

DESENVOLVIMENTO DE COMPLEMENTO EM SISTEMA DE INFORMAÇÃO GEOGRÁFICA PARA REALIZAÇÃO DE PROJETOS GEOMÉTRICOS DE ESTRADAS

⁽¹⁾Matheus Fillipe Araújo Galvão

⁽²⁾Reynaldo Furtado Faria Filho

⁽¹⁾Estudante de Engenharia Civil - Universidade Federal de Viçosa, Rio Paranaíba

⁽²⁾Professor Adjunto do curso de Engenharia Civil - Universidade Federal de Viçosa, Rio Paranaíba

Presidente da Banca: Prof. Reynaldo Furtado Faria Filho

Membro 1: Prof^ª. Adriana Zanella Martinhago

Membro 2: Prof. Vinícius Gonçalves de Almeida

16 de Dezembro de 2019

RESUMO: Os transportes são um fator fundamental para a economia de qualquer país e no caso do Brasil, o modal rodoviário é o mais utilizado e recebe mais investimentos. Em 2005 o Governo Federal estabeleceu prioridade para utilização do software livre com objetivo de aliar eficiência com custo-benefício. Nesse contexto se encaixa o QGIS como software de informações geográficas com código aberto e com a possibilidade de ser estendido por meio de complementos escritos na linguagem python, que é de alto nível e suporta múltiplos paradigmas de programação. Nesse artigo é apresentado o desenvolvimento de um complemento para o QGIS, desenvolvido em python, para auxiliar os passos básicos de um projeto geométrico de estradas, deste o traçado horizontal, curvas, até o cálculo de volumes de corte e aterro. A intenção foi desenvolver uma alternativa de código aberto e totalmente gratuita semelhante a softwares já conceituados no mercado como o Civil 3D da AutoDesk e o topoGRAPH de Bentley.

PALAVRAS-CHAVE: concordância horizontal, concordância vertical, complemento, python, QGIS.

GEOGRAPHIC INFORMATION SYSTEM PLUGIN DEVELOPMENT FOR THE GEOMETRIC DESIGN OF ROADS

ABSTRACT: Transport is a fundamental factor for the economy of any country and in Brazil road transportation is the most used mode and the one that receives more investments. In 2005 the Federal Government established the priority on using free software for cost efficiency. QGIS comes to this context as an open source geographic information system software which supports being extended by plugins written in the high-level programming language python. This paper presents the development of a QGIS plugin written in python with the goal of assisting in all the basic steps

35 of geometric road design, beginning with the horizontal layout definition, through curves, up to the
36 volumes of cut and embankment, with the ambition of developing a completely free and open
37 source alternative for similar and already popular software in the industry such as AutoDesk Civil
38 3D and Bentley topoGRAPH.

39 **KEYWORDS:** grade, plugin, python, transition elements, QGIS.

40 **INTRODUÇÃO**

41 O Brasil para ser competitivo frente aos demais países, nas diversas áreas da economia, deve
42 ter sua infraestrutura de transportes como um fator estratégico fundamental (VASCONCELOS &
43 BASSO, 2008). De acordo com o boletim estatístico da Confederação Nacional do Transporte
44 (CNT, 2018) o modal rodoviário possui atualmente, no Brasil, a maior representatividade entre os
45 modais existentes, com finalidade no transporte de cargas e passageiros (Tabela 1).

46 Tabela 1: Porcentagem de representatividade entre os modais existentes no Brasil.

Modal	Porcentagem
Rodoviário	61,1
Ferrovário	20,7
Aquaviário	13,6
Dutoviário	4,2
Aéreo	0,4

47 Fonte: CNT (2018).

48 O transporte rodoviário, por ser o mais utilizado, recebe investimentos constantes para sua
49 manutenção, ampliação, bem como criação de novas rodovias. Para a criação de novas rodovias
50 deve-se realizar diversos estudos/projetos: estudos de planejamento de transporte; estudos de
51 tráfego; estudos geológicos; estudos hidrológicos; estudos topográficos; projeto geométrico; projeto
52 de obras de terra; projeto de terraplenagem; projeto de pavimentação; projeto de drenagem; projeto
53 de obras de arte; projeto de viabilidade econômica; projeto de desapropriação; projetos de
54 interseções, retornos e acesso; projeto de sinalização; projeto de elementos de segurança, dentre
55 outros (PONTES FILHO, 1998).

56 O projeto geométrico de uma estrada é composto pelos elementos físicos, como
57 comprimento de retas, largura de pista, raios de curvas e rampas, que estão diretamente associadas
58 às características de operação dos veículos (frenagem, velocidade, distância de visibilidade, dentre
59 outros). Assim, na maioria dos casos, o desenvolvimento do projeto geométrico é uma sequência
60 estruturada por rotinas que visam definir os parâmetros geométricos, bem como calcular a
61 geometria. A repetição destas rotinas fornece várias soluções para análise e, a escolha de uma
62 considerada melhor possibilita o desenho do projeto (OLIVEIRA JUNIOR, 2007).

63 No mercado existem diversos softwares disponíveis para a geração de projeto geométrico de
64 estradas, entretanto o custo de aquisição, principalmente para o ensino nas Universidades, é
65 elevado. Pode-se citar alguns destes softwares disponíveis no mercado que possibilitam a realização
66 de projeto geométrico de estrada: TopoCal, Civil 3D, topoGRAPH, DataGeosis, TCP-MDT,
67 InRoads, Posição, dentre outros (MENEZES et al., 2014).

68 Em 2005 o Governo Federal estabeleceu prioridade para utilização do software livre com
69 objetivo de aliar eficiência com custo-benefício (GUIA LIVRE, 2005). Assim, além de um software
70 livre poder ser aplicado na geração de um projeto geométrico de uma rodovia que será implantada,
71 este poderá ainda ser de grande utilidade para o ensino de qualidade nas Universidades com um
72 custo baixo para às mesmas.

73 Neste contexto, está inserido o QGIS, que foi inicialmente chamado de Quantum GIS. O
74 QGIS é um Sistema de Informação Geográfica (SIG) de Código Aberto licenciado segundo a
75 Licença Pública Geral GNU. É um projeto oficial da *Open Source Geospatial Foundation* (OSGeo).
76 Funciona em Linux, Unix, Mac OSX, Windows e Android e suporta inúmeros formatos de vetores,
77 rasters e bases de dados e funcionalidades.

78 Assim, o objetivo deste trabalho foi o de aliar as capacidades de geoprocessamento do QGIS
79 com uma base de código que cobre cada uma das etapas de um projeto geométrico de estradas,
80 criando um complemento com fins educacionais a priori.

81

82 **2. REFERENCIAL TEÓRICO**

83 **2.1 Projeto geométrico de estradas**

84 Antes de projetar os parâmetros de uma rodovia deve-se esgotar todas as possibilidades de
85 desenvolvimento de um traçado. Para isso, o Departamento Nacional de Infra-estrutura de
86 Transportes (DNIT) disponibilizou a Instrução de Serviço IS-207 que trata dos Estudos
87 Preliminares de Engenharia para Rodovias, ou seja, estudos de traçado (DNIT, 2006). Para isso,
88 deve-se elaborar os estudos que abrangem as seguintes tarefas:

- 89 a) Definir as diretrizes tecnicamente possíveis da rodovia;
- 90 b) Determinar a viabilidade física das alternativas indicadas como sendo as adequadas pelos
91 estudos de tráfego para a ligação rodoviária proposta;
- 92 c) Definir certas soluções básicas para a elaboração desembaraçada dos trabalhos mais
93 detalhados nos estudos posteriores;
- 94 d) Estimar os custos aproximados de construção e de desapropriação para fins de avaliação
95 econômica e financeira.

96 Após a fase de estudo dos traçados faz-se a elaboração do projeto geométrico. De acordo
97 com a Instrução de Serviço IS-208, do DNIT, o projeto geométrico deve ser elaborado em duas
98 fases (DNIT, 2006):

99 a) FASE DE PROJETO BÁSICO: deve ser elaborado a partir dos estudos topográficos
100 realizados segundo o que dispõe a Instrução de Serviço IS-204. Já as características geométricas
101 mínimas do projeto geométrico, serão norteadas pelos Estudos de Tráfego, e deverão atender as
102 recomendações do Manual de Projeto Geométrico de Rodovias Rurais DNER - 1999. O projeto
103 geométrico deve ser elaborado detalhadamente de maneira a apresentar condição de ser locado na
104 fase de projeto. No projeto geométrico devem constar: a) Quadro de características técnicas e
105 operacionais; b) Projeto em planta, na escala de 1:2000 (composição das curvas horizontais;
106 elementos cadastrais; interseções; pontes e viadutos e bueiros); c) Projeto em perfil, nas escalas de
107 1:2.000 (H) e 1:200 (V) (composição das curvas verticais; rampas; pontes e viadutos e bueiros) e d)
108 Seções transversais típicas da plataforma.

109 b) FASE DE PROJETO EXECUTIVO: deve ser elaborado a partir dos estudos topográficos
110 realizados segundo o que dispõe a Instrução de Serviço IS-205. Este projeto deverá constituir-se de:
111 a) Projeto planialtimétrico, nas escalas de 1:2.000 (H) e 1:200 (V); b) Determinação das seções
112 transversais do projeto, nas escalas de 1:200 ou 1:100 e c) Detalhamento dos elementos especiais do
113 projeto como: retornos e acessos em nível; terceiras faixas de tráfego; tapers.

114 As rodovias por ser um ente tridimensional podem ter seus elementos geométricos
115 projetados nas três dimensões. A linha (eixo) que representa a via é projetada em um plano
116 horizontal e representada em uma planta, enquanto que o projeto em perfil tem seu
117 dimensionamento feito a partir de um plano vertical, possibilitando a definição do greide que é a
118 representação do eixo da pista no plano vertical (OLIVEIRA JUNIOR, 2007). Os principais
119 elementos geométricos de uma estrada são apresentados na Figura 1 - Anexo A (PONTES FILHO,
120 1998).

121

122 **2.2 Projeto geométrico de estradas desenvolvido com auxílio do computador**

123 Menezes et al. (2014) fizeram uma análise comparativa dos softwares topoGRAPH 98 SE e
124 TCP-MDT (versão 6.7) que são utilizados para o desenvolvimento de projetos geométrico de
125 estradas. O primeiro (topoGRAPH) possui mais de duas décadas de existência e é bastante
126 difundido no Brasil, enquanto que o segundo (TCP-MDT) é de origem Europeia. Ao final do
127 trabalho os autores concluíram que “*embora o software TCP-MDT, se encontra em adaptação às*
128 *normas brasileiras, demonstrou-se de forma geral, ser um programa simples com recursos robustos*
129 *e práticos*”, enquanto que o “*topoGRAPH não apresentou suas ferramentas de projeto tão*

130 *explícitas quanto ao TCP-MDT, além de deixar a desejar na sua visualização em 3D*". Os autores
131 salientam que o topoGRAPH está em conformidade com as normas brasileiras.

132 Campos Neto et al. (2012) compararam os softwares AutoCAD Civil 3D e o Microstation
133 V8i com o módulo InRoads para a geração do projeto geométrico de uma estrada. Os autores
134 fizeram ainda o projeto de forma manual. Ao final do trabalho ficou evidente que o processo
135 computacional é superior ao manual em relação ao tempo gasto para desenvolver um projeto e em
136 relação à qualidade do produto final. Quanto aos softwares os autores concluíram que o Bentley
137 InRoads se mostrou superior ao AutoCAD Civil 3D no que se diz respeito ao desempenho.

138 No que se refere a software Livre, para realização de projeto geométrico de estradas, o autor
139 Silva Júnior (2003) desenvolveu um software cuja metodologia divide a elaboração do projeto
140 geométrico automatizado em cinco etapas básicas:

141 Etapa 1 - Definição Topográfica para o Projeto;

142 Etapa 2 - Aquisição e processamento dos Dados Topográficos;

143 Etapa 3 - Criação do Modelo Digital do Terreno;

144 Etapa 4 - Criação do Modelo Digital do Projeto;

145 Etapa 5 - Emissão de relatórios, desenhos e medições.

146 O autor Oliveira Júnior (2007), implementou outras funções ao software desenvolvido por
147 Silva Júnior (2003). Como o trabalho de Silva Júnior (2003) centrou-se na aquisição de dados,
148 Oliveira Júnior (2007), procurou desenvolver seu sistema a partir desse ponto, ou seja, fez a
149 incorporação dos dados já levantados. Para isso, Oliveira Júnior (2007) utilizou a linguagem de
150 programação JAVA por ser uma linguagem portátil, flexível e possível de ser utilizada em diferentes
151 sistemas operacionais. Além da linguagem JAVA o referido autor utilizou o sistema JUMP por ser
152 um pacote com solução para alguns problemas, como a representação gráfica do projeto, além de
153 possibilitar a realização do projeto inteiramente georreferenciado, dentro de um sistema de
154 informações geográficas, aumentando consideravelmente sua potencialidade.

155 Oliveira Júnior (2007) relata que o seu software teve o intuito de:

156 [...] desenvolver um sistema novo, mas que não se esgotasse, ou seja, que não obrigasse a
157 um próximo pesquisador iniciar tudo novamente. Para tanto, ele teve como características
158 principais ser um sistema flexível, de fácil reutilização, possuir um modelo teórico, um
159 projeto de software independente da linguagem de implementação e ser documentado para
160 possibilitar o entendimento do que foi proposto. (JUNIOR, 2007)

161

162

163

164 **2.3 Ambiente de programação**

165 **2.3.1 Python**

166 Python é uma linguagem de programação criada por Guido Van Rossum em 1991. A
167 linguagem suporta múltiplos paradigmas de programação, como a programação estruturada
168 funcional e orientada a objetos. Criada sobre a licença de código livre e multiplataforma, o Python
169 está presente nos principais sistemas operacionais como Windows, Linux e OSX (PYSCIENCE,
170 2016).

