica, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Urbana.

Orientador: José Luis Menegotto

Rio de Janeiro
Fevereiro de 2025

Oliveira, Halan Bastos

Aplicação da metodologia BIM e de ferramentas GIS para automação de modelagem de redes de drenagem pluvial urbana / Halan Bastos Oliveira. – 2025.

107 f. : 67 il. ; 30 cm.

Dissertação (mestrado) – Universidade Federal do Rio de Janeiro, Escola Politécnica, Programa de Engenharia Urbana, Rio de Janeiro, 2025.

Orientador: José Luis Menegotto

1. BIM. 2. GIS. 3. automação. 4. drenagem pluvial. I. Menegotto, José Luis. II. Universidade Federal do Rio de Janeiro. Escola Politécnica. III. Aplicação da metodologia BIM e de ferramentas GIS para automação de modelagem de redes de drenagem pluvial urbana.

Halan Bastos Oliveira

APLICAÇÃO DA METODOLOGIA BIM E DE FERRAMENTAS GIS PARA
AUTOMAÇÃO DE MODELAGEM DE REDES DE DRENAGEM PLUVIAL
URBANA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Urbana, Escola Politécnica, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Urbana.

Aprovada pela banca:

Prof. José Luis Menegotto, D.Sc., Universidade Federal do Rio de Janeiro

Prof. Marcelo Gomes Miguez, D.Sc., Universidade Federal do Rio de Janeiro

Prof. Giuseppe Miceli Junior, D.Sc., Instituto Militar de Engenharia

Dedico essa dissertação a Deus, que me sustentou e me deu forças na caminhada, à minha amada esposa Evelyn por exercitar sua paciência nesta minha jornada de dedicação aos estudos, ao meu filho Tomé por me dar forças a cada choro e sorriso e a cada batalha vencida em sua vida, ao meu filho Pedro que já amo desde antes de seu nascimento, e aos meus pais e irmãos que me incentivam desde sempre.

RESUMO

BASTOS OLIVEIRA, Halan. **APLICAÇÃO DA METODOLOGIA BIM E DE FERRAMENTAS GIS PARA AUTOMAÇÃO DE MODELAGEM DE REDES DE DRENAGEM PLUVIAL URBANA**. Rio de Janeiro, 2025. Dissertação (Mestrado). Programa de Engenharia Urbana, Escola Politécnica, Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2025.

Apesar do grande potencial do BIM para o setor de infraestrutura, a metodologia ainda apresenta limitações, especialmente no gerenciamento de empreendimentos de maiores dimensões lineares. À medida que projetos se tornam mais extensos ou suas topografias mais complexas, o BIM necessita do apoio de ferramentas GIS, pois estas têm o potencial de quantificar situações do mundo real e de focar em informações geoespaciais externas às edificações e em suas interações com o ambiente. Elementos espaciais mais complexos são considerados parte dos modelos GIS, enquanto componentes de construção mais detalhados participam do universo BIM, e ambos complementam-se com a presença de objetos comuns como as *Parcels*. Como forma de possibilitar a criação de ferramentas de projeto personalizadas e que permitam a integração de informações entre os domínios, a indústria AEC desenvolveu aplicativos de programação incorporados aos softwares, inclusive de linguagens de programação visual. No caso de projetos de drenagem pluvial urbana, que não podem ser replicados em escala devido às características topográficas únicas de cada local, o desafio é elaborar modelagens personalizadas para cada empreendimento. Nessa atividade, os valores de parâmetros como o *runoff* são frequentemente definidos de forma manual, sem o auxílio de ferramentas computacionais, resultando em imprecisões e mais demora na conclusão da tarefa. Dessa forma, este trabalho busca apresentar um algoritmo de automação de projetos de redes de drenagem pluvial urbana por meio da integração dos softwares Autodesk Civil 3D e QGIS. Como metodologia, foram desenvolvidos 4 algoritmos de automação utilizando as ferramentas nativas dos dois softwares e as linguagens de programação visual do Dynamo do Civil 3D e do Modelador Gráfico do QGIS, capazes de determinar as *Parcels* das bacias hidrográficas do

terreno, de calcular os coeficientes de *runoff*, e de modelar as tubulações e estruturas da rede de drenagem. O resultado da implementação do algoritmo em 2 estudos de casos foi a obtenção automática supervisionada dos valores de *runoff* de cada bacia, geradas a partir de polígonos com a indicação dos pontos de acúmulo de água, e dos modelos de redes de drenagem, em que cada estrutura coletora foi posicionada nesses pontos indicados. Diferente do identificado na literatura, em que os autores justificaram a necessidade da integração BIM-GIS no caso de infraestruturas extensas ou em ambientes urbanos complexos, este autor empregou GIS por provar-se uma ferramenta mais eficiente para lidar com dados adimensionais e bidimensionais oriundos de objetos geoespaciais, e de devolver os resultados ao software BIM sem perder os níveis de detalhes e de informações.

Palavras-chave: BIM. GIS. Infraestrutura. Drenagem. Automação. VPL.

ABSTRACT

BASTOS OLIVEIRA, Halan. **APPLICATION OF BIM METHODOLOGY AND GIS TOOLS FOR URBAN STORM DRAINAGE NETWORKS MODELING AUTOMATION**. Rio de Janeiro, 2025. Master's Thesis. Urban Engineering Program, Polytechnic School, Federal University of Rio de Janeiro. Rio de Janeiro, 2025.

In spite of the great potential of BIM for the infrastructure sector, the methodology still presents limitations, especially in managing larger linear projects. As projects become larger or their topography become more complex, BIM requires support of GIS tools, as these have the potential to quantify real-world situations and focus on buildings' external geospatial information and their interactions with the environment. More complex spatial elements are considered part of GIS models, while more detailed construction components belong to BIM universe, and both complement each other with presence of common objects such as Parcels. To enable the creation of customized design tools which allow information integration between domains, AEC industry has developed programming applications embedded in software, including Visual Programming Languages (VPLs). In the case of urban stormwater drainage projects, which cannot be replicated on a large scale due to unique topographic features of each location, the challenge is to develop customized models for each project. In this activity, parameters such as runoff coefficient values are often defined manually, without computational tools assistance, resulting in inaccuracies and delays in completing the task. Thus, this work aims to present an automating algorithm for urban stormwater drainage networks design through the integration of Autodesk Civil 3D and QGIS software. As a methodology, 4 automation algorithms were developed using native tools of both software and Dynamo for Civil 3D and Graphic Modeler for QGIS VPLs, capable of determining the Parcels of terrain's watersheds, calculating runoff coefficients, and modeling drainage network pipes and structures. Implementation of the algorithm in 2 case studies resulted in supervised automatic obtainment of runoff values for each basin, modeled from polygons with water discharge points indication, and pipe networks models, in which each discharge structure was positioned at these

points. Unlike literature review, where authors justified BIM-GIS integration necessity in the cases of large infrastructure or complex urban environments, this author employed GIS as it proved to be a more efficient tool for handling non-dimensional and two-dimensional data from geospatial objects, and for returning the results to BIM software without losing levels of detail and information.

Keywords: BIM. GIS. Infrastructure. Drainage. Automation. VPL.

Sumário

Lista de Figuras.....	12
Lista de Tabelas	15
Lista de Siglas.....	16
Capítulo 1.....	18
1 Introdução.....	18
1.1 Objetivo	21
1.1.1 Objetivo Geral	21
1.1.2 Objetivo Específico	21
1.2 Estratégia de pesquisa.....	22
1.3 Estrutura do Trabalho	23
Capítulo 2.....	24
2 Revisão Bibliográfica	24
2.1 Infraestruturas críticas	24
2.1.1 Riscos e ameaças	25
2.1.2 Resiliência	25
2.1.3 Integração BIM-GIS na prevenção e mitigação de desastres	27
2.1.4 Gestão dos riscos de inundações urbanas	27
2.2 A utilização do BIM em projetos.....	28
2.2.1 Impactos de projetos inadequados e estratégias	29
2.2.2 Emprego do BIM em projetos de infraestrutura.....	31
2.2.3 Modelo Digital da Superfície (MDS) e do Terreno (MDT)	33
2.3 Nível de informação necessária (LOIN).....	34
2.4 Alinhamentos.....	37
2.5 Formato IFC 4.3	39
2.5.1 A entidade <i>IfcAlignment</i>	40

2.5.2	A entidade <i>IfcSite</i>	42
2.6	Formato CityGML.....	44
2.7	Formatos Shapefile e SDF.....	46
2.8	Sistemas de Informação Geográfica (GIS).....	47
2.8.1	A utilização de ferramentas GIS em projetos de infraestrutura.....	47
2.8.2	Modelagem de Informação Geoespacial (GIM).....	49
2.9	Integração BIM-GIM.....	51
2.9.1	Os desafios da integração BIM-GIM	53
2.10	Programação visual.....	54
2.10.1	Modelo de execução de uma VPL.....	55
2.10.2	Aplicativo Dynamo para Civil 3D.....	58
2.10.3	A linguagem <i>Design Script</i>	60
Capítulo 3.....		62
3	Metodologia.....	62
Figura 33 - Fluxograma de atividades em um projeto de drenagem pluvial urbana. As cinco etapas são as mesmas apresentadas no		63
3.1	Softwares empregados.....	67
3.1.1	Ferramenta GIS	68
3.1.2	Ferramentas BIM.....	68
3.1.3	AutoCAD.....	70
3.2	Os algoritmos.....	71
3.2.1	Algoritmo nº 1 - Obtenção das bacias	73
3.2.2	Algoritmo nº 2 - Obtenção dos coeficientes de <i>runoff</i>	75
3.2.3	Algoritmo nº 3 - Atribuição dos coeficientes de <i>runoff</i> às bacias	80
3.2.4	Algoritmo nº 4 - Criação automatizada de tubulações	82
Capítulo 4.....		85
4	Aplicações práticas	85

4.1	Estudo de caso 1	85
4.2	Estudo de caso 2	86
Capítulo 5.....		90
5	Avaliação dos Resultados	90
Capítulo 6.....		93
6	Conclusões.....	93
Referências Bibliográficas.....		96
Apêndice A		102

Lista de Figuras

Figura 1 - Diversos tipos de superfície para uma única bacia, como grama, árvores, telhado	21
Figura 2 - Análise das janelas de tempo antes, durante e após os eventos de inundação para altos níveis de resiliência com uma abordagem de gerenciamento sistêmica (a), altos níveis de resiliência com uma abordagem baseada em silo (b) e baixos níveis de resiliência com uma abordagem baseada em silo (c). O atraso ou a falta de reparo/substituição prejudica significativamente a recuperação pós-evento (c).	26
Figura 3 - Principais motivos para a paralisação das obras no Brasil.	30
Figura 4 - Estatísticas do Painel de Obras Paralisadas em novembro de 2024.	31
Figura 5 - Modelo Digital do Terreno (MDT).	33
Figura 6 - Esquema da diferença entre MDS e MDT.	34
Figura 7 - Exemplos de LOD conforme classificação da AIA.	35
Figura 8 - Diferentes LODs para um projeto de rodovia no Civil 3D.	38
Figura 9 – (a) Alinhamento de uma rede de tubulação no Civil 3D e (b) vista do perfil.	38
Figura 10 - Alinhamentos de redes de tubulações de drenagem selecionados no Civil 3D	41
Figura 11 - Perfil e vista de perfil longitudinais de um dos alinhamentos da Figura 10	42
Figura 12 - Posicionamento e elevações de um Site	43
Figura 13 – <i>Sites</i> (a) e <i>Parcels</i> (c) na extensão .dwg do Civil 3D; e dos <i>sites</i> (b) e <i>Parcels</i> (d) na extensão .ifc na versão IFC 4.3.	43
Figura 14 - Bacias como <i>Parcels</i> .	44
Figura 15 - Gráfico dos diferentes níveis LOD para o GIS conforme o padrão <i>CityGML</i> .	46
Figura 16 - Representação de um <i>shapefile</i> com os 3 formatos de arquivo (SHP, .shx e .dbf).	47
Figura 17 - Configurações de exportação do projeto para SDF.	47
Figura 18 - Sobreposição de camadas de informações num software GIS.	48
Figura 19 - Visualização de edificações no formato 3D em software GIS.	49
Figura 20 - GIM e BIM e seus objetos comuns.	50
Figura 21 - B-Rep (a), <i>Swept Solid</i> (b) e CSG (c).	51
Figura 22 - Estrutura básica do Dynamo para Civil 3D	55

Figura 23 - Instruções para operações matemáticas simples (a) e seu fluxograma equivalente (b).	56
Figura 24 - Nó do Dynamo para Civil 3D para criação de polígono a partir de pontos.	57
Figura 25 - Sequência de comandos em texto em caixas de código dentro do Dynamo para Civil 3D para criação de um polígono a partir de pontos em sequência.	58
Figura 26 - Interface do Dynamo para Civil 3D.	58
Figura 27 - Localização do Dynamo dentro do Civil 3D.	58
Figura 28 - Nós da biblioteca do Dynamo.	59
Figura 29 - Pacote Civil3DToolkit 1.1.32 disponível na biblioteca da Autodesk.	60
Figura 30 - Esquema de cores utilizado pelo autor (em ordem, da esquerda para a direita: entrada, obtenção de dados, processamento de dados, criação de objetos e saída).	60
Figura 31 - <i>Code Block</i> , a “unidade mínima” da linguagem visual do Dynamo.	61
Figura 32 - Algoritmo para obter <i>Parcels</i> e seus respectivos nomes.	61
Figura 33 - Fluxograma de atividades em um projeto de drenagem pluvial urbana. As cinco etapas são as mesmas apresentadas no	63
Figura 34 - Analogia da polia andando sobre uma corda numa tirolesa.	63
Figura 35 - <i>Catchment</i> com o perímetro em azul e a linha de fluxo de água em amarelo com os dados de comprimento, inclinação e tempo de concentração.	66
Figura 36 - Ferramentas de análise pluviométrica e de drenagem no Civil 3D.	66
Figura 37 - Superfície TIN com os pontos em vermelho e as arestas em azul.	67
Figura 38 - <i>Watersheds</i> no Civil 3D com as setas de indicação do fluxo de água.	67
Figura 39 - QGIS 3.34.13 - Prizren.	68
Figura 40 - Autodesk Civil 3D 2024.	69
Figura 41 - Interface do Civil 3D 2024.	70
Figura 42 – Arquivo .dwg com as formas geométricas representativas das superfícies do terreno.	71
Figura 43 - Fluxograma das tarefas executadas. Em ciano as entradas, em rosa os arquivos no formato .dwg do Civil 3D e do AutoCAD, e em roxo os arquivos do QGIS.	72
Figura 44 - Propriedade <i>Runoff</i> da <i>Parcel</i> .	73
Figura 45 - Algoritmo nº 1 de obtenção e exportação das bacias.	74
Figura 46 - Polígonos de bacias e pontos de acumulação de água.	75
Figura 47 – 1ª etapa do algoritmo nº 2 de exportação das formas das superfícies e exportação como <i>shapefile</i> .	76

Figura 48 – 2ª etapa do algoritmo nº 2 para a criação da propriedade <i>Runoff</i> e exportação das bacias hidrográficas como <i>shapefile</i>	77
Figura 49 - Feições carregadas no QGIS.	78
Figura 50 - Algoritmo do modelador gráfico do QGIS.	78
Figura 51 – 3ª etapa do algoritmo nº 2 de cálculo do atributo <i>Runoff</i> na feição <i>shapefile</i> com as bacias.	79
Figura 52 - Algoritmo nº 2 em linguagem Python.	80
Figura 53 - Feições carregadas no Civil 3D.	81
Figura 54 - Algoritmo nº 3 de atribuição dos valores de <i>Runoff</i> das feições SHP aos objetos do tipo <i>Parcel</i> .	82
Figura 55 - Algoritmo nº 3 em linguagem em texto <i>Design Script</i> .	82
Figura 56 - Polilinha na cor magenta indicando os locais de inserção dos objetos a serem modelados.	83
Figura 57 - Dados a serem inseridos como entradas para o Algoritmo nº 4.	84
Figura 58 - <i>Watersheds</i> observadas.	85
Figura 59 - Estruturas coletoras da rede de tubulação.	86
Figura 60 - Polígonos 3D das bacias com indicação dos pontos de acúmulo de água.	87
Figura 61 - Bacias como <i>shapefiles</i> no QGIS.	88
Figura 62 - Atributo <i>Runoff</i> calculado.	88
Figura 63 - Bacias com os valores de <i>Runoff</i> calculados e atribuídos.	89
Figura 64 - Modelagem finalizada das redes, em linha verde tracejada, no Civil 3D.	89
Figura 65 - Processamento demorado pela quantidade de <i>Watersheds</i> .	90
Figura 66 - Tabela de tubos do Civil 3D com indicação de estruturas à montante e à jusante de cada tubo.	91
Figura 67 - Vista de perfil horizontal de um trecho da rede de tubulação gerada.	92

Lista de Tabelas

Tabela 1 - Diferenças entre o BIM para infraestrutura e edificações.

32

Lista de Siglas

2D. *Dois dimensões*

3D. *Três dimensões*

ABNT. *Associação Brasileira de Normas Técnicas*

ADE. *Application Domain Extensions*

AEC. *Arquitetura, Engenharia e Construção*

AIA. *American Institute of Architects*

BIM. *Building Information Modeling*

B-Rep. *Boundary Representation*

BSI. *buildingSMART International*

CityGML. *City Geography Markup Language*

CSG. *Constructive Solid Geometry*

DNIT. *Departamento Nacional de Infraestrutura*

EUA. *Estados Unidos da América*

FIFO. *First In, First Out*

GIM. *Geospatial Information Modeling*

GIS. *Geographic Information System*

IA. *Inteligência Artificial*

IFC. *Industry Foundation Classes*

ISO. *International Organization for Standardization*

IUCs. *Infraestruturas Urbanas Críticas*

LOd. *Level of Detail*

LOD. *Level of Development*

LOI. *Level of Information*

MDS. *Modelo Digital da Superfície*

MDT. *Modelo Digital do Terreno*

MVDs. *Model View Definitions*

OAE. *Obra de Arte Especial*

OGC. *Open Geospatial Consortium*

Planejamento 4D. *Planejamento de tempo*

Planejamento 5D. *Planejamento de custo*

SDF. *Spatial Data File*

SIG. *Sistema de Informação Geográfica*

SIG 3D. *Special Interest Group 3D*

SINAPI. *Sistema Nacional de Pesquisa de Custos e Índices da Construção Civil*

TCU. *Tribunal de Contas da União*

TIN. *Triangular Irregular Networks*

UNECE. *United Nations Economic Commission for Europe*

VPL. *Visual Programming Language*

Capítulo 1

1 INTRODUÇÃO

As cidades modernas são atualmente definidas pela literatura como “sistemas de sistemas”, onde os “sistemas” são as Infraestruturas Urbanas Críticas (IUCs) (Arrighi; Pregnolato; Castelli, 2021).

Atualmente, 55% da população global é urbanizada, em comparação com 30% em 1950, o que resulta em desafios e pressões crescentes sobre os Sistemas de Infraestruturas Urbanas, exigindo medidas de sustentabilidade. Além disso, o aumento das demandas, a poluição das fontes, a deterioração da infraestrutura e as mudanças climáticas têm levado à limitação desses sistemas, juntamente com comportamentos sociais que são determinantes para a sustentabilidade das Infraestruturas Urbanas (Polonenko; Hamouda; Mohamed, 2020).

Para Arrighi, Pregnolato e Castelli (2021) quando os impactos sobre as IUCs são devidos ao contato físico com as águas da inundação, são denominados "impactos diretos". Quando os impactos não são devidos ao contato físico e/ou ocorrem fora da área inundada no espaço ou no tempo, são denominados "impactos indiretos ou em cascata". Impactos indiretos e efeitos em cascata são amplamente considerados como mais significativos quando há uma interconexão entre redes, porém poucos estudos abordam esses impactos e efeitos no tempo e no espaço, principalmente nos casos de estruturas conceituais complexas.

Ainda mais difícil do que encontrar espaço para construir arranha-céus cada vez mais altos é encontrar espaço para construir novas infraestruturas urbanas subterrâneas. O espaço subterrâneo urbano, quando iniciadas as construções das primeiras infraestruturas enterradas, foi inicialmente preenchido por canalizações, redes de água, coletores de esgoto e redes de túneis de fuga ou de acesso rápido. Além disso, parte dessas tubulações foi construída sem qualquer plano diretor ou registros em planta (Campos, *et al.*, 2006).

A aplicação da metodologia BIM (*Building Information Modeling*, ou Modelagem da Informação da Construção) em projetos de infraestrutura urbana tem possibilitado aos

engenheiros projetistas a criação de modelos virtuais precisos de construções, com geometria exata e dados relevantes para dar suporte à construção, além da extração dos insumos necessários para a realização da obra.

A partir de modelos criados, como a modelagem BIM de redes de drenagem pluvial existentes, os engenheiros podem traçar com exatidão novas infraestruturas urbanas enterradas, realizar simulações de diversos cenários possíveis e certificarem-se de que não haverá interferências ou outras surpresas durante a execução das obras, gerando desenhos bidimensionais, ou em 2D, precisos e consistentes em qualquer etapa do projeto (Eastman, *et al.*, 2014).

A falta de softwares BIM desenvolvidos para modelagem de infraestruturas lineares ressalta a necessidade de interoperabilidade no desenvolvimento de modelos, o que requer uma combinação de várias ferramentas para entregar uma modelagem completa. A interoperabilidade, definida como a capacidade dos aplicativos BIM de trocar dados, permite essa integração de softwares e a automação de fluxo de trabalho (Nielsen, *et al.*, 2024).

Quanto ao padrão de arquivos utilizados na modelagem BIM, a ISO (*International Organization for Standardization*) normatizou o *Industry Foundation Classes* (IFC) como um padrão internacional aberto, cujos direitos não podem ser detidos por nenhum desenvolvedor de software, sendo utilizável em diversos softwares e interfaces em diversas aplicações. Apesar de ser vocacionado à arquitetura e edificações, o padrão mais recente da extensão, o IFC 4.3, lançado pela BSI (*buildingSMART International*) em abril de 2020, é atualmente a principal atualização do esquema para a indústria de infraestrutura civil.

Ainda em contraste à essa vocação às edificações, o crescimento no número de publicações centradas em infraestrutura rodoviária ressalta o crescente interesse global em BIM dentro do setor de infraestrutura. Além disso, os números também refletem a crescente importância atribuída à metodologia por órgãos governamentais, universidades e outras organizações (Nielsen, *et al.*, 2024).

O BIM gerencia informações muito ricas e detalhadas dos pontos de vista geométrico e semântico de uma construção, no entanto, não inclui informações das construções do entorno, sendo que por vezes são necessárias informações espaciais para análises ambientais, de locação ou de segurança de sistemas maiores ou mais complexos. Por exemplo, informações topográficas georreferenciadas, que são essenciais para o traçado de rodovias e de redes de drenagem pluvial, são acessadas mais facilmente no ambiente GIS (*Geographic Information System*) (Liu, *et al.*, 2017).

Pelas características específicas de cada metodologia, também denominada domínio neste trabalho, a integração BIM-GIS possibilita o gerenciamento mais eficiente de informações nos diversos estágios do ciclo de vida de um projeto de infraestrutura.

Apesar das facilidades oferecidas pelas ferramentas BIM e GIS, o maior desafio de projetos de drenagem urbana e de outros projetos de saneamento é que não podem ser replicados em escala, visto que cada local, por possuir características únicas, como topografia, traçado de vias e infraestrutura enterrada existente, demanda um projeto único e extremamente personalizado (Barbosa, 2017).

Dentro desse desafio da elaboração de projetos personalizados, a determinação do coeficiente adimensional de escoamento superficial de bacias hidrográficas, também conhecido como coeficiente de *runoff*, é frequentemente realizada de forma manual, sem o auxílio de ferramentas computacionais. Esse procedimento pode resultar em imprecisões e aumentar a demora e complexidade na elaboração de um projeto de drenagem pluvial urbana.

Além do uso de ferramentas BIM para a modelagem de redes subterrâneas, que por si só aumentariam a produtividade dos projetos (Eastman, *et al.*, 2014), Lua e Milford (2019) notaram que é possível aplicar algoritmos em linguagem de programação visual que permitem automatizar o fluxo de atividades em um projeto de infraestrutura utilizando o aplicativo Dynamo do software Civil 3D da empresa Autodesk, o que pode aumentar ainda mais a produtividade.

Para o caso da definição de escoamento superficial de bacias hidrográficas, a complexidade na definição dessas áreas, conforme ilustrado na Figura 1, e a monotonia na repetição dessa tarefa ao longo do projeto podem desmotivar os projetistas, que frequentemente estimam os valores com base apenas na observação visual, sem a aplicação de técnicas estabelecidas na literatura. Esse parâmetro é obtido a partir das médias ponderadas dos coeficientes predefinidos para cada tipo de superfície, como grama, asfalto, concreto ou telhado cerâmico, e a área de cada um desses tipos dentro dos limites da bacia.

Figura 1 - Diversos tipos de superfície para uma única bacia, como grama, árvores, telhado



Fonte: Autor (2025).

1.1 OBJETIVO

1.1.1 Objetivo Geral

Este estudo tem por objetivo apresentar um algoritmo de modelagem e dimensionamento automatizados de redes de drenagem pluvial urbana por meio da integração da metodologia BIM e de ferramentas GIS utilizando linguagens de programação visual.

1.1.2 Objetivo Específico

Dentro desse objetivo geral, encontram-se as seguintes metas específicas:

- Obter de forma automatizada os coeficientes de escoamento superficial das bacias da área em estudo por meio dos softwares Autodesk Civil 3D e QGIS, lançando mão das linguagens de programação visual do Dynamo e do Modelador Gráfico, respectivamente;
- Apresentar um algoritmo de modelagem BIM automatizada de tubulações de drenagem pluvial urbana utilizando o Dynamo; e
- Verificar e analisar os resultados da aplicação dos algoritmos de automatização apresentados a 2 estudos de casos.

1.2 ESTRATÉGIA DE PESQUISA

O método desenvolvido para solucionar os problemas levantados no objetivo geral foi dividida nas seguintes etapas específicas:

- Na Etapa n° 1, foi criado um arquivo no formato .dwg no software Autodesk Civil 3D®, contendo o Modelo Digital do Terreno (MDT) do local que terá as bacias analisadas. Desse arquivo, a partir do uso de ferramentas nativas do Civil 3D, foram obtidos polígonos representativos das bacias e círculos representativos dos pontos de acúmulo de água por meio do algoritmo n° 1.
- Na Etapa n° 2, foi elaborado outro arquivo no formato .dwg, dessa vez no software AutoCAD®, contendo formas geométricas que representavam os diversos tipos de superfícies que compõem a área em estudo.
- Na Etapa n° 3, para cada polígono obtido no algoritmo n° 1 foi atribuída a propriedade denominada *runoff*, que armazenou valores arbitrários prévios para este parâmetro. Esses polígonos, juntamente com as formas geométricas representativas dos tipos de superfície, foram carregados no software QGIS como entradas para o algoritmo n° 2. No QGIS, o algoritmo n° 2, escrito na linguagem de programação visual do Modelador Gráfico do software, fez a leitura das entradas, realizou a partição dessas formas a partir dos polígonos e obteve o coeficiente de escoamento superficial ponderado de cada bacia.
- Na Etapa n° 4, os resultados obtidos no QGIS foram carregados no arquivo .dwg criado na primeira etapa, e o algoritmo n° 3, escrito na linguagem de programação visual do aplicativo Dynamo® para Civil 3D, atribuiu às bacias os novos valores calculados do parâmetro *runoff*. Com os novos valores desse parâmetro, a vazão da chuva em cada bacia pôde ser calculada.
- Na Etapa n° 5, ainda no arquivo criado no Civil 3D, foram desenhadas polilinhas sobre a superfície que representavam o traçado dos coletores tronco da rede de drenagem. Esses objetos, combinados com os círculos dos pontos de acúmulo de água, foram lidos pelo algoritmo n° 4, também escrito na linguagem de programação visual do Dynamo, que criou objetos parametrizados da ferramenta de criação de redes de drenagem pluvial do Civil 3D, sendo: bocas de lobo localizadas em cada círculo, poços de visita localizados nos vértices das polilinhas, tubulações nos ramais entre cada

boca de lobo e o poço de visita mais próximo, e tubulações conectando os poços de visita entre si. O algoritmo nº 4 também fez ajustes nas elevações das tubulações, de modo que o fluxo da água sempre escoasse por gravidade.

1.3 ESTRUTURA DO TRABALHO

A fim de alcançar os objetivos propostos, o trabalho foi dividido nos seguintes capítulos:

No Capítulo 2 é apresentada a revisão bibliográfica a respeito do assunto, onde são abordadas as definições de infraestruturas críticas, além da gestão dos riscos de inundações urbanas. No mesmo capítulo, são apresentados conceitos da metodologia BIM, incluindo as definições de nível de detalhe e de desenvolvimento, e de que forma ela é empregada em projetos de infraestrutura urbana, a integração entre as siglas BIM e GIS, além de uma introdução à Linguagem de Programação Visual, do inglês *Visual Programming Language* (VPL).

No Capítulo 3 é demonstrada a metodologia, com a apresentação dos algoritmos de VPL desenvolvidos, bem como das ferramentas dos softwares utilizados para a obtenção dos resultados.

No Capítulo 4 são apresentados dois estudos de caso, um realizado em Recife e outro para uma situação hipotética, com a comparação entre os resultados obtidos manualmente e os resultados gerados a partir dos algoritmos de automação.

