

ROGÉRIO RAMOS RODRIGUES DO CARMO

***AVALIAÇÃO DA QUALIDADE DAS MEDIDAS E
POSICIONAMENTO GNSS EM SMARTPHONES ANDROID***



Dissertação de Mestrado

Presidente Prudente/SP

Março/2023



ROGÉRIO RAMOS RODRIGUES DO CARMO

**AVALIAÇÃO DA QUALIDADE DAS MEDIDAS E POSICIONAMENTO GNSS EM
SMARTPHONES ANDROID**

Dissertação apresentada ao Programa de Pós-Graduação em Ciências Cartográficas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de Presidente Prudente.

Orientador:

Dr. Milton Hirokazu Shimabukuro

Coorientador:

Dr. João Francisco Galera Monico

Presidente Prudente/SP

Março/2023

Carmo, Rogério Ramos Rodrigues do
Avaliação da qualidade das medidas e posicionamento GNSS em
smartphones Android / Rogério Ramos Rodrigues do Carmo. --
Presidente Prudente, 2023

187 f.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp),
Faculdade de Ciências e Tecnologia, Presidente Prudente

Orientador: Milton Hirokazu Shimabukuro

1. Geodesia. 2. Satélites artificiais em navegação. 3.
Smartphones.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da Faculdade de
Ciências e Tecnologia, Presidente Prudente. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

Impacto potencial desta pesquisa

O principal impacto potencial dessa pesquisa é a ampliação dos conhecimentos, no contexto nacional, sobre os receptores GNSS dos smartphones Android. Com isso, abrem-se possibilidades que se tenham avanços em aplicativos para o posicionamento e a navegação por satélite utilizando-se dados GNSS brutos capturados por meio de smartphones com tal capacidade.

Potential impact of this research

The main potential impact of this research is the expansion of knowledge, in the national context, about the GNSS receivers of Android smartphones. This opens up possibilities for advances in applications for satellite positioning and navigation using raw GNSS data captured through smartphones with such capability.



UNIVERSIDADE ESTADUAL PAULISTA

Câmpus de Presidente Prudente

CERTIFICADO DE APROVAÇÃO

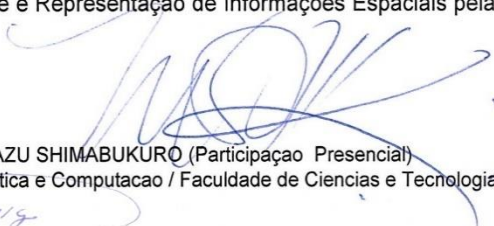
TÍTULO DA DISSERTAÇÃO: Avaliação da qualidade das medidas e posicionamento GNSS em smartphones Android

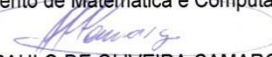
AUTOR: ROGÉRIO RAMOS RODRIGUES DO CARMO

ORIENTADOR: MILTON HIROKAZU SHIMABUKURO

COORIENTADOR: JOÃO FRANCISCO GALERA MONICO

Aprovado como parte das exigências para obtenção do Título de Mestre em Ciências Cartográficas, área: Aquisição, Análise e Representação de Informações Espaciais pela Comissão Examinadora:


Prof. Dr. MILTON HIROKAZU SHIMABUKURO (Participação Presencial)
Departamento de Matemática e Computação / Faculdade de Ciências e Tecnologia de Presidente Prudente


Prof. Dr. PAULO DE OLIVEIRA CAMARGO (Participação Presencial)
Departamento de Cartografia / Faculdade de Ciências e Tecnologia de Presidente Prudente

Prof. Dr. VINICIUS AMADEU STUANI PEREIRA (Participação Virtual)
UTFPR / Universidade Tecnológica Federal do Paraná

Presidente Prudente, 10 de março de 2023

DEDICATÓRIA

*Aos meus pais Aduino e Valdelice, que sempre
trataram a minha educação como prioridade e
permitiram que eu a alcançasse.*

AGRADECIMENTOS

Agradeço ao meu orientador Milton Hirokazu Shimabukuro, que há mais de seis anos me permite aprender sobre muitas coisas relacionadas à Cartografia, sem perder de vista as habilidades de Cientista da Computação adquiridas durante a graduação e no mercado de trabalho.

Ao meu coorientador João Francisco Galera Monico, pelos ensinamentos, críticas e sugestões que me fizeram evoluir meus conhecimentos em Geodésia e GNSS. Viabilizando o desenvolvimento desse trabalho.

Ao Programa de Pós-Graduação em Ciências Cartográficas (PPGCC), a Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP) e ao Instituto Nacional de Ciência e Tecnologia “Tecnologia GNSS no Suporte à Navegação Aérea” (INCT GNSS-NavAer, processos CNPq n° 465648/2014-2, FAPESP n° 2017/50115-0 e CAPES n° 88887.137186/2017-00), por proporcionar ambientes e ferramentas adequadas para o desenvolvimento desse projeto.