171 A linguagem Python é amplamente utilizada em projetos científicos devido a facilidade de
172 desenvolvimento e integração com sistemas legados e bibliotecas providas por outras linguagens.
173 Além disto, é possível ter uma boa performance sendo muitas vezes comparáveis a C++, pelo fato
174 da plataforma ser simples fazendo com que consiga fazer otimizações que seriam mais trabalhosas
175 em outras linguagens (PYSCIENCE, 2016).

176

177 **2.3.2 Qt**

178 Utilizado por diversas empresas e projetos de código aberto, o Qt é considerado um dos
179 mais versáteis *frameworks* gráficos. Desenvolvido em 1995 pela empresa norueguesa Trolltech,
180 posteriormente pela equipe da Nokia e atualmente pela Digia. Tem como principal atrativo a
181 premissa de código universal, no qual o programador não precisara se preocupar como funciona a
182 gestão de elementos em interfaces de cada sistema operacional. Pois o mesmo é feito pela
183 plataforma, necessitando apenas ter um adaptador para cada sistema, para gerar os executáveis para
184 cada um deles. Como se trata de um projeto, que embora tenha fins comerciais, possuir código livre
185 para melhorias de integração e funcionamento faz com que ele seja largamente utilizado em
186 soluções de código aberto e em diversas mídias (NOKIA, 2009).

187 Desenvolvido em C++ com suporte a diversas linguagens como Python, C#, Java, R, Ruby
188 entre outros. Além do suporte para dispositivos embarcados e móveis, faz do *framework* uma das
189 soluções mais rápidas e simples para desenvolvimento de software multi-plataforma. Sua
190 arquitetura se baseia na criação de interfaces que utilizam o QtDesigner, na qual gera-se um código
191 específico para a linguagem escolhida. A linguagem C++ é gerada nativamente, e com isso é
192 possível fazer todo monitoramento de eventos e respostas da interface, sem programar a mesma de
193 maneira direta. Em seguida pode-se gerar executáveis para cada arquitetura de hardware e de
194 sistema operacional (HÜBSCHLE et al., 2011).

195 Essa biblioteca também é utilizada como a base de toda a interface gráfica para o QGIS e
196 conseqüentemente a mais recomendada e viável para o desenvolvimento das interfaces para
197 complementos dentro desse software, por já fazer parte de seu ambiente padrão.

198 2.3.3 PyQtGraph

199 *Pyqtgraph* é uma biblioteca para representação gráfica e criação de interfaces desenvolvida
200 em *python* usando PyQt5 e *numpy*. Seu objetivo é a utilização em aplicações matemáticas,
201 científicas ou na engenharia. Apesar de ter sido completamente escrita em python, seu desempenho
202 é bom graças a ajuda do *numpy* para o processamento de números e a *Framework GraphicsView* do
203 Qt para visualização acelerada. Essa biblioteca é distribuída na licença *Open Source* MIT
204 (PYQTGRAPH, 2018).

205 Com o auxílio dessa biblioteca podem ser gerados gráficos e interfaces de desenho em 2D
206 que o QGIS, por si só não disponibiliza tais como soluções para a realização de traçados verticais e
207 para a elaboração das sessões tipo e visualizações de perfis.

208

209 2.3.4 QGIS

210 O QGIS, antes conhecido como Quantum GIS, é um software gratuito, *open source* e de
211 multiplataforma, sendo compatível com os sistemas operacionais Windows, Linux, Mac e Android
212 (em estado de desenvolvimento) sendo dedicado à visualização, edição e análise de dados
213 geoespaciais. É um projeto oficial da “*Open Source Geospatial Foundation*” (OSGeo) licenciado
214 sobre a GNU suportando várias modalidades de computação vetorial com o auxílio de *rasters* e
215 outros formatos de banco de dados.

216 Foi criado por Gary Sherman em 2002 e escrito em C++, com uso extensivo da biblioteca
217 Qt, utilizando-se de programa de manutenção e lançamento composto majoritariamente por
218 programadores voluntariados até os dias atuais. A partir do lançamento da versão 2, o QGIS obteve
219 suporte ao desenvolvimento de *plugins* com a linguagem de programação python e conta com um
220 repositório oficial *online* com os mais diversos tipos de ferramentas que podem ser facilmente
221 instaladas (QGIS, 2018).

222 O QGIS não possui atualmente aplicações específicas para o desenvolvimento de um
223 projeto geométrico de estradas. Entretanto, possui diversas ferramentas implementadas que podem
224 permitir o seu desenvolvimento. O projeto QGIS iniciou com o intuito de ser um visualizador
225 gratuito de bases de dados geográficas que pudesse ser instalado em sistemas operacionais livres. O
226 projeto foi iniciado em 2002 (MANGHI et al., 2011).

227

228

229

230

231

232 3. METODOLOGIA

233 Para implementação do novo sistema foi necessário o estudo dos *plugins* existentes para o
234 QGIS e também o domínio na criação desse tipo de software. Por padrão os *plugins* para essa
235 plataforma são desenvolvidos na linguagem *python* com interface gráfica Qt. O uso do QGIS como
236 base desse projeto, ao invés de uma solução autônoma, deve-se à quantidade de ferramentas de
237 geoprocessamento que já se encontram na API do QGIS como por exemplo: cálculos geométricos
238 diversos no plano, leitura de pixels em *rasters*, que são de fácil acesso para o desenvolvedor.

239 O projeto foi escrito na versão 3.6 da linguagem *python* para ser compatível com a terceira
240 versão do QGIS. A biblioteca de interface gráfica (GUI) utilizada foi a PyQt5 que está presente
241 dentro do próprio ambiente *python* padrão do QGIS, o que facilita o desenvolvimento e reduz o
242 tamanho final do *plugin*. Essa biblioteca contém os *bindings* do Qt para *python*, tornando possível a
243 criação de interfaces gráficas em *plugins*.

244 Inicialmente foram identificados os requisitos necessários para o desenvolvimento do
245 projeto geométrico de uma estrada e em seguida foi feita uma modelagem gráfica da solução para
246 facilitar o trabalho na etapa seguinte que é a implementação da mesma. Esta implementação foi
247 basicamente a transformação da modelagem para código “python” integrando com a interface
248 gráfica “Qt”.

249 Foi desenvolvida a leitura dos dados de altimetria ou cotas. Nela foi utilizado um modelo
250 digital no formato “Geotiff” que é um *raster* onde os valores dos pixels indicam os valores das
251 elevações. Os valores extraídos desses arquivos podem, opcionalmente, ser interpolados durante a
252 leitura de acordo com a intenção do usuário. Esta interpolação pode acarretar em maiores gastos
253 computacionais.

254 Por fim foi realizado o teste das funcionalidades, por meio de estudos de casos com testes
255 desenvolvidos em conjunto com o profissional da área. A partir dos relatos dos usuários foram
256 realizadas as alterações para melhorar o sistema desenvolvido. O complemento está baseado na
257 possibilidade de uso em múltiplos sistemas operacionais, visto que foi testado no Windows 10,
258 Windows 7, Linux Ubuntu 16.04 e 18.04, tendo somente o QGIS 3 como dependência.

259

260 3.1 Gestão e armazenamento dos dados de projeto

261 A modelagem no plano horizontal baseou-se principalmente nas configurações básicas e na
262 criação de estacas e curvas. Estas pertencem ao mesmo controle, por se tratar de recursos
263 semelhantes em nível de controle, no entanto, se separam em outras camadas, pois as ações e a
264 interação ocorre de maneira diferente. A modelagem do perfil vertical se deu a partir da tabela
265 greide e o perfil transversal a partir do relevo da sessão, sessão tipo e perfil transversal.

266 Para o armazenamento dos dados foi definida a necessidade de ser um sistema portátil que
267 possa ser facilmente manipulado sem gastar muitos recursos computacionais. Seguindo esses
268 requisitos a melhor solução encontrada foi o SQLite, por atender as necessidades, possuir fácil
269 integração com a plataforma QGIS e por ser leve, mesmo sendo um sistema gerenciador de banco
270 de dados robusto. Como apresentado na Figura 2 (Anexo A), a estratégia utilizada para manipulação
271 de dados foi separá-los em entidades relacionadas representadas pelos quadros dessa imagem, sendo
272 elas: estacas, curvas horizontais, sessão tipo, greide, relevo, curva vertical. Os dados das
273 configurações principais são armazenados na tabela “PROJECT”, tais como a classe de projeto e as
274 limitações de inclinação e velocidade.

275 A fim de facilitar o carregamento e edição do perfil horizontal e suas curvas, adotou-se em
276 redundância o formato de arquivo *geopackage*, que também contém um banco de dados SQLite que
277 pode ser instanciado dentro do QGIS no formato de uma *layer*. Com o mesmo objetivo de
278 otimização para as sessões transversais desenvolveu-se uma biblioteca para armazenar os dados
279 geométricos prismoides que fosse capaz de ser serializada pela biblioteca padrão do python
280 “pickle”. Com isso pode-se gerar um arquivo único que possa ser salvo no disco e instanciado na
281 memória rapidamente com todas as informações massivas da sessão transversal. Esse arquivo
282 recebeu a extensão “.prism”.

283 A fim de simplicidade e praticidade, todos esses arquivos ficam armazenados em uma pasta
284 compactada “.zip” que pode ser compartilhada e importada em outra máquina com o QGIS e com o
285 complemento. Esse arquivo compactado também fica vinculado e com seu caminho salvo no
286 arquivo de projeto próprio do QGIS. A árvore de arquivos de projeto é representada pela Figura 3
287 (Anexo A). Vale observar que o nome dos arquivos é dado em função do id do traçado
288 correspondente que é armazenado pela “TABLEESTACA” (Figura 2 - Anexo A), já que o mesmo
289 projeto pode possuir múltiplos traçados. O complemento faz uso das pastas temporárias do sistema
290 operacional para descompactar esses arquivos durante suas diversas operações.

291

292 **3.2 Traçado e curvas horizontais**

293 A definição do traçado horizontal pode ser feita criando uma ou importando uma *layer* no
294 QGIS. Uma *layer* pode ser gerada por meio de arquivos do tipo *shapefile*, *dxf* (CAD), *geopackage*,
295 enfim, qualquer arquivo que seja suportado pelo QGIS será suportado pelo complemento desde que
296 seja do tipo linha vetorial, mesmo possuindo múltiplos atributos. O QGIS se encarrega de converter
297 todos esses formatos geométricos para um tipo único “QgsVectorLayer” de onde podem ser
298 extraídas todas as informações geométricas necessárias.

299 A Tabela 2 representa os tipos de dados que estão contidos na *layer* de desenho. A coluna
 300 atributo mostra o identificador para cada parte da geometria que pode possuir campos diferentes, o
 301 que é comum dos tipos de arquivos com informação geográfica. O complemento pode importar
 302 *layers* com esses campos já preenchidos ou gerar a partir de outra *layer*. O campo tipo representa o
 303 tipo de geometria. Qualquer valor de texto que inicia com “T” será interpretado como tangente, “C”
 304 para curva circular simples e “E” para espiral.

305 O algoritmo de importar *layers* segmenta o desenho do traçado horizontal entre o intervalo
 306 de estacas definido pelo usuário nas configurações, que é por padrão 20 metros. Em cada ponto
 307 desses fica definido uma estaca sem descrição. Em cada ponto inicial, final ou de interseção entre
 308 duas tangentes define-se uma estaca intermediária “PI” (ponto de interseção). Em pontos notáveis
 309 tais como nas mudanças de tipo de atributo do desenho, o que pode significar o início da parte de
 310 transição de uma curva ou da parte circular, define-se uma estaca intermediária que recebe a
 311 descrição “TS”, “SC”, “CS” ou “ST”, onde “T” significa tangente, “S” para espiral e “C” para
 312 trecho circular. O campo descrição é opcional com a intenção de permitir o usuário fazer simples
 313 anotações textuais para cada trecho caso seja necessário.

314 Para cada uma das estacas obtidas calcula-se as coordenadas de acordo com a fonte CRS
 315 configurada no QGIS e os azimutes. Os elementos de curva horizontal circular e espiral também
 316 podem ser computados e desenhados pelo complemento para serem recalculados e inseridos a uma
 317 tabela de horizontais já existente. Se forem adicionados pelo complemento esses dados serão
 318 armazenados na tabela “CUVAS_SIMPLES” do banco de dados, para que possam ser alterados pelo
 319 usuário. O desenho final é armazenado em um arquivo no formato *geopackage* para que se possa
 320 usar o mesmo algoritmo de conversão em tabela de estacas.

321 Tabela 2: Exemplo de tabela de atributos para uma *layer* de traçado horizontal usada pelo
 322 complemento no formato *geopackage*. Todas as unidades estão em metros e graus.

Atributo	Tipo	Descrição	Raio	Angulo de Deflexão (Delta)	Tangente Externa (T)	Desenvolvimento (D)
1	T					
2	E		520	16,67	128,073	120
3	C		520	16,67	128,073	120
4	E		520	16,67	128,073	120
5	T					
6	E		400	9,382	239,073	200
7	C		400	9,382	239,073	200
8	E		400	9,382	239,073	200
9	T					
10	E		683,1	33,56	201,235	180
11	C		683,1	33,56	201,235	180
12	E		683,1	33,56	201,235	180

323 Todo esse processo é representado pelo fluxograma da Figura 4 (Anexo A). Destaca-se o
324 processo iterativo feito para cada atributo contido na *layer vetorial*, o que resulta em um elemento
325 geométrico isolado para cada atributo ou, colocando de outra forma, para cada linha na Tabela 2, do
326 qual se faz a segmentação em estacas. Essas estacas são armazenadas na tabela “ESTACA” do
327 banco de dados dentro do arquivo compactado de projeto.

328

329 3.3 Perfil e curvas verticais

330 Para a parte da sessão vertical de um projeto de estradas o complemento desenvolvido
331 realiza inicialmente a tradução do perfil horizontal em um perfil longitudinal do relevo do terreno
332 onde possa ser traçado, pelo usuário, por meio de uma interface simples e intuitiva, o greide e
333 definidas as curvas verticais. Esse processo é mostrado pelo fluxograma na Figura 6 (Anexo A).