No Capítulo 5 é apresentada a análise dos resultados da aplicação dos algoritmos desenvolvidos.

Por fim, no Capítulo 6 são apresentadas as conclusões e as propostas para trabalhos futuros. Em seguida, são apresentadas as referências bibliográficas utilizadas no estudo.

Capítulo 2

2 REVISÃO BIBLIOGRÁFICA

2.1 INFRAESTRUTURAS CRÍTICAS

Infraestruturas críticas podem ser definidas como “instalações, serviços, bens e sistemas cuja interrupção ou destruição, total ou parcial, provoque sério impacto social, ambiental, econômico, político, internacional ou na segurança do Estado e da sociedade”. Os danos ou destruição de infraestruturas críticas por catástrofes naturais, terrorismo e atividades criminosas podem ter consequências negativas para a segurança do país e o bem-estar dos seus cidadãos (Brasil, 2020).

Essas estruturas estão sujeitas a riscos associados a ameaças físicas e desastres naturais, e mais recentemente também expostas a riscos cibernéticos. Esses riscos resultam de uma integração crescente de tecnologias de informação e comunicação com infraestrutura crítica e adversários focados na exploração de potenciais vulnerabilidades cibernéticas. À medida que a infraestrutura física se torna mais dependente de sistemas cibernéticos complexos para operações, a infraestrutura crítica pode se tornar mais vulnerável a certas ameaças cibernéticas, incluindo ameaças estrangeiras (EUA, 2019).

Dentre as infraestruturas críticas de uma nação podem estar: rede de comunicações, fábricas, barragens, indústria de defesa, serviços de emergência, redes de distribuição de energia elétrica, serviços financeiros, alimentos e agricultura, instalações do governo, sistema de saúde, tecnologia da informação, usinas e materiais nucleares e radioativos, transportes e sistemas de abastecimento e tratamento de água. Uma infraestrutura crítica deve ser resiliente, robusta, ágil e adaptável (Justo, *et al.*, 2021)..

Nos Estados Unidos da América (EUA), provavelmente em razão do desenvolvimento econômico e do histórico de guerras e ameaças terroristas, comumente enfrentadas por aquele país, a conscientização sobre o tema de infraestrutura crítica iniciou-se bem antes que no Brasil. O governo norte-americano, logo após os atentados terroristas de 11 de setembro de 2001, publicou uma série de diretrizes de segurança interna, entre as quais havia a elaboração de um

plano nacional abrangente para garantir a segurança de infraestruturas críticas por meio de cooperação das autoridades e das agências federais, regionais e locais, além do setor privado e de outras entidades (Brasil, 2020).

Rocha (2019) destaca que, devido à cultura pacifista de seu povo e ao fato de o país não ser alvo de ações terroristas graves, no Brasil não há uma conscientização forte em prevenção, segurança e defesa. Apesar de felizmente o estado não vivenciar os dissabores de uma guerra, ainda assim é importante a defesa dessas infraestruturas, pois, mesmo que não haja oponentes atacando ativos para gerar dificuldades logísticas e operacionais, causando efeitos psicológicos negativos na população, há a necessidade do envolvimento do Exército Brasileiro na defesa dessas estruturas, ainda que por meio do apoio em combate aos efeitos de desastres materiais e humanos causados por inundações.

2.1.1 Riscos e ameaças

Os EUA (EUA, 2019) referem-se ao risco como o potencial de materialização de um resultado indesejado decorrente de um incidente, acidente ou evento determinado pela sua probabilidade e pelas consequências associadas. O gerenciamento de risco é o processo de identificar, analisar e comunicar esses riscos, passando pela decisão de aceitá-los, evitá-los, transferi-los ou controlá-los a um nível aceitável, alocando recursos nas ações destinadas a prevenir ou mitigar os efeitos das ameaças e perigos que têm maior probabilidade de causar resultados indesejados significativos em uma infraestrutura.

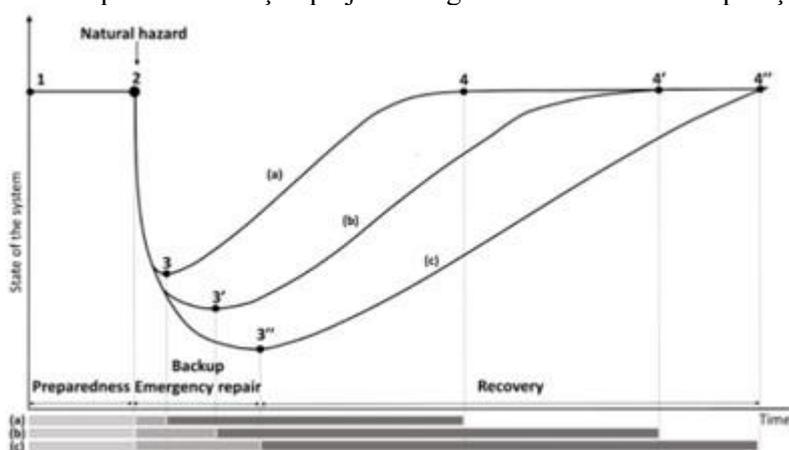
Arrighi, Pregolato e Castelli (2021) elucidam que quatro fatores compõem os riscos: ameaça, exposição, vulnerabilidade e consequências. Uma infraestrutura pode responder a uma ameaça de várias maneiras: (i) absorvendo o impacto e minimizando as consequências; (ii) ajustando-se a condições não ótimas e fornecendo um desempenho inferior; (iii) falhando em algumas ou todas as suas partes, sem afetar outros serviços; e (iv) falhando em algumas ou todas as suas partes e causando falhas em cascata em outros serviços.

2.1.2 Resiliência

Resiliência é entendida como a capacidade de se preparar e se adaptar a mudanças e perturbações em um sistema, ou seja, ser capaz de resistir e se recuperar rapidamente de interrupções, ataques, acidentes ou incidentes naturais (EUA, 2019). Para Liu e Zhaoyang (2020), resiliência refere-se à capacidade de se adaptar a condições em constante mudança, resistir e se recuperar no menor espaço de tempo de interrupções causadas por emergências.

Segundo Arrighi, Pregnotato e Castelli (2021), o tempo de resposta é crucial para a resiliência às ameaças naturais; especialmente em um contexto de inundação, uma intervenção adequada pode seguir planos de emergência que são ativados por sistemas de alerta precoce. A identificação de estações de bombeamento e de ruas mais vulneráveis a alagamentos, por exemplo, permite que se planeje com antecedência a reforma de estações de bombeamento ou que se desenvolvam planos de emergência, diminuindo os efeitos em cascata (Figura 2).

Figura 2 - Análise das janelas de tempo antes, durante e após os eventos de inundação para altos níveis de resiliência com uma abordagem de gerenciamento sistêmica (a), altos níveis de resiliência com uma abordagem baseada em silo (b) e baixos níveis de resiliência com uma abordagem baseada em silo (c). O atraso ou a falta de reparo/substituição prejudica significativamente a recuperação pós-evento (c).



Fonte: Adaptado de Arrighi, Pregnotato e Castelli (2021).

Conforme observado por Mattsson e Jenelius (2015) e Elco *et al.* (2019), a resiliência da infraestrutura crítica está fortemente relacionada à sua capacidade de se adaptar às consequências das chuvas intensas. Em projetos de construção rodoviária ou manutenção de estradas vicinais, estima-se que a adaptação do sistema de drenagem para uma maior frequência pluviométrica, ou para reduzir pela metade os danos financeiros esperados, resulte em um impacto financeiro de aproximadamente 2% do orçamento. No entanto, para estradas pavimentadas existentes, o redimensionamento dos parâmetros de chuva exigiria a reconstrução de seções inteiras da rodovia apenas para substituir os sistemas de drenagem (Elco, *et al.*, 2019).

Head (2014) destaca que a literatura sobre desenvolvimento sustentável sugere que a extensão e a urgência na resolução dos problemas são intensificadas em uma era de mudanças climáticas, juntamente com as demandas crescentes por água, energia e segurança alimentar em toda a comunidade global.

Além do tempo de resposta após ameaças, a inércia institucional e as práticas incorporadas limitam o reconhecimento antecipado dos problemas, restringem as escolhas e inibem novos pensamentos. Essa tendência para a institucionalização de soluções preferidas e o conceito de

"sempre foi feito assim" pode ser desafiada por choques externos ou pelo surgimento de novos problemas para os quais o antigo paradigma não oferece flexibilidade e proteção suficientes. Essas oportunidades para novos pensamentos e aprendizagem adaptativa nem sempre são aproveitadas, dependendo das circunstâncias locais, incluindo o grau de abertura no sistema de políticas e o envolvimento das partes interessadas na gestão de questões. Assim, o problema do pensamento de resiliência levanta uma série de questões importantes sobre a eficácia e a adaptabilidade do sistema político.

2.1.3 Integração BIM-GIS na prevenção e mitigação de desastres

As inundações sempre representaram uma séria ameaça à segurança das áreas urbanas. Essas ameaças geram como consequências tanto danos físicos às infraestruturas e populações urbanas quanto subsequente contaminação de pessoas afetadas pelos poluentes das cheias (Cao, *et al.*, 2023).

Do ponto de vista da prevenção e mitigação de desastres, é fundamental prever os eventos, as consequências e enumerar as ações a realizar de maneira antecipada.

Com base em capacidades de simulação proporcionadas pelo BIM e demonstração ambiental espacial 3D em GIS, a aplicação integrada BIM-GIS pode simular cenários de desastres de diferentes intensidades de forma eficaz, o que auxilia as autoridades competentes na realização de avaliações dos danos causados e no delineamento das áreas afetadas (Cao, *et al.*, 2023).

2.1.4 Gestão dos riscos de inundações urbanas

O risco de inundação se concretiza quando ocorrem chuvas intensas. Quando esses eventos acontecem em áreas urbanas, geram consequências negativas para a população afetada, tanto em termos materiais quanto na ameaça à saúde humana (Bertilsson, *et al.*, 2019). No Brasil, as enchentes causaram danos no valor de US\$ 10,16 bilhões e resultaram em 3.591 mortes entre 2001 e 2024 (EM-DAT, 2024). Vale ressaltar que nessa estatística ainda não haviam sido contabilizadas as perdas relativas aos eventos climáticos no sul do país em meados de 2024.

A gestão do risco de inundações urbanas preocupa-se com as variáveis socioeconômicas impactadas pela inundação, combinando medidas estruturais e não estruturais para prevenir, mitigar, preparar, responder e recuperar-se de inundações, resultando em economia de gastos públicos e em cidades mais saudáveis, com uma relação mais harmoniosa entre os ambientes

natural e construído, além de melhor qualidade de vida para a população (Bertilsson, *et al.*, 2019).

Costin *et al.* (2018) mencionam que o BIM pode ser aplicado como uma ferramenta sistemática de gestão de riscos e de controle e melhoria da segurança. A gestão de riscos e a análise de segurança são viáveis por meio da fase de planejamento de tempo do projeto, ou 4D, onde a ordem e a duração das tarefas são definidas e justificadas. A aplicação do BIM na fase de projeto pode auxiliar na identificação de incertezas na construção de infraestruturas, como redes de drenagem e esgoto, evitando o método da "tentativa e erro" na fase de construção, aumentando a produtividade e diminuindo os riscos associados ao tempo e custo.

2.2 A UTILIZAÇÃO DO BIM EM PROJETOS

Durante muito tempo, a indústria da construção civil enfrentou desafios como a redução do tempo de construção, o aumento da precisão dos projetos e a resolução de problemas de compatibilização entre diferentes disciplinas. A metodologia BIM surgiu como uma revolução tecnológica, oferecendo os recursos necessários para superar esses desafios. O BIM, conforme definido pelo Comitê de Projeto da *National BIM Standard* dos EUA, é “uma representação digital das características físicas e funcionais de uma instalação; um recurso de conhecimento compartilhado para informações sobre uma instalação, formando uma base confiável para decisões durante seu ciclo de vida, desde a concepção inicial até a demolição”.

Desde seu surgimento, o BIM tem sido amplamente utilizado na indústria de edificações devido à conscientização da importância de sua implementação. Além dos principais benefícios da metodologia para modelagem de projetos, estudos de viabilidade ou detecção de interferências, ele contribui significativamente para outros setores da construção, como segurança, gerenciamento de instalações ou reabilitação de edifícios históricos. As principais contribuições ocorrem nas fases de projeto, orçamentação, construção e gestão de instalações.

De acordo com Costin *et al.* (2018), o termo BIM tem vários usos e conotações, sendo definido como um produto, um processo colaborativo e um requisito de gerenciamento do ciclo de vida de uma instalação. O objetivo do BIM era capturar todas as informações e aspectos do projeto e construção de uma edificação, para que pudessem ser utilizados na operação e manutenção. O BIM não deve ser visto apenas como um modelo 3D de uma construção com recursos e funções adicionais. O BIM trata de informações da construção, e o modelo 3D é apenas uma forma de representar essas informações. Portanto, a criação, o compartilhamento e o gerenciamento centralizados de informações introduziram uma nova mudança de paradigma

na indústria de Arquitetura, Engenharia e Construção (AEC), movendo a abordagem de projeto tradicional para uma colaboração mais centrada em informações.

Ainda segundo Basta, Serror e Marzouk (2020), os modelos BIM têm a capacidade de mostrar as relações entre os elementos de construção. À medida que o projetista atualiza o modelo, todas as relações entre os elementos de um edifício são automaticamente atualizadas, facilitando para as disciplinas de projeto a atualização de seus modelos e a correção de erros de compatibilização.

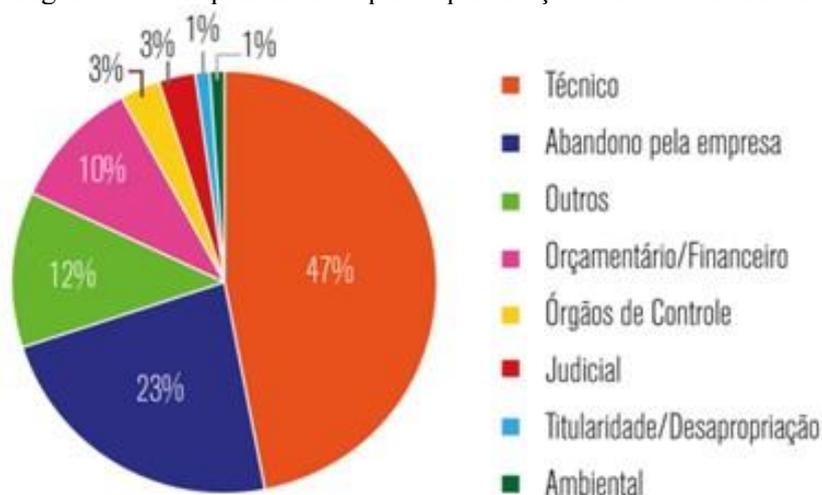
Da mesma forma, o BIM contribui diretamente para a modelagem detalhada dos elementos e componentes com todas as informações necessárias na fase de construção, e fornece melhor comunicação e envolvimento de todas as partes interessadas do projeto no início do processo de modelagem, garantindo que a construtibilidade seja integrada ao projeto.

Outra vantagem significativa do uso do BIM é que ele permite aos projetistas e empreiteiros a identificação permanente de todos os componentes da obra. Além disso, aumenta a capacidade de enriquecer o nível de informações do modelo, permitindo a inclusão de mais parâmetros ou categorias para cada elemento, como fabricante, peso ou preço. Cálculos automatizados de desempenho e mudanças de modelo permitem coordenar e estudar diferentes opções de projeto, possibilitando tomar a decisão correta com antecedência, simulando diferentes aspectos do comportamento da construção.

2.2.1 Impactos de projetos inadequados e estratégias

Uma das causas para a falta de infraestrutura urbana adequada é a paralisação de obras públicas. Segundo o Acórdão 1.079/2019 do Tribunal de Contas da União (TCU) (Brasil, 2019), o principal fator para a paralisação de obras públicas são questões técnicas envolvendo projetos básicos deficientes ou inadequados (Figura 3).

Figura 3 - Principais motivos para a paralisação das obras no Brasil.



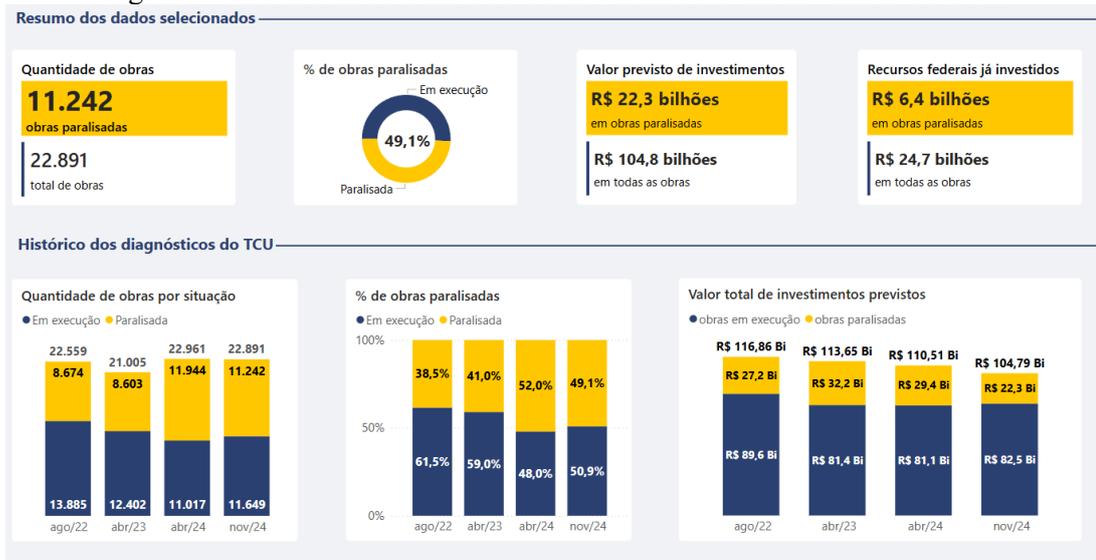
Fonte: Brasil (2019).

Além de projetos deficitários serem o principal motivo para a paralisação, em um universo de 1.688 auditorias realizadas pelo TCU em obras públicas entre 2008 e 2018, o termo "Projeto básico deficiente, inexistente ou desatualizado" foi mencionado em 1.158 dessas auditorias, sendo o segundo problema mais frequente em todas as obras públicas no país. Os danos ao erário e a limitação do crescimento econômico do país são consequências diretas dos projetos deficitários e, conseqüentemente, das obras paralisadas no Brasil (Brasil, 2019).

Os prejuízos decorrentes da interrupção de uma obra compreendem, além dos valores já aplicados em sua execução, outros que são de difícil mensuração. Uma obra não concluída no tempo certo consome os recursos nela aplicados sem gerar retorno para a sociedade. Além de limitar o crescimento econômico do país, por interromper a movimentação da economia local com a restrição de empregos diretos e indiretos gerados (Brasil, 2019).

O levantamento do "Painel de Obras Paralisadas" do TCU (Brasil, 2025), que reúne informações atualizadas sobre a execução de obras custeadas com recursos federais, aponta que, das 22.891 obras contratadas em andamento no Brasil em novembro de 2024, o percentual de empreendimentos paralisados era de 49,1%, ou 11.242 obras. A ineficiência na execução dos objetos contratados poderia gerar, à época da última atualização do Painel, um prejuízo aos cofres públicos federais de R\$ 6,4 bilhões, relativo aos valores já medidos em obras inacabadas (Figura 4).

Figura 4 - Estatísticas do Painel de Obras Paralisadas em novembro de 2024.



Fonte: Brasil (2025).

Uma das estratégias para aumentar a eficiência dos projetos é a utilização do BIM. De acordo com Eastman *et al.* (2014), pesquisas sobre ganhos de produtividade na produção de plantas de detalhamento de projeto de estruturas utilizando o BIM indicaram ganhos entre 21% e 59%, dependendo das características das estruturas. Em um estudo de caso da implementação dessa metodologia no projeto de construção de um centro médico em *Mountain View*, Califórnia, EUA, mesmo com atrasos causados pela inexperiência dos projetistas, que resultaram em detalhamentos inadequados e erros de modelagem 3D, houve um aumento de produtividade entre 15% e 30% acima do padrão de projetos semelhantes. Apesar das dificuldades enfrentadas, a maioria dos participantes considerou que o esforço de colaboração e os processos de trabalho enxutos foram bem-sucedidos.

2.2.2 Emprego do BIM em projetos de infraestrutura

O BIM foi inicialmente projetado para aplicação no setor de edificações da construção civil, mas expandiu-se para outras áreas da engenharia, incluindo a infraestrutura civil. Considerando que o setor público detém a maior parte do sistema de infraestrutura rodoviária e de transporte, o BIM pode ser particularmente benéfico para governos em projetos relacionados a esse tipo de construção. Portanto, o governo e as empreiteiras são as entidades que mais podem se beneficiar da implementação dessa metodologia em projetos de infraestrutura. Entretanto, devido ao atraso na adoção do BIM em projetos de infraestrutura, estima-se que sua aplicação nessa área esteja cerca de três anos atrás em comparação com seu uso em edificações (Costin, *et al.*, 2018).

Bradley *et al.* (2016) explicam que a pesquisa limitada em projetos de infraestrutura provavelmente se deve ao fato de que a maior parte da pesquisa transferível do BIM (aplicável tanto no setor de edificações quanto no de infraestrutura) já foi concluída no campo das edificações. Outro fator é que os principais *stakeholders*, ou atores participantes, do setor de infraestrutura (geralmente órgãos da administração pública), por possuírem recursos avançados de gerenciamento de ativos em comparação com stakeholders privados do setor de edificações, criam obstáculos para a adoção do BIM na fase operacional. No setor de edificações, por outro lado, onde a força motriz das inovações são empresas e projetistas, o uso do BIM foi rapidamente estendido a todas as fases da construção.

Estudos recentes indicam que o BIM é usado predominantemente em projetos de edificações, com menos desenvolvimento no setor de infraestrutura. Essa discrepância leva a uma lacuna de maturidade no avanço da metodologia para infraestruturas, conforme apresentado por Nielsen *et al.* (2024) na Tabela 1.

Tabela 1 - Diferenças entre o BIM para infraestrutura e edificações.

Principais diferenças	Infraestrutura	Edificações
Complexidade do modelo	Baixa	Alta
Opções de softwares para modelagem	Poucas	Muitas
Interoperabilidade	Baixa	Razoável
Nível de maturidade	Baixo	Alto
Pesquisa acadêmica	Limitada	Razoável
Principal cliente	Governo	Iniciativa privada
Cultura	Execução de obras	Projetos
Integração GIS	Dependente	Independente
Principal tipo de informação	Não gráfica	Gráfica

Fonte: Nielsen *et al.* (2024).

O DNIT cita que a aplicação de normas e conceitos BIM a elementos de infraestrutura rodoviária, viária e Obras de Arte Especiais (OAEs), se comparadas à classe de projetos de edificações, ainda se encontra em estágio preliminar. O órgão ressalta ainda que o mercado ainda não está preparado para suportar a nova demanda de projetos em BIM, visto que, são poucos os profissionais habilitados em BIM na área de infraestrutura (Brasil, 2022).

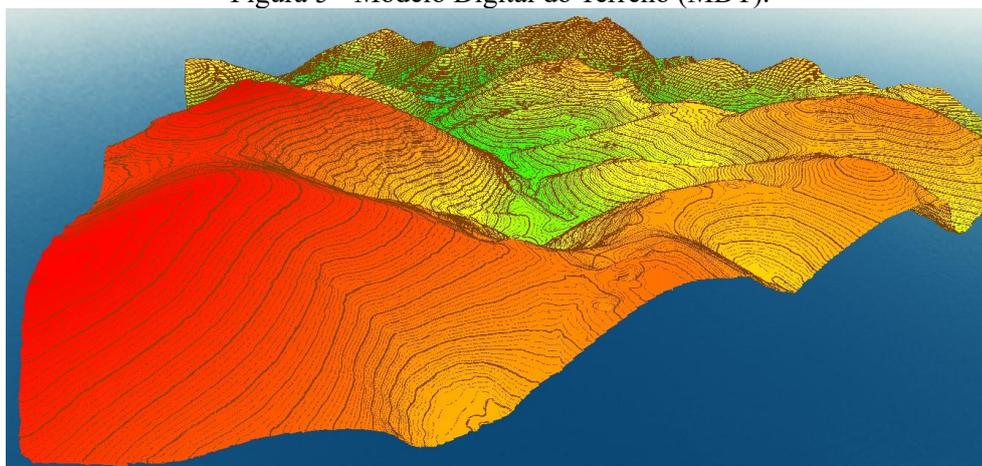
De acordo com Costin *et al.* (2018), a aplicação do BIM desde a fase de planejamento pode simplificar a comunicação e aprimorar a colaboração entre as diferentes partes interessadas. A utilização do planejamento 4D e do planejamento de custo, ou 5D, em grandes projetos de infraestrutura pode agregar mais valor à medida que os projetos se tornam maiores e mais complexos. Quanto mais complexo e extenso o projeto, maiores serão os benefícios obtidos.

2.2.3 Modelo Digital da Superfície (MDS) e do Terreno (MDT)

Na maioria dos projetos de engenharia, a localização é baseada em sistemas de coordenadas cartesianas. No entanto, como infraestruturas são colocadas em um contexto geoespacial, discrepâncias surgem a partir da transformação da superfície curva da Terra para um sistema de coordenadas ortogonais e as dimensões de um modelo podem não corresponder às do mundo real, mas estarem distorcidas de acordo com o sistema de referência de coordenadas utilizado. Dessa forma, um elemento essencial para qualquer modelo BIM de infraestrutura é sua geolocalização, ou colocação de um objeto na superfície da Terra. Assim, georreferenciar refere-se ao processo de definição de uma geolocalização, além de incluir a definição dos parâmetros do sistema de coordenadas e, portanto, as consequentes transformações geodésicas.

Para que o georreferenciamento faça sentido num processo de modelagem BIM de um projeto de infraestrutura, é necessário criar classes de objetos que serão a base para a modelagem de todos os demais objetos: o Modelo Digital da Superfície (MDS) e o Modelo Digital do Terreno (MDT).

Figura 5 - Modelo Digital do Terreno (MDT).



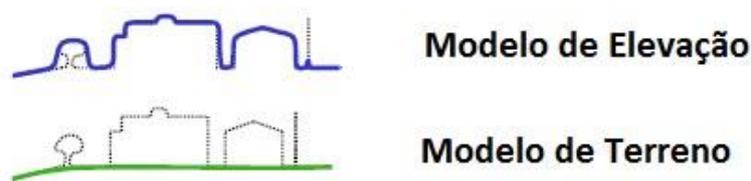
Fonte: Engesat (2024).

O MDS é a representação das altitudes de uma área específica da superfície, levando em consideração edificações, árvores, veículos e outros objetos naturais ou artificiais. Este modelo é amplamente utilizado em ambientes urbanos, pois as construções, vegetações e

pavimentações são essenciais para o estudo da volumetria e do comportamento do escoamento da água, por exemplo.

Em contraste, o MDT (Figura 5) representa as altitudes de uma área da superfície sem considerar edificações ou outros objetos presentes acima dela. Os MDTs são empregados em cálculos hidrográficos, estudos de relevo ou no planejamento de traçados de rodovias, entre outras aplicações. O MDT pode ser gerado por meio de um processo de filtragem do MDS, destacando o perfil do terreno e excluindo os objetos sobrepostos ao nível do solo (Figura 6).

Figura 6 - Esquema da diferença entre MDS e MDT.



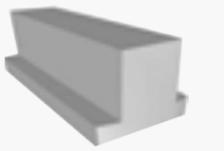
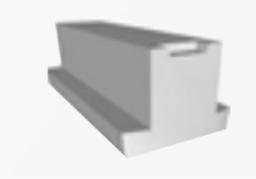
Fonte: Engesat (2024).

2.3 NÍVEL DE INFORMAÇÃO NECESSÁRIA (LOIN)

A definição de Nível de Desenvolvimento, do inglês *Level of Development* (LOD), é um conceito adotado pela indústria arquitetônica, originalmente proposto em 2008 pelo *American Institute of Architects* (AIA). O conceito de LOD abrange diversos aspectos e determinava, entre outras coisas: o nível de detalhe, do inglês *Level of Detail* (LOD), da modelagem; a quantidade, qualidade e relevância dos dados não gráficos e informações, em inglês *Level of Information* (LOI); e o tipo de dados não gráficos.

O nível de detalhe (LOD) refere-se à precisão geométrica do modelo em relação à aparência real do elemento e, muitas vezes, é confundido com o nível de desenvolvimento (LOD). O LOD representa o grau em que as informações sobre um elemento são consideradas confiáveis para fins de tomada de decisões em um determinado momento, como exemplificado na Figura 7. Em outras palavras, o nível de desenvolvimento significa a modelagem no nível de detalhe adequado para o uso eficiente do BIM.

Figura 7 - Exemplos de LOD conforme classificação da AIA.

LOD	Nível de detalhe	Nível de informação
LOD 100	sem modelagem 3D	Descrição básica para compreensão da proposta, podendo não ter informações.
LOD 200	 representação básica aproximada	Informações básicas com definição de material nível genérico. Ex: Parede.
LOD 300	 representação geométrica precisa	Informações suficientes que permitam a realização do 3D, planejamentos 4D e 5D, e compatibilização dos projetos. Ex: Identificação do elemento com o código do elemento, definição do material e seus insumos com códigos SINAPI ¹ , dentre outros.
LOD 400	 representação geométrica detalhada	Informações suficientes que permitam a realização do 3D, planejamentos 4D e 5D, e execução das obras. Ex: Além das informações já existentes, inserção da fase em que elemento será executado dentro do cronograma; manual explicativo de instalação do elemento modelado, dentre outros.
LOD 500	 representação geométrica detalhada	Informações suficientes que permitam o entendimento do empreendimento no estado atual e o planejamento e execução de sua manutenção durante seu ciclo de vida. Ex: Além das informações já inseridas, manual explicativo de instalação e de manutenção do elemento modelado, previsão para manutenção, dentre outros.