Aos membros da banca examinadora: Dr. Paulo de Oliveira Camargo/UNESP, Dr. Vinicius Amadeu Stuaní Pereira/UTFPR (Universidade Tecnológica Federal do Paraná), Dr. Paulo Sergio de Oliveira Junior/UFPR (Universidade Federal do Paraná) e Dr. Haroldo Antonio Marques/UNESP.

Aos membros do GEGE (Grupo de Estudos em Geodésia Espacial), pelo auxílio nos momentos mais difíceis.

Agradeço em especial aos amigos Weverton da Costa Silva e Ana Lucia Christovam de Souza, que contribuíram em muito nos momentos finais deste projeto. Agradeço também aos membros do INCT Ítalo Tsuchiya e Bruno César Vani que me auxiliaram durante todo o projeto.

Ao Engenheiro Cartógrafo Thiago Tiedtke dos Reis pela disponibilização do ortofotomosaico utilizado nesse projeto.

Aos amigos Bruno de Moura Morceli, Gustavo Grassi, Letícia Rosim Porto e Viviane Aparecida dos Santos.

Aos professores e funcionários da Pós-Graduação e da UNESP, que direta ou indiretamente, contribuíram para o desenvolvimento desse projeto.

Agradeço aos meus pais, Aduino e Valdelice, que fizeram o possível e impossível para que eu tivesse a oportunidade de completar os estudos. Afirmando, sem sombra de dúvidas, que o título obtido a partir desse projeto é muito mais deles do que meu.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

EPÍGRAFE

“Eu não luto achando que vou vencer, eu luto porque tenho que vencer, é por isso que continuo lutando e sempre continuarei”

Ichigo Kurosaki

RESUMO

Constelações de satélites vêm sendo cada vez mais utilizadas, nas últimas décadas, em aplicações de posicionamento e localização. Dentre as mais conhecidas estão a norte-americana *Global Positioning System* (GPS) e a russa *Global Navigation Satellite System* (GLONASS), sendo o GPS de amplo conhecimento da mídia em geral. É importante destacar que nos últimos anos houve um crescente aprimoramento nas constelações de satélites, como a adição da onda portadora L5 ao GPS e a implementação dos sistemas BeiDou e Galileo. Os receptores utilizados em solo que são capazes de coletar dados enviados por esses satélites são conhecidos como receptores do tipo *Global Navigation Satellite System* (GNSS). Em paralelo aos avanços das constelações de satélites e receptores GNSS, houve um aumento expressivo no número de usuários de dispositivos móveis, como no caso dos smartphones com o sistema operacional Android, maioria do mercado mundial. Estes dispositivos possuem receptores GNSS e utilizam recursos de localização em diversas aplicações, como nos aplicativos para entregas e mobilidade urbana. Um ponto de mudança importante nesse contexto ocorreu em 2016, com o lançamento da versão 7.0 do Android, conhecido como Android Nougat. A partir dessa versão é possível obter os dados GNSS brutos, sem processamentos, registrados pelos smartphones, desde que equipados com *chipsets* apropriados. Com isso, é possível obter as observáveis GNSS, amplamente utilizadas em aplicações de geolocalização, como medições de pseudodistância e fase da onda portadora. Os dados brutos podem ser utilizados nas aplicações Android ou compartilhados em formatos de arquivos comuns a softwares de processamento de dados GNSS, como no formato *Receiver INdependent EXchange Format* (RINEX). Esse trabalho possui como objeto de estudo os dados GNSS brutos provenientes dos smartphones Android. Nesse contexto, são discutidos nesse trabalho, os recursos de software e hardware para a captura, processamento e compartilhamento desses dados. Nesse trabalho é utilizado o método de Posicionamento por Ponto Simples (PPS), o método de cálculo de pseudodistâncias por tempo de recepção comum e técnicas relacionadas a bancos de dados em nuvem. Dessa forma, foram desenvolvidos neste trabalho, aplicativos Android e softwares desktop para coletar, armazenar, processar e efetuar o controle de qualidade de processamentos utilizando dados GNSS brutos provenientes de smartphones Android. Ao serem processados em softwares de referência, como no RTKPOST, verifica-se que tais dados podem atingir resultados similares àqueles de receptores GNSS comuns, uma vez tratadas as particularidades do processamento desses dados. Um dos smartphones utilizados nesse projeto foi o Huawei P30 Pro. Ao processar os dados GNSS brutos, registrados pelo Huawei P30 Pro no software desenvolvido nessa pesquisa, utilizando o método PPS, foi possível obter *Root Mean Square Error* (RMSE) 2D e 3D em torno de 20 e 22 metros, respectivamente. Além disso, ao utilizar o RTKPOST os valores de RMSE variaram entre 13 e 17 metros. Além disso, ao processar os dados no RTKPOST utilizando um tipo de método PPS aprimorado, foi possível obter erro horizontal inferior à 5 m.