334 Para cada estaca criada no passo anterior deve-se extrair uma elevação, relativa às suas
335 coordenadas, a partir da qual o terreno para a elaboração do perfil vertical será definido. Esse
336 terreno é basicamente a vista de perfil da sessão horizontal e sua utilidade é servir de base para o
337 greide de projeto. Essas elevações são obtidas através de *rasters* e podem ser opcionalmente
338 interpolados por um algoritmo que foi construído especificamente para o complemento. A partir das
339 coordenadas da estaca localiza-se o pixel no qual esse ponto está contido (ponto em vermelho na
340 Figura 5 - Anexo A). Existem 8 pixels ao redor do central e são escolhidos os três com os centros
341 mais próximos do ponto em questão. A elevação interpolada é calculada com as equações 1 e 2:

$$342 \quad P_i = \frac{(D - L_i)}{D} \quad (1)$$

$$343 \quad C = \frac{\sum v_i P_i}{\sum P_i} \quad (2)$$

344 Onde D é o comprimento da diagonal (Figura 5 - Anexo A), P os pesos da média, v a
345 elevação de cada pixel associado, L a distância do ponto ao centro do pixel central e L₁, L₂ e L₃ as
346 distâncias aos centros dos pixels mais próximos. Vale ressaltar que os comprimentos a e b de cada
347 pixel podem ser convertidas para metros para que esse processo seja feito e o valor de v é
348 proporcional ao valor do pixel, que é o módulo de um vetor tridimensional RGB. Os valores da
349 elevação associados às estacas são armazenados na tabela “RELEVO_SESSÃO”.

350 Para uma melhor interação com usuário foi necessário implementar uma interface de
351 desenho e edição do greide sobre o relevo. Como a parte de perfis expandiria as classes atuais do
352 projeto além do limite considerado legível, optou-se por isolar suas funcionalidades em uma classe
353 isolada denominada “Ui_Perfil” que cuida da interface de edição interativa do greide buscando
354 permitir ao usuário o desenho sobre o relevo feita a partir da *pyqtgraph*. Essa mesma interface

355 serviu de base para a sessão transversal e o Diagrama de Bruckner. Procurou-se manter o mesmo
356 padrão de entidades e adicionar subclasses da biblioteca *pyqtgraph* com destaque a subclasses ROI.
357 Esta subclasse é útil para os desenhos com as adicionais e métodos para determinação da posição
358 dos vértices e com métodos carregados com sinais Qt que abrem os diálogos de edição de vértices e
359 rampas.

360 A subclasse “CustomViewBox” foi criada para tornar a movimentação e comandos de
361 arrastar do mouse mais familiares para usuários de softwares CAD. Destaca-se que a classe
362 “Perfil_cv” encapsula os cálculos básicos para determinar as cotas em cada ponto da curva vertical
363 dado uma progressiva e a inclinação das rampas adjacentes ao vértice. Na Figura 7 (Anexo A)
364 apresenta-se o diagrama de classes da interface gráfica para o perfil. Essas classes possuem também
365 os métodos onde são realizadas todas operações relacionadas à manipulação dos vértices,
366 salvamento, restauração e cálculos auxiliares tais como a determinação da inclinação de rampas,
367 comprimentos e cotas.

368 Passada a fase de modelagem, o passo seguinte foi aplicação dos conceitos na plataforma
369 através de código, orientados pelos padrões da plataforma. A primeira parte foi o desenvolvimento
370 das interfaces através do *framework* Qt baseado em um padrão clássico. Para tal buscou-se trabalhar
371 de forma mais próxima da utilizada pelo sistema, com o uso extensivo do sistema de *signals e slots*
372 para a intercomunicação entre as classes do perfil e Estacas. O *pyqt5* facilitou esse processo o que
373 pode tornar a execução mais reativa e assíncrona, o que também foi usado para que o programa
374 apresente quaisquer alterações de valores refletidas nos desenhos em tempo real.

375

376 **3.4 Perfil transversal e Diagrama de Bruckner**

377 Antes da definição do perfil transversal é necessário combinar a tabela de estacas horizontais
378 com as verticais, já que as estacas intermediárias de curvas horizontais e de curvas verticais nem
379 sempre irão coincidir ou serão estacas inteiras. Esse processo é feito com a organização das duas
380 tabelas em ordem de progressivas, que é a distância horizontal acumulada, somando-se as duas
381 listas de estacas, eliminando-se as repetições e reorganizando-as em ordem crescente de estacas. O
382 resultado final é uma tabela de interseção entre o perfil vertical e o horizontal.

383 Para cada estaca na tabela de interseção calcula-se uma coordenada e também qual é a reta
384 perpendicular ao ponto dado o ângulo de azimute da estaca. Essa reta é o que define a sessão
385 transversal para aquela estaca e, em um procedimento semelhante ao da obtenção das elevações
386 para o traçado horizontal, calcula-se todas as cotas ao longo desse reta. A largura a ser considerada e
387 precisão são configuráveis.

388 Foi criada uma interface semelhante à do perfil vertical para definir a sessão tipo que
389 representa o corte transversal da pista que deseja-se projetar. Essa interface permite o usuário
390 definir os taludes de corte e aterro, valetas e a pista em si. Aqui é utilizada a subclasse “ssROI” que
391 gerencia os vértices editáveis para a sessão tipo. Todo esse algoritmo está apresentado no
392 fluxograma da Figura 8 (Anexo A).

393 O próximo passo é calcular a interseção entre a sessão tipo definida pelo usuário e o terreno.
394 Optou-se por criar uma biblioteca própria para esse processamento e para os cálculos de volume ao
395 invés de usar uma solução já reconhecida como *scipy* ou a própria API do QGIS. Essa escolha se
396 justifica pela carência dessas outras opções no cálculo de interseção entre curvas e outras funções
397 que geralmente se encontram em núcleos CAD. Para a criação dessa biblioteca procurou-se separar
398 os problemas em partes simples pela orientação a objetos, iniciando-se com a definição de um
399 ponto. Depois uma reta a partir de dois pontos e limitada por eles, definindo-se métodos para a
400 obtenção do ponto de interseção. Com uma lista de retas interligadas definiu-se a classe curva, com
401 métodos para a interseção com outras curvas. A partir de duas curvas se define uma face, desde que
402 essas curvas possuam mais de duas interseções, com métodos para o cálculo de sua área e por fim
403 um prismoide a partir de múltiplas faces.

404 As áreas das faces são determinadas pela fórmula de Gauss, já que todas as faces são
405 perpendiculares ao traçado horizontal, da seguinte maneira:

$$A = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| \quad (3)$$
$$A = \frac{1}{2} |x_1 y_2 + x_2 y_3 + \dots + x_{n-1} y_n + x_n y_1 - x_2 y_1 - x_3 y_2 - \dots - x_n y_{n-1} - x_1 y_n|$$

407 Onde x_i e y_i são as coordenadas do i -ésimo pixel em um polígono e A a área total desse
408 polígono. O cálculo de volume do prismoide se dá a partir do somatório do produto das distâncias
409 entre as faces e as médias de suas áreas. Esse método é conhecido como semissoma. O objeto
410 prismoide pode ser armazenado diretamente no disco a partir da biblioteca *pickle*, que permite
411 serializar um objeto python armazenando todos seus atributos de maneira que ele possa ser
412 recarregado na memória e instanciado novamente. Porém essa solução pode falhar e então o
413 complemento irá recalculer novamente o prismoide do traçado a partir dos dados no banco de
414 dados. Os relevos são armazenados na tabela “RELEVO_SESSAO” e as sessão tipo em
415 “TRANSVERSAL”.

416 A partir desse prismoide pode-se calcular o volume total de aterro e de corte para o traçado,
417 e gerar o Diagrama de Bruckner que se trata dos volumes acumulados ao longo do traçado. Também
418 foi criada uma interface baseada na “Ui_Perfil” para visualizar esse diagrama em um certo intervalo

419 de estacas e para fazer medições de distâncias e volume sobre ele. Para uma melhor visualização
420 dos volumes adiciona-se 1000 m³ ao menor volume do gráfico. A interface também permite a
421 determinação da onda de Bruckner sendo esta definida pelo retângulo de mesma área que a área
422 abaixo do trecho delimitado do diagrama de massas, dado que o usuário escolha a base desse
423 retângulo que é a distância média de transporte. Essa ferramenta é útil para que seja feito o balanço
424 de volumes a partir do cálculo do momento de transporte.

425

426 **4. RESULTADOS**

427 O projeto está mantido publicamente no *github* e pode ser visualizado através do endereço:
428 <https://github.com/matheusfillipe/Topografia> onde também se encontram instruções para a
429 instalação. A intenção é que também esteja disponível no repositório padrão do QGIS. A licença
430 escolhida para o software foi a GPL versão 2.0, que é a mesma licença de código aberto utilizada
431 pelo QGIS.

432 A árvore de arquivos final do código fonte (Figura 9 - Anexo A) se baseia em três principais
433 diretórios. Primeiramente “view” contém a lógica da interface gráfica. Por exemplo o arquivo
434 “config.py” contém a classe para a janela de configurações e “estacas.py” os vários diálogos com as
435 tabelas de estacas. Os *layouts* das janelas e diálogos foi modelada usando o formato *Designer ui* que
436 pode ser editado com o software Qt Designer. A pasta “model” contém a interface entre os dados na
437 memória e o banco de dados e armazenamento. A pasta “controller” contém os algoritmos descritos
438 na sessão anterior. Alguns destaques são os diretórios “controler/Geometria”, que contém a
439 biblioteca para cálculo de interseções de curvas e cálculos relacionados ao prismoide, e o arquivo
440 “model/utills.py” que contém várias utilidades relacionadas ao uso da API do QGIS dentre elas a
441 função de interpolação de rasters.

442 O complemento se apresenta em estado funcional, cobrindo o básico do projeto geométrico
443 de estradas e permitindo a extração dos desenhos em formato CAD, de maneira simplificado, e das
444 tabelas em *csv* (Planilha com valores separados por vírgula). A seguir será apresentado de forma
445 resumida o funcionamento do complemento mostrando suas capacidades abordando um projeto
446 simples dentro do QGIS.

447

448 **4.1 O fluxo de projeto com o complemento**

449 Na Figura 10 (Anexo A) é apresentada a interface principal do complemento na barra de
450 ferramentas do QGIS, sendo composta por seis ícones, sendo eles em ordem partindo da esquerda:
451 (1) criação do arquivo de projeto; (2) abertura de arquivo um arquivo de projeto (.zip); (3) salvar o

452 projeto atual; (4) configurações de projeto; (5) mostrar imagens de satélite; (6) o gestor dos
453 arquivos de projetos.

454 Para criação, abertura e armazenamento de arquivos topográficos foi utilizada a interface
455 padrão do sistema operacional no que se refere à seleção de arquivos e pastas, com a
456 implementação específica feita internamente, o que facilita o uso do sistema. Para criação de
457 arquivos topográficos, o usuário apenas irá escolher o local e o nome a qual serão salvas as
458 configurações do projeto. Este poderá ser aberto com a mesma lógica, e quando isso ocorrer todas
459 as configurações e traçados estarão importados para o QGIS, sem a necessidade de armazenar em
460 arquivos separados. Para seleção dos arquivos topográficos do tipo *shapefile* (.shp), também foi
461 utilizada a interface padrão, com a diferença que é escolhida a pasta e não os arquivos separados.

462 Vale ressaltar a necessidade de se selecionar o sistema de projeção cartográfica adequado
463 nas configurações de projeto do QGIS e nas *layers* importadas antes que se inicie qualquer
464 procedimento que utilize o complemento. O primeiro passo é criar um novo arquivo topográfico por
465 meio do primeiro ícone. O segundo passo consiste em fazer as configurações básicas do projeto
466 através do quarto ícone. Apesar de existir o botão de salvar (terceiro ícone), todas as modificações
467 são salvas no momento em que editadas e aplicadas dentro do complemento, exceto pelas janelas de
468 perfil vertical, transversal e diagrama de Bruckner, que são implementadas com a *pyqtgraph*.

469 Todas configurações do projeto geométrico são realizadas na janela de configurações
470 (Figura 11 - Anexo A). Nela pode ser configurada a classe de projeto que influencia nas inclinações
471 limite do greide e velocidade que são mostrados abaixo. A velocidade é por padrão para terreno
472 ondulado, mas todas essas inclinações para os três tipos de terreno e a velocidade podem ser
473 editados independentemente da classe de projeto caso seja necessário. Outros valores que podem ser
474 interessantes nessa janela são o *checkbox* interpolar rasters, que se verificado performará a
475 interpolação dos pixels nas operações de extração de elevações; a distância entre as estacas, que é
476 por padrão de 20 metros; EMAX que é a máxima taxa de superelevação admissível em m/m,
477 utilizada para calcular o raio mínimo recomendado em curvas horizontais; o separador CSV
478 utilizado nas planilhas a serem exportadas e as configurações para a extração de elevação nas
479 sessões transversais, que são a largura transversal total de cada lado a partir do eixo, medido em
480 metros, e o número de pontos a serem distribuídos igualmente ao longo dessa largura, de onde
481 se extrai valores de elevação. Esse último valor deve ser mantido como mínimo possível para que
482 os procedimentos com a sessão transversal carreguem mais rápido.

483 A partir das configurações realizadas, o próximo passo é a criação dos traçados e
484 desenvolvimento das curvas horizontais. Qualquer *layer* vetorial que pode ser criada no QGIS ou
485 extraída de um programa CAD pode ser utilizada. O quinto ícone da barra de ferramentas carrega

486 um *tilemap* com as imagens de satélite e imagens de elevação do terreno, fornecidas pela Google,
487 com o intuito de auxiliar na definição do traçado. Também existe um botão que serve como um
488 atalho para gerar a layer com o traçado horizontal dentro do QGIS (Figura 12 - Anexo A). Quando a
489 layer estiver definida (Figura 13 - Anexo A), para criar uma tabela de estacas basta usar o botão
490 “Novo arquivo de estacas” da janela da Figura 12 (Anexo A) e selecionar a layer com o traçado. A
491 tabela aparecerá já calculada como na Figura 14 (Anexo A). Ressalta-se que múltiplos traçados
492 podem ser definidos dentro de um mesmo arquivo topográfico e eles aparecerão em uma lista
493 dentro da janela de gestão de traçados (Figura 15 - Anexo A), onde eles podem ser abertos,
494 excluídos ou duplicados.