Fonte: Adaptado de Brasil (2022).

Para a *United Nations Economic Commission for Europe (UNECE)*, embora os critérios mais comuns de classificação de LOD, como a classificação da AIA, possam ser utilizados para diferenciar regras e métodos de processamento no contexto da modelagem de dados de construção, as características únicas de projetos de infraestrutura, como no caso de rodovias,

¹ Sistema Nacional de Pesquisa de Custos e Índices da Construção Civil: principal fonte de referência brasileira de custos para obras e serviços de engenharia

exigem a formulação de diretrizes mais adequadas às suas características específicas (UNECE, 2021).

A Secretaria de Estado de Infraestrutura e Logística do Paraná (Paraná, 2022) afirma ser possível que um elemento em um projeto de infraestrutura apresente pouco detalhamento geométrico, como em LOD 200, e uma grande quantidade de informações, como em LOI 500, contendo marca, modelo, fabricante e tipo de revestimento asfáltico, por exemplo, sem necessariamente progredir nesses níveis simultaneamente ou seguir a mesma lógica de desenvolvimento.

Miceli Junior (2019) observou que a adoção de um LOD único em todos os elementos de um projeto, apesar de possível, poderia torná-lo inviável, uma vez que é improvável que existam bibliotecas de todos os componentes para o LOD definido. Assim, os níveis a serem adotados em cada fase do empreendimento deveriam ser definição mútua entre a organização responsável pelo projeto e o contratante, e motivados pela necessidade de maior exatidão de orçamentos e pela melhora na gestão de informações do modelo, não devendo haver relação direta entre o estágio em que um projeto se encontra, e os níveis LOD e LOI adotados.

De modo geral, as informações necessárias para criar LODs mais elevados tornam-se gradualmente disponíveis à medida que o projeto avança nas etapas subsequentes. Simultaneamente, o desenvolvimento dos elementos individuais no processo de modelagem pode progredir em diferentes velocidades. Por exemplo, os elementos de uma estrada no projeto final podem estar em LOD 300, enquanto os serviços individuais podem ser modelados em LOD 200. Portanto, o LOD é utilizado apenas para descrever um elemento específico do modelo, não o modelo completo (UNECE, 2021).

Como forma de preencher essas lacunas entre os conceitos de LOD, LOd e LOI, em 2018, a ISO 19650-1, adaptada ao Brasil por meio da norma ABNT NBR ISO 19650-1:2022 (ABNT, 2022) introduziu o conceito, de maneira curta e genérica, de Nível de Informação Necessária, em inglês *Level of Information Need* (LOIN), com o objetivo de definir as necessidades de informação dos autores envolvidos na elaboração e gerenciamento ao longo de um projeto.

Para a nova norma da ISO e da Associação Brasileira de Normas Técnicas (ABNT), e na contramão das definições de LOD da AIA, qualquer coisa além do mínimo necessário em cada etapa do projeto é considerada desperdício. O LOIN solicita apenas o que é necessário em um determinado momento para um propósito específico, além de definir bem as entregas contratuais, incluindo requisitos geométricos e documentação.

2.4 ALINHAMENTOS

O termo "alinhamento" descreve três conceitos distintos, mas intimamente interligados: a definição de um sistema de referência para posicionamento linear; a salvaguarda e a otimização do movimento de veículos sob uma perspectiva cinemática; e a construção geométrica de estradas, ferrovias ou outras infraestruturas lineares.

Um alinhamento é utilizado para definir um sistema de referência com o propósito de posicionar elementos principalmente em obras de construção linear, como estradas, trilhos, pontes e outras. O posicionamento relativo ao longo do alinhamento é definido pela metodologia de referenciação linear (Buildingsmart International, 2024a).

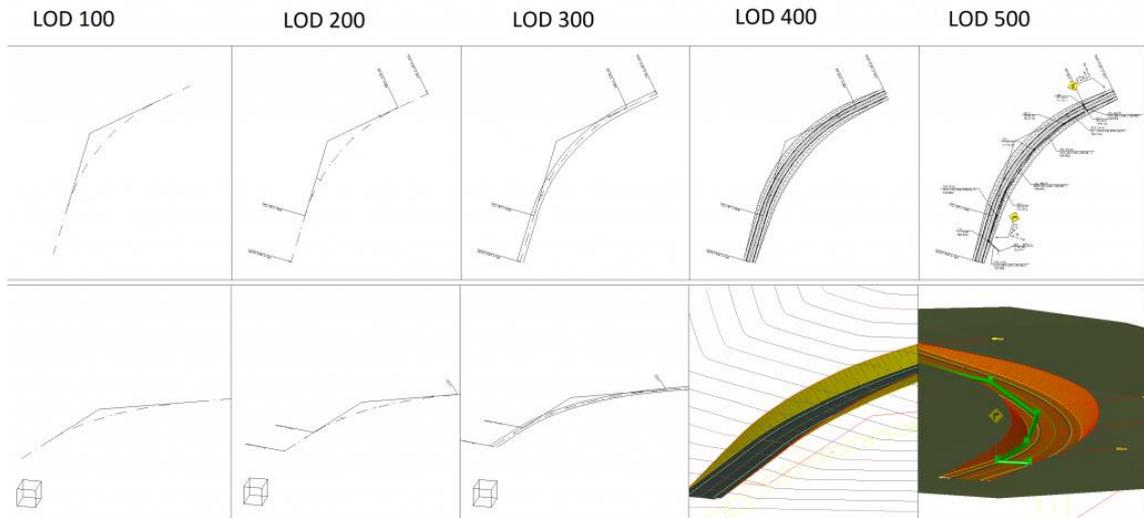
Embora não necessitem de um nível de detalhamento tão alto quanto um projeto de estação de tratamento de efluentes, por exemplo, projetos de infraestrutura, como pavimentação, sistemas tubulares de drenagem e esgoto, demandam altos níveis de informações sobre volumes de escavação e aterro, ou seções transversais específicas em diversos locais ao longo de trechos quilométricos. Para o caso da modelagem dessas obras lineares no Civil 3D, essa riqueza de informações fica armazenada no elemento alinhamento.

A definição de alinhamentos é feita seguindo as seguintes etapas, que podem mudar de sequência de caso para caso e pode ser repetida uma ou mais vezes para atingir os objetivos econômicos e cumprir os requisitos das normas (Buildingsmart International, 2024a):

- Primeiro um traçado horizontal é definido em um plano;
- Em uma segunda etapa, o traçado vertical é adicionado;
- Na maioria dos casos, superelevações são adicionadas às curvas horizontais para compensar uma parte da aceleração lateral indesejada;
- Na etapa final, o traçado proposto é verificado em relação a um conjunto definido de regras, fórmulas e limites para garantir a conformidade com a regulamentação.

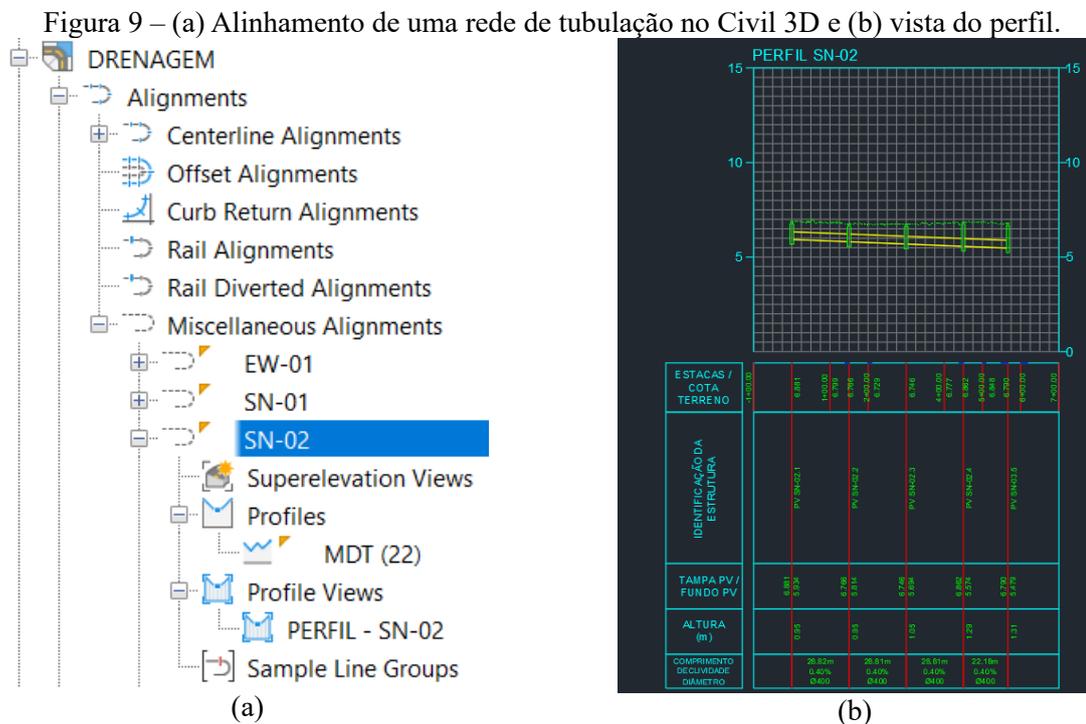
Soethe (2017) explica que com o Civil 3D é possível modelar objetos com diferentes LODs simultaneamente, desde que o *template* utilizado esteja previamente configurado com as normas do Departamento Nacional de Infraestrutura (DNIT), por exemplo. Isso quer dizer que para reduzir a quantidade de dados alfanuméricos de um alinhamento, seria necessário desligar a opção de estaqueamento, por exemplo, pois o padrão *default* do software é apresentar essas informações em planta (Figura 8).

Figura 8 - Diferentes LODs para um projeto de rodovia no Civil 3D.



Fonte: Soethe (2017).

Fazendo uma analogia com o entendimento de Soethe (2017), o LOD de um projeto de rede de tubulação no mesmo software aumenta à medida que mais informações são armazenadas na entidade, como os perfis, ou *Profiles* em inglês, e as vistas de cada perfil, ou *Profile Views* em inglês (Figura 9 (a)); e mais detalhes são exibidos nas vistas dos perfis (Figura 9 (b)).



Fonte: Autor (2025).

Alinhada com o que a NBR ISO 19650-1:2022 coloca como premissa, a UNECE (2021) alerta que se deve também estar atento à prática conhecida como "modelagem excessiva", que pode resultar em desperdício de recursos do projeto. Dessa forma, é crucial alinhar os níveis

necessários de desenvolvimento em uma fase inicial com todas as disciplinas que fornecem componentes 3D para o modelo BIM geral.

2.5 FORMATO IFC 4.3

A BSI (2024c) define o IFC como uma descrição digital padronizada do ambiente construído, abrangendo edificações e infraestrutura civil. Trata-se de um padrão internacional aberto, normatizado pela ISO 16739-1:2018 (ISO, 2018), cujos direitos não podem ser detidos por nenhum desenvolvedor de software, sendo utilizável em uma ampla gama de dispositivos de hardware, softwares e interfaces para diversos fins.

O IFC não é apenas um formato, mas um esquema padronizado que codifica, de maneira lógica, a identidade e semântica, características ou atributos e relacionamentos de objetos, conceitos abstratos, processos e pessoas. Além de responder ao que está sendo construído, o IFC também aborda como será feito e como a construção será operada e utilizada.

Ainda de acordo com a BSI (2024c), o formato é frequentemente empregado na troca de informações entre os diversos atores envolvidos em um projeto de edificação, podendo também ser utilizado para arquivar informações do projeto de forma incremental durante as fases de projeto, aquisição e construção, ou como uma coleção de informações *as-built* (“como construídas”, em português) para fins de preservação e operações de longo prazo.

O IFC 4.1 introduziu o alinhamento como um sistema de referência linear que facilita a inserção de elementos e a interligação de diferentes infraestruturas em uma rede de transporte. A versão seguinte, o IFC 4.2, trouxe pontes, mas essa entidade foi retirada após a BSI decidir compilar todas as entidades de infraestrutura em um único lançamento, em vez de publicar versões individuais para cada uma (Justo, *et al.*, 2021).

Já o formato IFC 4.3 foi desenvolvido para estender os benefícios do IFC às construções horizontais, como rodovias, ferrovias, portos, hidrovias, pontes e elementos comuns entre eles. Em comparação com as versões anteriores, incluiu mudanças de hierarquia e nomenclatura no esquema, juntamente com outros recursos, como a inclinação do perfil horizontal para alinhamento (Kelly, 2022).

A primeira fase do desenvolvimento do IFC 4.3 definiu as entidades contempladas, incluindo referência de alinhamento, terraplenagem, estrutura de pontes, rede elétrica, sinalização, telecomunicações e trilhos ferroviários, drenagem, hidrovias, e geotecnia.

A segunda fase produziu instruções de teste e validou o software do formato IFC 4.3. Além disso, as Definições do Conjunto de Propriedades (*Property Set Definitions*) foram feitas para

entrada no dicionário de dados BSI, parte do padrão. Essa fase também incluiu a definição de três Definições de Vista do Modelo, do inglês *Model View Definitions* (MVDs): a *Reference View*, a *Alignment Based Reference View* e a *Design Transfer View*.

Justo *et al.* (2021) explicam que, com as extensões do esquema IFC 4.3, as informações geométricas e semânticas dos ativos de infraestrutura podem ser combinadas em um modelo BIM que serve de base para a tomada de decisões e monitoramento.

Neste trabalho, os arquivos trabalhados no formato IFC, mais especificamente o IFC 4.3, utilizaram como extensão o *.ifc*.

2.5.1 A entidade *IfcAlignment*

A entidade alinhamento na abordagem IFC, ou apenas *IfcAlignment*, define três conceitos separados, mas intimamente interconectados (BSI, 2024a):

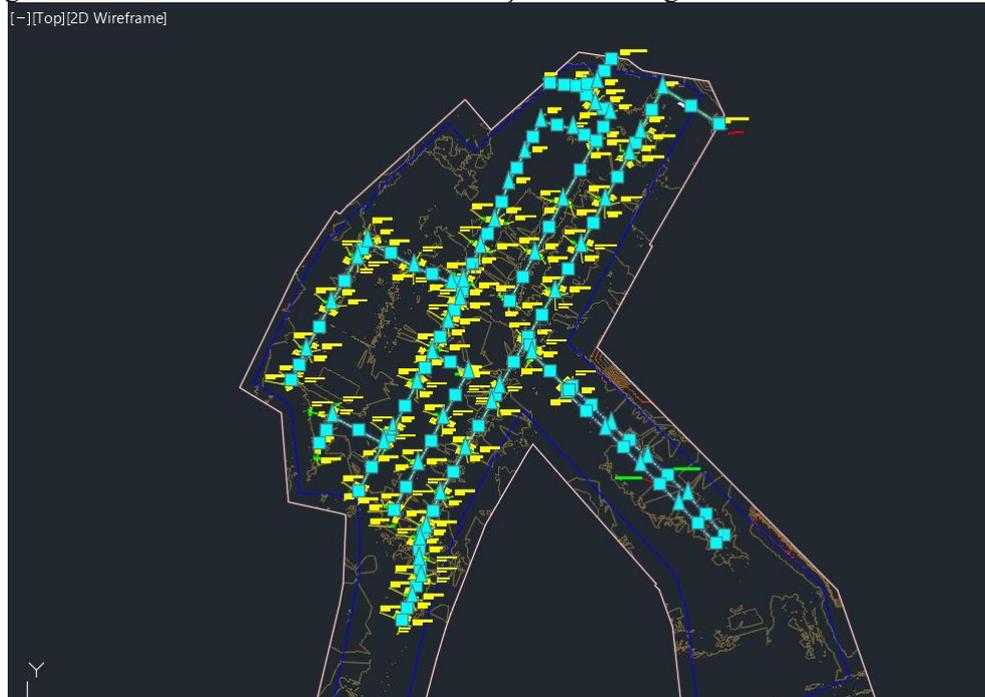
- um sistema de referência para posicionamento linear;
- requisitos de segurança e otimização do tráfego de veículos (perspectiva cinemática); e
- traçado de rodovias, ferrovias ou outras infraestruturas lineares.

Um único alinhamento pode ser descrito por um ou mais dos seguintes traçados:

- um traçado horizontal (*IfcAlignmentHorizontal*), definido no plano XY;
- um traçado vertical acompanhante (*IfcAlignmentVertical*), definido ao longo do traçado horizontal na distância ao longo do espaço de coordenadas Z; e
- um traçado de superelevação (*IfcAlignmentCant*), definido como inclinação lateral ao longo do traçado horizontal.

No Civil 3D, os *Alignments* podem representar eixos de rodovias, redes de tubulação e outras infraestruturas lineares (Figura 10). A geometria de um alinhamento pode ser criada a partir de uma simples polilinha e, em seguida, transformada nessa entidade. Se um alinhamento existir em um *Site*, sua geometria interage com a de outros objetos do mesmo *Site*, como *Parcels* e linhas de feições, ou *Feature lines*. Essas entidades serão mais bem detalhadas no tópico 2.5.2.

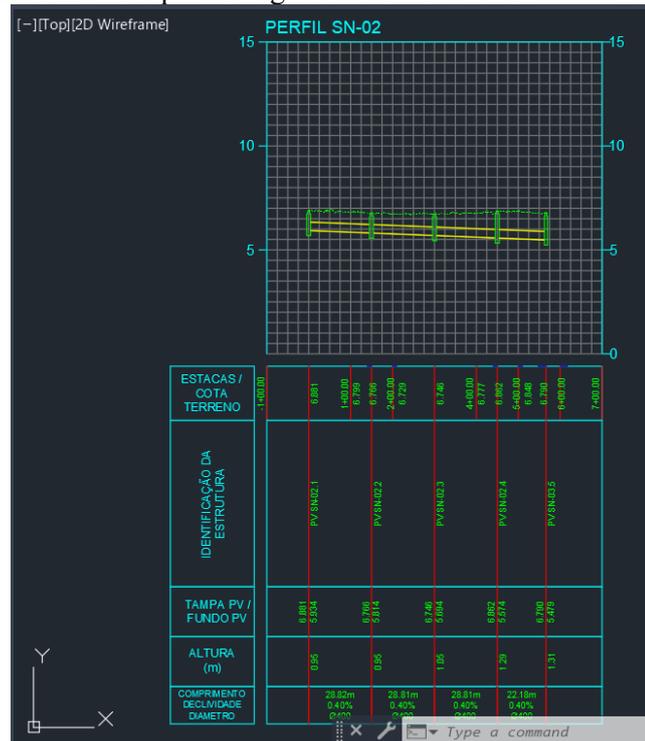
Figura 10 - Alinhamentos de redes de tubulações de drenagem selecionados no Civil 3D



Fonte: Autor (2025).

Os alinhamentos no Civil 3D podem ser objetos autônomos ou objetos superiores (em termos de herança) para perfis, vistas de perfis, corredores e seções longitudinais (Figura 11). Se uma dessas entidades for editada, as alterações serão refletidas automaticamente em quaisquer objetos herdados.

Figura 11 - Perfil e vista de perfil longitudinais de um dos alinhamentos da Figura 10



Fonte: Autor (2025).

2.5.2 A entidade *IfcSite*

A BSI (2024b) define *Site* como uma área definida de terra, que também pode ser coberta de água, na qual a construção do projeto deve ser concluída. Um local pode ser usado para erguer, reformar ou requalificar edifícios, ou para outros desenvolvimentos relacionados à construção.

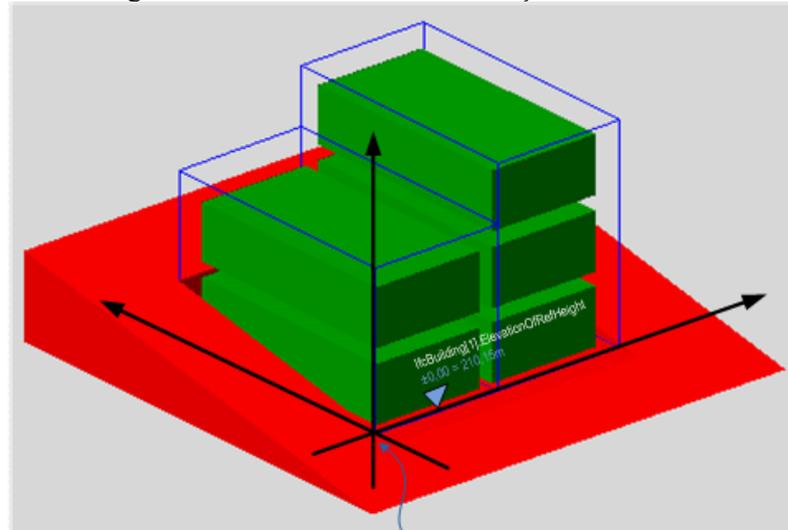
Informações geoespaciais precisas de um *Site* devem ser herdadas das entidades de georreferenciamento que relacionam o *IfcProject* ao mundo real, usando um sistema de coordenadas georreferenciado. Em projetos de pequena escala, quando o georreferenciamento não é definido, os atributos Longitude, Latitude e Elevação do *IfcSite* podem ser usados para indicação aproximada da localização do *Site*. Se esses atributos forem definidos, estabelecem o ponto no sistema de coordenadas onde o ponto 0.,0.,0. do *LocalPlacement* do *IfcSite* está situado.

O posicionamento geométrico do *Site*, definido pelo *IfcLocalPlacement* (Figura 12), deve ser sempre relativo à origem do projeto, no qual este *Site* está incluído; ou absoluto, ou seja, posicionado no sistema de coordenadas geográfico, conforme estabelecido pelo contexto de representação geométrica do projeto. O sistema de coordenadas global, estabelecido no *IfcProject.RepresentationContexts*, pode incluir uma definição do norte verdadeiro dentro do

plano XY desse sistema, e que, se fornecido, pode ser obtido em *IfcGeometricRepresentationContext.TrueNorth*.

Para o georreferenciamento preciso do projeto, as entidades *IfcCoordinateReferenceSystem* e *IfcMapConversion* devem ser usadas para definir um posicionamento exato do sistema de coordenadas local do projeto no sistema de coordenadas geográficas.

Figura 12 - Posicionamento e elevações de um Site

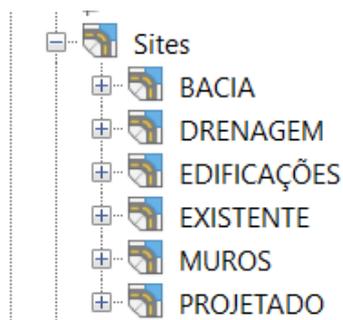


IfcSite.ObjectPlacement = IfcLocalPlacement
for information purpose equal to: RefLongitude, RefLatitude, RefHeight
Referring to degree, minute, seconds (with fractions) given in WGS84: 15°,52',23.34"; 53°,21',12.34",210.15m

Fonte: BSI (2024b).

O Autodesk Civil 3D usa a entidade *Site* como uma espécie de pasta para armazenar entidades do tipo “segmentos de localização”, como *Alignments*; linhas de recurso, ou *Feature lines*, que são espécies de “linhas interativas e de referência” com/para outros objetos do software; grupos de nivelamento, ou *Grading groups*, que armazenam linhas de construção de banquetas e taludes; e lotes, ou *Parcels* (Figura 13).

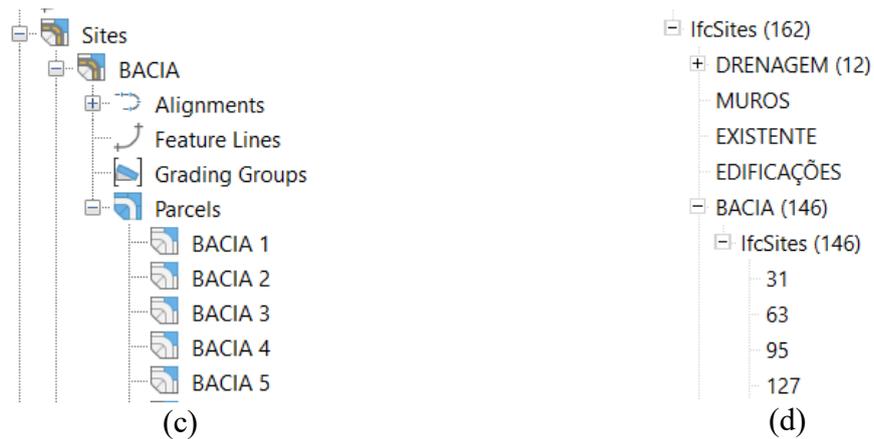
Figura 13 – *Sites* (a) e *Parcels* (c) na extensão .dwg do Civil 3D; e dos *sites* (b) e *Parcels* (d) na extensão .ifc na versão IFC 4.3.



(a)



(b)



Fonte: Autor (2025).

Os objetos *Parcels* são normalmente usados para representar parcelas de imóveis, como lotes em uma subdivisão. Esses objetos também podem representar outras entidades de perímetro fechado, como bacias hidrográficas e divisões do solo (Figura 14).

Figura 14 - Bacias como *Parcels*.



Fonte: Autor (2025).

2.6 FORMATO CITYGML

City Geography Markup Language (CityGML) é o padrão internacional da *Open Geospatial Consortium (OGC)* para a representação e troca de modelos de cidades tridimensionais. Ele define a geometria tridimensional, a topologia, a semântica e a aparência dos objetos topográficos mais relevantes em contextos urbanos ou regionais, fornecendo diferentes níveis de detalhe bem definidos (Gröger; Plümer, 2012).

Deng, Cheng e Anumba (2016) explicam que o formato foi desenvolvido na Alemanha em 2002 pelo Special Interest Group 3D (SIG 3D), da iniciativa *Geodata Infrastructure North*

Rhine-Westphalia, e aceito como padrão oficial do OGC em 2008. O *CityGML* é o primeiro esquema GIS a oferecer suporte avançado à informação semântica, permitindo que diferentes componentes de construção GIS recebam IDs, nomes e descrições exclusivos. Ele também suporta cinco níveis de desenvolvimento (LOD) e níveis de detalhe (LOd) variando de LOD0 até LOD4, com resoluções diferentes para aplicações distintas. Além disso, o *CityGML* oferece suporte a extensões de domínio de aplicativo, ou *Application Domain Extensions* (ADE), permitindo que usuários criem suas próprias extensões para aplicativos específicos.

Neste trabalho, é importante realçar que há de se fazer distinção entre as siglas LOD, que se refere ao Nível de Desenvolvimento, e LOd, com d minúsculo, que se refere ao Nível de Detalhe.

Os formatos IFC e *CityGML* são os dois principais modelos semânticos não proprietários atualmente usados para representar modelos BIM e GIS, respectivamente. O *CityGML* tornou-se o formato padrão para modelar áreas urbanas, enquanto o IFC evoluiu para um modelo de referência para modelar edifícios e espaços (Wang; Pan; Luo, 2019).

Biljecki, Ledoux e Stoter (2016) explicam que o padrão do formato *CityGML* categoriza o conceito de LOD para GIS em cinco níveis. Deng, Cheng e Anumba (2016) também acrescentam que esses diferentes LOds também são uma forma eficiente de reduzir a demanda por capacidade de renderização e por espaço de armazenamento. A classificação destina-se principalmente a diferenciar o grau de dados resultante de diferentes fluxos de trabalho de modelagem, e foi motivada tanto pelas diversas formas de apresentação das informações do modelo, quanto pela geometria.

Na indústria e na comunidade de pesquisa, a classificação foi aceita sob a óptica da classificação geométrica, em parte devido à falta de aplicativos que exijam a leitura de informações. A classificação aplica-se a várias classes temáticas de objetos, mas é principalmente focada em edifícios (Figura 15):

- **LOD0:** Representação de manchas e, opcionalmente, polígonos da projeção do telhado marcando a transição de 2D para 3D;
- **LOD1:** Modelo prismático grosseiro geralmente obtido pela extrusão de um modelo LOD0;
- **LOD2:** Modelo com uma forma de telhado simplificada e onde as partes do objeto podem ser modeladas em várias classes semânticas (por exemplo, telhado, parede);

- **LOD3:** Modelo arquitetonicamente detalhado com janelas e portas, sendo consideravelmente mais complexo do que o nível anterior;
- **LOD4:** Completa o LOD3 incluindo elementos internos.

Figura 15 - Gráfico dos diferentes níveis LOD para o GIS conforme o padrão *CityGML*.



2.7 FORMATOS SHAPEFILE E SDF

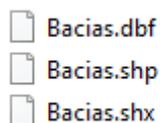
A ESRI (1998) define *shapefiles* como arquivos vetoriais que armazenam geometria não topológica e informações de atributos de feições geoespaciais em um conjunto de dados. A geometria de uma feição é armazenada como uma forma composta por um conjunto de coordenadas vetoriais.

Como os *shapefiles* não possuem a sobrecarga de processamento de uma estrutura de dados topológica, eles apresentam vantagens sobre outras fontes de dados, como maior velocidade de modelagem, melhor capacidade de edição e menor demanda por espaço de armazenamento. *Shapefiles* podem ainda comportar feições únicas que se sobrepõem ou que não são adjacentes, suportando recursos de ponto, linha e área.