Palavras-chave: Dados GNSS brutos. Android. Smartphone. Nougat.

ABSTRACT

Satellite constellations have been increasingly used in the past few decades in positioning and location applications. Among the best known are the American Global Positioning System (GPS) and the Russian Global Navigation Satellite System (GLONASS), the GPS being widely known in the media in general. It is important to highlight that in recent years there has been an increasing improvement in satellite constellations, such as the addition of the L5 carrier phase to GPS and the implementation of the BeiDou and Galileo systems. Receivers on the ground that are capable of collecting data sent by these satellites are known as receivers of the Global Navigation Satellite System (GNSS) type. In parallel with the advances in the constellations of satellites and GNSS receivers, there has been a significant increase in the number of users of mobile devices in recent years, as in the case of smartphones with the Android operating system, most of the world market. These devices have GNSS receivers and use location features in several applications, such as applications for delivery and urban mobility. An important turning point in this context occurred in 2016, with the launch of Android version 7.0, known as Android Nougat. Starting from this version of Android it is possible to obtain the raw GNSS data, unprocessed, captured by smartphones, provided as long as they are equipped with appropriate chipsets. Thus, it is possible to obtain observable GNSS, widely used in geolocation applications, such as pseudorange and carrier phase measurements. The raw data can be used in Android applications or shared in file formats common to GNSS data processing software, such as the Receiver INdependent EXchange Format (RINEX). This work has as its object of study the raw GNSS data from Android smartphones. In this context, the software and hardware resources for capturing, processing and sharing these data are discussed in this work. In this work, the Simple Point Positioning (SPP) method, the method of calculating pseudorange by common reception time and techniques related to cloud databases are applied. In this way, Android applications and desktop software were developed in this work to collect, store, process and perform quality control of processing using raw GNSS data from Android smartphones. When processed in reference software, such as RTKPOST, it appears that such data can achieve results similar to those of common GNSS receivers, once the particularities of processing such data are dealt with. One of the smartphones used in this project was the Huawei P30 Pro. When processing the raw GNSS data, registered by the Huawei P30 Pro in the software developed in this research, using the PPS method, it was possible to obtain 2D and 3D Root Mean Square Error (RMSE) around 20 and 22 meters, respectively. In addition, when using RTKPOST the RMSE values ranged between 13 and 17 meters. Furthermore, when processing the data in RTKPOST using an enhanced PPS type of method, it was possible to obtain horizontal error of less than 5 m.

Keywords: Raw GNSS Data. Android. Smartphone. Nougat.

LISTA DE FIGURAS

Figura 1: Número de usuários de smartphones Android em bilhões.....	22
Figura 2: Número de aplicativos disponíveis na Google Play Store.	23
Figura 3: Porcentagem dos usuários Android por versão.....	24
Figura 4: Tela principal do aplicativo GnssLogger.	33
Figura 5: Trecho de arquivo gerado pelo GnssLogger.	34
Figura 6: Tela principal do Gnss Analysis App.	35
Figura 7: Gráficos gerados pelo GNSS Analysis App.	36
Figura 8: Configurações do GNSS Analysis App.	37
Figura 9: Tela do GNSS Analysis App para comparação de arquivos.....	38
Figura 10: Gráfico com os valores de C/N0 para dois arquivos.....	38
Figura 11: Visualização do resultado do processamento no Gnss Analysis App.....	39
Figura 12: Tela inicial do Geo++ RINEX Logger.....	40
Figura 13: Tela inicial do GNSS Compare.....	41
Figura 14: Tela inicial do PPP WizLite.....	42
Figura 15: Princípio básico de posicionamento por satélite.....	43
Figura 16: Princípio da medida de pseudodistância.	44
Figura 17: Ilustração da observável de fase da onda portadora.....	46
Figura 18: Ilustração da ambiguidade da fase da onda portadora.	46
Figura 19: Mudanças no pacote Location da versão 23 para a versão 24 da API Android.....	55
Figura 20: Decodificação de mensagem de navegação GPS no código fonte do GnssLogger.....	57
Figura 21: Ilustração do conceito de A-GNSS.	59
Figura 22: Transmissão do código e dados pelo sistema A-GNSS.	59
Figura 23: Sistema de referência RSW.....	61
Figura 24: Trecho de mensagem SUPL para a solicitação de dados A-GNSS.	62
Figura 25: Mensagens RTCM obtidas via Ntrip em uma aplicação Java.	63
Figura 26: Efemérides transmitidas obtidas via servidor SUPL em uma aplicação Java.....	64
Figura 27: Exemplo de arquivo CSV.	67
Figura 28: Exemplo de arquivo CSV importado para um software de edição de planilhas.....	68
Figura 29: Exemplo de arquivo GPX no OpenStreet Map.	69
Figura 30: Trecho de arquivo GPX.	69
Figura 31: Trecho de arquivo KML para criação de uma linha.	70