495 As curvas horizontais podem ser criadas a partir do botão curvas (Figura 14 - Anexo A). O
496 complemento irá criar uma *layer* interna do traçado com curvas e mostrá-la. A janela de edição de
497 curvas aparecerá conforme a Figura 16 (Anexo A). Nessa interface podem ser definidas dois tipos
498 de curvas: circular simples e circular com transição simétrica. Os principais parâmetros das curvas
499 podem ser editados e a curva pode ser armazenada com o botão “salvar”. O botão “Gerar Todas”
500 gera todas as curvas para cada ponto de interseção usando os valores mínimos recomendados para o
501 desenvolvimento (D) arredondados para cima e sendo múltiplos da distância entre estacas. O último
502 passo é dado com o botão “Calcular” que importa a *layer* modificada e recalcula a tabela de
503 horizontais.

504 De posse do traçado horizontal é necessário calcular os dados de elevação para cada estaca
505 por meio de um arquivo *geotiff*. Isso é feito a partir do botão “Obter Cotas GeoTiff” (Figura 14)
506 onde deve-se selecionar esse arquivo. A coluna cotas da tabela será preenchida.

507 A partir do traçado horizontal e suas respectivas cotas, o complemento permite a criação do
508 perfil longitudinal e as curvas verticais por meio do botão “perfil de trecho”. Por padrão é traçado
509 um greide que segue a mediana do terreno, os vértices das extremidades têm suas posições
510 horizontais fixadas de acordo com o início e término do perfil horizontal e o terreno também está
511 plotado. Conforme pode ser observado na Figura 17 (Anexo A), os círculos azuis representam as
512 estacas, com o intuito de facilitar a determinação do traçado vertical ideal. O greide pode ser
513 editado adicionando vértices com o clique do botão esquerdo do mouse e arrastando-os para a
514 posição desejada. A rampa de maior inclinação do trecho fica sempre sublinhada de vermelho e a
515 velocidade recomendada para essa inclinação mostrada na parte superior.

516 Podem ser inseridos valores para as rampas quando clicadas com o botão direito. As curvas
517 podem ser inseridas ao se clicar com o botão direito sobre um vértice e acessar a “opção editar
518 vértice” conforme mostra Figura 18. Essa interface apresenta todos os parâmetros da curva e das
519 rampas adjacentes. “L utilizado” é o valor do comprimento da curva em metros que pode ser

520 editado e visualizado em tempo real assim como na interface de curvas horizontais. Nessa interface
521 a escala do desenho pode ser alterado ao se arrastar o mouse segurando o botão direito. O botão
522 zoom retorna a imagem ao zoom e escala originais a qual inserem todo o trecho na tela. O último
523 passo é usar o botão “Aplicar” ou “Fechar” e salvar o perfil longitudinal.

524 Em seguida pode-se retornar à janela da Figura 14 (Anexo A) e usar o botão “Abrir
525 verticais” que irá calcular a tabela de verticais e abrir a janela da Figura 19 (Anexo A). Nessa janela
526 pode ser calculada também a tabela de interseções, necessária para definir a sessão transversal que é
527 acessada através do botão “Definir sessão tipo” através da janela mostrada pela Figura 21. Na
528 primeira vez o complemento irá calcular as sessões do terreno para cada estaca. As sessões tipo
529 podem ser aplicadas para diferentes trechos de estacas pelo botão “Aplicar”. Elas também podem
530 ser extraídas ou importadas usando o formato dxf com os botões “Importar” e “Exportar”. O
531 volume total de corte e aterro pode ser calculado com o botão “Volume”.

532 De volta à janela da Figura 19 (Anexo A), o diagrama de Bruckner pode ser calculado e o
533 resultado mostra é apresentado na Figura 24 (Anexo A) com uma interface semelhante à da sessão
534 transversal e perfil vertical. Pode ser selecionado o intervalo de estacas de interesse para calcular
535 uma curva de Bruckner com a distância de transporte editável conforme apresentado na Figura 25
536 (Anexo A).

537

538 **4.2 Validação do complemento**

539 Os resultados das tabelas e do cálculo de volumes foram comparados para um mesmo
540 traçado, greide e sessão transversal entre o complemento e o Bentley TOPOgraph. Os resultados das
541 tabelas foram idênticos.

542 Os testes realizados com o complemento demonstraram que o mesmo configura um grande
543 avanço na área, pois permite a criação de traçados que utilizam como referência arquivos vetoriais
544 disponibilizados pelo IBGE e imagens de satélite. Esse fato agiliza o trabalho, pois a partir das
545 imagens têm-se uma noção mais fidedigna dos usos do solo existentes na área de projeto.

546 Ressalta-se que o complemento foi aplicado pelos estudantes da disciplina de projeto
547 geométrico de estradas no período 2019 II, do curso de Engenharia Civil. Foi proposto à eles que
548 apliquem na prática o complemento validando os cálculos e indicando melhorias. Com isso foram
549 feitas várias correções na interface do complemento, além de se descobrir também alguns
550 problemas de lógica que já foram corrigidos.

551

552

553

554 **5. CONCLUSÃO**

555 Pode-se dizer que esse projeto tem sua originalidade e pioneirismo com relação ao
556 desenvolvimento de softwares desta natureza sob a licença livre com funcionalidades semelhantes
557 aos pagos. Com relação à interface, as mesmas foram inspiradas a partir dos sistemas mais
558 utilizados da área como o Bentley TOPOgraph, buscando-se assim ter uma curva de aprendizagem
559 mais rápida. Também buscou-se inovar em alguns aspectos que agilizem o processo de projeto por
560 meio de interfaces mais enxutas e com capacidade de resposta mais aguçada. Tudo com o intuito de
561 promover um fluxo de trabalho o mais direto possível, sem perder a capacidade de correções, mas
562 evitando resservidos.

563 O complemento possui código seguindo os padrões mais utilizados na engenharia de
564 software no mercado, facilitando a manutenção e inclusão de novas funcionalidades sob a demanda
565 necessária. Ressalta-se que devido à linguagem python ser de alto nível e interpretada, alguns
566 processos tais como o cálculo da sessão transversal e os cálculos volumétricos tem uma certa
567 lentidão no processamento devido sua profundidade de iteração.

568 Como é de costume quando se trata de software nunca há um verdadeiro fim do
569 desenvolvimento já que novas funcionalidades sempre podem ser adicionadas. A expectativa é que
570 o projeto continue a ser desenvolvido e que novas funções como o cálculo de superelevação e
571 superlargura seja possível em próximas versões.

572

573 **6. REFERÊNCIAS BIBLIOGRÁFICAS**

574 CAMPOS NETO, S.; PAMBOUKIAN, S. V. D.; CRAVEIRO, M. V.; BARROS, E. A. R. Estudo
575 comparativo de ferramentas computacionais que utilizam tecnologia BIM para desenvolvimento de
576 projetos de engenharia civil. In: XI Congresso Brasileiro de Educação em Engenharia, Belém.
577 **Anais...** Pará, 2012.

578

579 CNT - CONFEDERAÇÃO NACIONAL DO TRANSPORTE. **Boletim Estatístico**. Brasília, 2018.
580 Disponível em: <<http://www.cnt.org.br/Boletim/boletim-estatistico-cnt>>. Acesso em: 17 de maio
581 2018.

582

583 DNIT - DEPARTAMENTO NACIONAL DE INFRA-ESTRUTURA DE TRANSPORTES.

584 **Diretrizes básicas para elaboração de estudos e projetos rodoviários escopos**

585 **básicos/instruções de serviço**. Brasília, 2006. Disponível em <

586 http://www1.dnit.gov.br/arquivos_internet/ipr/ipr_new/manuais/diretrizes_basicas_instrucoes_servi
587 [cos.pdf](http://www1.dnit.gov.br/arquivos_internet/ipr/ipr_new/manuais/diretrizes_basicas_instrucoes_servi)>. Acesso em: 11 nov. 2015.

588
589 GUIA LIVRE. **Referência de Migração para Software Livre do Governo Federal**. Organizado
590 por Grupo de Trabalho Migração para Software Livre. Brasília, 2005.
591
592 MAIA, R. O.; SILVA, F. A.; PAZOTI, M. A.; ALMEIDA, L. L.; PEREIRA, D. R.
593 **Desenvolvimento de um dispositivo para apoio ao ensino de computação e robótica**. Revista
594 Colloquium Exactarum, v. 6, 2014.
595
596 MANGHI, G.; CAVALLINI, P.; NEVES, V. **Quantum GIS: Um desktop potente e amigável**.
597 Revista FOSSGIS, v. 2, 2011.
598
599 MENEZES, R. R. V.; SANTOS, A. P.; CARVALHO, C. A. B.; FERRAZ, A. S. **Emprego dos**
600 **softwares TCP-MDT e topoGRAPH para projeto geométrico de estrada**. In: V Simpósio
601 Brasileiro de Ciências Geodésicas e Tecnologias da Geoinformação, Recife. **Anais...** Pernambuco,
602 2014.
603
604 OLIVEIRA JUNIOR, M. G. **Projeto geométrico de vias com modelagem digital do terreno em**
605 **ambiente de software livre e sistema de informações geográficas**. 2007. 186 f.. Dissertação
606 (Mestrado em Engenharia de Transportes) - Instituto Militar de Engenharia, 2007.
607
608 OSGEO (org.). **Descubra o QGIS: QGIS - A liderança do SIG de código aberto**. [S. l.], 20--.
609 Disponível em: https://QGIS.org/pt_BR/site/about/index.html. Acesso em: 11 nov. 2019.
610
611 PONTES FILHO, G. **Estrada de rodagem - projeto geométrico**. São Paulo, Editora Eletrônica,
612 1998.
613
614 SILVA JÚNIOR, J. A. D. **Metodologia para a utilização de dados topográficos em projetos**
615 **geométricos automatizados de vias**. 2003. 144 f.. Dissertação (Mestrado em Engenharia de
616 Transportes) - Instituto Militar de Engenharia, 2003.
617
618 VASCONCELOS, G. M. & BASSO, L. **Sistemas de Transportes Brasileiros: Origens, Estado**
619 **Atual e Desafios**. Revista Jovens Pesquisadores Ano V, v. 8, 2008.

ANEXO A - FIGURAS

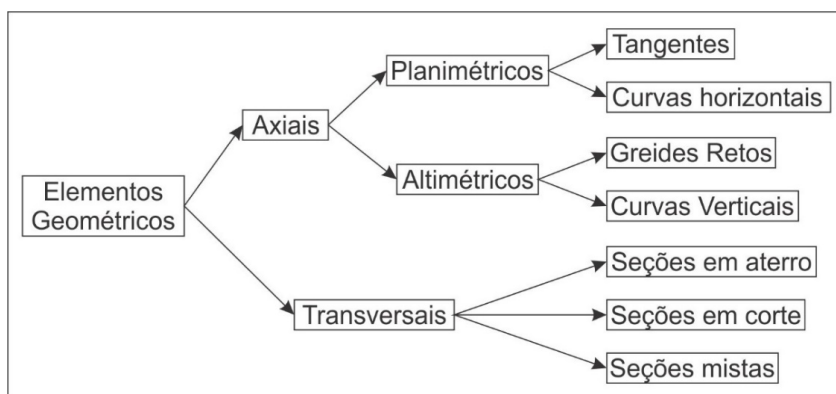


Figura 1: Elementos geométricos de uma estrada.

Fonte: Pontes Filho (1998, p. 30).

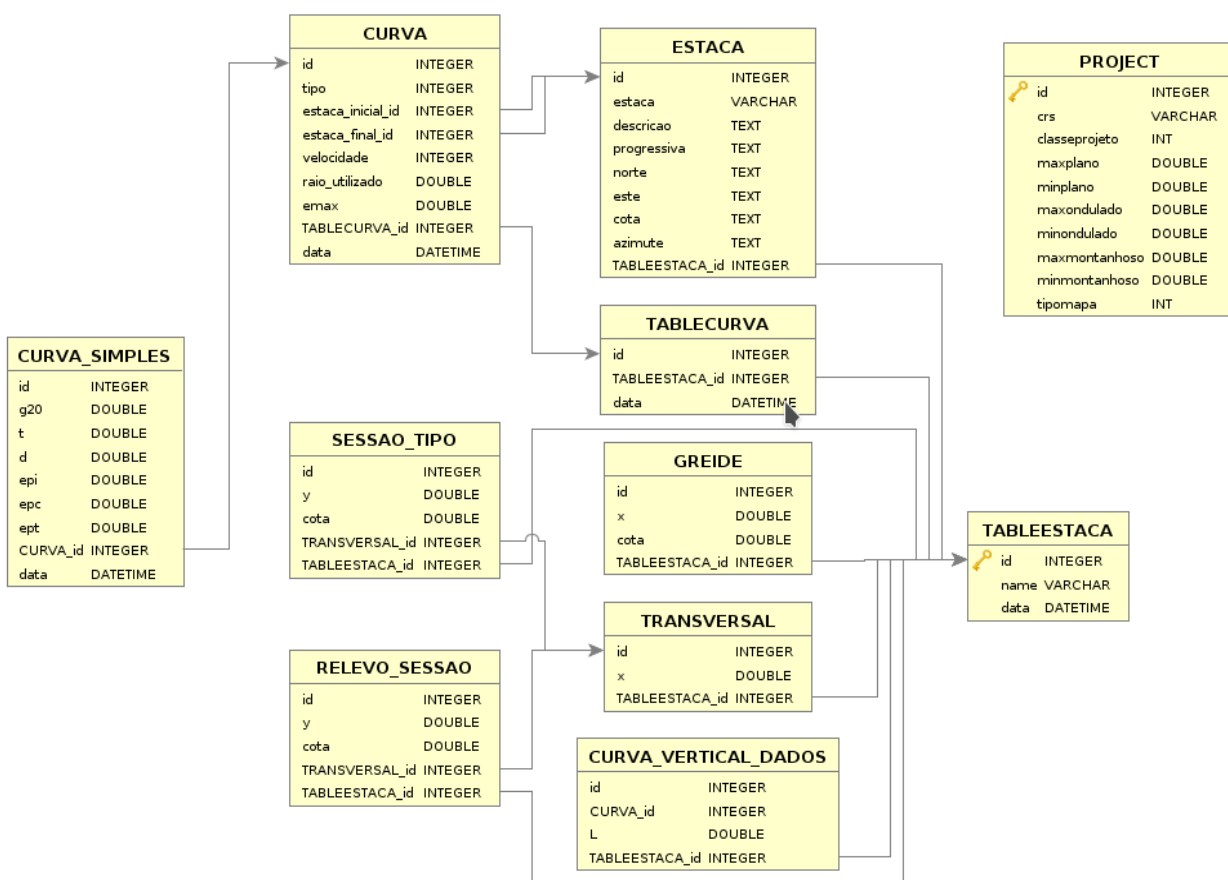


Figura 2: Diagrama entidade relacionamento para os tipos de dados do complemento.