As feições de área são representadas como polígonos de dupla digitação em circuito fechado, onde a primeira e a última coordenadas são iguais (ESRI, 1998).

Um *shapefile* é composto por um arquivo principal (sufixo SHP), um arquivo de índice (sufixo .shx) e uma tabela de atributos no formato dBASE (sufixo .dbf), conforme indicado na Figura 16. O arquivo principal é um arquivo de acesso direto, com comprimento de registro variável, no qual cada registro descreve uma forma com uma lista de seus vértices. No arquivo de índice, cada registro contém o deslocamento do registro correspondente do arquivo principal desde o início dele. A tabela dBASE contém informações das feições com um registro por feição.

Figura 16 - Representação de um *shapefile* com os 3 formatos de arquivo (SHP, .shx e .dbf).

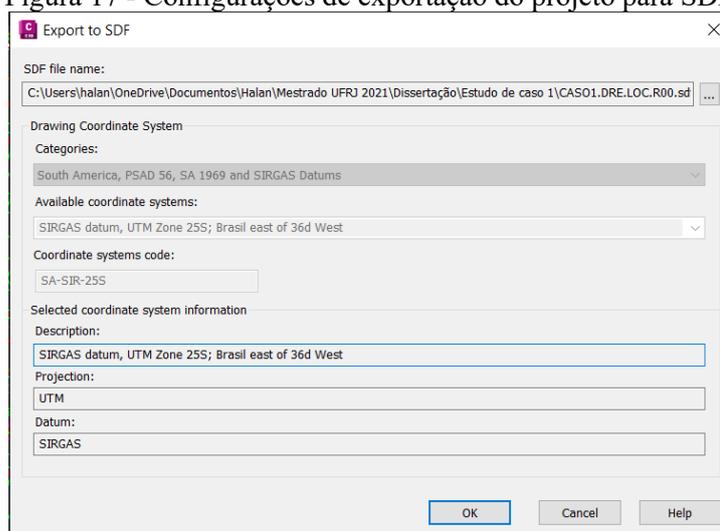


Fonte: Autor (2025).

Já o SDF (*Spatial Data File*) é um formato geoespacial nativo da Autodesk otimizado para armazenar grandes conjuntos de dados. O SDF é semelhante ao formato *shapefile*, pois contém dados espaciais e dados de atributos, no entanto, diferentemente do SHP, ele armazena todos os tipos de dados em um único arquivo em vez de um conjunto de arquivos (Autodesk, 2024a).

Quando as feições do projeto do Civil 3D são exportados para SDF, o sistema de coordenadas geográficas predefinido é mantido, uma vez que se trata de um formato geoespacial (Figura 17).

Figura 17 - Configurações de exportação do projeto para SDF.



Fonte: Autor (2025).

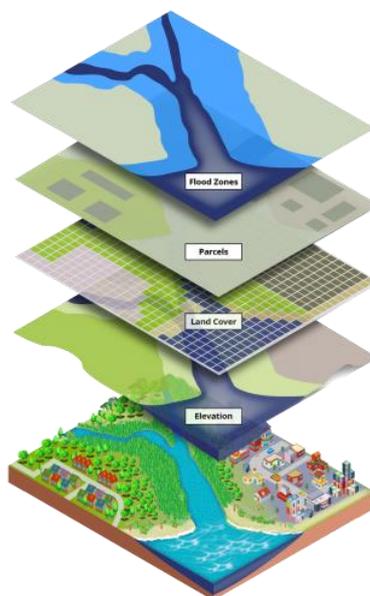
2.8 SISTEMAS DE INFORMAÇÃO GEOGRÁFICA (GIS)

2.8.1 A utilização de ferramentas GIS em projetos de infraestrutura

De acordo com a definição de Wang, Pan e Luo (2019), Sistemas de Informação Geográfica (SIG), ou *Geographic Information Systems* (GIS) em inglês, são sistemas computadorizados baseados em geografia, cartografia e tecnologia de sensoriamento remoto. Esses sistemas são capazes de coletar, armazenar, gerenciar, calcular, analisar, exibir e descrever informações espaciais e dados sobre a superfície da Terra, além de digitalizar e visualizar informações abstratas. Um conjunto de ferramentas GIS inclui um sistema de banco de dados com funções

específicas para dados de referência espacial e um conjunto de operações para o processamento específico de dados (Figura 18).

Figura 18 - Sobreposição de camadas de informações num software GIS.



Fonte: ESRI (2019).

Enquanto o BIM se concentra nas próprias edificações, prestando mais atenção aos projetos de construção e seus detalhes internos que constituem dados em nível micro, o GIS é focado em informações geoespaciais externas às edificações e é utilizado para gerar informações de nível macro. Devido às suas capacidades de processamento de dados espaciais, o GIS pode ser aplicado em uma ampla gama de áreas, como monitoramento de desastres, agricultura, manutenção de infraestrutura, levantamentos topográficos, gerenciamento cadastral, gerenciamento ambiental e outros propósitos (Wang; Pan; Luo, 2019).

As informações armazenadas em bancos de dados GIS são normalmente apresentadas em formato bidimensional, geralmente na forma de mapas para cada camada de informação. Entretanto, devido à crescente necessidade de estudar os diversos tipos de impactos causados e sofridos por edificações em ambientes urbanos, as informações espaciais do GIS passaram a ser exibidas em formato tridimensional com o uso de recursos de outras tecnologias, como o BIM, como na Figura 19.

Figura 19 - Visualização de edificações no formato 3D em software GIS.



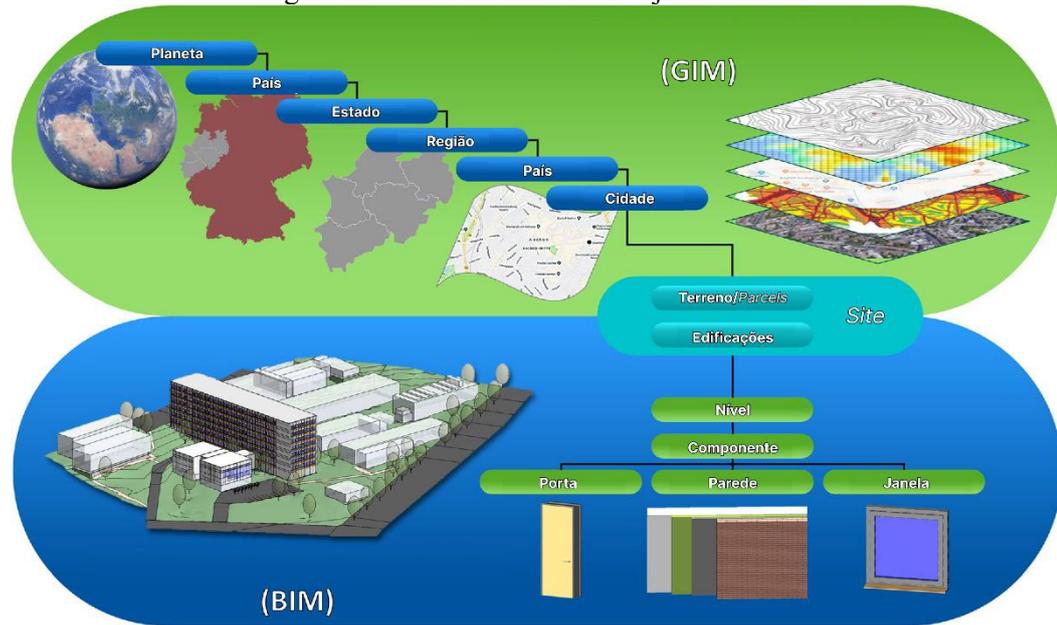
Fonte: ESRI (2019).

2.8.2 Modelagem de Informação Geoespacial (GIM)

Herle *et al.* (2020) definem Modelagem de Informação Geoespacial, ou *Geospatial Information Modeling* (GIM) em inglês, como o método de modelagem digital de fenômenos relacionados ao espaço do mundo real. É caracterizada por descrições multidimensionais de feições geoespaciais por localização e orientação no Sistema de Referência Espacial, geometria e topologia vetorial, dados de atributos e tempo. Assim, o GIM é utilizado como uma documentação digital dos estados do mundo real e pode ser aplicado a uma variedade de questões espacialmente relacionadas.

Em outras palavras, enquanto GIS refere-se a uma ferramenta, o termo GIM é utilizado como um conceito semelhante ao BIM, porém em referência à modelagem no domínio geoespacial.

Figura 20 - GIM e BIM e seus objetos comuns.



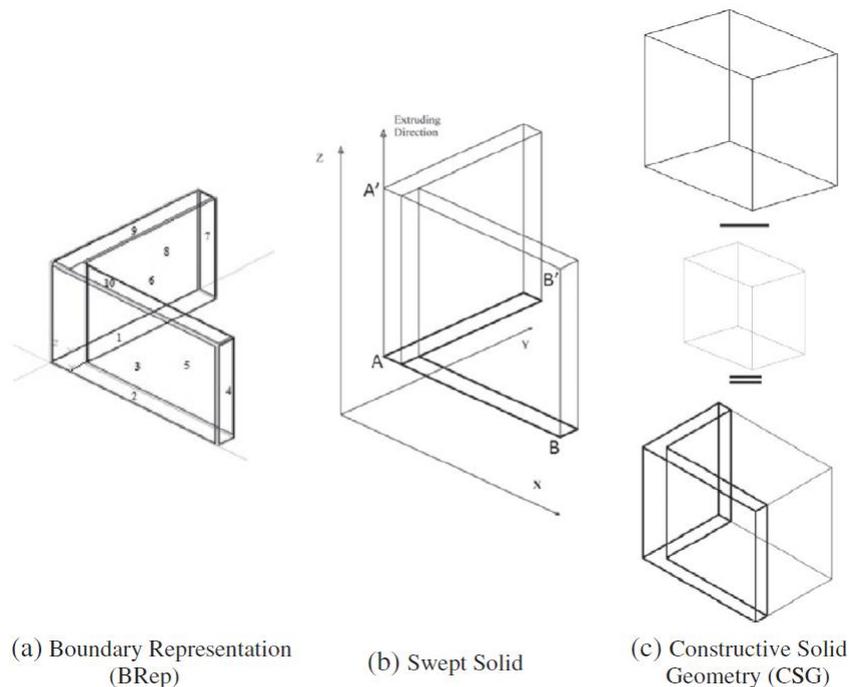
Fonte: Adaptado de Herle *et al.* (2020).

Conforme ilustrado na Figura 20, o BIM e o GIM se complementam com a presença de objetos comuns em ambos os tipos de modelagem, especialmente no Terreno/*Parcels* e no nível da edificação. Cada elemento geoespacial de maior extensão é considerado parte dos modelos GIM, enquanto componentes de construção altamente detalhados fazem parte do universo BIM (Herle, *et al.*, 2020).

Diferentemente do BIM, no GIM as formas das geometrias são descritas explicitamente pelos seus nós, arestas e superfícies caracterizadas por coordenadas. A chamada representação de limites, ou *Boundary Representation* (B-Rep) em inglês, é um dos modelos tridimensionais mais utilizados para definir geometrias por suas superfícies, sendo o *CityGML* um dos padrões usados para codificar modelos de cidades com geometrias B-Rep.

A representação B-Rep utiliza superfícies limite para definir a forma dos objetos. Na representação *Swept Solid*, a seção transversal dos objetos é primeiramente definida e posteriormente extrudada em determinadas direções até um comprimento específico para criar a forma sólida. Por outro lado, a representação CSG resulta de operações booleanas aplicadas a objetos (Figura 21).

Figura 21 - B-Rep (a), *Swept Solid* (b) e CSG (c).



Fonte: Deng, Cheng e Anumba (2016).

2.9 INTEGRAÇÃO BIM-GIM

BIM e GIM pertencem a domínios distintos: o primeiro foi desenvolvido para AEC e Gestão de Instalações, enquanto o segundo atende à indústria geoespacial. O BIM pode expandir o escopo da modelagem GIM, por exemplo, utilizando modelos BIM para criar redes internas de resposta a emergências, avaliar o impacto do ruído em um edifício e analisar a influência de inundações em uma sala.

Han *et al.* (2022) explicam que, apesar do grande potencial do BIM na modelagem de projetos e gestão de obras de infraestrutura, a metodologia ainda apresenta limitações, especialmente no gerenciamento de malhas rodoviárias, que geralmente possuem dimensões lineares de centenas de quilômetros. À medida que projetos se tornam mais extensos ou suas topografias mais complexas, o BIM necessita do apoio do GIM, pois este último, sendo uma ferramenta de gerenciamento de dados 2D, tem o potencial de quantificar diversas situações do mundo real. No entanto, o nível de detalhamento no GIM é insuficiente para visualizar ou simular o processo de construção de uma rodovia, algo que o BIM facilita.

Para Yang *et al.* (2021), enquanto o BIM envolve informações abundantes do ciclo de vida das construções, atendendo melhor a análises específicas, os sistemas GIS capturam uma gama de entidades georreferenciadas do sistema de infraestrutura e suas propriedades, fornecendo poderosas ferramentas de análise geoespacial. As concessionárias de rodovias, por exemplo,

não estão apenas interessadas nas informações de planejamento e gerenciamento de tráfego, mas também trabalham com informações rodoviárias de modelagem e dimensionamento, que não estão disponíveis em bancos de dados geoespaciais tradicionais, mas sim armazenadas em um sistema BIM.

No contexto de gerenciamento de informações viárias, o BIM pode fornecer projetos com alto nível de desenvolvimento e detalhamento de redes de tubulações em ambientes urbanos, com informações tão detalhadas quanto o fabricante de cada peça de uma rede de esgoto. Essas informações complementam o gerenciamento tradicional do GIM e a análise de funcionamento das tubulações.

Wang, Pan e Luo (2019) dividem a integração BIM-GIM em três modelos:

- **BIM coordena e GIM apoia:** O BIM é utilizado principalmente para lidar com edifícios, mas quando necessário considerar vários edifícios em uma cidade ou região, a quantidade de informações dos modelos detalhados é muito alta. Portanto, um aplicativo GIS é usado para reduzir o respectivo modelo de casca ou simplificar o modelo interno para diversas finalidades, como simulação de ruído ou gerenciamento de ações de emergência.
- **GIM coordena e BIM apoia:** Como o GIS pode fornecer dados espaciais sobre projetos de construção e devido à interoperabilidade de informações de outros softwares, inclusive de modelagem BIM, as informações de dados podem ser importadas diretamente para o software GIS. Um modelo BIM sempre representa um edifício e suas informações relacionadas, mas carece de uma relação geoespacial com outros objetos do terreno. Assim, é desejável converter um modelo BIM em um modelo superficial GIM para estudo em ambiente GIS.
- **BIM e GIM em níveis iguais de coordenação:** Para alcançar a interoperabilidade entre modelos BIM e GIM, são geradas regras para integração entre os formatos IFC e *CityGML*, que representam aplicações em dois ambientes diferentes, integrando dados das dimensões de cidades, derivados de modelos GIM, com edificações individuais derivadas de modelos BIM.

Embora na maior parte dos casos o GIM bidimensional seja suficiente, o requisito de lidar com edificações complexas impulsionou o desenvolvimento de um modelo 3D mais detalhado. O *CityGML*, entre todos os esforços, foi, e tem sido até então, o padrão de troca mais abrangente de informações urbanas no domínio geoespacial (Liu, *et al.*, 2017).

2.9.1 Os desafios da integração BIM-GIM

Embora existam objetos similares que interagem em suas modelagens, a troca de dados e modelos entre os domínios BIM e GIM não é trivial. Isso ocorre principalmente devido às diferenças nos objetivos e nas perspectivas das duas abordagens de modelagem (Herle, *et al.*, 2020).

As diferenças de conteúdo e estruturas de dados entre os esquemas, mais especificamente entre os formatos IFC e *CityGML*, refletem em quatro desafios para sua integração (Deng; Cheng; Anumba, 2016):

- **Transformação do sistema de referenciamento local em sistema de coordenadas geográficas:** O IFC utiliza um sistema de posicionamento local relativo a um objeto do projeto e que não se refere ao sistema de coordenadas geográficas. No *CityGML*, entretanto, todos os objetos são definidos em um sistema de coordenadas geográficas, cuja transformação é necessária durante o processo de integração;
- **Compatibilização de detalhes durante o processo de integração:** Os LODs em BIM e GIM, apesar de fortemente correlacionados e de compartilharem diversas semelhanças em termos de informações arquitetônicas e atributos não geométricos (Wang; Pan; Luo, 2019), não atingem uma integração completa, pois não existem estruturas de transformação completas entre os cinco LODs do *CityGML* com os LODs do IFC;
- **Perda de dados durante o processo de integração:** Como padrão de dados BIM, o IFC foi projetado para capturar as informações da edificação, incluindo fabricantes, materiais e custos. No entanto, o foco do *CityGML* é definir as propriedades comuns do GIM, como localização, coordenadas e altura. Em geral, as informações contidas no BIM são muito mais ricas que as do GIM, e muitas entidades no formato IFC não conseguem encontrar as entidades correspondentes no *CityGML*; e
- **Transformação do *Swept Solid*/CSG em B-Rep:** O IFC suporta os três tipos de representações geométricas comuns (Figura 21), todavia, particularmente para formas cuboides, como paredes e lajes, o *Swept Solid* é comumente utilizado, pois pode requerer menor armazenamento de dados. Para formas complexas em arquivos IFC, o *Constructive Solid Geometry*

(CSG) é uma forma eficiente de representação. No *CityGML*, todas as formas são representadas usando BRep; portanto, a transformação de *Swept Solid/CSG* para BRep deve ser desenvolvida para a integração entre IFC e *CityGML*.

Para o caso específico deste trabalho, como as informações a serem compatibilizadas entre os aplicativos BIM e GIS são bidimensionais, o último desafio não se aplica, pois este aborda apenas casos em que são trabalhadas informações tridimensionais.

Apesar da difícil integração entre as siglas, Wang, Pan e Luo (2019) acreditam que a relação BIM-GIM destaca a importância da visualização de modelos para resolver problemas práticos. Exemplos incluem a avaliação de áreas sujeitas a inundações por meio da visualização tridimensional para gerenciar riscos de danos a edificações, bem como a análise da relação entre riscos de inundação e urbanização, resultando na otimização da gestão urbana.

As normas ISO, pelo fato de a integração dos dois modelos ser muito complexa e poder causar uma variedade de problemas de ambiguidade, não pretendem desenvolver um modelo de informação unificado entre BIM e GIM, apesar de afirmar que a unificação é um objetivo desejável (ISO, 2021).

2.10 PROGRAMAÇÃO VISUAL

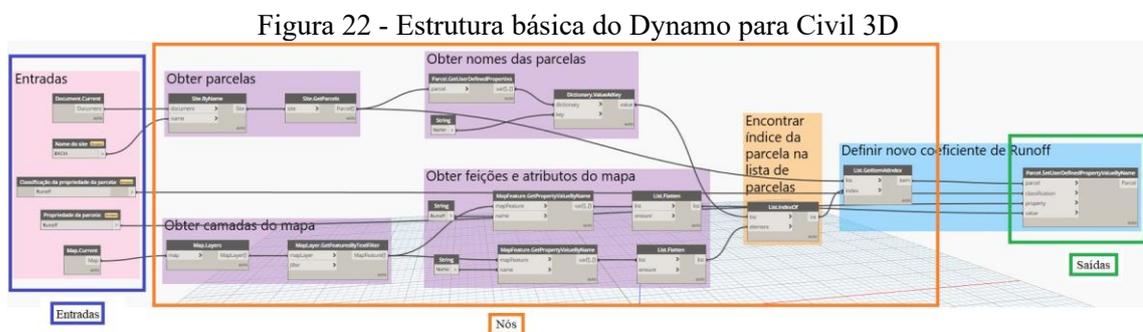
Segundo Collao *et al.* (2021), no início do desenvolvimento da metodologia BIM não foram contempladas ferramentas como as de modelagem parametrizada e de geração de gráficos de análise. Como forma de possibilitar a criação de comandos e ferramentas personalizados e específicos, a indústria AEC desenvolveu aplicativos de programação em linhas de código, incorporados aos softwares BIM, que geram algoritmos definindo diferentes sequências de tarefas por meio de linhas de comando de instruções. No entanto, a interface de trabalho complexa e as habilidades avançadas de programação exigidas por esses aplicativos representam um desafio para os profissionais da indústria AEC. Para resolver essa questão, as empresas de desenvolvimento de softwares BIM complementaram as interfaces com ambientes VPLs.

De acordo com Serra e Hage (2019), a programação visual é o processo de estabelecer regras que descrevem as relações entre as partes de um projeto, formalizadas por meio de um algoritmo. Um algoritmo, por sua vez, é um conjunto de instruções que devem ser seguidas para realizar uma tarefa. Essas instruções podem ser expressas em texto em diversos idiomas ou de maneira gráfica. Ghannad *et al.* (2019) complementam a definição citando que VPL pode

ser descrita como uma linguagem formal com notação gráfica, que emprega um sistema modular de sinais e regras com elementos visuais em vez de textuais no nível semântico e sintático.

As VPLs geram algoritmos baseados em expressões visuais para criar scripts de sequência de processos e são representados como grafos de processo, em vez de linhas de código da programação convencional. A sequência de nós é semelhante a um fluxograma (Figura 22), proporcionando aos usuários legibilidade e facilidade na auditoria de erros no algoritmo, sendo composta pelos seguintes elementos Collao *et al.* (2021):

- **Entrada:** Informações iniciais requeridas pelos algoritmos VPL para desenvolver o fluxo de instruções.
- **Nós:** Operadores que geram o fluxo de instruções do algoritmo, submetendo os dados de entrada a funções específicas.
- **Conectores:** Linhas que unem os diferentes nós para criar o algoritmo. Os conectores que fluem em direção a um nó são considerados conectores de entrada, enquanto aqueles que fluem para fora são considerados conectores de saída desse nó.
- **Saída:** Informação final entregue pelos nós durante a execução do algoritmo.



Fonte: Autor (2025).

2.10.1 Modelo de execução de uma VPL

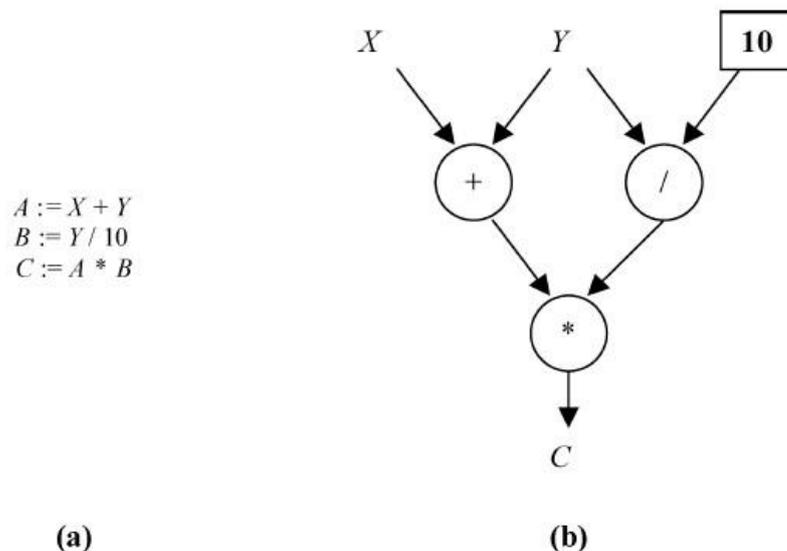
No modelo de execução em fluxograma, os dados fluem como pacotes ao longo dos conectores, operando segundo o princípio *First In, First Out* (FIFO). Uma fila FIFO é uma estrutura que executa as solicitações na ordem em que chegam até o nó. Quando o programa se inicia, nós de ativação especiais inserem dados nos conectores-chave de entrada, acionando o restante do programa. Sempre que um conjunto específico de conectores de entrada de um nó (denominado conjunto gatilho) contém dados, o nó é considerado disparável. Um nó disparável é executado em um determinado momento após o acionamento do conjunto gatilho. Como

resultado, ele remove um pacote de dados de cada conector no conjunto gatilho, executa sua operação e insere um novo pacote de dados (Johnston; Hanna; Millar, 2004).

A principal vantagem é que, no modelo em fluxograma, mais de uma instrução pode ser executada simultaneamente. Assim, se várias instruções se tornarem disparáveis ao mesmo tempo, elas podem ser executadas em paralelo. Esse princípio simples oferece o potencial para execução paralela massiva no nível da instrução.

Um exemplo de execução em fluxograma versus um algoritmo sequencial em texto é ilustrado na Figura 23. A Figura 23(a) apresenta instruções sequenciais para três operações matemáticas simples, enquanto a Figura 23(b) mostra a representação dessas operações em forma de fluxograma. As setas representam os conectores e os círculos representam os nós de instrução. O quadrado representa um valor constante, definido como entrada no programa. As letras designam as entradas e a saída do algoritmo. Quando mais de um conector sai de uma determinada entrada, isso indica que o valor único é duplicado e transmitido em cada caminho.

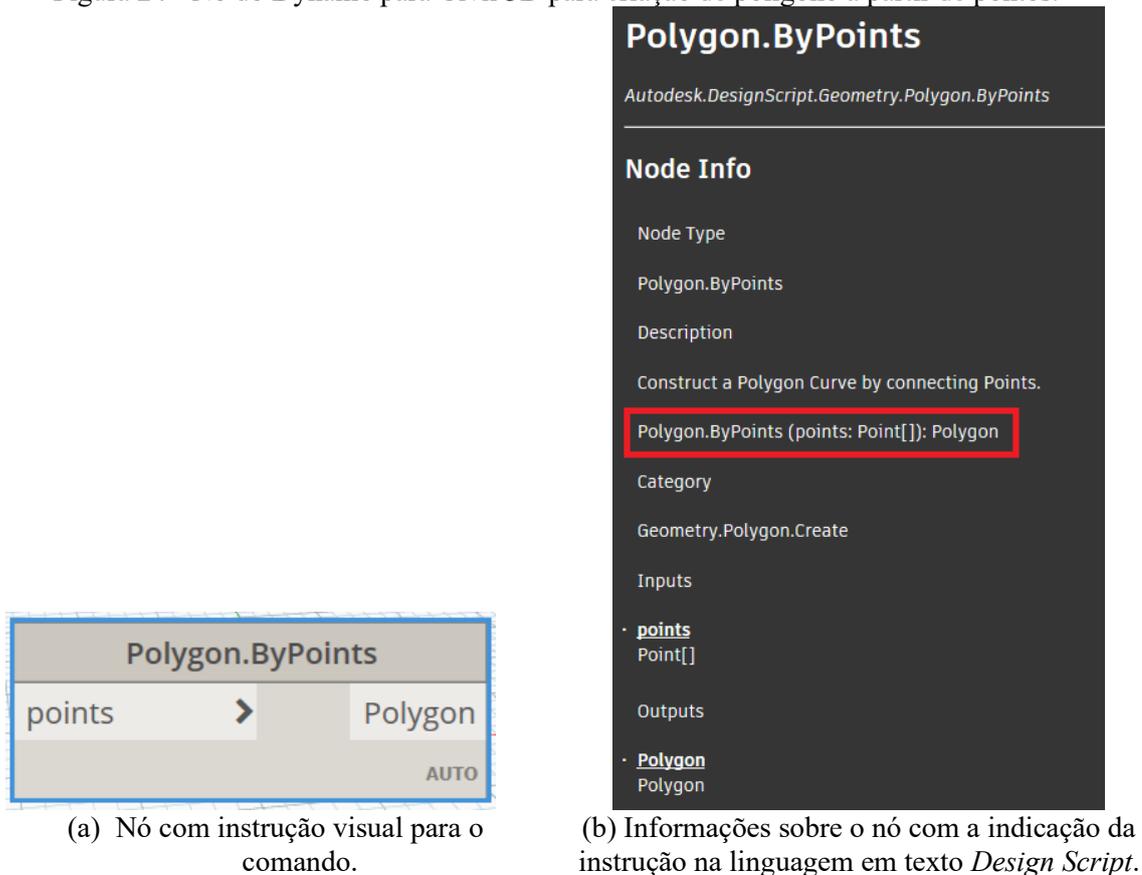
Figura 23 - Instruções para operações matemáticas simples (a) e seu fluxograma equivalente (b).



Fonte: Johnston, Hanna e Millar (2004).

Cada nó geralmente representa uma parte de um código modularizado, assim como uma unidade básica de programação. O nó da VPL do Dynamo para Civil 3D mostrado na Figura 24, por exemplo, tem como saída um polígono criado a partir da ligação sequencial de pontos. Esta notação gráfica também possui a capacidade de representar regras de projeto em uma linguagem inteligível tanto por máquinas quanto por humanos (Ghannad, *et al.*, 2019).

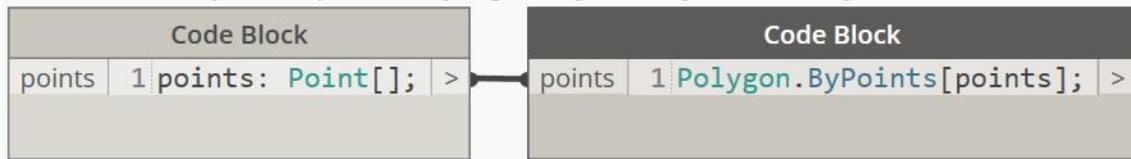
Figura 24 - Nó do Dynamo para Civil 3D para criação de polígono a partir de pontos.



Fonte: Autor (2025).

Collao *et al.* (2021) acreditam que, ao contrário das VPLs, que são baseadas em conceitos básicos de programação de computadores para facilitar a interpretação com mais rapidez e facilidade, a complexidade das linguagens de programação baseadas em texto dificulta o acesso de iniciantes em programação. No exemplo do nó mostrado na Figura 24, por exemplo, caso se desejasse instruir a criação de um polígono a partir de pontos no Civil 3D por meio de linguagem em texto, seria necessário digitar linhas de código em caixas de código conforme instruído na Figura 24(b) (Figura 25).

Figura 25 - Sequência de comandos em texto em caixas de código dentro do Dynamo para Civil 3D para criação de um polígono a partir de pontos em sequência.

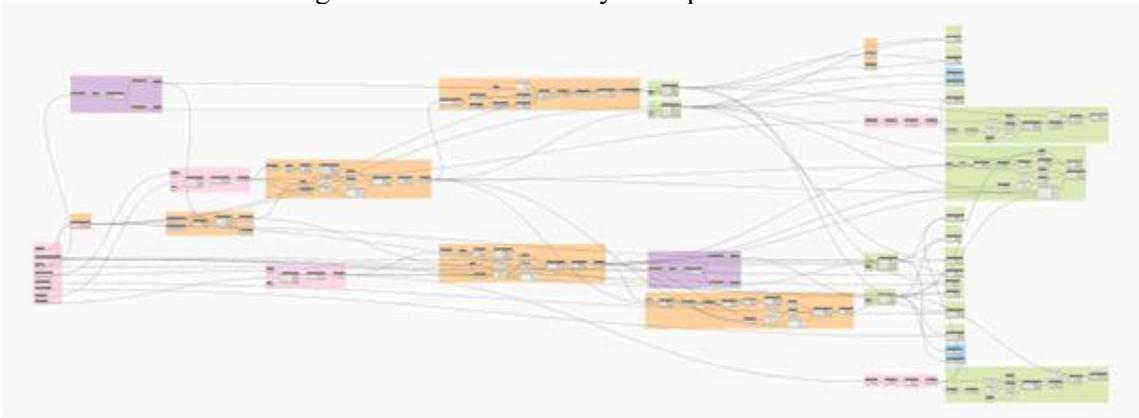


Fonte: Autor (2025).

2.10.2 Aplicativo Dynamo para Civil 3D

O Dynamo é uma ferramenta de programação visual que interage com o Civil 3D e outros softwares da Autodesk. Esta ferramenta permite aos usuários criarem programas ou códigos por meio da manipulação gráfica de comandos do programa, em vez de programá-los por meio de linhas de código. A ferramenta é baseada no conceito de "caixas e setas", onde caixas e outros objetos são considerados entidades conectadas por setas ou linhas que representam suas relações (Figura 26). Sua interface de comunicação é de fácil compreensão e uso intuitivo, graças à estrutura de caixas e setas. Com o uso dessa ferramenta, mesmo usuários com pouca experiência podem ter a possibilidade de automatizar tarefas repetitivas em seus projetos.

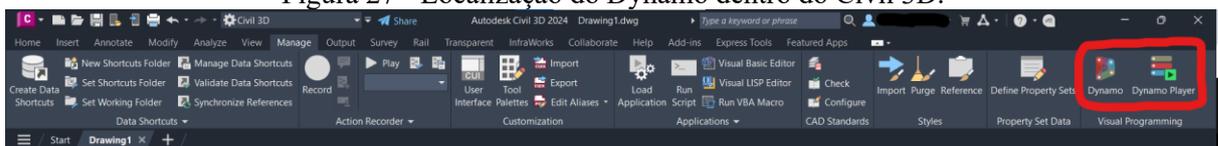
Figura 26 - Interface do Dynamo para Civil 3D.



Fonte: Autor (2025).

O Dynamo pode ser localizado dentro do Civil 3D (Figura 27) na aba *Manage > Visual Programming Panel > Dynamo*:

Figura 27 - Localização do Dynamo dentro do Civil 3D.

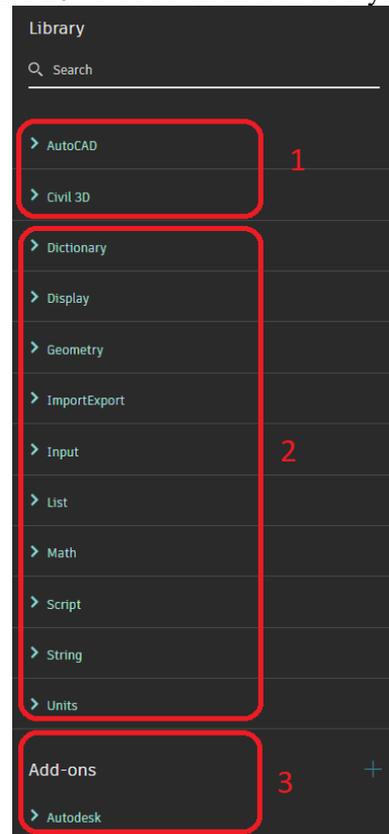


Fonte: Autor (2025).

A biblioteca do Dynamo contém um vasto número de nós (ou entidades). Os nós funcionam como comandos, e cada nó realiza uma operação específica. A categoria nº 1 (Figura 28) inclui

nós específicos para objetos do AutoCAD e do Civil 3D. A categoria nº 2 faz parte do núcleo das funções básicas do Dynamo, abrangendo operações numéricas, importação e exportação de dados. Nessa mesma categoria estão armazenados os nós personalizados pelo usuário ou os pacotes baixados da biblioteca da Autodesk.

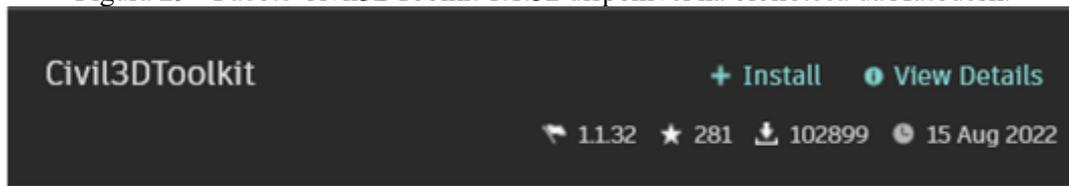
Figura 28 - Nós da biblioteca do Dynamo.



Fonte: Autor (2025).

Um pacote do Dynamo e um conjunto de nós personalizados foram criados a partir dos nós das bibliotecas nativas da ferramenta, ou desenvolvidos do zero utilizando a linguagem de programação C# (Serra; Hage, 2019). Para a elaboração do algoritmo objeto deste estudo, foi baixado da biblioteca da Autodesk o pacote de nós Civil3DToolkit (Figura 29), na versão 1.1.32, desenvolvido por Serra e Hage (2022).

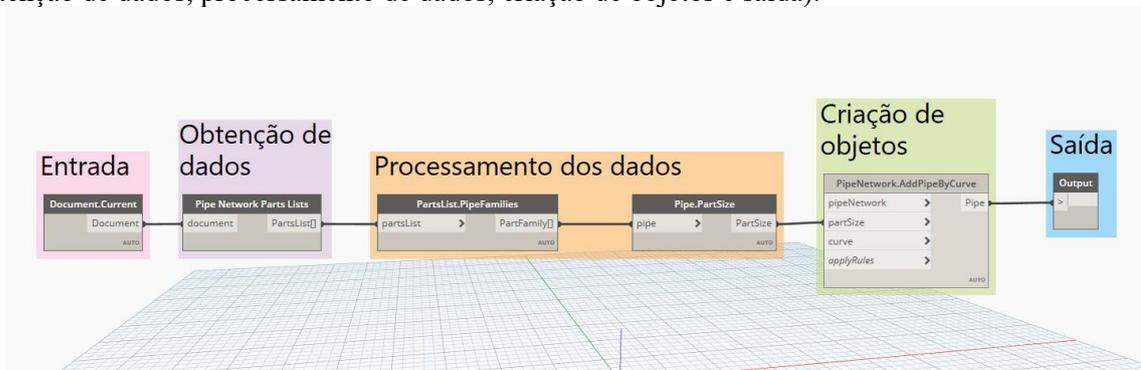
Figura 29 - Pacote Civil3DToolkit 1.1.32 disponível na biblioteca da Autodesk.



Fonte: Autor (2025).

Como demonstrado na Figura 30, o Dynamo permite a utilização de diferentes esquemas de cores para representar as funções de cada grupo de nós. Também é possível nomear cada grupo para indicar sua função ou outras informações adicionais.

Figura 30 - Esquema de cores utilizado pelo autor (em ordem, da esquerda para a direita: entrada, obtenção de dados, processamento de dados, criação de objetos e saída).



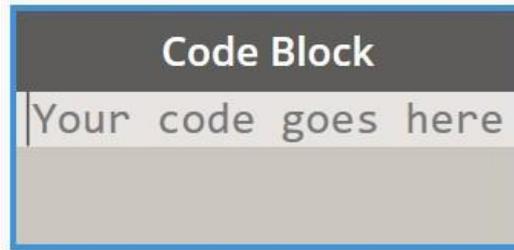
Fonte: Autor (2025).

2.10.3 A linguagem *Design Script*

O Dynamo utiliza a linguagem, compilador e processador de execução da *Design Script* como seu mecanismo computacional. A *Design Script* é uma linguagem legível e concisa que oferece resultados imediatos para pequenos trechos de código, assim como para interações grandes e complexas. A *Design Script* também constitui a base do mecanismo que impulsiona a maioria dos aspectos operacionais das "caixinhas" do Dynamo (Aish, 2016).

Ao utilizar o ambiente de programação de fluxo de dados visuais do Dynamo, o usuário pode não perceber, mas cada nó com instrução visual, ou "caixinha", é, na verdade, uma instrução em *Design Script*. A "unidade mínima" dos nós é o *Code Block*, onde o usuário pode inserir desde textos e operações matemáticas simples até instruções em *Design Script* (Figura 31).

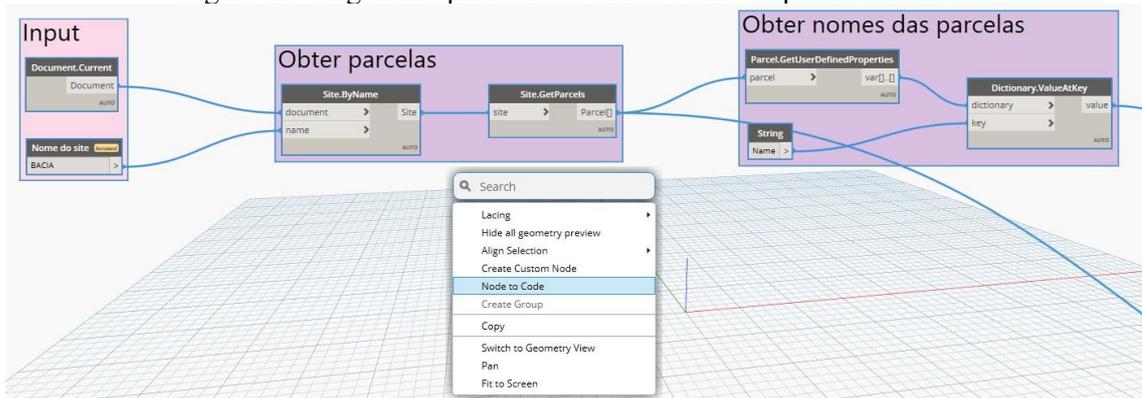
Figura 31 - *Code Block*, a “unidade mínima” da linguagem visual do Dynamo.



Fonte: Autor (2025).

Aish (2016) explica que a funcionalidade *Node to Code*, ou "Nó para Código" em português, pode ser utilizada para criar a instrução equivalente em *Design Script* dentro de um nó *Code Block*. Em outras palavras, o *Node to Code* expõe o código *Design Script* que opera nos bastidores dos nós, fornecendo uma tradução conveniente e acessível da linguagem de programação visual para a linguagem baseada em texto, como ilustrado na Figura 32.

Figura 32 - Algoritmo para obter *Parcels* e seus respectivos nomes.



(a) Execução do comando *Node to Code*.

```

Code Block
1 document1 = Autodesk.Document.Current;
2 t1 = "BACIA";
3 site1 = Site.ByName(document1, t1);
4 parcel1 = Site.GetParcels(site1);
5 t2 = Parcel.GetUserDefinedProperties(parcel1);
6 t3 = "Name";
7 t4 = Dictionary.ValueAtKey(t2, t3);

```

(b) Instruções em texto equivalentes na linguagem *Design Script*.

Fonte: Autor (2025).

Aish (2016) por fim destaca que a linguagem *Design Script* é independente do Dynamo. Por exemplo, o Dynamo implementa funções geométricas que, embora possam ser programadas usando o *Design Script*, utilizam classes e métodos geométricos que não fazem parte da linguagem. Outras bibliotecas, incluindo bibliotecas de geometria, podem ser desenvolvidas utilizando ferramentas de desenvolvimento em C# ou Python, por exemplo.

Capítulo 3

3 METODOLOGIA

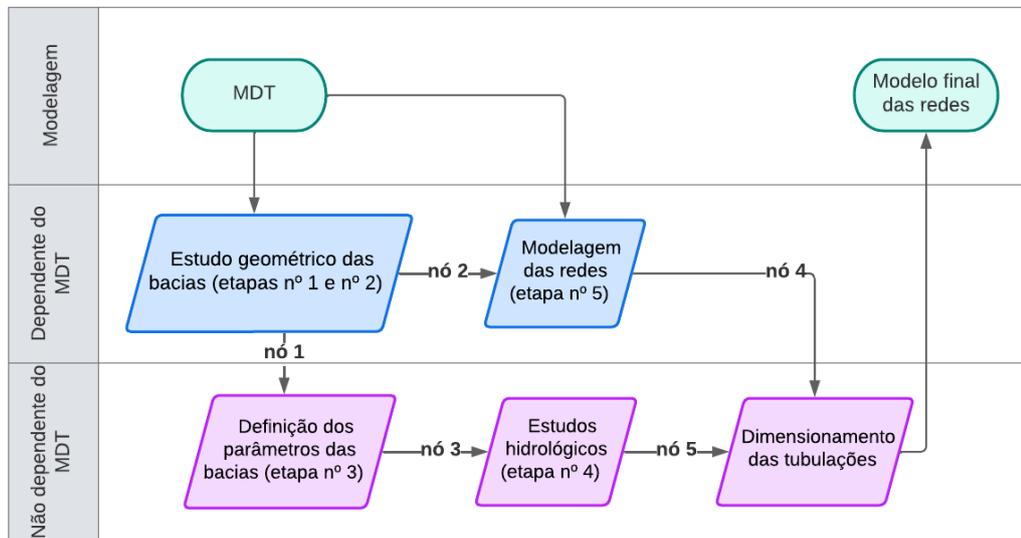
O processo de modelagem de tubulações e de estruturas para um projeto de drenagem pluvial urbana no Civil 3D segue uma sequência de tarefas encadeadas que deve começar pela obtenção de um MDT georreferenciado.

As informações sobre as elevações do terreno são essenciais para que dados extraídos e objetos modelados estejam sempre situados no exato local da superfície terrestre. Caso não seja carregado um MDT no início da modelagem, as ferramentas BIM de definição de bacias e de modelagem de redes de drenagem (caixas azuis da Figura 33) não poderão ser utilizadas de forma adequada, uma vez que essas duas entidades são dependentes de coordenadas tridimensionais.

Outras etapas no processo de modelagem não necessitam diretamente do MDT ou limitam-se apenas ao plano XY, e podem ser executadas sem cotas extraídas da superfície, como a definição dos parâmetros das bacias e o dimensionamento das tubulações da Figura 33; entretanto, alguns dados de entrada e condições de contorno para a realização dessas tarefas ainda dependem dos dados volumétricos do terreno, como a etapa dos estudos hidrológicos na Figura 33.

É importante salientar que este estudo não abordará o método empregado na confecção do MDT. Partir-se-á do ponto em que esse modelo já se encontra finalizado. Ainda assim, vale ressaltar que um dos MDTs utilizados em estudos de caso representava uma superfície já existente e foi elaborado a partir da combinação do levantamento topográfico convencional, utilizando estação total e GPS, com a aerofotogrametria realizada com drone. Os dados levantados foram posteriormente tratados em softwares específicos para essa tarefa, permitindo obter uma superfície retificada, com a remoção de pontos que pudessem impedir uma triangulação adequada. Em outro estudo de caso, o MDT utilizado foi modelado a partir de um projeto de loteamento de um condomínio.

Figura 33 - Fluxograma de atividades em um projeto de drenagem pluvial urbana. As cinco etapas são as mesmas apresentadas no Capítulo 1.



Fonte: Autor (2025).

A motivação para este estudo surgiu a partir da análise do fluxograma apresentado na Figura 33 em um projeto real. O autor constatou que o caminho das setas que conectam as atividades, embora transmita a ideia de um fluxo contínuo de ações sem interrupções, assemelha-se mais à descida de uma polia em uma tirolesa, como na Figura 34, porém com uma corda cheia de nós. Ou seja, em cada nó da corda, o fluxo é interrompido e a polia só pode prosseguir sua descida caso esse nó seja desatado.

Figura 34 - Analogia da polia andando sobre uma corda numa tirolesa.



Fonte: Parque Bondinho do Pão de Açúcar (2023).

Deve-se considerar, portanto, que:

- No fluxo, ou corda, na analogia deste trabalho, que conecta a obtenção do MDT até o Modelo final das redes, existe uma série de nós (essa corda será denominada “corda MDT-Modelo”);
- A corda MDT-Modelo está inclinada a favor da gravidade, similar a uma tirolesa;
- O processo de modelagem pode ser comparado a um "pacote" preso a uma polia que deve ser transportado pela corda MDT-Modelo (Figura 34);
- O fluxo natural do pacote a favor da gravidade representa a execução de tarefas automáticas pelos softwares, sem ou com mínima interferência do projetista;
- Cada nó na corda representa um evento que impede o prosseguimento contínuo do fluxo de modelagem por meio de tarefas automáticas; e
- Sempre que o fluxo natural é interrompido por um nó, é necessário desatar o nó por meio de intervenção humana.

Os nós da corda MDT-Modelo foram contabilizados em 5 (cinco), como apresentado na Figura 33, e definidos da seguinte forma:

- **Nó 1 (Obtenção dos polígonos representativos das bacias do MDT):** Esta atividade é geralmente executada por meio da definição manual das bacias, criando-se polígonos 2D sobre o MDT com a ferramenta *Create Catchment from Object* do Civil 3D, que são posteriormente transformados em objetos do tipo *Catchment*. Alternativamente, pode-se realizar a definição automática de cada bacia mediante a seleção individual de cada ponto de acúmulo de água sobre o MDT utilizando a ferramenta *Create Catchment from Surface* do Civil 3D. Em ambos os processos, há a necessidade de repetição para cada bacia a ser obtida;
- **Nó 2 (Obtenção dos pontos de acumulação de água nas bacias obtidas do MDT):** Esta atividade é normalmente executada em conjunto com a atividade número 1. Quando a bacia é obtida com a ferramenta *Create Catchment from Object*, é necessário indicar manualmente o ponto de acumulação de água após a criação da bacia. Quando a bacia é obtida com a ferramenta *Create Catchment from Surface*, o ponto de acumulação de água deve ser indicado antes da criação da bacia. Em ambos os processos, há a repetição de tarefas para cada bacia a ser obtida;

- **Nó 3 (Atribuição dos valores de *runoff* às bacias, que possibilitará o cálculo da vazão da chuva):** Esta atividade é realizada de forma recursiva e manual para cada bacia *Catchment*, acessando as propriedades do objeto e alterando a propriedade *Runoff coefficient*. Nesta etapa, ocorre a maior probabilidade de erros de dimensionamento no projeto, pois, como não há ferramenta para mensuração desse coeficiente, frequentemente o projetista acaba estimando os valores do *runoff* apenas por meio de observação visual. Uma estimativa incorreta pode resultar em erros no dimensionamento das tubulações;
- **Nó 4 (Modelagem dos objetos das redes de drenagem, incluindo ajustes nas elevações das tubulações):** Nesta atividade, o projetista exercita sua criatividade na resolução de problemas e propõe uma solução para o melhor traçado da rede. Como explicado no Capítulo 1, cada projeto demanda uma solução personalizada e, ainda assim, não há uma solução única de traçado; e
- **Nó 5 (Carregamento das Bacias Parametrizadas):** Carregamento das bacias parametrizadas na ferramenta de dimensionamento automático de redes de tubulações no Civil 3D. Esta atividade pode ser automática se a etapa de obtenção das bacias do tipo *Catchment* e definição dos parâmetros for feita manualmente; contudo, neste estudo, as bacias criadas serão do tipo *Watershed*, o que não permite a execução automática da atividade.

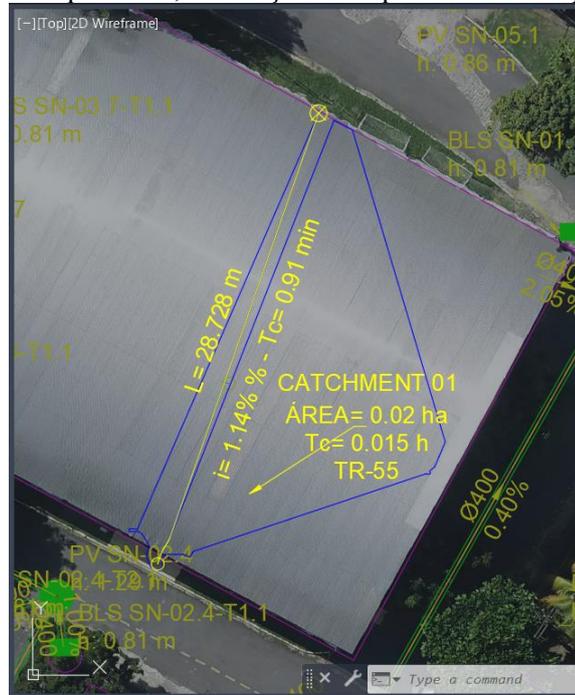
Com base nessa analogia, o método desenvolvido no presente estudo visa solucionar os nós 1, 2, 3 e 4 por meio da implementação de quatro algoritmos de automação. Esta abordagem elimina a necessidade de intervenção humana recorrente no processo de modelagem de redes de drenagem pluvial urbana. O problema referente ao nó 5 será apresentado neste estudo; no entanto, sua solução poderá ser explorada em uma pesquisa futura. Caso o quinto nó também seja resolvido, espera-se alcançar um processo totalmente automatizado de modelagem de tubulações de drenagem pluvial urbana.

Outro fato essencial para compreensão da metodologia deste trabalho é que existem dois tipos de bacias hidrográficas que podem ser definidas no Civil 3D: as *Catchments* e as *Watersheds*.

Uma bacia do tipo *Catchment* (Figura 35) é composta de um perímetro, representa a área da bacia hidrográfica que produz o escoamento, e de uma linha do fluxo de água representando

os dados de tempo de concentração do escoamento, que, no caso do Civil 3D, podem ser calculados automaticamente usando o método TR-55.

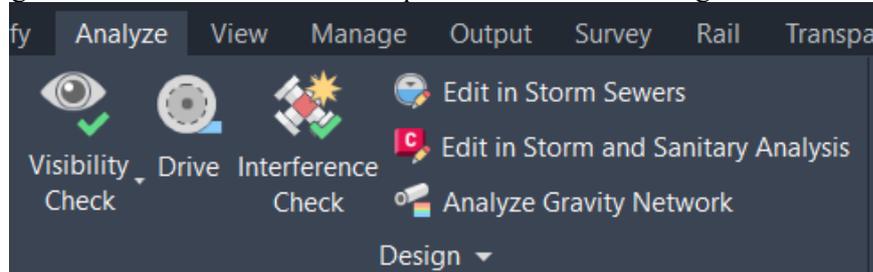
Figura 35 - *Catchment* com o perímetro em azul e a linha de fluxo de água em amarelo com os dados de comprimento, inclinação e tempo de concentração.



Fonte: Autor (2025).

Quando uma bacia *Catchment* é associada a uma estrutura coletora da rede de drenagem, os dados da captação podem ser analisados dentro do próprio Civil 3D usando a ferramenta *Analyze Gravity Network* ou exportados para o *Storm and Sanitary Analysis* para posterior análise hidrológica e dimensionamento do projeto (Figura 36).

Figura 36 - Ferramentas de análise pluviométrica e de drenagem no Civil 3D.



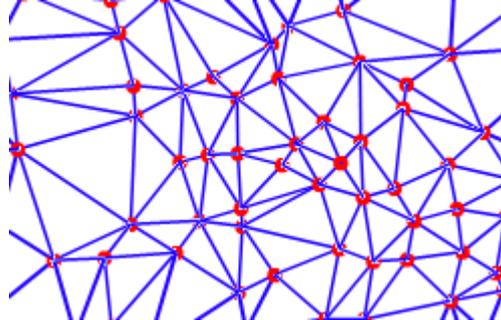
Fonte: Autor (2025).

Já em uma bacia do tipo *Watershed*, que é o tipo de bacia que será estudado neste trabalho, as arestas que formam as superfícies TIN² (*Triangular Irregular Networks*) (Figura 37) são

² Superfícies TIN são uma forma de dados geográficos digitais baseados em vetores e construídos pela triangulação de um conjunto de pontos. Os vértices são conectados com uma série de arestas para formar uma rede de triângulos (ESRI, 2024).

usadas para calcular as áreas em que a água flui ao longo da superfície (Autodesk, 2024b). A partir dessas áreas, os locais de acúmulo das águas e as bacias são determinados.

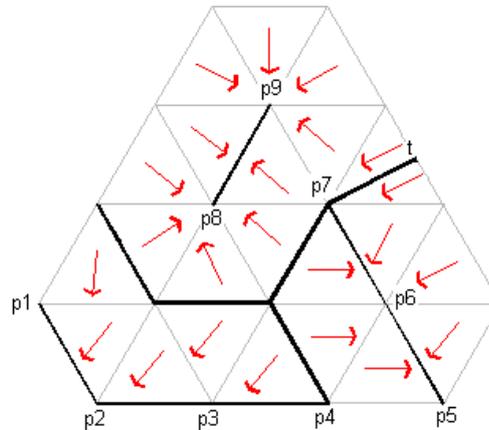
Figura 37 - Superfície TIN com os pontos em vermelho e as arestas em azul.



Fonte: ESRI (2024).

Diferentes tipos de locais de acúmulo de água são calculados para as *Watersheds* de uma superfície TIN, como: um ponto no perímetro da bacia que recebe a água de um talvegue (ponto p5 na Figura 38); talvegue para onde a água flui no meio da bacia (segmento p8-p9 na Figura 38); ou ainda segmentos no perímetro da superfície (segmento p1-p2-p3-p4 na Figura 38).

Figura 38 - *Watersheds* no Civil 3D com as setas de indicação do fluxo de água.



Fonte: Autodesk (2024b).

3.1 SOFTWARES EMPREGADOS

Como descrito por Wang, Pan e Luo (2019), e apontado por este autor no tópico 2.9.1, BIM e GIM, ou suas ferramentas BIM e GIS correspondentes, podem estar em um mesmo nível de coordenação ou com uma das duas com ascendência. Neste estudo, notou-se que, para cumprimento dos objetivos, foi necessário que a ferramenta BIM coordenasse a ferramenta GIS, uma vez que os parâmetros necessários foram essencialmente identificados e gerados pela primeira, tendo a segunda apoiado na determinação dos valores desses parâmetros por meio de suas ferramentas de programação.

3.1.1 Ferramenta GIS

Diferentemente do que apontaram diversos autores durante a revisão bibliográfica, softwares GIS, quando integrados ao BIM em um mesmo projeto, não necessariamente são utilizados apenas para a manipulação de um grande volume de informações geoespaciais, mas também devem ser empregados quando se perceber que são ou possuem ferramentas mais adequadas que as BIM.

No caso de uma quantidade limitada de dados espaciais de um loteamento, por exemplo, certos atributos bidimensionais poderiam ser tanto trabalhados em um software como o Civil 3D, quanto no QGIS. Porém no primeiro software precisariam ser, ou manipulados com suas coordenadas *z*, ou inicialmente planejados antes de serem utilizados, o que nos dois casos implicaria em maiores demandas por processamento. Uma saída viável é exportar as informações BIM para um formato GIS compatível entre os dois domínios e estudá-los no segundo software.

O software escolhido para a manipulação dos dados bidimensionais das bacias foi o QGIS 3.34.13 - Prizren (Figura 39) por ser um aplicativo gratuito, capaz de manipular tanto arquivos SDF, quanto *shapefiles*, e também por permitir desenvolver algoritmos de automação usando sua VPL.

Figura 39 - QGIS 3.34.13 - Prizren.



Fonte: Autor (2025).

3.1.2 Ferramentas BIM

O Civil 3D é o software de projetos de infraestrutura civil da Autodesk que oferece suporte à Modelagem BIM. Ele possui recursos integrados para aprimorar a documentação de desenhos, projetos e construção (Figura 40).