Figura 3: Árvore de arquivos dentro do arquivo compactado de projeto.

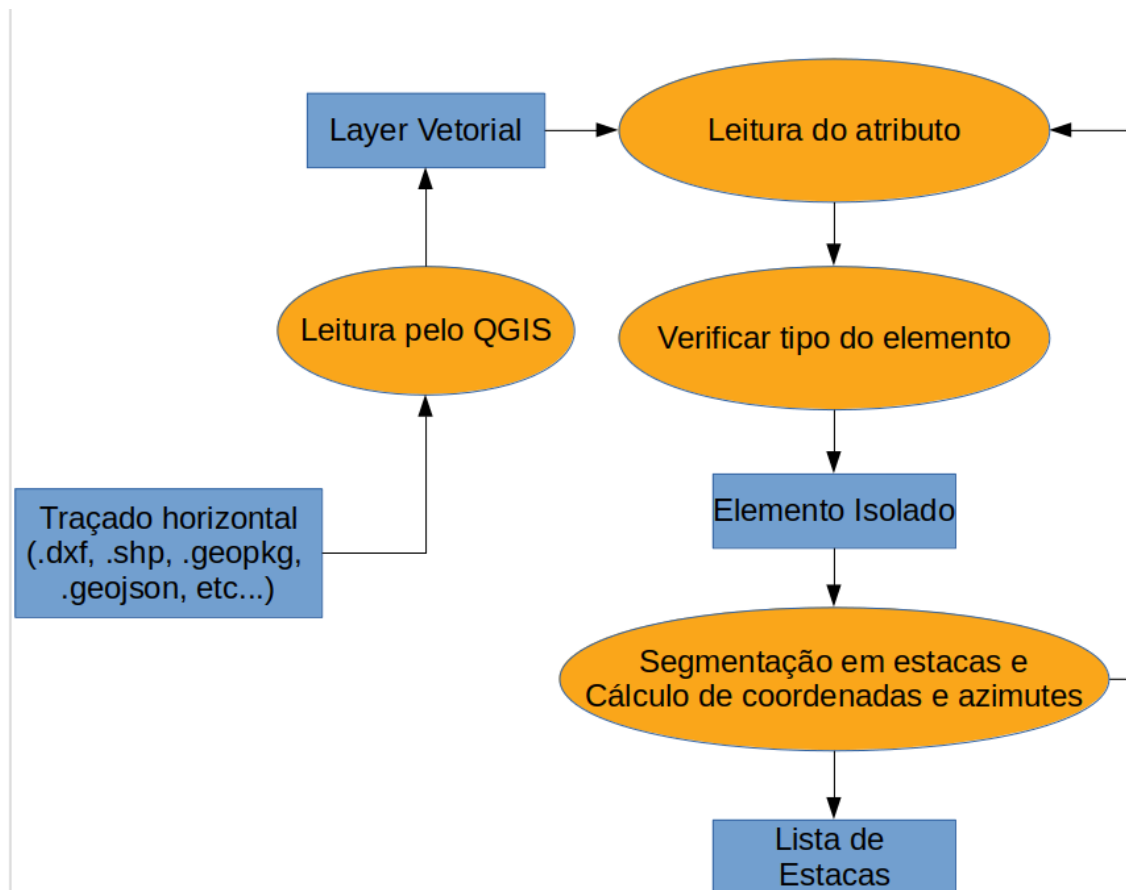


Figura 4: Algoritmo de segmentação do traçado horizontal a partir de um desenho

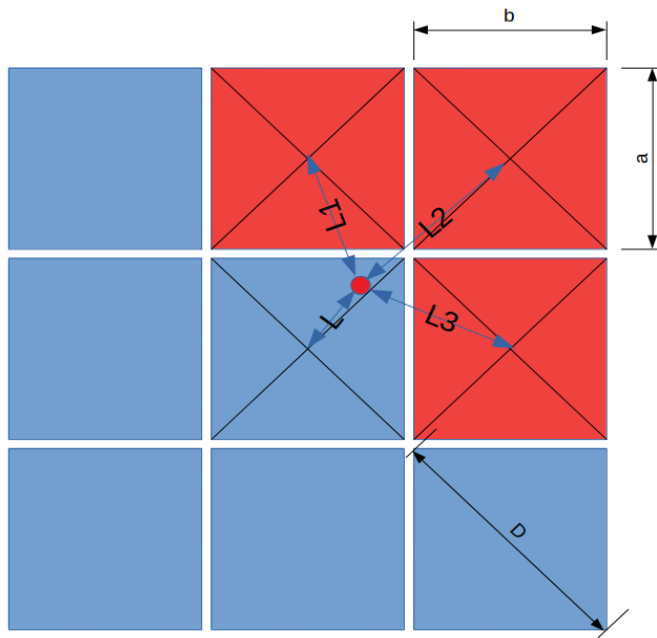


Figura 5: Interpolação de *rasters*. Cada quadro representa um pixel. Os pixels em vermelho são os escolhidos para a interpolação da elevação do ponto em vermelho.

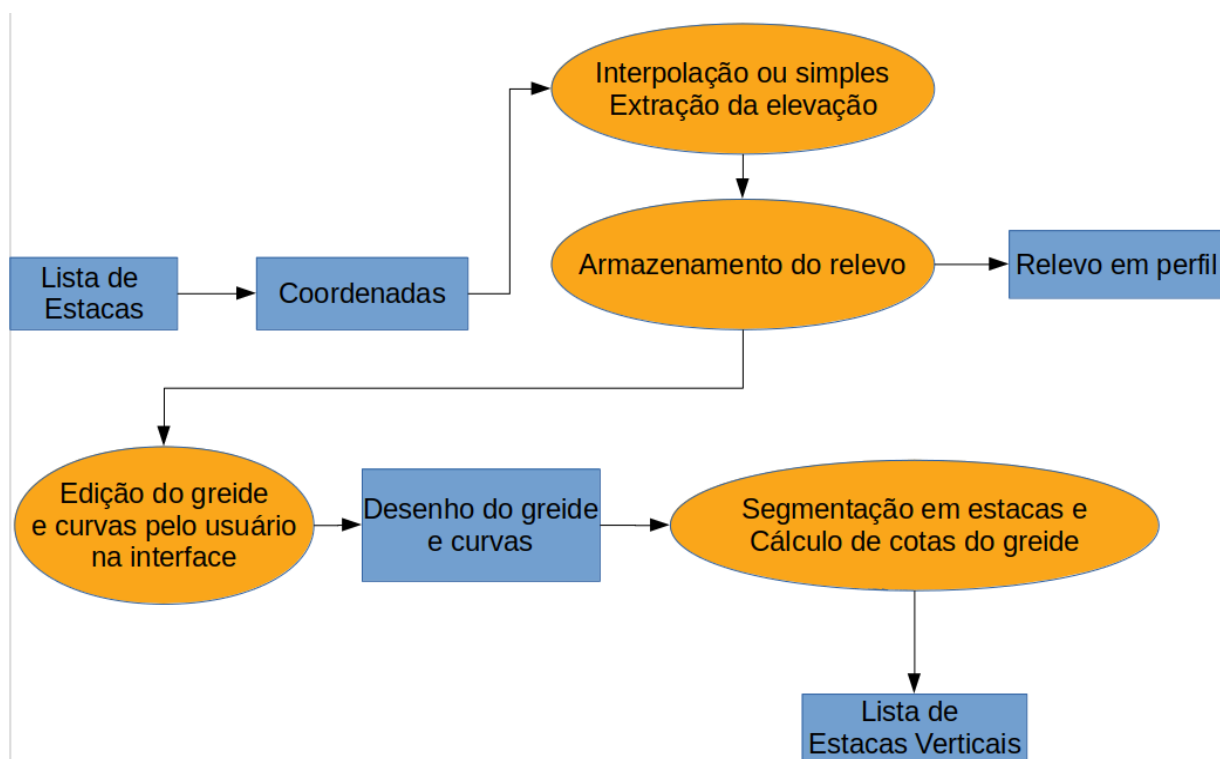


Figura 6: Processo de definição do perfil vertical.

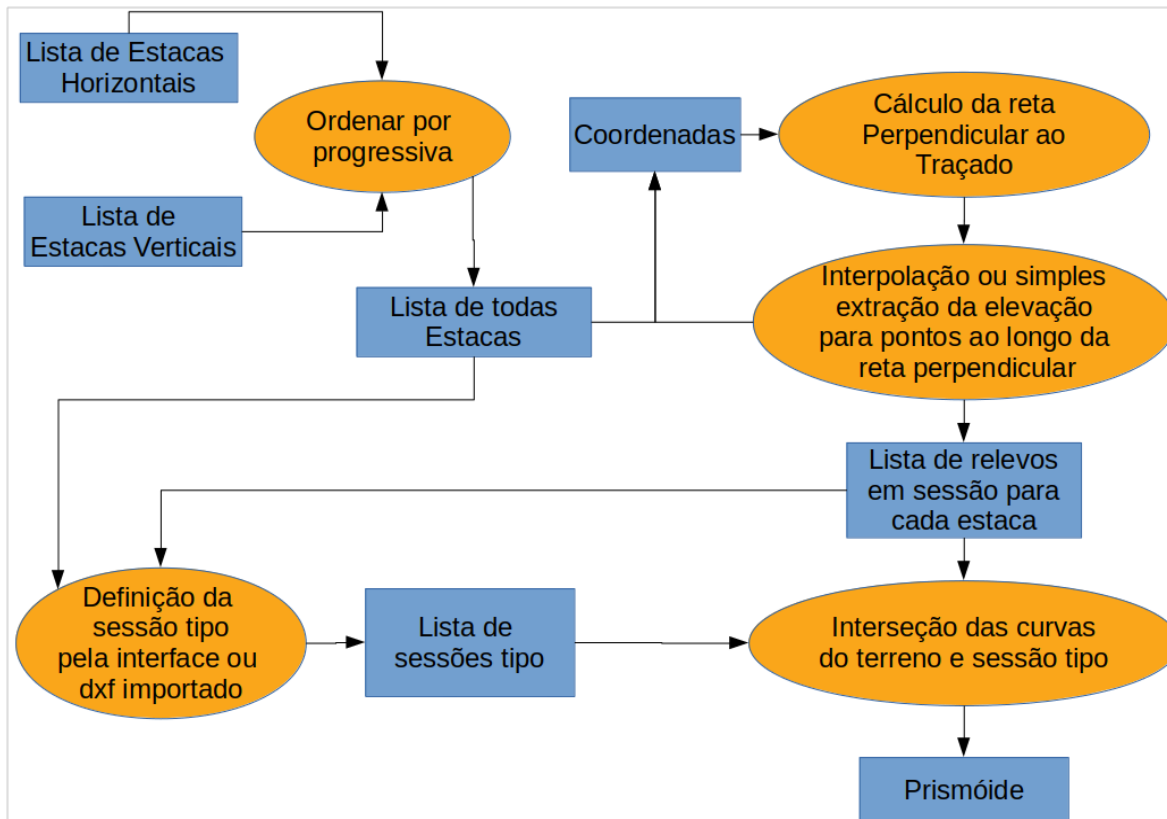


Figura 8: Combinação das estacas verticais e horizontais na tabela de interseções para a definição das sessões transversais.