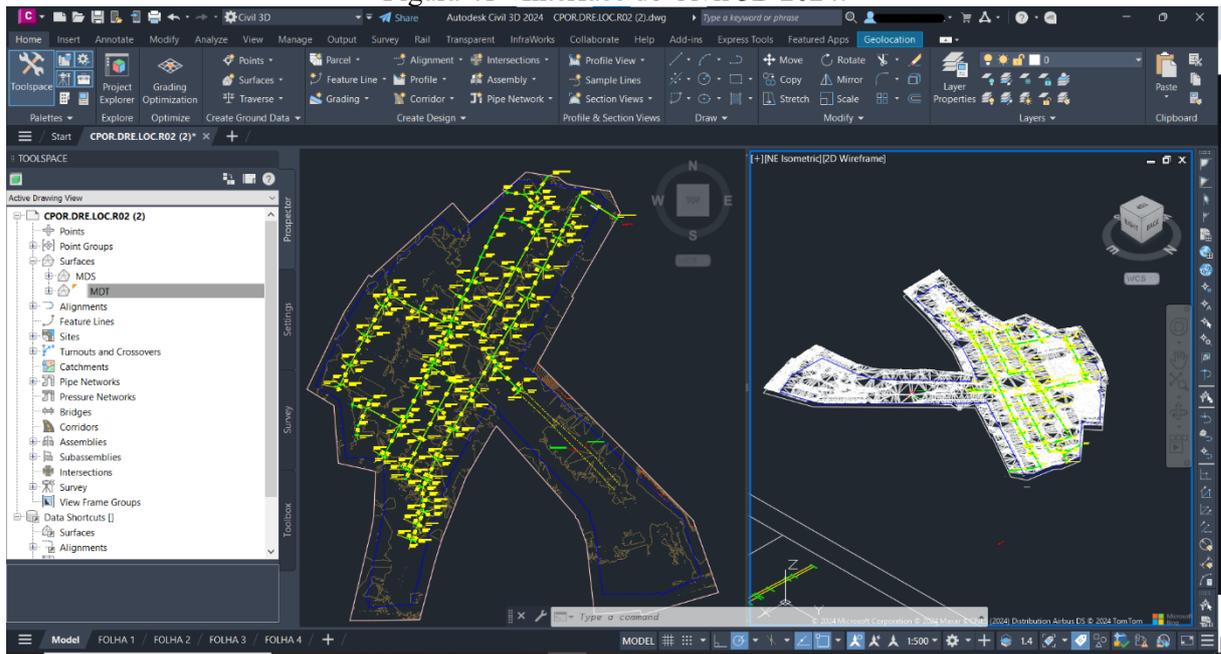
Figura 40 - Autodesk Civil 3D 2024.



Fonte: Autor (2025).

Apesar de nenhum software, de maneira isolada, representar o conceito de BIM, o Civil 3D é uma ferramenta que aplica conceitos da metodologia, porém voltados para projetos de infraestrutura, como: a criação de objetos parametrizados, incluindo a possibilidade de atribuição de novos parâmetros e a exportação de modelos IFC; a atualização automática de vistas e de quantitativos; a interoperabilidade com outros softwares; e a possibilidade de compartilhamento de informações em tempo real entre as diversas equipes envolvidas no mesmo projeto e que estejam utilizando o software.

Figura 41 - Interface do Civil 3D 2024.



Fonte: Autor (2025).

3.1.3 AutoCAD

Como ferramenta complementar, foi utilizado o AutoCAD para definição das formas geométricas representativas das superfícies do terreno (Figura 42).

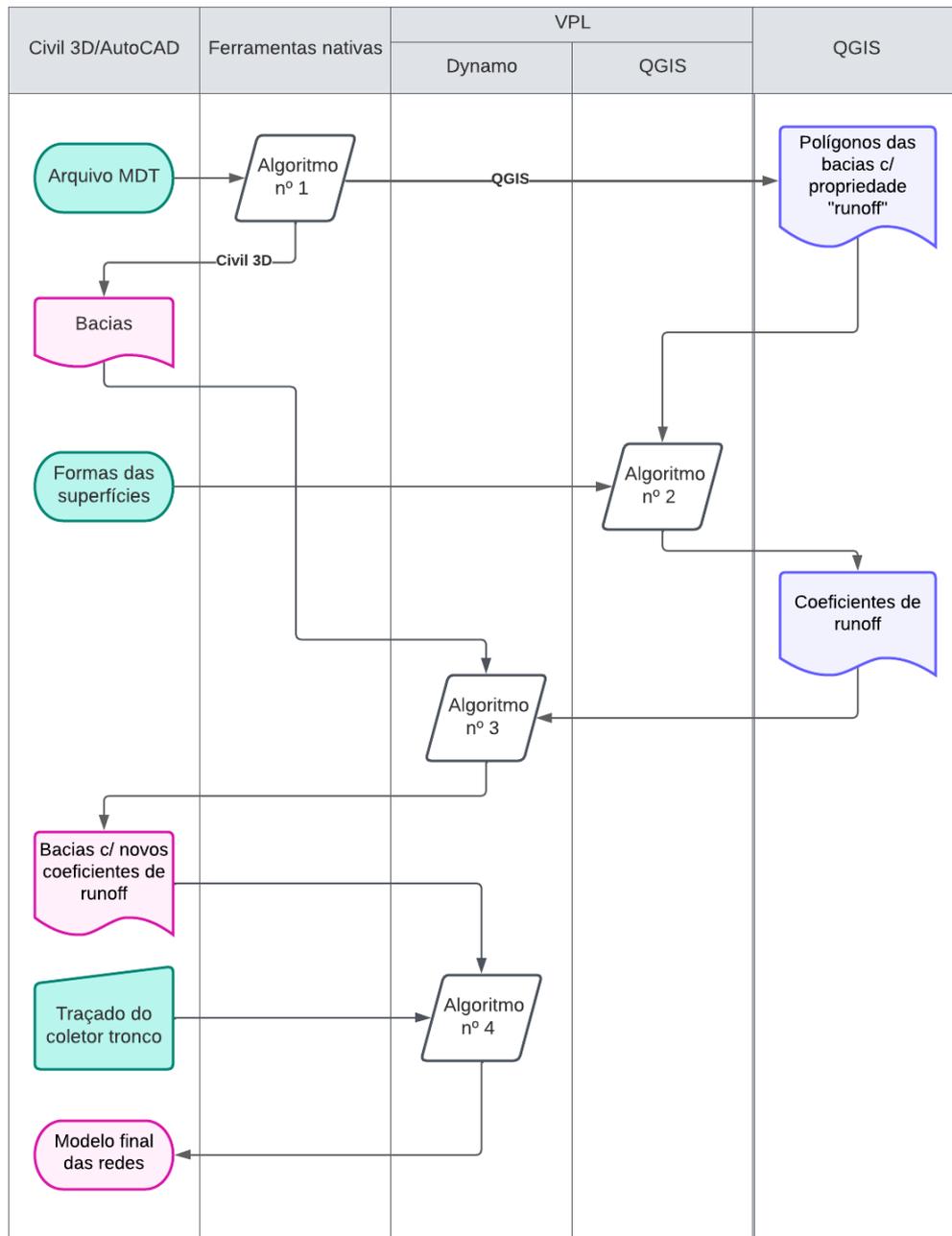
Figura 42 – Arquivo .dwg com as formas geométricas representativas das superfícies do terreno.



3.2 OS ALGORITMOS

O fluxograma das atividades realizadas pelos algoritmos desenvolvidos e implementados neste estudo foi explicitado na Figura 43:

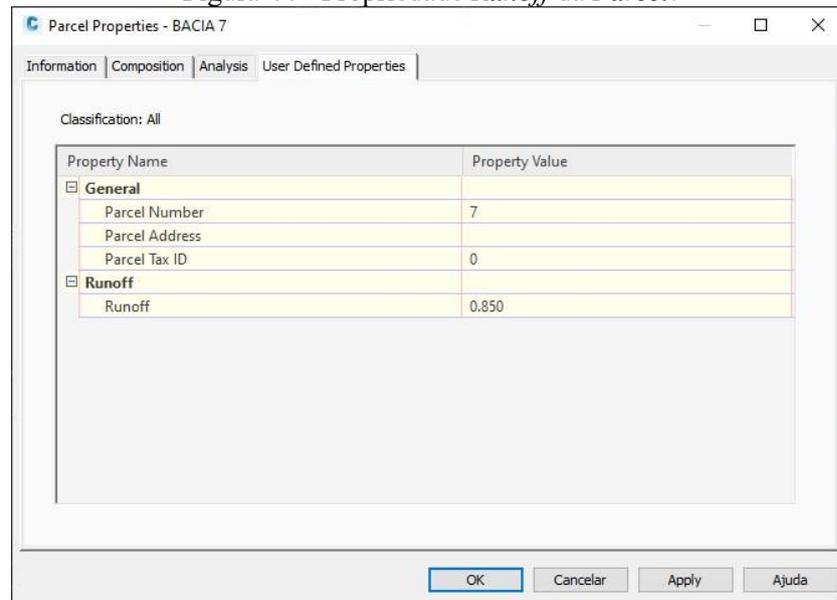
Figura 43 - Fluxograma das tarefas executadas. Em ciano as entradas, em rosa os arquivos no formato .dwg do Civil 3D e do AutoCAD, e em roxo os arquivos do QGIS.



Fonte: Autor (2025).

Diferentemente do apresentado na Figura 33, que focava em dividir as etapas pela dependência dos dados do MDT, a Figura 43 classifica os algoritmos, suas entradas e saídas, de acordo com o software ou ferramenta utilizada para o processamento.

Inicialmente um MDT é carregado no Civil 3D, e, por meio de ferramentas do próprio software, o algoritmo nº 1 é compilado, resultando nos polígonos, como *Parcels*, e nos pontos de acúmulo de água das bacias no Civil 3D e entregando como saída um arquivo SDF com esses polígonos, já com a propriedade *Runoff* atribuída (Figura 44).

Figura 44 - Propriedade *Runoff* da *Parcel*.

Fonte: Autor (2025).

O arquivo SDF resultante é carregado no QGIS junto com as formas dos diversos tipos de superfície, onde são calculados no algoritmo nº 2, por meio da VPL do software, os valores do atributo *Runoff* e resultando como saída um arquivo SHP, porém já com esse atributo calculado.

Esse arquivo SHP é dessa vez carregado no Civil 3D no mesmo arquivo resultante do algoritmo nº 1, e os valores de *Runoff* são atribuídos às *Parcels* das bacias por meio do algoritmo nº 3, compilado na VPL do *Dynamo*.

Por fim, o traçado bidimensional da rede de drenagem é definido no mesmo arquivo do Civil 3D em que os pontos de acúmulo de água, definidos no algoritmo nº 1, indicam o local onde devem ser inseridas as estruturas coletoras de água. Esse traçado é processado no algoritmo nº 4, também compilado na VPL do *Dynamo*, e resulta em uma rede de tubulações tridimensional, posicionada relativa ao MDT, da entidade *Pipe Network*.

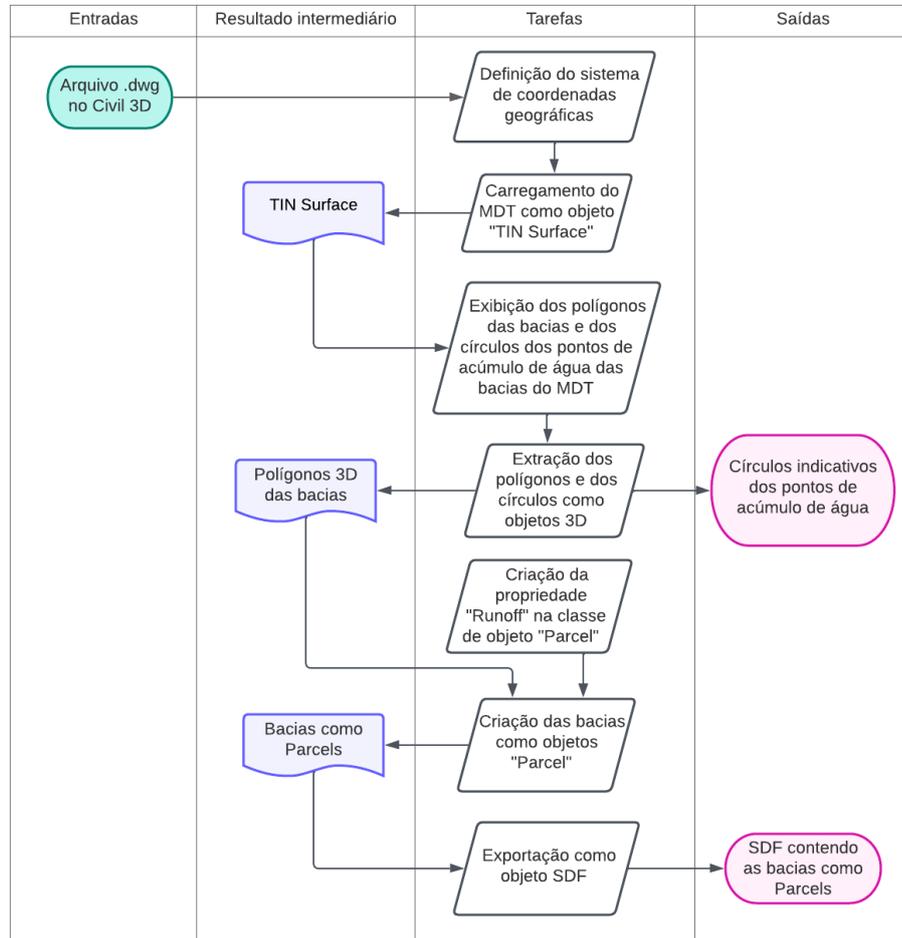
A saída do algoritmo nº 3, apesar de resultar na propriedade hidrológica *Runoff* das *Watersheds* e estar inserida no fluxograma, não foi diretamente empregada no algoritmo nº 4, pois a propriedade seria necessária em um caso em que a rede de tubulações também fosse dimensionada automaticamente.

3.2.1 Algoritmo nº 1 - Obtenção das bacias

O Algoritmo nº 1 tem como objetivo resolver o problema da obtenção das bacias de uma superfície de forma automática, sem a aplicação do método de definição manual das bacias a partir de polígonos. Nesse primeiro algoritmo, não foi utilizada nenhuma linguagem de programação, apenas o encadeamento lógico de tarefas realizadas com ferramentas do próprio

Civil 3D, sem a automação do processo. A sequência de instruções do algoritmo no Civil 3D foi apresentada no fluxograma da Figura 45.

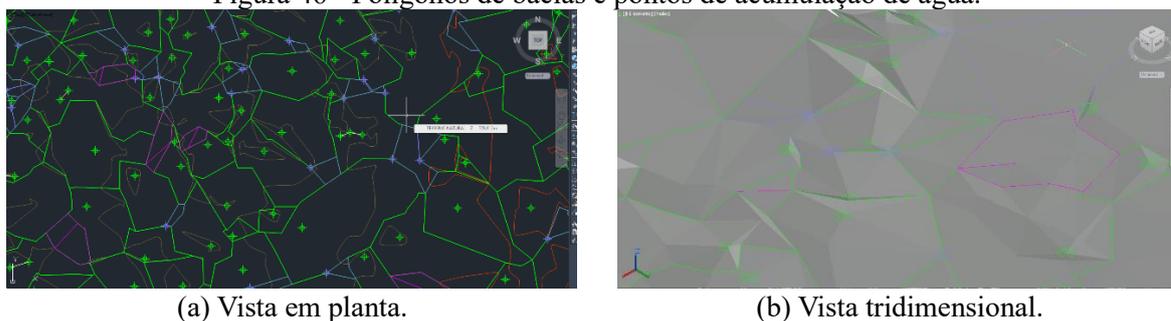
Figura 45 - Algoritmo nº 1 de obtenção e exportação das bacias.



Fonte: Autor (2025).

Após compreender o conceito de *Watershed*, explicado no início do tópico 3, torna-se mais fácil a interpretação do fluxograma da Figura 45 até a etapa da obtenção dos polígonos tridimensionais das bacias e dos círculos indicativos dos pontos de acúmulo de água, como apresentado na Figura 46.

Figura 46 - Polígonos de bacias e pontos de acumulação de água.



Fonte: Autor (2025).

Após sua obtenção, os polígonos podem, então, ser transformados em objetos do tipo *Parcel* no Civil 3D, já com a propriedade *Runoff* criada e atribuída à classe. Após essa etapa, as bacias podem ser exportadas para um arquivo no formato SFD (formato aberto de base de dados geoespaciais desenvolvido pela Autodesk).

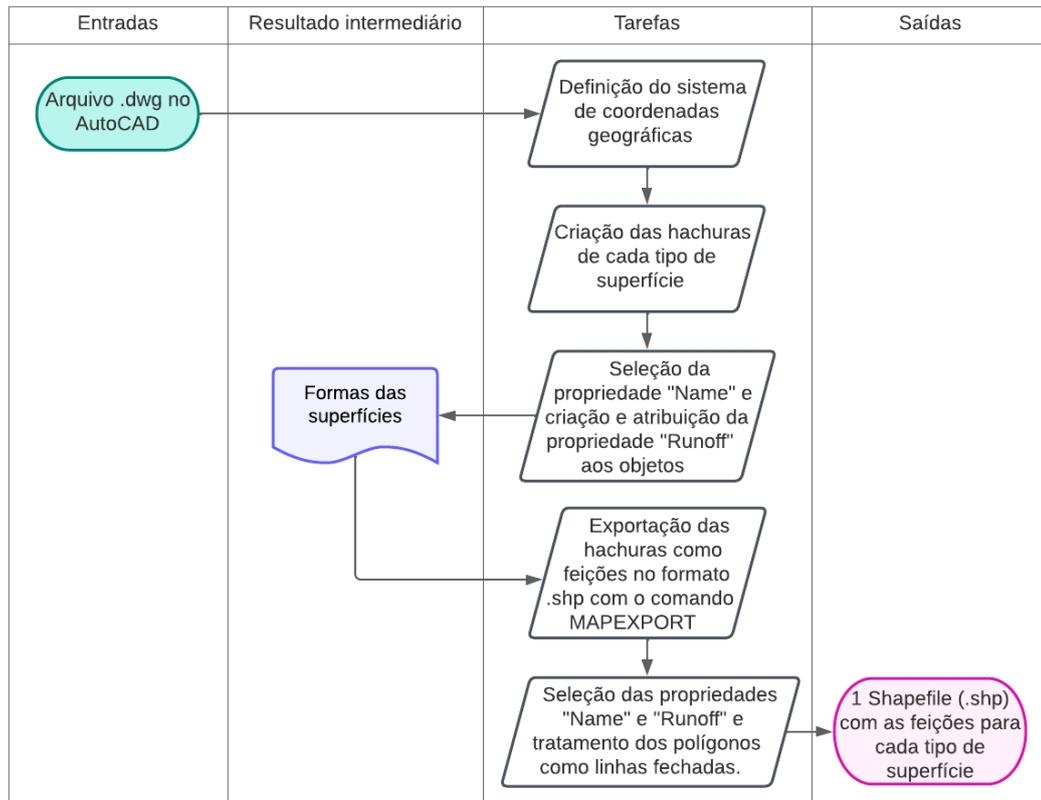
Outra observação importante é que, quanto mais pontos levantados a superfície tiver, mais e menores bacias serão obtidas. No caso de um MDT obtido a partir de levantamento topográfico com drone, que gera nuvens com milhares de pontos por metro quadrado, a quantidade de bacias calculadas utilizando o algoritmo aqui apresentado pode, inclusive, tornar o estudo hidrológico mais trabalhoso. Portanto, é necessário fazer uma filtragem dos pontos do MDT antes da aplicação do Algoritmo nº 1.

3.2.2 Algoritmo nº 2 - Obtenção dos coeficientes de *runoff*

O Algoritmo nº 2 é dividido em 3 etapas e tem como objetivo resolver o problema do cálculo do coeficiente de escoamento superficial por meio da combinação de ferramentas nativas do Civil 3D e do AutoCAD com um algoritmo de programação visual desenvolvido no Modelador Gráfico do QGIS, possibilitando a automação do processo de cálculo da propriedade em estudo.

A primeira etapa do Algoritmo nº 2 é semelhante à do Algoritmo nº 1, com a exceção de que não há necessidade de utilizar um MDT, ou seja, apenas a etapa de manipulação de formas será usada agora. Nesta fase, as hachuras representativas dos diversos tipos de superfícies (concreto, asfalto, grama, telhado, entre outras) são exportadas como *shapefiles*. No momento da exportação, a propriedade *Runoff* é atribuída aos objetos já com o valor predefinido de cada tipo de superfície (Figura 47).

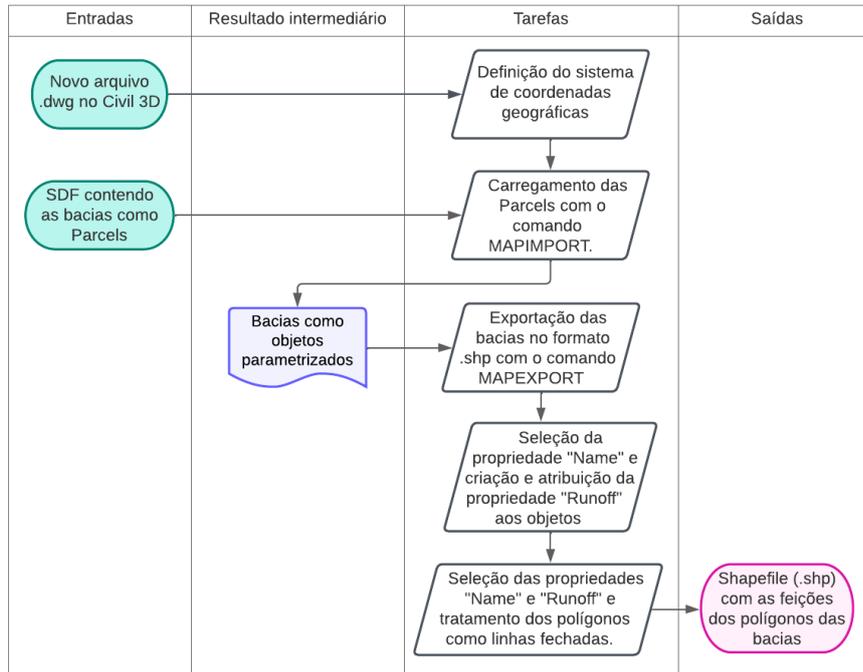
Figura 47 – 1ª etapa do algoritmo nº 2 de exportação das formas das superfícies e exportação como *shapefile*.



Fonte: Autor (2025).

Na segunda etapa, o arquivo SDF obtido pelo Algoritmo nº 1 é carregado em um novo arquivo DWG do Civil 3D, selecionando-se apenas a classe de objeto *Parcels* no momento da confirmação da importação. Após essa tarefa, é atribuída aos objetos parametrizados a propriedade *Runoff*, com valores vazios, e a forma de cada tipo de superfície é exportada como um *shapefile* individual (Figura 48).

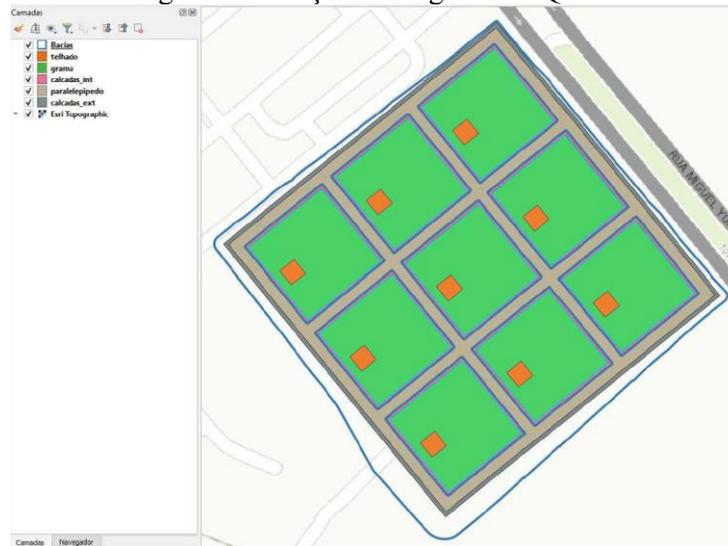
Figura 48 – 2ª etapa do algoritmo nº 2 para a criação da propriedade *Runoff* e exportação das bacias hidrográficas como *shapefile*



Fonte: Autor (2025).

Por fim, no QGIS, os diversos arquivos *shapefile* obtidos até então são carregados no software como feições (Figura 49). O primeiro passo após o carregamento é a exportação de cada polígono da feição das bacias como uma feição individual. Em seguida, todas as feições, com exceção daquela da bacia inicialmente importada, são lidas como entradas para o algoritmo de automação escrito no Modelador Gráfico do QGIS (Figura 50).

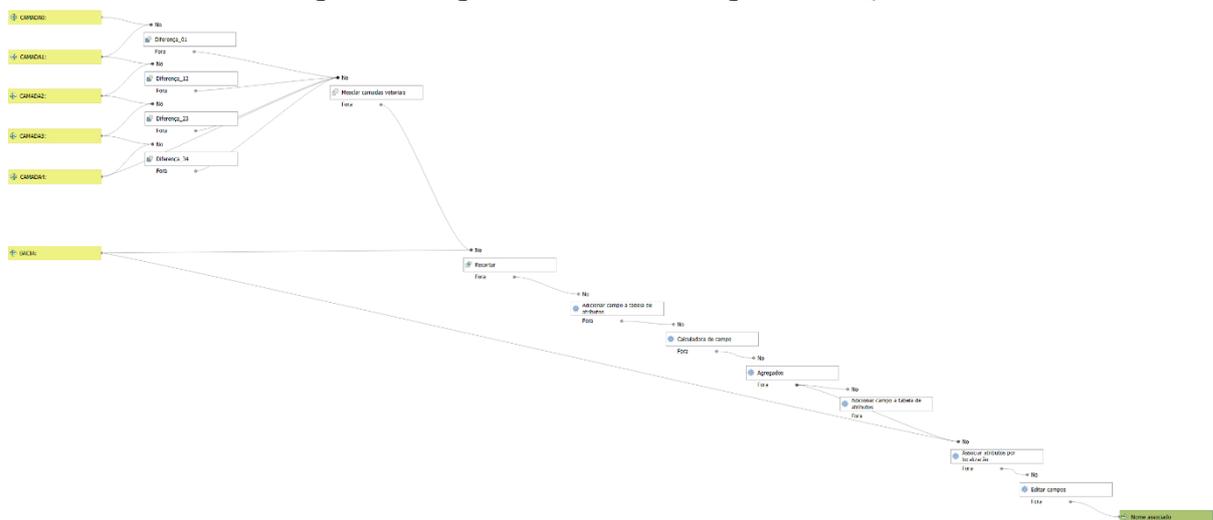
Figura 49 - Feições carregadas no QGIS.



Fonte: Autor (2025).

Nessa etapa, algoritmo recorta inicialmente as feições das superfícies para remover sobreposições de camadas e mescla o resultado, formando uma superfície única. Logo após, o algoritmo gráfico, de posse da superfície mesclada e das feições individuais das bacias, usa esses polígonos das bacias como ferramenta de recorte da superfície mesclada, obtendo assim vários pedaços de superfície, cada um formado por uma mistura de tipos de superfície (concreto, asfalto, grama, telhado, entre outras).

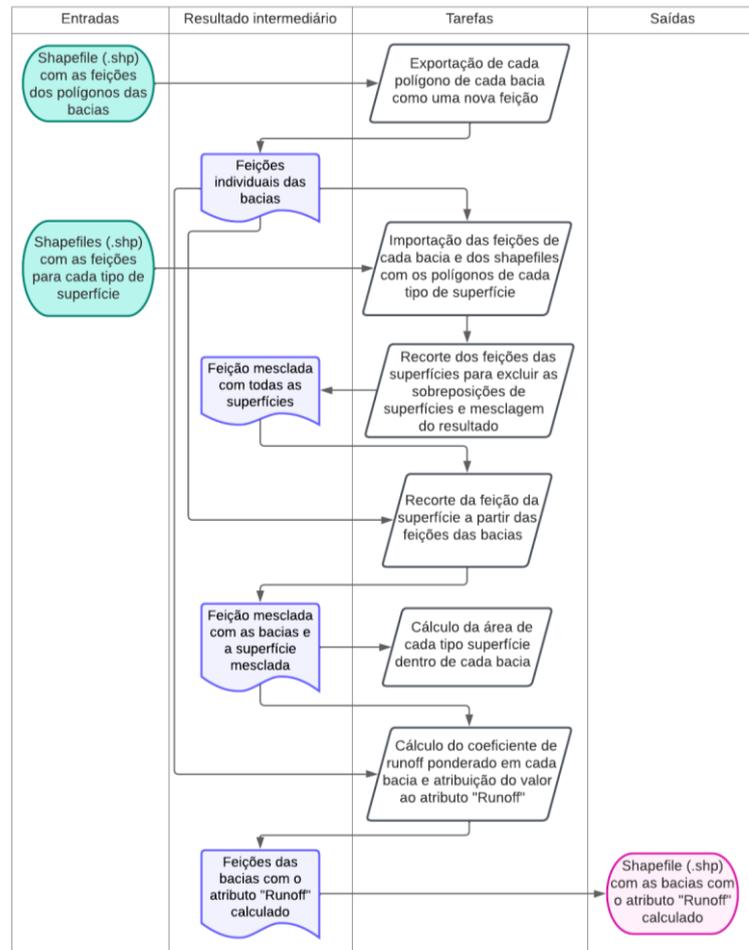
Figura 50 - Algoritmo do modelador gráfico do QGIS.



Fonte: Autor (2025).

A partir daí, o algoritmo calcula em cada superfície recortada, e a partir dos valores da propriedade *Runoff*, a área de cada tipo de superfície e faz o cálculo do *Runoff* ponderado pela área. O valor calculado é inserido na propriedade *Runoff* da feição de cada bacia correspondente, e essas feições são finalmente exportadas como *shapefile* (Figura 51).