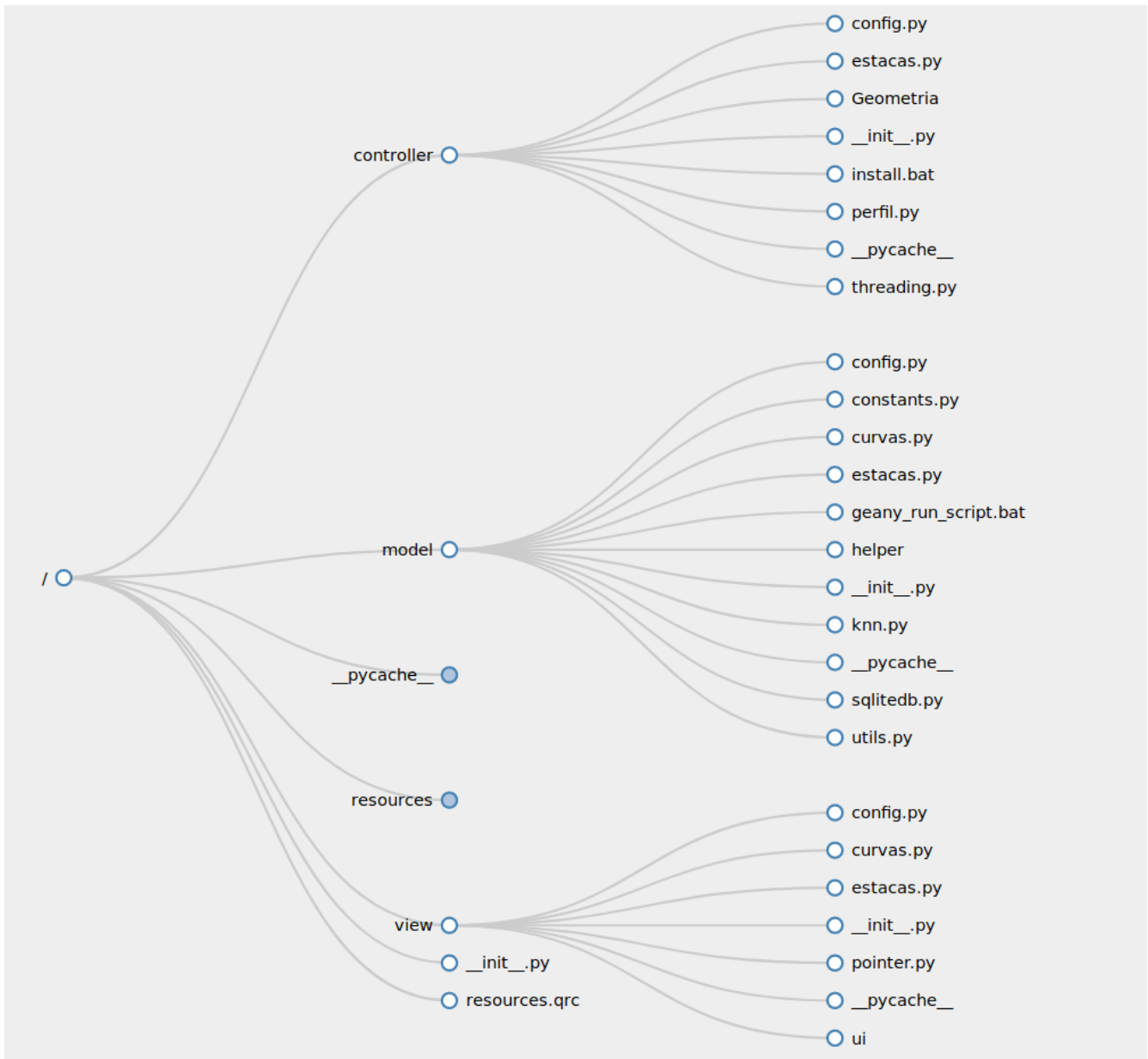


Figura 9: Árvore de arquivos do código fonte.

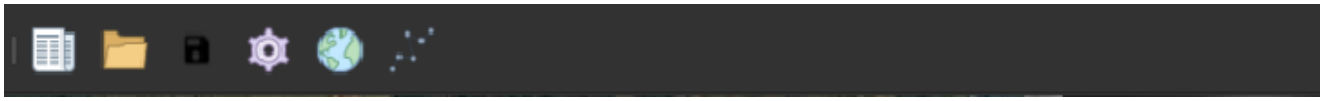


Figura 10: Interface e acesso do complemento desenvolvido dentro da barra de ferramentas do QGIS.

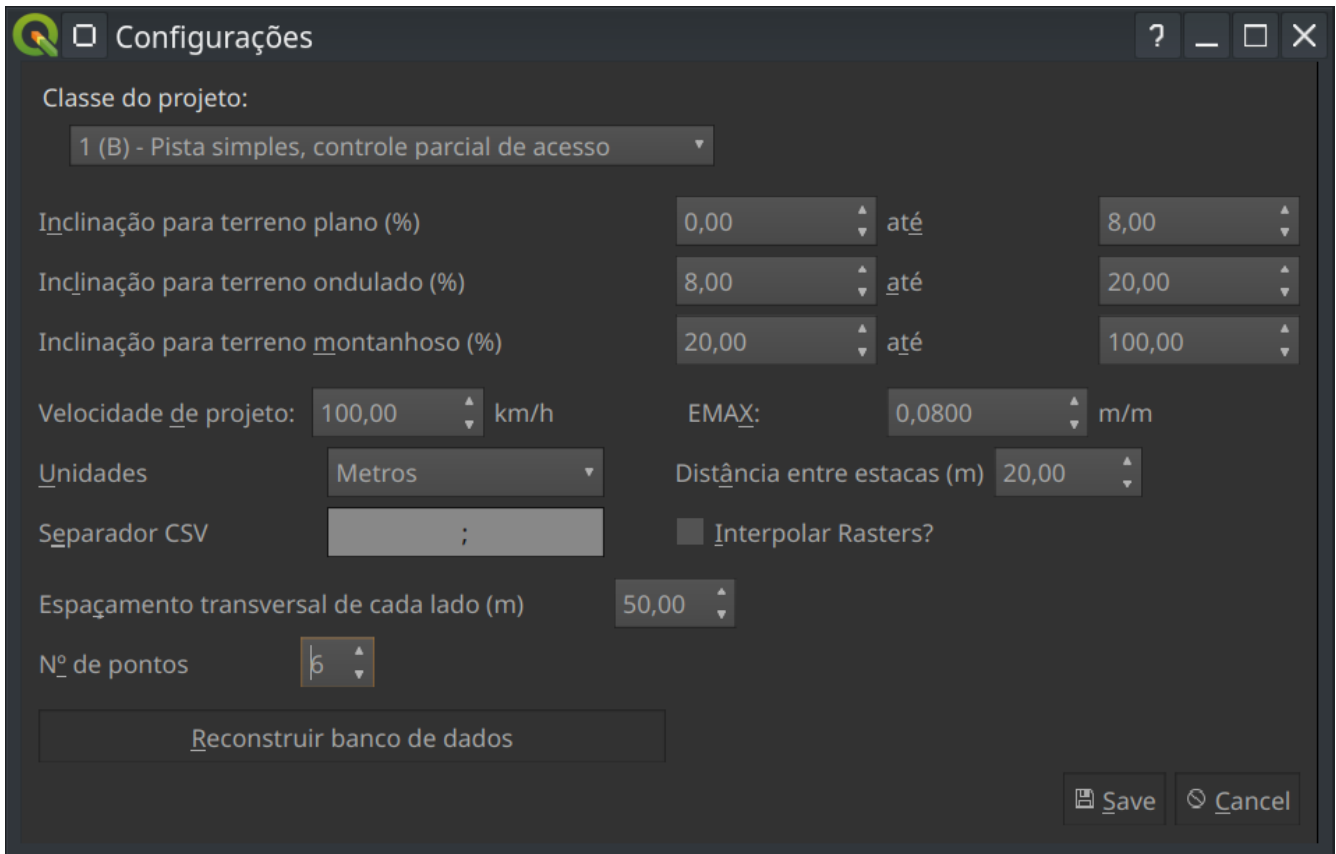


Figura 11: Janela de configurações do projeto.

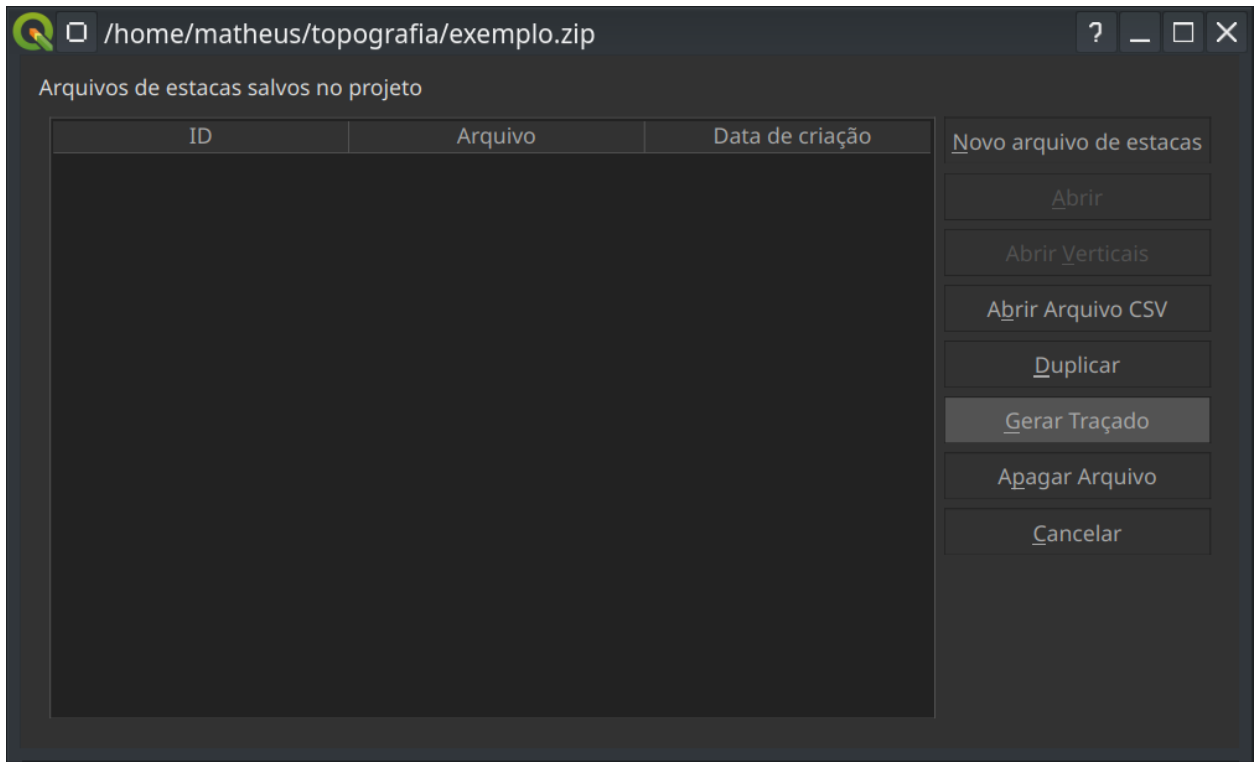


Figura 12: Janela para manipulação de traçados. O botão destacado é um atalho para criar uma nova layer no QGIS que irá servir como traçado horizontal.

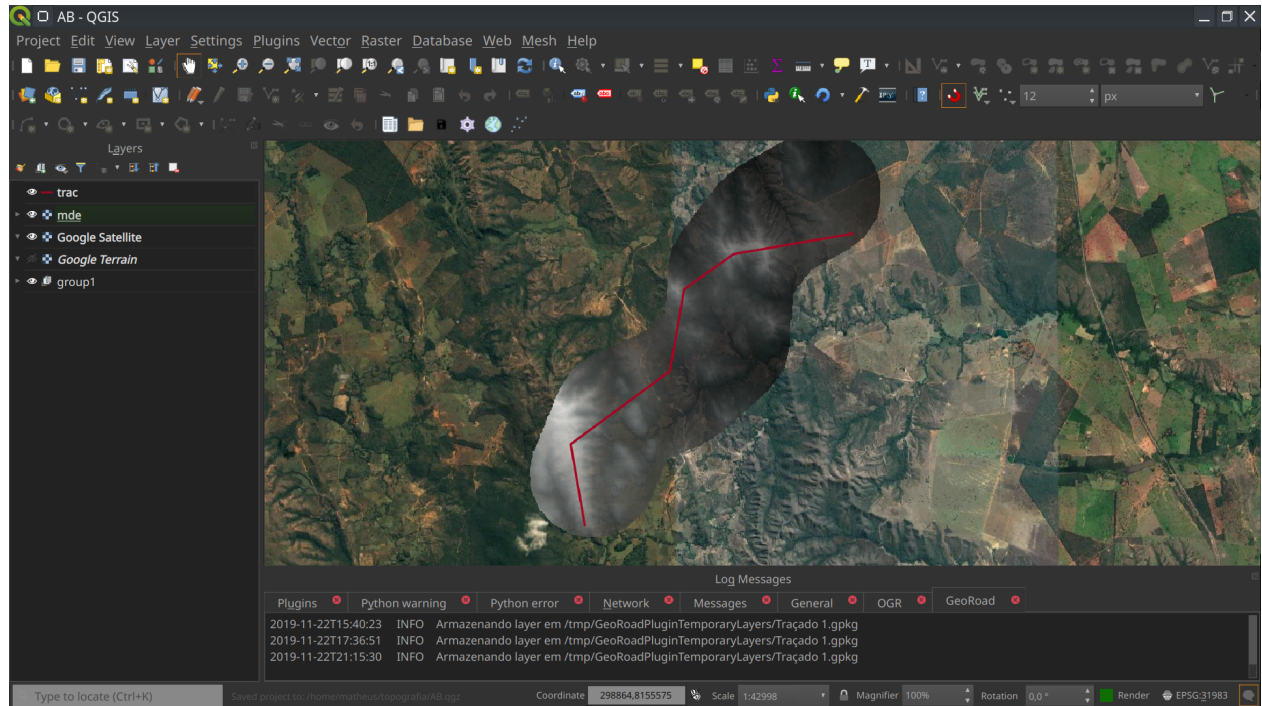


Figura 13: Exemplo meramente ilustrativo de um traçado horizontal com as tangentes que um usuário pode importar como uma layer do QGIS.