Figura 51 – 3ª etapa do algoritmo nº 2 de cálculo do atributo *Runoff* na feição *shapefile* com as bacias.



Fonte: Autor (2025).

O Modelador Gráfico do QGIS permite que os algoritmos escritos sejam exportados na linguagem Python, conforme mostrado na Figura 52 e detalhado no Apêndice A.

Figura 52 - Algoritmo nº 2 em linguagem Python.

```

1 = """
2 Model exported as python.
3 Name: Extrair runoff
4 Group:
5 With QGIS: 32601
6 """
7
8 from qgis.core import QgsProcessing
9 from qgis.core import QgsProcessingAlgorithm
10 from qgis.core import QgsProcessingMultiStepFeedback
11 from qgis.core import QgsProcessingParameterVectorLayer
12 from qgis.core import QgsProcessingParameterFeatureSink
13 import processing
14
15
16 = class ExtrairRunoff(QgsProcessingAlgorithm):
17
18 =     def initAlgorithm(self, config=None):
19         self.addParameter(QgsProcessingParameterVectorLayer('bacia', 'BACIA',
20 self.addParameter(QgsProcessingParameterVectorLayer('camada0', 'CAMADA0
21 self.addParameter(QgsProcessingParameterVectorLayer('camada1', 'CAMADA1
22 self.addParameter(QgsProcessingParameterVectorLayer('camada2', 'CAMADA2
23 self.addParameter(QgsProcessingParameterVectorLayer('camada3', 'CAMADA3
24 self.addParameter(QgsProcessingParameterVectorLayer('camada4', 'CAMADA4
25 self.addParameter(QgsProcessingParameterFeatureSink('NomeAssociado', 'N
26
27 =     def processAlgorithm(self, parameters, context, model_feedback):
28         # Use a multi-step feedback, so that individual child algorithm progres

```

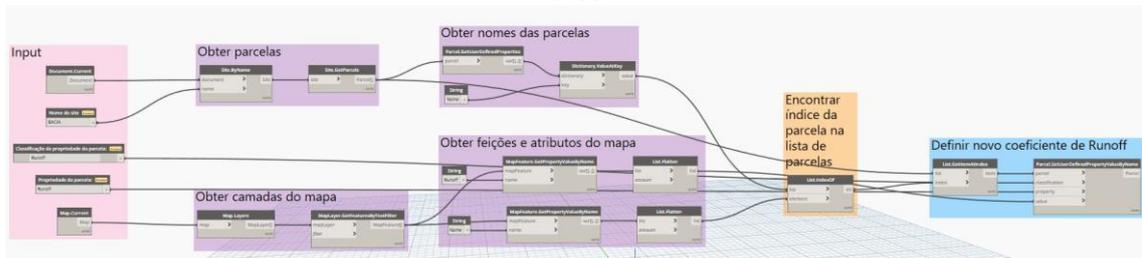
Fonte: Autor (2025).

3.2.3 Algoritmo nº 3 - Atribuição dos coeficientes de *runoff* às bacias

O Algoritmo nº 3 foi desenvolvido em linguagem *Design Script* do Dynamo para Civil 3D para atribuir os valores de *Runoff*, calculados no Modelador Gráfico do QGIS e exportados como arquivos *shapefile* por meio do Algoritmo nº 2, às bacias do tipo *Parcel* obtidas no Algoritmo nº 3.

As feições das bacias no formato Shapefile resultantes do Algoritmo nº 2 foram inicialmente carregadas no Civil 3D utilizando a ferramenta MAPCONNECT (Figura 53). Em seguida, foi executado o algoritmo de programação visual.

Figura 54 - Algoritmo nº 3 de atribuição dos valores de *Runoff* das feições SHP aos objetos do tipo *Parcel*.



Fonte: Autor (2025).

Figura 55 - Algoritmo nº 3 em linguagem em texto *Design Script*.

```

Code Block
1 document1 = Autodesk.Document.Current;
2 t1 = "BACIA";
3 site1 = Site.ByName(document1, t1);
4 parcel1 = Site.GetParcels(site1);
5 t2 = Parcel.GetUserDefinedProperties(parcel1);
6 t3 = "Name";
7 t4 = Dictionary.ValueAtKey(t2, t3);
8 map1 = Map.Current();
9 maplayer1 = Map.Layers(map1);
10 mapfeature1 = MapLayer.GetFeaturesByTextFilter(maplayer1, "");
11 t5 = "Name";
12 t6 = MapFeature.GetPropertyByName(mapfeature1, t5);
13 t7 = List.Flatten(t6, -1);
14 num1 = List.IndexOf(t4, t7);
15 t8 = List.GetItemAtIndex(parcel1, num1);
16 t9 = "Runoff";
17 t10 = "Runoff";
18 t11 = "Runoff";
19 t12 = MapFeature.GetPropertyByName(mapfeature1, t11);
20 t13 = List.Flatten(t12, -1);
21 parcel1 = Parcel.SetUserDefinedPropertyValueByName(t8, t9, t10, t13);

```

Fonte: Autor (2025).

3.2.4 Algoritmo nº 4 - Criação automatizada de tubulações

O Algoritmo nº 4 permite a modelagem automática das peças que compõem o sistema de tubulação de drenagem. A linguagem de programação utilizada foi o *DesignScript* do Dynamo para Civil 3D. Para sua execução, é necessário que o Algoritmo nº 1 tenha sido previamente aplicado, obtendo-se os polígonos das bacias e os pontos de acúmulo de água de cada uma, não sendo necessário executar os Algoritmos nº 2 e nº 3.

Antes da aplicação do Algoritmo nº 4, é imprescindível criar as polilinhas que representam os coletores troncos das ruas, onde cada vértice da polilinha indica um ponto obrigatório de inserção de uma estrutura do tipo Poço de Visita. Não é necessária a criação de polilinhas tridimensionais, visto que o algoritmo corrige as elevações pela superfície. Dessa maneira, além dos vértices criados para mudanças de direção em curvas, também foram adicionados vértices em cada esquina, de modo que uma estrutura seja inserida para receber as estruturas engolidoras de água do tipo Boca de Lobo (Figura 56).

Figura 56 - Polilinha na cor magenta indicando os locais de inserção dos objetos a serem modelados.



Fonte: Autor (2025).

Após a criação da polilinha, o Algoritmo nº 4 inicia-se com a definição do nome da rede a ser criada, a escolha da superfície de referência, a seleção das polilinhas dos troncos coletores e dos círculos dos pontos de acúmulo de água, bem como a definição das condições de criação da rede, tais como a distância máxima entre poços de visita, o cobrimento sobre as tubulações e quais tubos e estruturas das listas de peças do Civil 3D serão utilizados na criação dos objetos (Figura 57).

Figura 57 - Dados a serem inseridos como entradas para o Algoritmo nº 4.

Dynamo Player

← ↻ ?

↳ Criar rede de drenagem_v11
Inputs needed

✓ Nome da rede :
DRENAGEM

✓ Selecione a superfície de referência :
SUPERFÍCIE FINAL

↳ Selecione os círculos do posicionamentos das bocas de lobo :
Select
Nothing selected.

✓ Distância máxima entre PV's (m) :
40

✓ Escolha o índice da lista de peças de tubulação (compile uma vez para carregar a lista) :
3

✓ Escolha o índice do tubo a ser utilizado pela lista a seguir (compile uma vez para carregar a lista) :
29

✓ Escolha o índice do PV a ser utilizado pela lista a seguir (compile uma vez para carregar a lista) :
8

✓ Escolha o índice da BL a ser utilizada pela lista a seguir (compile uma vez para carregar a lista) :
2

✓ Cobertura sobre os tubos (m) :
1

↳ Selecione a polilinha do tronco coletor (cada vértice tonar-se-á um PV) :
Select
Nothing selected.

C:\Users\halan\OneDrive\Documentos\Halan\Mestrado UFRJ 2021\Disse

Fonte: Autor (2025).

O algoritmo inicia ao sobrepor as polilinhas dos troncos coletores à superfície de referência, criando polilinhas 3D, incluindo a adição de vértices caso as distâncias entre os vértices existentes excedam o valor especificado na entrada "Distância máxima entre PVs (m)". Em seguida, essas polilinhas são ajustadas em relação à superfície a uma distância correspondente ao valor informado na entrada "Cobertura sobre os tubos (m)". Após isso, poços de visita são inseridos nos vértices das polilinhas 3D. Na etapa seguinte, o algoritmo posiciona bocas de lobo nos círculos de acúmulo de água selecionados e verifica qual o poço de visita ou boca de lobo mais próximo de cada boca de lobo. Ao localizar essa estrutura (poço de visita ou boca de lobo), ele conecta as duas estruturas com um objeto do tipo tubulação. Finalmente, o algoritmo ajusta as elevações das tubulações para que sigam a direção montante-jusante.

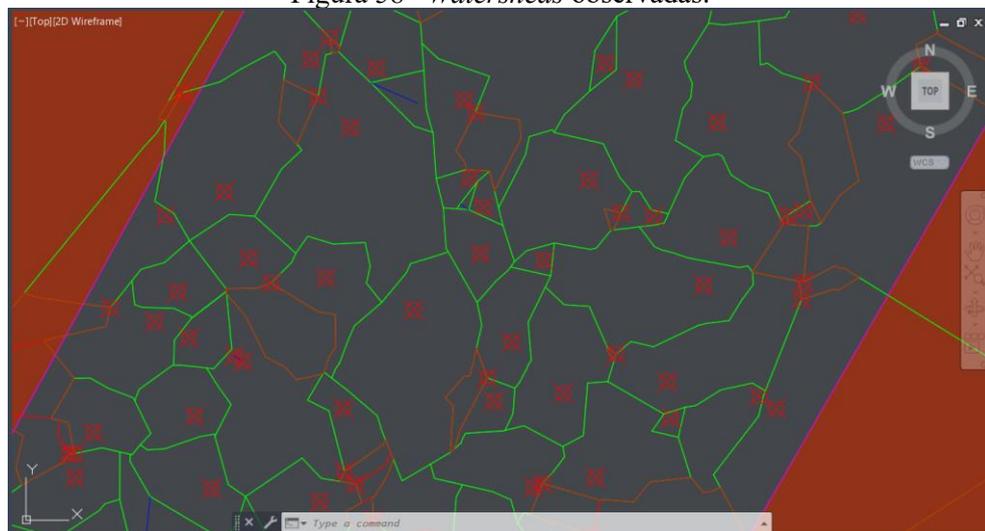
Capítulo 4

4 APLICAÇÕES PRÁTICAS

4.1 ESTUDO DE CASO 1

Neste estudo de caso, os algoritmos foram aplicados em um projeto de drenagem pluvial urbana real, localizado em um quartel do Exército Brasileiro na cidade de Recife, Pernambuco, com um MDT resultante com cerca de 155.000 pontos em uma área de 7 hectares (Figura 58).

Figura 58 - *Watersheds* observadas.

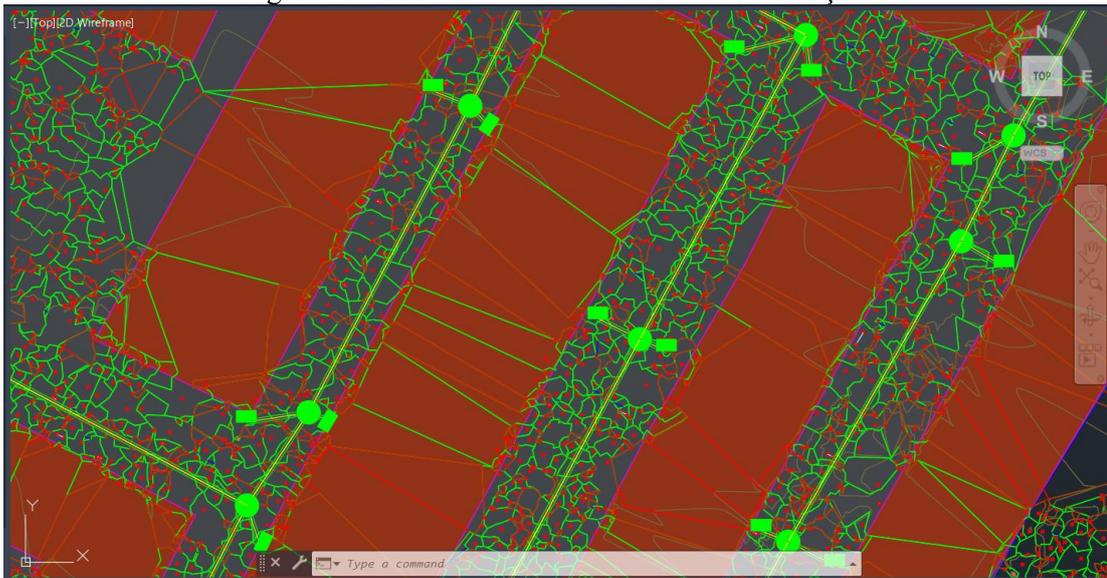


Fonte: Autor (2025).

O algoritmo nº 2 não enfrentou grandes problemas para calcular o parâmetro *Runoff* de cada uma das bacias e nem o algoritmo nº 3 para atribuir os valores calculados a cada *Parcel*.

Por fim, para o algoritmo nº 4, quando foi necessário selecionar os círculos marcados no projeto pelo algoritmo nº 1 (Figura 58) para definir os locais de criação das estruturas coletoras de água, foi preciso tomar cuidado para não marcar todos os círculos e criar uma quantidade desnecessária de bocas de lobo, uma vez que esse tipo de estrutura pode ser concebida para engolir a água escoada superficialmente de um conjunto de bacias, como demonstrado na Figura 59.

Figura 59 - Estruturas coletoras da rede de tubulação.



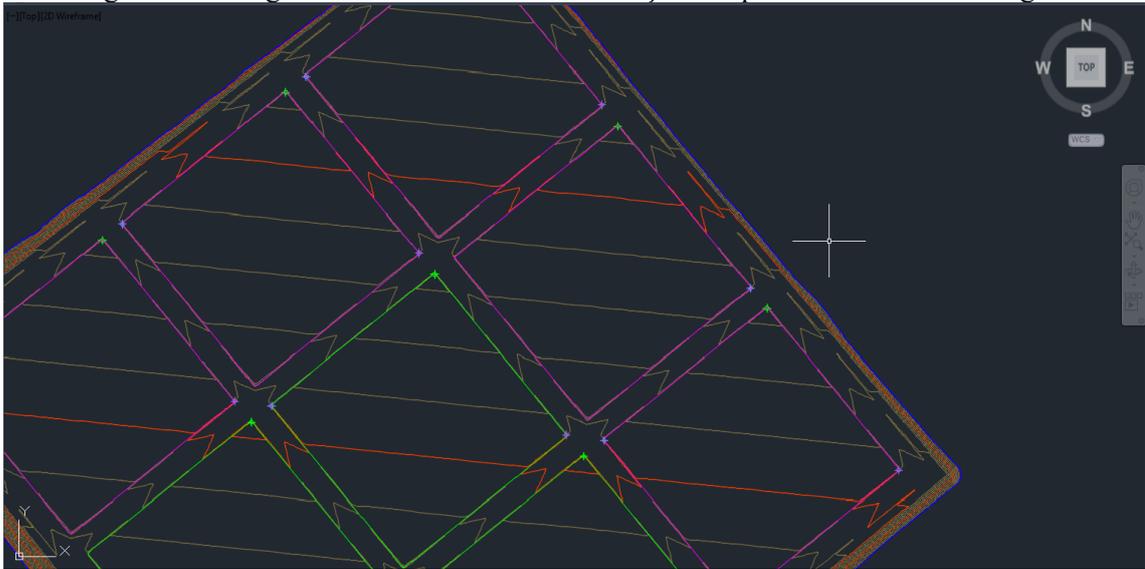
Fonte: Autor (2025).

4.2 ESTUDO DE CASO 2

Neste estudo de caso, os algoritmos foram aplicados em um projeto hipotético, utilizando um MDT modelado a partir de corredores de vias em loteamento de um condomínio urbano. Pelo fato de ter sido modelado a partir de corredores, a quantidade de pontos é reduzida a apenas pontos necessários, como o eixo, bordo direito e bordo esquerdo da via.

Dessa forma, as bacias resultantes do algoritmo nº 1 acabam sendo apenas os lotes e cada um dos lados das vias em cada quarteirão, e os pontos de acúmulo de água as esquinas (Figura 60).

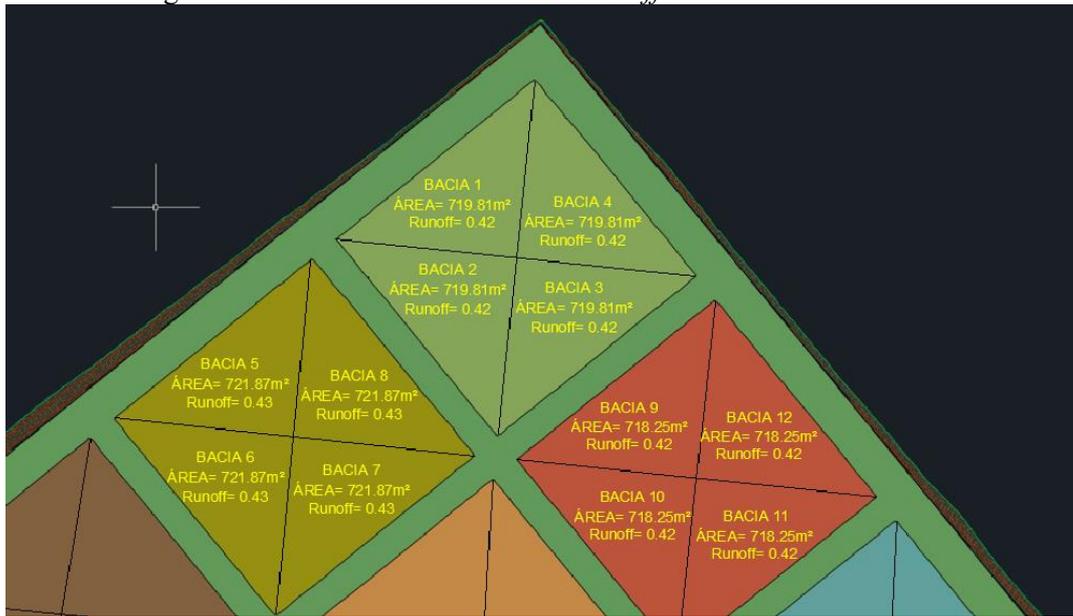
Figura 60 - Polígonos 3D das bacias com indicação dos pontos de acúmulo de água.



Fonte: Autor (2025).

O algoritmo nº 2, como observado no Caso 1, não tem desempenho diferente pela quantidade de bacias (Figura 61), e rapidamente calculou os valores de *Runoff* de cada *Parcel* importada do arquivo SDF (Figura 62).

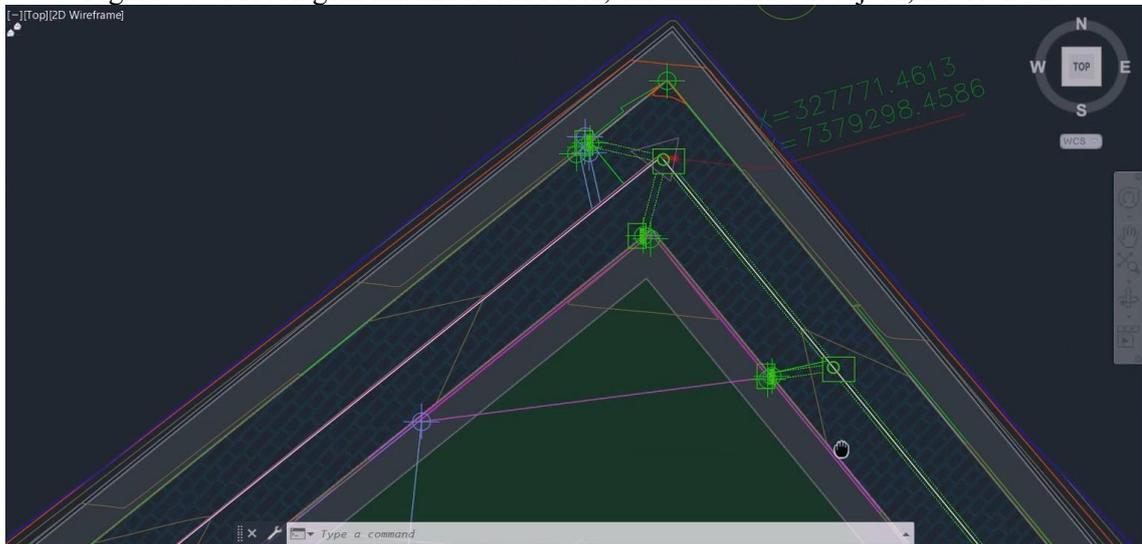
Figura 63 - Bacias com os valores de *Runoff* calculados e atribuídos.



Fonte: Autor (2025).

Por fim, no algoritmo nº 4, como a quantidade de pontos de acúmulo de água foi bem reduzido em relação ao Caso 1, apenas poucos círculos foram descartados, já que as indicações de posicionamento de bocas de lobo basicamente encontravam-se nas esquinas das vias (Figura 64).

Figura 64 - Modelagem finalizada das redes, em linha verde tracejada, no Civil 3D.



Fonte: Autor (2025).

Capítulo 5

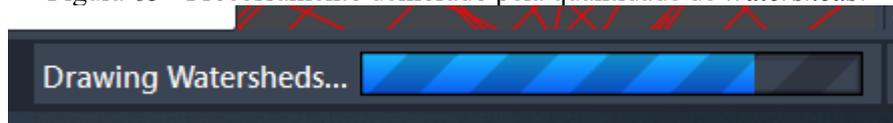
5 AVALIAÇÃO DOS RESULTADOS

O levantamento topográfico com estação total, GPS e aerofotogrametria de drone realizado no Estudo de caso 1 resultou em uma nuvem com mais de 1 milhão de pontos. Dessa forma, de modo a evitar a modelagem excessiva, comentada no tópico 2.4, foi necessária uma filtragem prévia de cerca de 90% dos pontos para a geração do MDT.

Ainda assim, como explicado no tópico 3.2.1, apesar da filtragem prévia dos pontos do levantamento para o MDT, a alta densidade resultou, após a aplicação do algoritmo nº 1, em uma grande quantidade de bacias do tipo *Watershed*, como apresentado na Figura 58 e pontuado pela UNECE no tópico 2.4.

Outro problema da quantidade excessiva de pontos foi que, a cada vez que a escala do projeto precisou ser ajustada, o processamento das bacias demorava muito para ser concluído (Figura 65).

Figura 65 - Processamento demorado pela quantidade de *Watersheds*.



Fonte: Autor (2025).

O algoritmo nº 2, uma vez que recebia como entrada o arquivo SHP exportado do Civil 3D, apresentou bom desempenho em ambos os casos, não apresentando problemas.

Os algoritmos nº 2 e nº 3 haviam sido inicialmente experimentados utilizando apenas o Civil 3D e o Dynamo como ferramentas para o recorte das formas geométricas do arquivo criado no AutoCAD. No entanto, o algoritmo único desenvolvido era extremamente confuso e os resultados obtidos não eram satisfatórios. Percebeu-se que a utilização dos polígonos tridimensionais das bacias, combinados com formas bidimensionais em um software BIM, não era a solução mais adequada para o problema.

Posteriormente, notou-se também que não havia a necessidade de trabalhar com dados tridimensionais para obter o coeficiente de *runoff*, uma vez que esse coeficiente é adimensional e função apenas de áreas e coeficientes predeterminados, como apresentado na Equação 1:

Equação 1

$$c_{runoff} = \frac{\sum_i A_i c_i}{\sum_i A_i}, \text{ onde:}$$

- c_{runoff} é o *runoff* ponderado de uma determinada bacia;
- A_i é a área de cada tipo de superfície nessa bacia; e
- c_i é o coeficiente de *runoff* para cada tipo de superfície.

Uma verificação importante acerca do algoritmo nº 4 é que ele foi capaz de, após criar as estruturas coletoras e de passagem de água, conectar cada tubo às estruturas à montante e à jusante correspondentes. Pode parecer um passo pouco perceptível, mas para o funcionamento das ferramentas de dimensionamento de tubulações do Civil 3D, não basta apenas modelar as tubulações e as estruturas com suas inclinações e cotas compatibilizadas, é necessário também fazer a conexão dessas peças entre si, inclusive no sentido montante-jusante correto.

A tabela da Figura 66, extraída do Civil 3D com a indicação das estruturas à montante e à jusante de cada tubo, só tem as duas últimas colunas preenchidas automaticamente pelo software caso esse passo seja executado corretamente.

Figura 66 - Tabela de tubos do Civil 3D com indicação de estruturas à montante e à jusante de cada tubo.

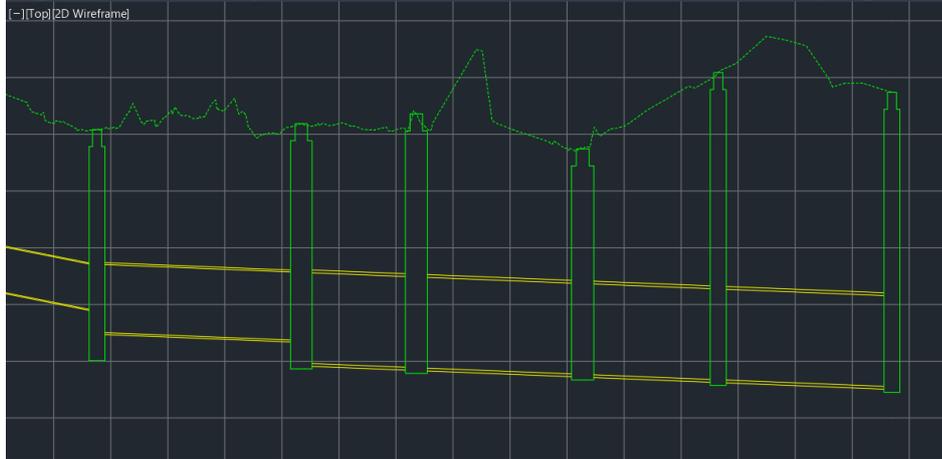
TABELA DE TUBOS DA REDE DRENAGEM								
TUBO	DN (mm)	DESCRIÇÃO	COMPRIMENTO (m)	DECLIVIDADE (%)	CS	CC	NÓ A MONTANTE	NÓ A JUSANTE
EW-01.1	600	TUBO DN 600 mm PEAD	28.43	0.40%	5.477	5.357	PV EW-01.1	PV EW-01.2
EW-01.1-T1.1	400	TUBO DN 400 mm PEAD	1.44	0.40%	5.488	5.477	BLS EW-01.1-T1.1	PV EW-01.1
EW-01.1-T2.1	400	TUBO DN 400 mm PEAD	0.93	0.40%	5.486	5.477	BLS EW-01.1-T2.1	PV EW-01.1
EW-01.2	600	TUBO DN 600 mm PEAD	8.14	0.40%	5.357	5.320	PV EW-01.2	PV SN-01.11
EW-01.2-T1.1	400	TUBO DN 400 mm PEAD	2.68	5.05%	6.062	5.867	BLS EW-01.2-T1.1	PV EW-01.2
EW-01.2-T2.1	400	TUBO DN 400 mm PEAD	4.66	0.40%	6.064	6.041	BLS EW-01.2-T2.1	PV EW-01.2
SN-01.1	400	TUBO DN 400 mm PEAD	13.63	0.40%	6.232	6.173	PV SN-01.2	PV SN-01.1

Fonte: Autor (2025).

Por fim, outra verificação importante sobre o algoritmo nº 4 é que ele também foi eficaz na etapa de compatibilização de cotas de topo de cada estrutura à cota da superfície equivalente ao centro de cada uma delas; e o alinhamento das tubulações pela geratriz superior, desde que essa regra tenha sido previamente configurada. A observação dessa compatibilização pôde ser observada após a definição do alinhamento, do perfil, e da vista do perfil do trecho analisado,

conforme indicado na Figura 67. A tarefa também pode parecer mero detalhe, mas é essencial para o correto cálculo dos volumes de escavação, por exemplo.

Figura 67 - Vista de perfil horizontal de um trecho da rede de tubulação gerada.



Fonte: Autor (2025).

Como explicado no tópico 3, as bacias trabalhadas ao longo do trabalho foram do tipo *Watersheds* e não *Catchments*. O motivo se deu pelo fato de o Civil 3D ainda não permitir a automação da criação de bacias do tipo *Catchment*, ou seja, ainda é necessário criar bacia por bacia recursivamente de forma manual, caso opte-se por criá-las usando o comando *Create Catchment from Surface*, que é comando mais preciso para definição de bacias a partir do local de acúmulo de água.

Caso fosse possível criar *Catchments* automaticamente, depois exportá-las em SDF e SHP, posteriormente calcular e atribuir os valores de *runoff* novamente aos objetos no Civil 3D, seria possível também automatizar o nó 5, de carregamento das bacias parametrizadas nas ferramentas nativas de dimensionamento das tubulações do software da Autodesk, apresentadas na Figura 36.

Capítulo 6

6 CONCLUSÕES

A automação com as ferramentas BIM do Dynamo para Civil 3D e com o Modelador Gráfico do QGIS na definição de bacias e no cálculo do coeficiente de escoamento superficial delas em superfícies definidas por MDT, no universo limitado deste estudo, provou ser eficiente. A tarefa de determinar essas entidades e parâmetros, ao contrário do apresentado na introdução do trabalho, deixou de ser realizada de maneira manual e intuitiva, passando a ser apenas supervisionada pelo autor.

Não foi possível determinar o grau de eficiência dessa automação e nem o aumento da produtividade dos projetos, porém, a partir do que foi observado nos estudos de casos, e pela lógica de que quanto maior a superfície trabalhada, maior a quantidade de bacias a serem estudadas, pode-se dizer que o emprego dos algoritmos certamente pode reduzir o tempo gasto por projetistas de redes de drenagem pluvial urbanas nas quatro etapas aqui encaradas como nós a serem desatados.

O método GIM empregado neste trabalho, trabalhando em nível igual de coordenação com o BIM, foi diferente do identificado na literatura, em que os autores justificaram a necessidade do uso de aplicativos GIS em complemento aos BIM quando os projetos tratassem de infraestruturas lineares extensas ou situados em ambientes urbanos de alta complexidade e com número grande de edificações. Este autor empregou o QGIS por provar-se uma ferramenta mais eficiente para lidar com dados bidimensionais, como as bacias tipo *Watershed*, e adimensionais, como o *runoff*. A tentativa inicial de desatar o nó 3 utilizando apenas o Civil 3D e o Dynamo, mostrou-se muito confusa e demandava maior capacidade de processamento do computador.

O trabalho observou que a definição e compatibilização de LODs, LOds e LOIs não foram obstáculos em momento nenhum, o que evitou gastar tempo com correções na organização de informações e com a compatibilização de detalhes geométricos. Com efeito, observou-se que, apesar da exportação de informações de uma modelagem inicial BIM para um formato de dados geoespaciais, como o SDF, e depois a importação por meio do formato análogo SHP, a

interoperabilidade entre os domínios provou ser eficiente, uma vez que os parâmetros e seus valores foram corretamente correlacionados.

Apesar da nova definição de LOIN da norma NBR ISO 19650-1:2022 mostrar ter potencial para satisfazer às necessidades particulares do gerenciamento de projetos de infraestrutura, em que o conteúdo alfanumérico desempenha papel de liderança na especificação, criação, entrega e validação das informações, o LOD ainda é usado com mais frequência na prática. No entanto, espera-se que isso mude à medida que a norma torne-se mais popular e o conceito mais esmiuçado em novas normas congêneres.

Outra diferença importante entre este trabalho e a literatura acerca da integração BIM-GIM é que os autores apontam o formato IFC como o mais adequado para o fluxo de dados da modelagem BIM e o formato *CityGML* para a modelagem GIM. No entanto, e como observado nesta conclusão de que não necessariamente grandes quantidades de edificações ou infraestruturas muito extensas serão comutadas entre os dois domínios, observou-se que nem a exportação da modelagem inicial do Civil 3D no formato IFC, nem tampouco no *CityGML*, seriam úteis para o objetivo da manipulação de dados das bacias no QGIS. O uso dos formatos SDF e SHP, apesar de siglas pouco mencionadas no quesito interoperabilidade, foi suficiente quando se precisou trabalhar com dados apenas de superfícies de MDT.

A implementação dos algoritmos por meio das VPLs do Dynamo e do Modelador Gráfico do QGIS é outro ponto a ser ressaltado no que diz respeito à acessibilidade da automação de tarefas em ambientes BIM e GIM, respectivamente. Como as VPLs utilizam o esquema de “caixas e setas”, não houve a necessidade de o autor utilizar uma linguagem de programação específica, bastando apenas compreender os conceitos de lógica de programação e o encadeamento correto das tarefas a serem executadas em sequência.

Como já abordado no tópico 2.9.1, o desafio da Transformação do *Swept Solid/CSG* em B-Rep para o formato IFC não se aplicou a este trabalho, uma vez que foram manipulados apenas dados adimensionais e bidimensionais entre BIM e GIM. No entanto, os outros três desafios foram encarados no percurso até estas conclusões.

A transformação do sistema de referenciamento local em sistema de coordenadas geográficas já deve ser sempre o primeiro passo quando um novo projeto de infraestrutura é iniciado no Civil 3D, e esse sistema não se altera quando os dados são exportados para o QGIS, portanto, o primeiro desafio foi facilmente superado na etapa inicial do trabalho.

Quanto à compatibilização de detalhes e perda de dados, considerando-se que os dados das bacias comutados não prezavam pela exibição gráfica, e assim não foram incompatíveis e nem

causaram perda de dados durante o processo de integração, apesar de transitarem em três formatos distintos.

Como mencionado na Avaliação dos Resultados, o fato de o Civil 3D não possibilitar a exportação de bacias *Catchment* no arquivo SDF e impossibilitar a automação da criação dessa entidade por meio do Dynamo, impediu, dentro dos objetivos e da metodologia deste estudo, que o nó 5, de carregamento das bacias parametrizadas nas ferramentas de dimensionamento de tubulações do software. O fato, sob a análise do autor, é um exemplo da desvantagem de se trabalhar com o formato proprietário fechado da Autodesk. Em um provável *software* de código aberto, em que fosse possível acessar e manipular cada classe de objetos, teria sido possível modificar o padrão de exportação para um arquivo geoespacial, permitindo a saída dos dados das *Catchments* como feições, e posteriormente permitindo o uso de uma ferramenta análoga ao *Analyze Gravity Network* para o dimensionamento dos tubos juntamente com as saídas do algoritmo nº 3.

A forma como esta dissertação integrou as siglas BIM e GIM, ou BIM e GIS, limitou-se a um espaço micro, se comparada às aplicações em ambientes urbanos mais complexos, todavia, ainda assim os produtos desenvolvidos podem potencializar as capacidades dessa integração de prevenir desastres de inundações e mitigar efeitos das ameaças, uma vez que uma ferramenta que acelera e otimiza a produção de projetos de microdrenagem pluvial urbana gera como consequência imediata uma maior quantidade, em um menor intervalo de tempo, de soluções para o problema do acúmulo e da destinação correta das águas escoadas superficialmente.

Por fim, como sugestão para próximos trabalhos, o autor propõe que sejam utilizadas as nuvens de pontos dos levantamentos de drone, combinadas com ferramentas de Inteligência Artificial (IA) de classificação de cores, para a classificação automática dos tipos de superfície. Por exemplo: utilizar uma IA para gerar um arquivo de formato geoespacial que, a partir da leitura das cores dos pontos da nuvem, divida e classifique as feições das partições do terreno em grama, telhados ou asfalto. Outra sugestão é que os temas da interoperabilidade e da automação de projetos de infraestrutura por meio de VPLs sejam mais explorados, uma vez que diversos autores mencionaram que o nível de maturidade de projetos BIM de infraestrutura está abaixo do nível para edificações.

Referências Bibliográficas

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS – ABNT. **NBR ISO 19650-1:2022: Organização da informação acerca de trabalhos da construção - Gestão da informação usando a modelagem da informação da construção - Parte 1: Conceitos e princípios.** ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. Rio de Janeiro. 2022.

AISH, R. **DesignScript User Manual**, 2016. Disponível em: https://www.researchgate.net/publication/320346998_DesignScript_User_Manual. Acesso em: 2 dez. 2024.

ARRIGHI, C.; PREGNOLATO, M.; CASTELLI, F. Indirect flood impacts and cascade risk across interdependent linear infrastructures. **Natural Hazards and Earth System Sciences**, n. 21, junho 2021. p. 1955-1969.

AUTODESK. To Export To SDF. **AutoCAD Map 3D 2024**, 2024. Disponível em: <https://help.autodesk.com/view/MAP/2024/ENU/?guid=GUID-D9C51D50-86B8-4662-BD3C-46AB610AA6AF>. Acesso em: 2 dez. 2024.

AUTODESK. Types of Watersheds. **Autodesk Civil 3D 2025**, 2024. Disponível em: <https://help.autodesk.com/view/CIV3D/2025/ENU/?guid=GUID-E021850D-0749-42CC-A49E-6C8BB6CC1DC8>. Acesso em: 2 dez. 2024.

BARBOSA, E. **IV-275 - BIM e saneamento - Soluções técnicas para a concepção de estudos e projetos de sistemas lineares de esgotamento sanitário e abastecimento de água.** Congresso ABES FENASAN 2017. São Paulo, Brasil: [s.n.]. 2017. p. 1-12.

BASTA, A.; SERRA, M. H.; MARZOUK, M. A BIM-based framework for quantitative assessment of steel structure deconstructability. **Automation in Construction**, n. 111, 2020. p. 1-12.

BERTILSSON, L.; WIKLUND, K.; TEBALDI, I. D. M. et al. Urban flood resilience – A multi-criteria index to integrate flood resilience into urban planning. **Journal of Hydrology**, n. 573, 2019. p. 970-982.

BILJECKI, F.; LEDOUX, H.; STOTER, J. An improved LOD specification for 3D building models. **Computers, Environment and Urban Systems**, n. 59, 2016. p. 25-37.

BRADLEY, A.; LI, H.; LARK, R. et al. BIM for infrastructure: An overall review and constructor perspective. **Automation in Construction**, n. 71, 2016. p. 139-152.

BRASIL. Tribunal de Contas da União – TCU. **Acordão nº 1079/2019 – Plenário**, 2019. Disponível em: <https://portal.tcu.gov.br/biblioteca-digital/auditoria-operacional-sobre-obras-paralisadas.htm>. Acesso em: 2 dez. 2024.

BRASIL. Decreto nº 10.569, de 9 de dezembro de 2020. Aprova a estratégia nacional de segurança de infraestruturas críticas, Brasília, DF, 2020. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2019-2022/2020/decreto/D10569.htm. Acesso em: 2 dez. 2024.

BRASIL. Departamento Nacional de Infraestrutura de Transportes – DNIT. **Caderno de Requisitos Técnicos BIM do DNIT**, 2022. Disponível em: https://www.gov.br/dnit/pt-br/assuntos/planejamento-e-pesquisa/bim-no-dnit/mosaico-de-servicos/documentos-tecnicos-bim/CRTBIM_V012022.pdf. Acesso em: 2 dez. 2024.

BRASIL. Tribunal de Contas da União – TCU. **Painel de Obras Paralisadas**, 2025. Disponível em: <https://paineis.tcu.gov.br/pub/?workspaceId=8bfbd0cc-f2cd-4e1c-8cde-6abfdffa6a8&reportId=013930b6-b989-41c3-bf00-085dc65109de&filterPaneEnabled=false&navContentPaneEnabled=false>. Acesso em: 2 dez. 2024.

BUILDINGSMART INTERNATIONAL – BSI. 5.4.3.1 IfcAlignment. **buildingSMART International**, 2024. Disponível em: <https://ifc43-docs.standards.buildingsmart.org/IFC/RELEASE/IFC4x3/HTML/lexical/IfcAlignment.htm>. Acesso em: 2 dez. 2024.

BUILDINGSMART INTERNATIONAL – BSI. 5.4.3.63 IfcSite. **buildingSMART International**, 2024. Disponível em: <https://ifc43-docs.standards.buildingsmart.org/IFC/RELEASE/IFC4x3/HTML/lexical/IfcSite.htm>. Acesso em: 2 dez. 2024.

BUILDINGSMART INTERNATIONAL – BSI. Industry Foundation Classes (IFC) – An Introduction. **buildingSMART International**, 2024. Disponível em: <https://technical.buildingsmart.org/standards/ifc/>. Acesso em: 2 dez. 2024.

CAMPOS, G.; IYOMASA, W.; SANTOS, A. et al. O “invisível” espaço subterrâneo urbano. **São Paulo em Perspectiva**, v. 20, n. 2, abr./jun. 2006. p. 147 - 157.

CAO, Y.; XU, C.; MARDHIYAH AZIZ, N. et al. BIM–GIS Integrated Utilization in Urban Disaster Management: The Contributions, Challenges, and Future Directions. **Remote Sensing**, n. 15, 2023.

COLLAO, J.; LOZANO-GALANT, F.; LOZANO-GALANT, J. et al. BIM Visual Programming Tools Applications in Infrastructure Projects: A State-of-the-Art Review. **Applied Sciences**, n. 11, 2021. p. 1-24.

COSTIN, A.; ADIBFAR, A.; HU, H. et al. Building Information Modeling (BIM) for transportation infrastructure – Literature review, applications, challenges, and recommendations. **Automation in Construction**, n. 94, 2018. p. 257-281.

DENG, Y.; CHENG, J.; ANUMBA, C. Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison. **Automation in Construction**, n. 67, 2016.

DENG, Y.; CHENG, J.; ANUMBA, C. Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison. **Automation in Construction**, n. 67, 2016.

EASTMAN, C.; TEICHOLZ, P.; SACKS, R. et al. **Manual de BIM: Um guia de modelagem da informação da construção**. Porto Alegre: Bookman, 2014.

ELCO, K.; ROZENBERG, J.; ZORN, C. et al. A global multi-hazard risk analysis of road and railway infrastructure assets. **Nature Communications**, v. 10, n. 2677, 2019.

EM-DAT. EM-DAT Public. **The International Disaster Database**, 2024. Disponível em: <https://public.emdat.be/data>. Acesso em: 2 dez. 2024.

ENGESAT. Conversão de MDS para MDT: A solução definitiva. **Soluções em imagens de satélite e geoprocessamento**, 2024. Disponível em: <https://www.engesat.com.br/conversao-de-mds-para-mdt-a-solucao-definitiva/>. Acesso em: 2 dez. 2024.

ESRI. ESRI Shapefile Technical Description. **Environmental Systems Research Institute – ESRI**, 1998. Disponível em: <https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf>. Acesso em: 2 dez. 2024.

ESRI. Authoring 3D Layers and Creating WebScenes in ArcGIS Online. **Environmental Systems Research Institute – ESRI**, 2019. Disponível em: <https://www.esri.com/arcgis-blog/products/arcgis-online/3d-gis/authoring-3d-layers-and-creating-web-scenes-in-arcgis-online/>. Acesso em: 2 dez. 2024.

ESRI. What is a TIN surface? **Environmental Systems Research Institute – ESRI**, 2024. Disponível em: <https://desktop.arcgis.com/en/arcmap/latest/manage-data/tin/fundamentals-of-tin-surfaces.htm>. Acesso em: 2 dez. 2024.

ESTADOS UNIDOS DA AMÉRICA – EUA. A guide to critical infrastructure security and resilience, 2019. Disponível em: <https://cncpic.mai.gov.ro/sites/default/files/2020-03/Guide-Critical-Infrastructure-Security-Resilience-110819-508v2.pdf>. Acesso em: 02 dez. 2024.

GHANNAD, P.; LEE, Y.-C.; DIMYADI, J. et al. Automated BIM data validation integrating open-standard schema with visual programming language. **Advanced Engineering Informatics**, v. 40, 2019. p. 14-28.

GRÖGER, G.; PLÜMER, L. CityGML – Interoperable semantic 3D city models. **Journal of Photogrammetry, Remote Sensing and Geoinformation Science**, n. 71, 2012.

HAN, C.; TANG, F.; MA, T. et al. Construction quality evaluation of asphalt pavement based on BIM and GIS. **Automation in Construction**, n. 141, 2022. p 1-14.

HEAD, B. Managing urban water crises: adaptive policy responses to drought and flood in Southeast Queensland, Australia. **Ecology and Society**, v. 19, n. 2, 2014.

HERLE, S.; BECKER, R.; WOLLENBERG, R. et al. GIM and BIM. Integration of BIM and GIS: Geometry from IFC to shapefile using open-source technology. **Journal of Photogrammetry, Remote Sensing and Geoinformation Science**, n. 88, 2020.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION – ISO. **ISO 16739-1:2024 - Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries — Part 1: Data schema**. International Organization for Standardization. Genebra, Suíça. 2018.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION – ISO. **ISO/TS 19166 - Geographic information - BIM to GIS conceptual mapping (B2GM)**. International Organization for Standardization. Genebra, Suíça. 2021.

JOHNSTON, W.; HANNA, J. R. P.; MILLAR, R. Advances in Dataflow Programming Languages. **ACM Computing Surveys**, v. 36, n. 1, 2004.

JUSTO, A.; SOILÁN, M.; SÁNCHEZ-RODRÍGUEZ, A. et al. Scan-to-BIM for the infrastructure domain: Generation of IFC-compliant models of road infrastructure assets and semantics using 3D point cloud data. **Automation in Construction**, n. 127, 2021.

KANG, T. W.; HONG, C. H. A study on software architecture for effective BIM/GIS-based facility management data integration. **Automation in Construction**, n. 54, 2015.

KELLY, R. IFC 4.3 – A Major Milestone Introduction and Overview of the Current Status. How we got there. **buildingSMART International**, 2022. Disponível em: <https://www.buildingsmart.org/ifc-4-3-a-major-milestone-introduction-and-overview-of-the-current-status/>. Acesso em: 2 dez. 2024.

LIU, W.; ZHAOYANG, S. Review of studies on the resilience of urban critical infrastructure networks. **Reliability Engineering and System Safety**, n. 193, ago. 2020.

LIU, X.; WANG, X.; WRIGHT, G. et al. A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS). **International Journal of Geo-Information**, v. 6, n. 53, 2017.

LUA, J.; MILFORD, A. Dynamo in Civil 3D Introduction - Unlocking the Mystery of Scripting. **Autodesk University**, 2019. Disponível em: <https://www.autodesk.com/autodesk->

university/class/Dynamo-Civil-3D-Introduction-Unlocking-Mystery-Scripting-2019. Acesso em: 2 dez. 2024.

MATTSSON, L.-G.; JENELIUS, E. Vulnerability and resilience of transport systems – A discussion of recent research. **Transportation Research Part A**, v. 81, 2015. p. 16-34.

MICELI JUNIOR, G. **Modelagem da Informação da Construção para gestão de projetos de obras de infraestruturas de Defesa**. Doutorado em Engenharia de Defesa. Instituto Militar de Engenharia - IME. Rio de Janeiro, p. 263. 2019.

NIELSEN, O.; MICELI JR, G.; FERREIRA FILHO, A. et al. A Review of Global Efforts in BIM Adoption for Road Infrastructure. **Infrastructures**, v. 9, n. 126, 2024.

PARANÁ. Secretaria de Estado de Infraestrutura e Logística. **Caderno BIM - Infraestrutura Rodoviária**, 2022. Disponível em: https://www.bim.pr.gov.br/sites/bim/arquivos_restritos/files/documento/2022-06/Caderno%20BIM%20-%20Infraestrutura%20Rodovi%C3%A1ria_VersaoPortal%20030622%20pags_0.pdf. Acesso em: 2 dez. 2024.

PARQUE BONDINHO DO PÃO DE AÇÚCAR. O que é tirolesa? Tudo sobre essa aventura nos ares. **Portal da tirolesa do Parque Bondinho do Pão de Açúcar**, 2023. Disponível em: <https://tirolesa.bondinho.com.br/blog/o-que-e-tirolesa/>. Acesso em: 2 dez. 2024.

POLONENKO, L.; HAMOUDA, M.; MOHAMED, M. Essential components of institutional and social indicators in assessing the sustainability and resilience of urban water systems: Challenges and opportunities. **Science of the Total Environment**, 708, mar. 2020.

ROCHA, P. **A relação entre a gestão de riscos integrada em uma organização com infraestrutura crítica e as questões de Defesa Nacional**. Trabalho de Conclusão de Curso (Curso de Altos Estudos em Defesa). Escola Superior de Guerra. Brasília, DF. 2019.

SERRA, P.; HAGE, S. Computational Design for Civil Engineers. **Autodesk University**, 2019. Disponível em: <https://www.autodesk.com/autodesk-university/class/Computational-Design-Civil-Engineers-2019>. Acesso em: 2 dez. 2024.

SERRA, P.; HAGE, S. Civil 3D Toolkit for Dynamo for Civil 3D - 1.1.32. **Pacote para Dynamo para Civil 3D disponível na biblioteca da Autodesk**, 2022.

SOETHE, P. Desmistificando o BIM com foco em Infraestrutura. **Mundo AEC - Blog Oficial sobre soluções da Autodesk Brasil**, 2017. Disponível em: <https://blogs.autodesk.com/mundoaec/desmistificando-o-bim-com-foco-em-infraestrutura/>. Acesso em: 2 dez. 2024.

UNITED NATIONS ECONOMIC COMMISSION FOR EUROPE – UNECE. **Building Information Modelling (BIM) for road infrastructure: TEM requirements and**

recommendations. United Nations Economic Commission for Europe – UNECE. Organização das Nações Unidas – ONU. Genebra, Suíça. 2021.

WANG, H.; PAN, Y.; LUO, X. Integration of BIM and GIS in sustainable built environment: A review and bibliometric analysis. **Automation in Construction**, n. 103, 2019. p. 141-52.

YANG, Y.; NG, T.; DAO, J. et al. BIM-GIS-DCEs enabled vulnerability assessment of interdependent infrastructures – A case of stormwater drainage-building-road transport Nexus in urban flooding. **Automation in Construction**, n. 125, 2021. p. 1-21.

Apêndice A

A seguir foi inserido o código do algoritmo nº 2, de obtenção dos coeficientes de escoamento superficial das bacias, em linguagem Python:

```

"""
Model exported as python.
Name : Extrair runoff
Group :
With QGIS : 32601
"""

from qgis.core import QgsProcessing
from qgis.core import QgsProcessingAlgorithm
from qgis.core import QgsProcessingMultiStepFeedback
from qgis.core import QgsProcessingParameterVectorLayer
from qgis.core import QgsProcessingParameterFeatureSink
import processing

class ExtrairRunoff(QgsProcessingAlgorithm):

    def initAlgorithm(self, config=None):
        self.addParameter(QgsProcessingParameterVectorLayer('bacia', 'BACIA:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterVectorLayer('camada0', 'CAMADA0:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterVectorLayer('camada1', 'CAMADA1:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterVectorLayer('camada2', 'CAMADA2:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterVectorLayer('camada3', 'CAMADA3:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterVectorLayer('camada4', 'CAMADA4:',
optional=True, types=[QgsProcessing.TypeVectorPolygon], defaultValue=None))
        self.addParameter(QgsProcessingParameterFeatureSink('NomeAssociado', 'Nome
associado', type=QgsProcessing.TypeVectorAnyGeometry, createByDefault=True,
supportsAppend=True, defaultValue='TEMPORARY_OUTPUT'))

    def processAlgorithm(self, parameters, context, model_feedback):
        # Use a multi-step feedback, so that individual child algorithm progress reports are
adjusted for the
        # overall progress through the model

```

```

feedback = QgsProcessingMultiStepFeedback(14, model_feedback)
results = {}
outputs = {}

# Diferença_34
alg_params = {
    'INPUT': parameters['camada3'],
    'OVERLAY': parameters['camada4'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['Diferena_34'] = processing.run('native:difference', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(1)
if feedback.isCanceled():
    return {}

# Diferença_12
alg_params = {
    'INPUT': parameters['camada1'],
    'OVERLAY': parameters['camada2'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['Diferena_12'] = processing.run('native:difference', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(2)
if feedback.isCanceled():
    return {}

# Diferença_01
alg_params = {
    'INPUT': parameters['camada0'],
    'OVERLAY': parameters['camada1'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['Diferena_01'] = processing.run('native:difference', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(3)
if feedback.isCanceled():
    return {}

# Diferença_23
alg_params = {
    'INPUT': parameters['camada2'],
    'OVERLAY': parameters['camada3'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}

```

```

    outputs['Diferena_23'] = processing.run('native:difference', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

    feedback.setCurrentStep(4)
    if feedback.isCanceled():
        return {}

    # Mesclar camadas vetoriais
    alg_params = {
        'CRS': None,
        'LAYERS':
[outputs['Diferena_12']['OUTPUT'],outputs['Diferena_01']['OUTPUT'],parameters['camada4']
,outputs['Diferena_34']['OUTPUT'],outputs['Diferena_23']['OUTPUT']],
        'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
    }
    outputs['MesclarCamadasVetoriais'] = processing.run('native:mergevectorlayers',
alg_params, context=context, feedback=feedback, is_child_algorithm=True)

    feedback.setCurrentStep(5)
    if feedback.isCanceled():
        return {}

    # Recortar
    alg_params = {
        'INPUT': outputs['MesclarCamadasVetoriais']['OUTPUT'],
        'OVERLAY': parameters['bacia'],
        'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
    }
    outputs['Recortar'] = processing.run('native:clip', alg_params, context=context,
feedback=feedback, is_child_algorithm=True)

    feedback.setCurrentStep(6)
    if feedback.isCanceled():
        return {}

    # Adicionar campo à tabela de atributos
    alg_params = {
        'FIELD_LENGTH': 10,
        'FIELD_NAME': 'Área_m2',
        'FIELD_PRECISION': 2,
        'FIELD_TYPE': 1, # Decimal (double)
        'INPUT': outputs['Recortar']['OUTPUT'],
        'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
    }
    outputs['AdicionarCampoTabelaDeAtributos'] =
processing.run('native:addfieldtoattributetable', alg_params, context=context,
feedback=feedback, is_child_algorithm=True)

    feedback.setCurrentStep(7)
    if feedback.isCanceled():

```

```

return {}

# Calculadora de campo
alg_params = {
    'FIELD_LENGTH': 0,
    'FIELD_NAME': 'Área_m2',
    'FIELD_PRECISION': 0,
    'FIELD_TYPE': 0, # Decimal (double)
    'FORMULA': '$area ',
    'INPUT': outputs['AdicionarCampoTabelaDeAtributos']['OUTPUT'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['CalculadoraDeCampo'] = processing.run('native:fieldcalculator', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(8)
if feedback.isCanceled():
    return {}

# Adicionar campo à tabela de atributos
alg_params = {
    'FIELD_LENGTH': 10,
    'FIELD_NAME': 'Runoff ponderado',
    'FIELD_PRECISION': 2,
    'FIELD_TYPE': 1, # Decimal (double)
    'INPUT': outputs['CalculadoraDeCampo']['OUTPUT'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['AdicionarCampoTabelaDeAtributos'] =
processing.run('native:addfieldtoattributetable', alg_params, context=context,
feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(9)
if feedback.isCanceled():
    return {}

# Calculadora de campo
alg_params = {
    'FIELD_LENGTH': 0,
    'FIELD_NAME': 'Runoff ponderado',
    'FIELD_PRECISION': 0,
    'FIELD_TYPE': 0, # Decimal (double)
    'FORMULA': '"Runoff" * $area /sum($area)',
    'INPUT': outputs['AdicionarCampoTabelaDeAtributos']['OUTPUT'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['CalculadoraDeCampo'] = processing.run('native:fieldcalculator', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(10)

```

```

if feedback.isCanceled():
    return {}

# Agregados
alg_params = {
    'AGGREGATES': [{'aggregate': 'sum','delimiter': ',', 'input': "'Área_m2'", 'length':
10, 'name': 'Area', 'precision': 2, 'sub_type': 0, 'type': 6, 'type_name': 'double
precision'}, {'aggregate': 'sum','delimiter': ',', 'input': "'Runoff ponderado'", 'length': 10, 'name':
'Runoff', 'precision': 2, 'sub_type': 0, 'type': 6, 'type_name': 'double precision'}],
    'GROUP_BY': 'NULL',
    'INPUT': outputs['CalculadoraDeCampo']['OUTPUT'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['Agregados'] = processing.run('native:aggregate', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(11)
if feedback.isCanceled():
    return {}

# Associar atributos por localização
alg_params = {
    'DISCARD_NONMATCHING': False,
    'INPUT': outputs['Agregados']['OUTPUT'],
    'JOIN': parameters['bacia'],
    'JOIN_FIELDS': [],
    'METHOD': 0, # Criar feição separada para cada feição correspondente (um-para-
muitos)
    'PREDICATE': [2,0], # igual, interseccionam
    'PREFIX': "",
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}
outputs['AssociarAtributosPorLocalizacao'] =
processing.run('native:joinattributesbylocation', alg_params, context=context,
feedback=feedback, is_child_algorithm=True)

feedback.setCurrentStep(12)
if feedback.isCanceled():
    return {}

# Adicionar campo à tabela de atributos
alg_params = {
    'FIELD_LENGTH': 10,
    'FIELD_NAME': 'Name',
    'FIELD_PRECISION': 0,
    'FIELD_TYPE': 2, # Texto (string)
    'INPUT': outputs['Agregados']['OUTPUT'],
    'OUTPUT': QgsProcessing.TEMPORARY_OUTPUT
}

```

```

        outputs['AdicionarCampoTabelaDeAtributos']
processing.run('native:addfieldtoattributetable', alg_params, context=context,
feedback=feedback, is_child_algorithm=True)

        feedback.setCurrentStep(13)
        if feedback.isCanceled():
            return {}

        # Editar campos
        alg_params = {
            'FIELDS_MAPPING': [{ 'expression': "'Name'", 'length': 10, 'name':
'Name', 'precision': 0, 'sub_type': 0, 'type': 10, 'type_name': 'text'}, { 'expression': "'Area'", 'length':
10, 'name': 'Area', 'precision': 2, 'sub_type': 0, 'type': 6, 'type_name': 'double
precision'}, { 'expression': "'Runoff'", 'length': 10, 'name': 'Runoff', 'precision': 2, 'sub_type':
0, 'type': 6, 'type_name': 'double precision'}],
            'INPUT': outputs['AssociarAtributosPorLocalizacao']['OUTPUT'],
            'OUTPUT': parameters['NomeAssociado']
        }
        outputs['EditarCampos'] = processing.run('native:refactorfields', alg_params,
context=context, feedback=feedback, is_child_algorithm=True)
        results['NomeAssociado'] = outputs['EditarCampos']['OUTPUT']
        return results

    def name(self):
        return 'Extrair runoff'

    def displayName(self):
        return 'Extrair runoff'

    def group(self):
        return ""

    def groupId(self):
        return ""

    def createInstance(self):
        return ExtrairRunoff()

```