Traçado 1: Horizontal

	Estaca	Descrição	Progressiva	Norte	Este	Cota	Azimute
42	41		820	8155043.162	295262.7616	0	350.1342
43	42		840	8155062.86...	295259.3348	0	350.1342
44	43		860	8155082.57...	295255.908	0	350.1342
45	44		880	8155102.27...	295252.4811	0	350.1342
46	45		900	8155121.979	295249.0543	0	350.1342
47	46		920	8155141.68...	295245.6275	0	350.1342
48	47		940	8155161.38...	295242.2007	0	350.1342
49	48		960	8155181.09...	295238.7739	0	350.1342
50	49		980	8155200.79...	295235.347	0	350.1342
51	50		1000	8155220.50...	295231.9202	0	350.1342
52	51		1020	8155240.20...	295228.4934	0	350.1342
53	52		1040	8155259.90...	295225.0666	0	350.1342
54	53		1060	8155279.61...	295221.6397	0	350.1342
55	54		1080	8155299.31...	295218.2129	0	350.1342
56	55		1100	8155319.02...	295214.7861	0	350.1342
57	56		1120	8155338.72...	295211.3593	0	350.1342
58	57		1140	8155358.42...	295207.9324	0	350.1342
59	58		1160	8155378.134	295204.5056	0	350.1342
60	59		1180	8155397.83...	295201.0788	0	350.1342

Recalcular Estacas

Plotar

Curvas

Perfil de trecho

Salvar

Salvar em CSV

Obter Cotas via GeoTIFF

Obter Cotas via DXF

Procurar:

PT0

Figura 14: Janela com a tabela de estacas horizontais

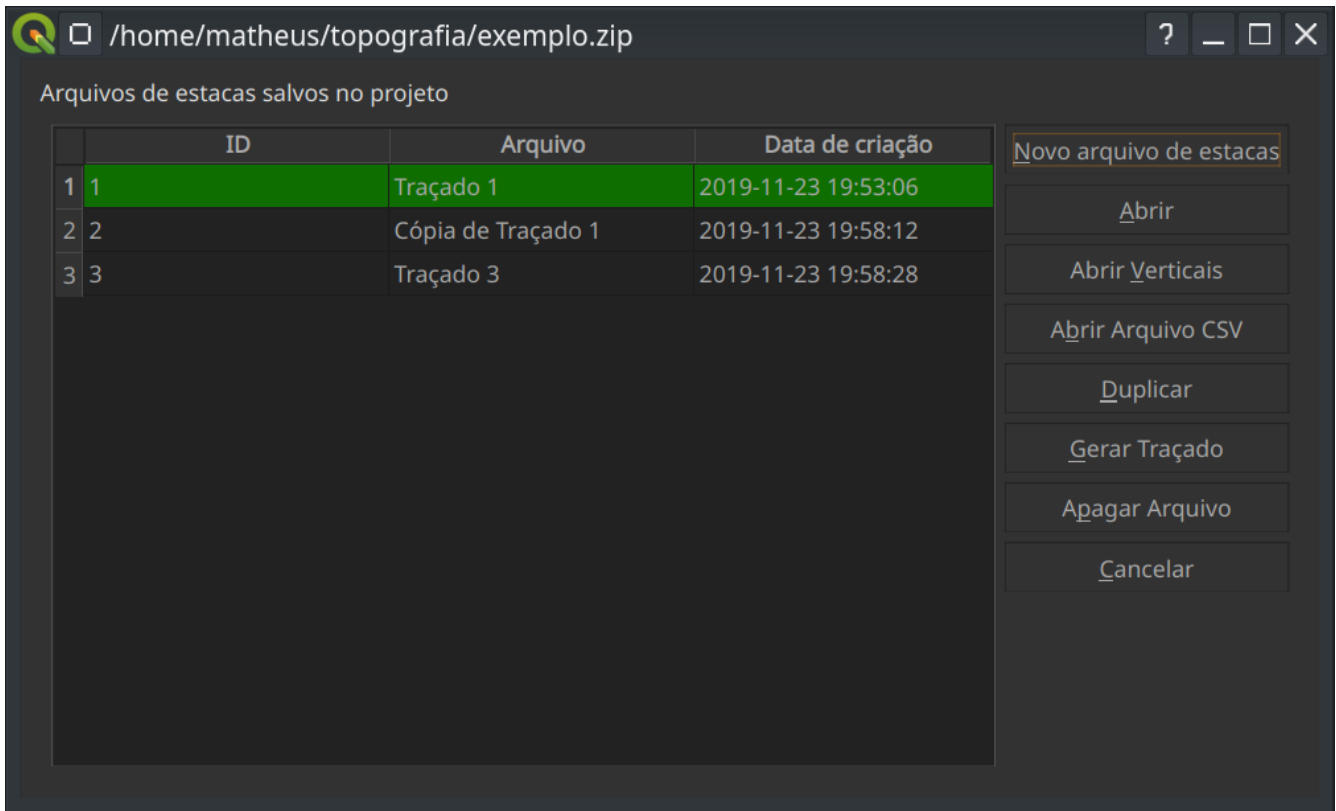


Figura 15: Gerenciamento de arquivos de projeto com múltiplos traçados.

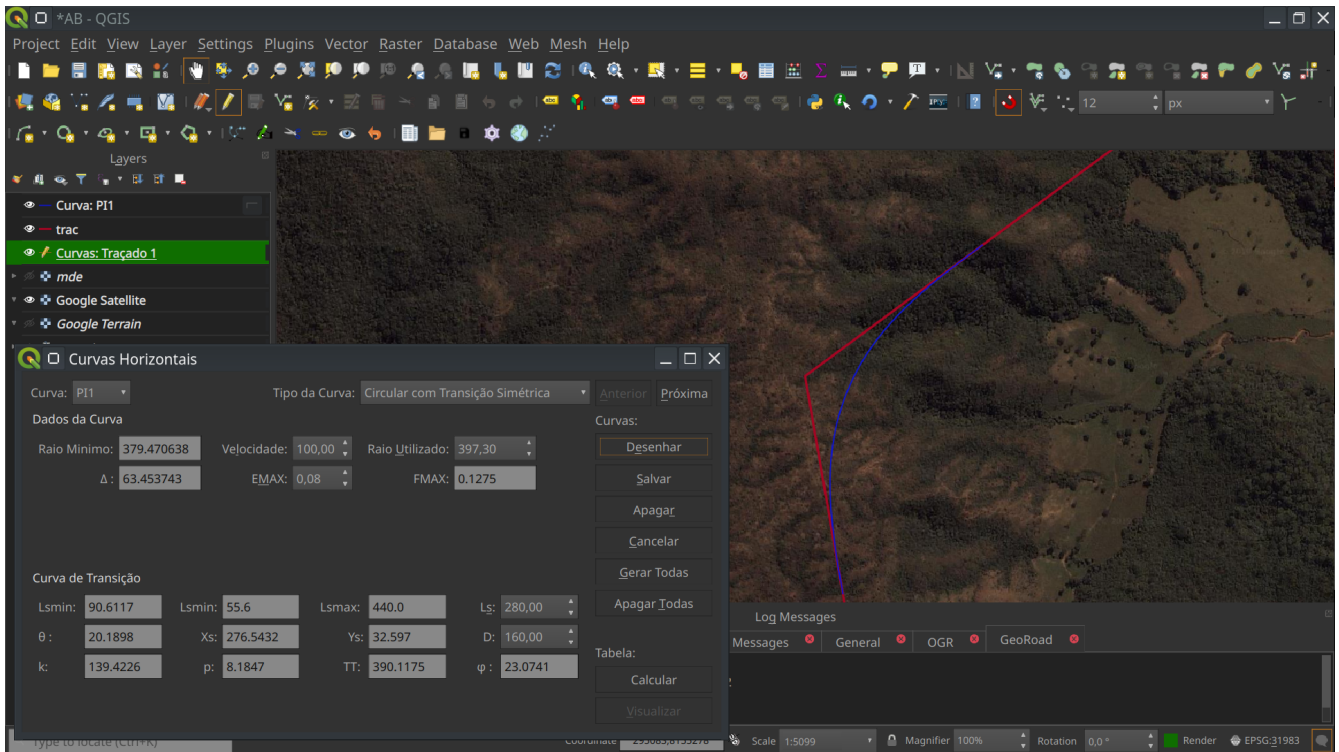


Figura 16: Exemplo de curva horizontal.

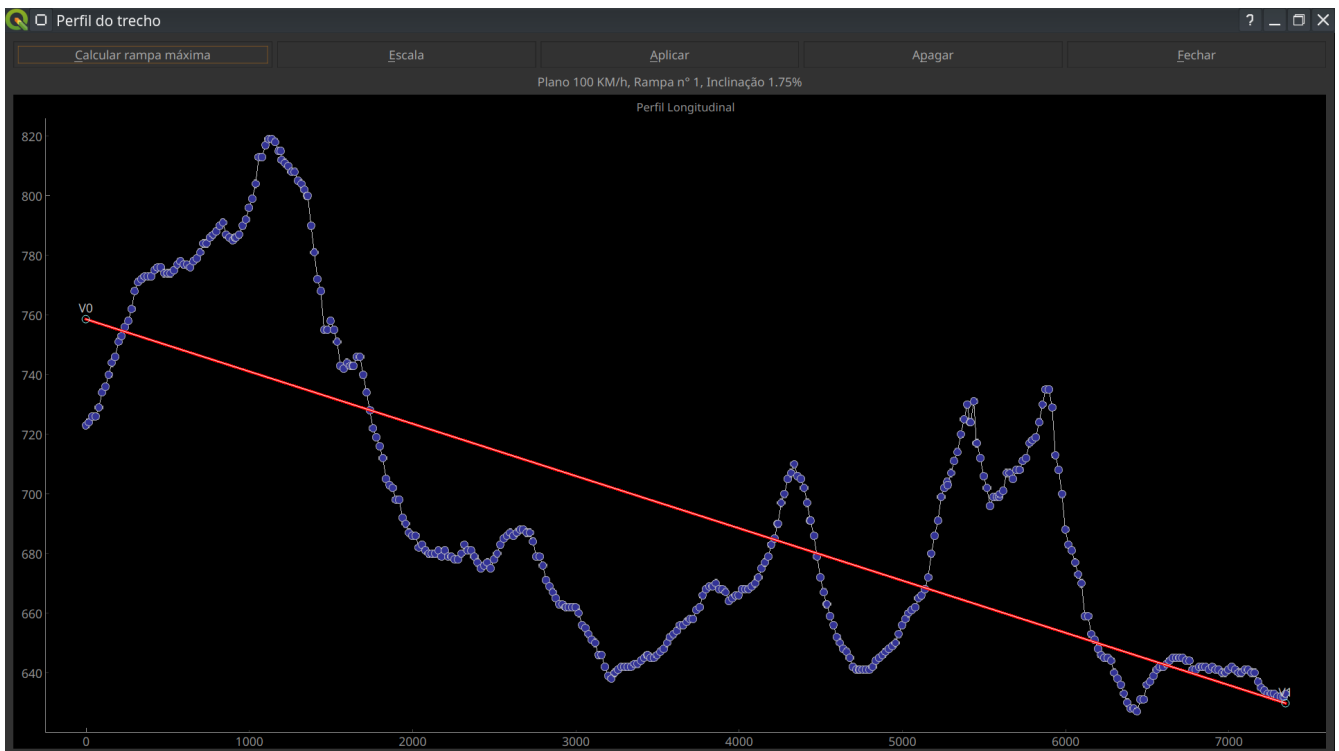


Figura 17: Interface para edição do greide. Os pontos azuis representam estacas horizontais no terreno, a linha em vermelho é o greide. O eixo das abscissas representa a progressiva em metros.

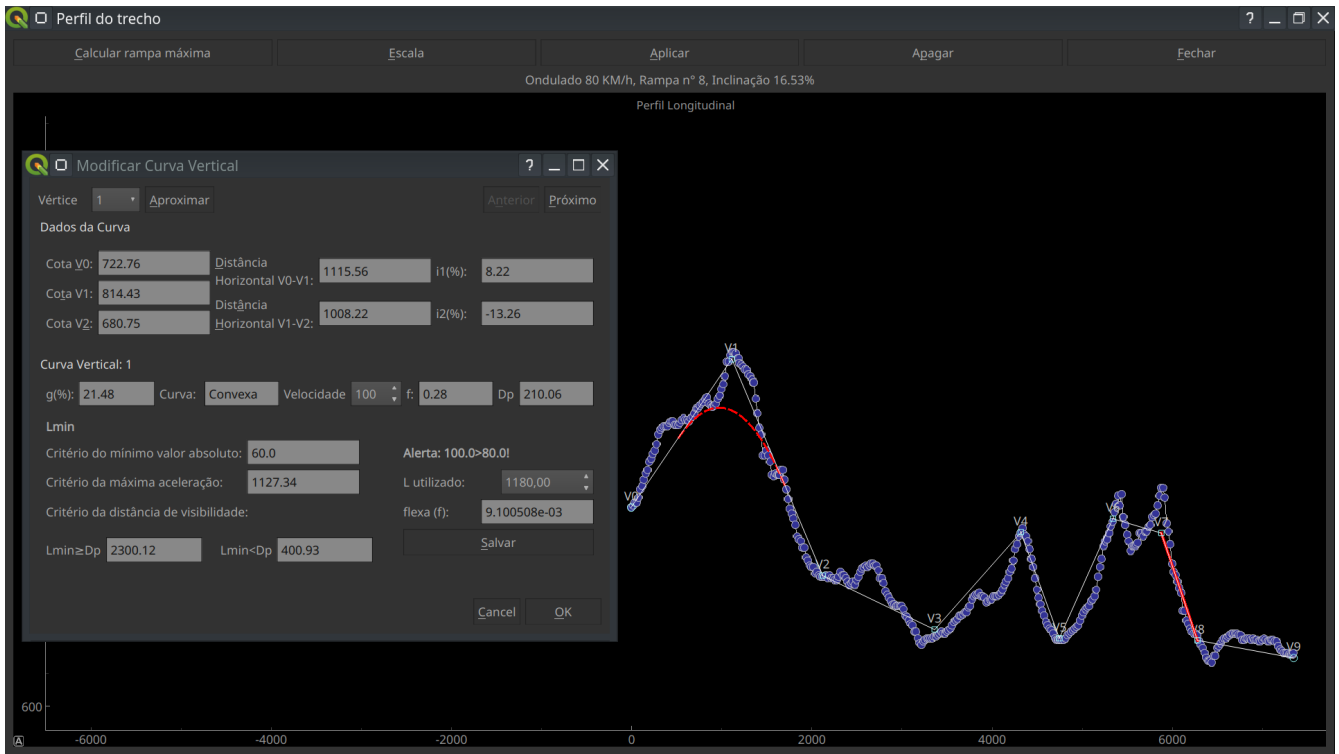


Figura 18: Definição de curvas verticais.

Traçado 1: Vertical

	Estaca	Descrição	Progressiva	Greide
1	0	V0	0	722.7643
2	1		20	724.4078
3	2		40	726.0513
4	3		60	727.6947
5	4		80	729.3382
6	5		100	730.9817
7	6		120	732.6252
8	7		140	734.2687
9	8		160	735.9121
10	9		180	737.5556
11	10		200	739.1991
12	11		220	740.8426
13	12		240	742.4861
14	13		260	744.1295
15	14		280	745.773

Recalcular

Tabela de Interseção

Definir Sessão Tipo

Diagrama de Bruckner

Exportar DXF

Exportar CSV

Procurar:

V0

Comprimento total:

Figura 19: Tabela de estacas verticais.

Traçado 1: Estacas

	Estaca	Descrição	Progressiva	Norte	Este	Greide	Cota	Azimute
1	0	PI0 + V0	0	8154235.2...	295403.26...	722.7643	723.0	350.1342
2	1		20.0	8154254.9...	295399.83...	724.4078	724.0	350.1342
3	2		40.0	8154274.6...	295396.40...	726.0513	726.0	350.1342
4	3		60.0	8154294.4...	295392.98...	727.6947	726.0	350.1342
5	4		80.0	8154314.1...	295389.55...	729.3382	729.0	350.1342
6	5		100.0	8154333.8...	295386.12...	730.9817	734.0	350.1342
7	6		120.0	8154353.5...	295382.70...	732.6252	736.0	350.1342
8	7		140.0	8154373.2...	295379.27...	734.2687	740.0	350.1342
9	8		160.0	8154392.9...	295375.84...	735.9121	744.0	350.1342
10	9		180.0	8154412.6...	295372.42...	737.5556	746.0	350.1342
11	10		200.0	8154432.3...	295368.99...	739.1991	751.0	350.1342
12	11		220.0	8154452.0...	295365.56...	740.8426	753.0	350.1342
13	12		240.0	8154471.7...	295362.13...	742.4861	756.0	350.1342
14	13		260.0	8154491.4...	295358.71...	744.1295	758.0	350.1342
15	14		280.0	8154511.1...	295355.28...	745.773	762.0	350.1342

Recalcular

Tabela de verticais

Definir Sessão Tipo

Diagrama de Bruckner

Exportar DXF

Exportar CSV

Procurar:

PI0 + V0

Comprimento total:

7372.2336 m

Figura 20: Tabela de interseção apresentando a totalidade das estacas verticais e horizontais.

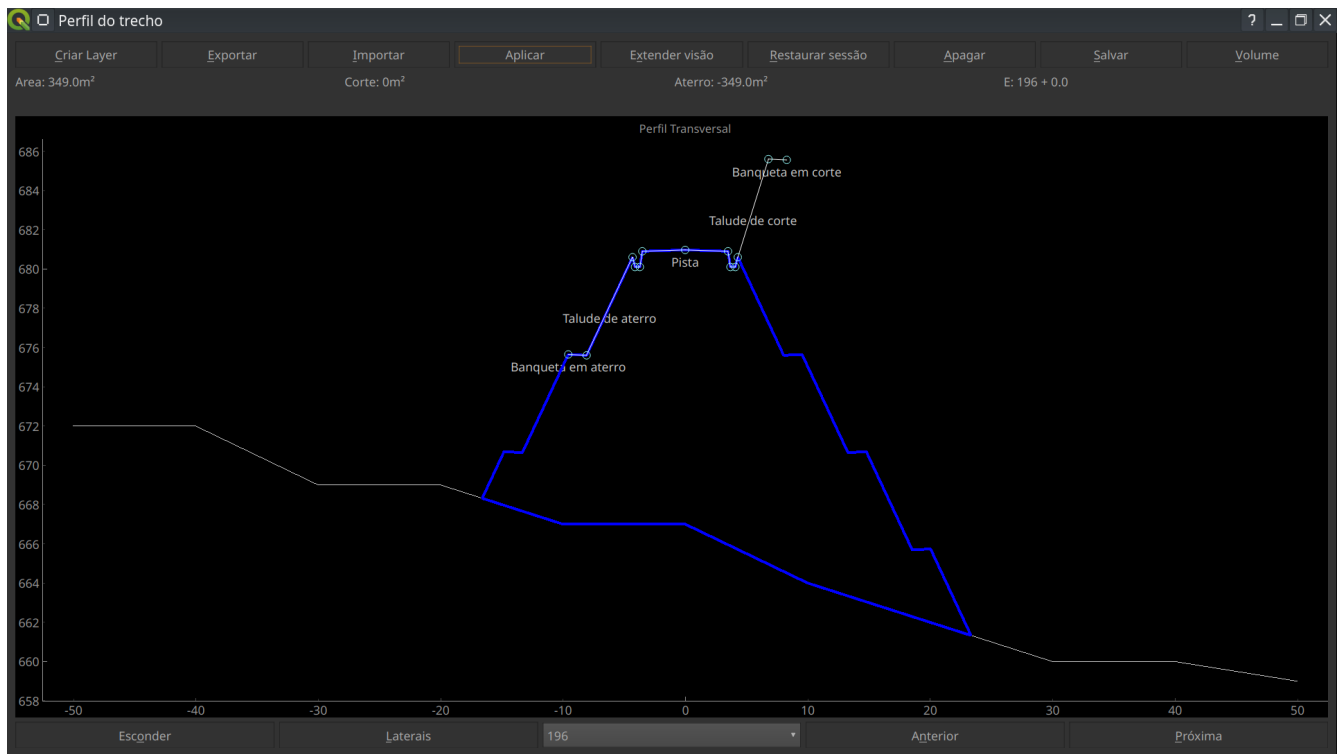


Figura 21: Interface de edição da sessão tipo.

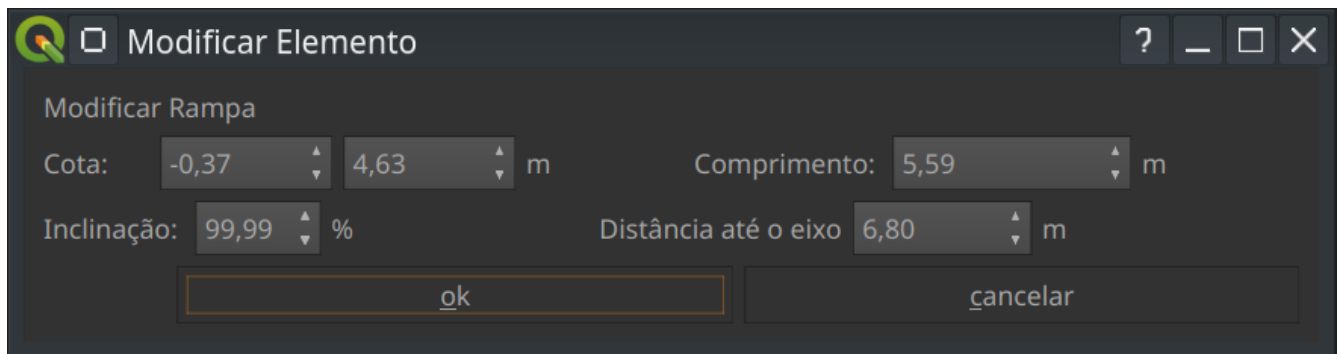


Figura 22: Janela de edição de rampas usada em ambas as interfaces de perfil vertical e sessão transversal.

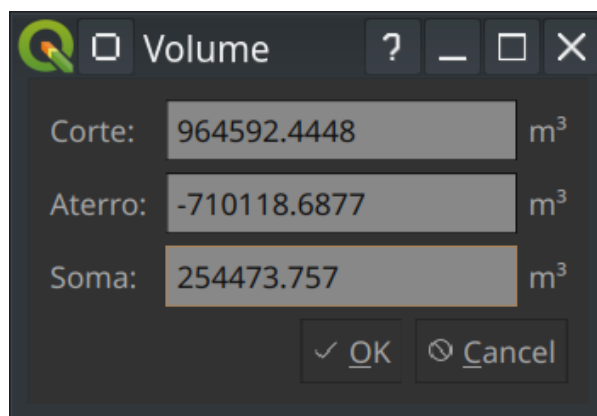


Figura 23: Resultados do cálculo de volume.

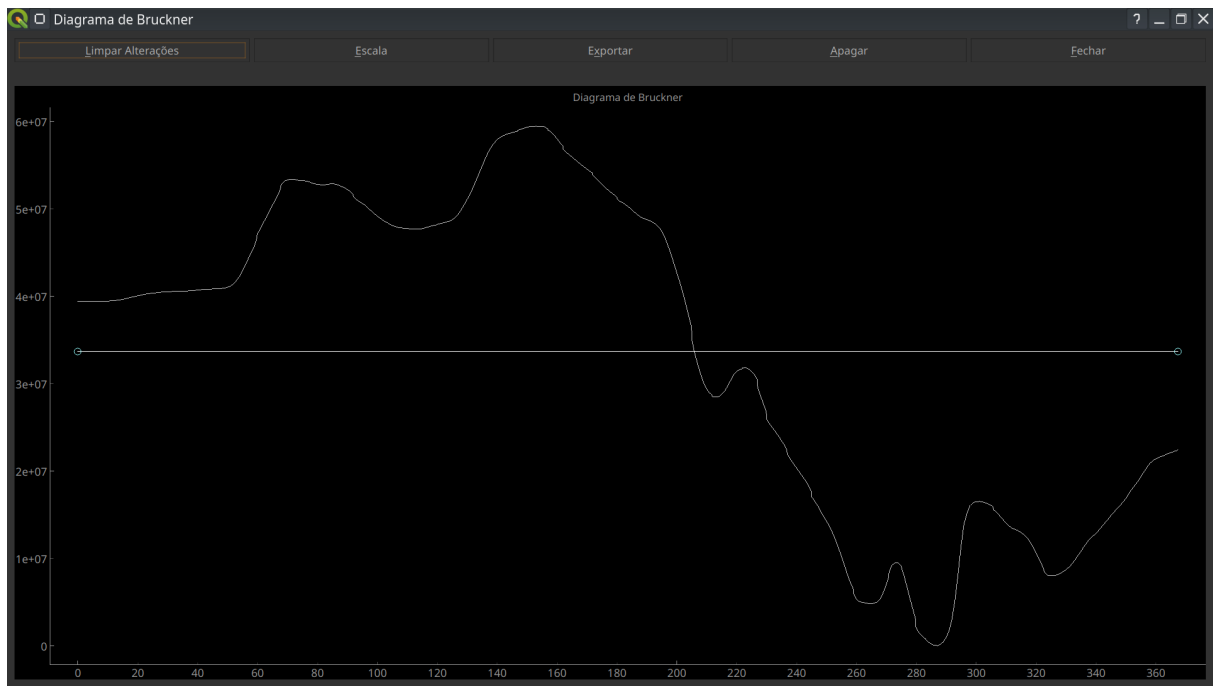


Figura 24: Diagrama de Bruckner para todo o trecho. As abscissas são as estacas e as ordenadas o volume em m^3 .

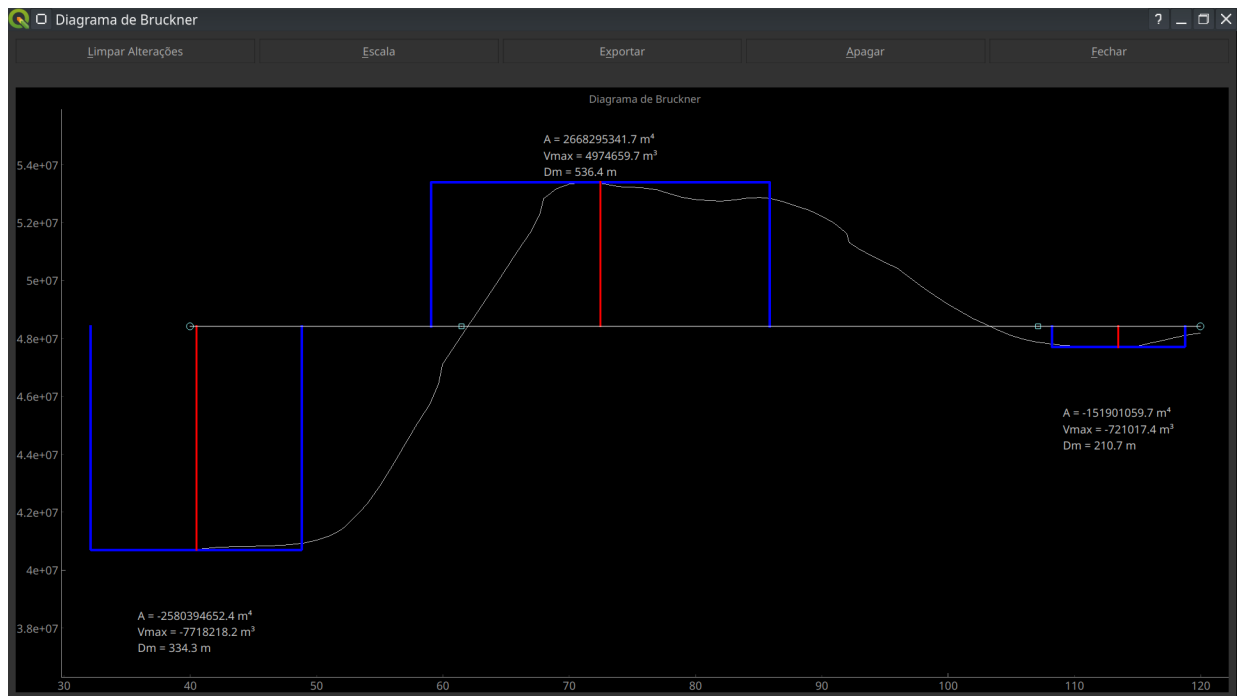


Figura 25: Curva de Bruckner e distância média.