Figura 32: Trecho de arquivo KML para criação de um conjunto de pontos.....	70
Figura 33: Arquivo KML visualizado no Google Earth.....	71
Figura 34: Trecho de arquivo RINEX de observação na versão 3.	73
Figura 35: Relação entre os conceitos de SaaS, PaaS e IaaS.	77
Figura 36: Efeito de <i>duty cycle</i> durante o tempo.	80
Figura 37: Opção para desativar o efeito de <i>duty cycle</i> via software no Android 9.0.	80
Figura 38: Representação dos centros de fase em L1 e L2.	81
Figura 39: Diagrama de blocos genérico para representar um receptor GNSS.....	83
Figura 40: Sincronização com o código C/A.....	85
Figura 41: Estrutura da mensagem de navegação GPS L1 C/A.....	85
Figura 42: Cálculo de pseudodistâncias através do método de tempo de recepção comum. ...	87
Figura 43: Exemplo em código Matlab para o cálculo de pseudodistâncias a partir de dados GNSS brutos capturados por smartphone Android.	93
Figura 44: Tela Principal do INCT-GNSS-APP.	97
Figura 45: Tela do INCT-GNSS-APP coletando dados brutos.	98
Figura 46: Trecho de arquivo de dados brutos gerado pelo INCT-GNSS-APP.....	98
Figura 47: Tela do INCT-GNSS-APP durante criação de arquivo RINEX na versão 2.11.	99
Figura 48: Trecho de arquivo RINEX gerado pelo INCT-GNSS-APP.....	100
Figura 49: Tela do INCT-GNSS-APP para a execução do PPS em tempo real.....	101
Figura 50: Telas de configurações do PPS em tempo real.	102
Figura 51: Tela do INCT-GNSS-APP executando o PPS em tempo real.	103
Figura 52: Arquivo gerado pelo INCT-GNSS-APP para compartilhamento.....	104
Figura 53: Trecho de arquivo CSV de resultados, gerado pelo INCT-GNSS-APP.....	105
Figura 54: Tela inicial do INCT-GNSS-APP Uploader.	106
Figura 55: Seleção de arquivo para upload no INCT-GNSS-APP Uploader.....	107
Figura 56: Arquivo armazenado no banco de dados em nuvem do Firestore.	108
Figura 57: Arquivo .zip gerado pelo INCT-GNSS-APP e armazenado no banco de dados em nuvem.	108
Figura 58: Interface gráfica do INCT-GNSS-APP Post Processing.	109
Figura 59: Trecho de arquivo de resíduos gerado pelo INCT-GNSS-APP Post Processing..	111
Figura 60: Trecho de arquivo de coordenadas gerado pelo INCT-GNSS-APP Post Processing.	112

Figura 61: Visualização dos arquivos GPX e KML gerados pelo INCT-GNSS-APP Post Processing.....	113
Figura 62: Trecho de arquivo de resultado de pós processamento gerado pelo INCT-GNSS-APP Post Processing.....	114
Figura 63: Interface gráfica do INCT-GNSS-APP Post Processing Analysis.	115
Figura 64: Arquivos criados pelo FCT-INCT-GNSS Post Processing Analysis.	115
Figura 65: Trecho de arquivo criado pelo FCT-INCT-GNSS Post Processing Analysis.....	116
Figura 66: Arquivo de RMSEs criado pelo INCT-GNSS-APP Post Processing Analysis. ...	117
Figura 67: Gráfico de discrepâncias interativo criado pelo INCT-GNSS-APP Post Processing Analysis, a) gráfico estático, b) gráfico com análise exploratória através do mouse e c) gráfico com seleção de área para aplicação de zoom.	117
Figura 68: Gráficos gerados pelo INCT-GNSS-APP Post Processing Analysis, a) gráfico do erro do relógio do receptor, b) gráfico de resíduos do ajustamento, c) gráfico do números de satélites utilizados, d) gráfico de discrepâncias e e) gráfico de precisões.	119
Figura 69: Fluxograma da interação entre os softwares do ecossistema INCT-GNSS-APP.	121
Figura 70: Imagens dos aparelhos Huawei P30 Pro a) e Xiaomi Mi 9 b).	123
Figura 71: Estrutura montada próxima a PRU2 para a coleta de dados.	124
Figura 72: Ortofotomosaico utilizado para visualização de resultados.	126
Figura 73: Círculos centrados na coordenada de referência da estação PRU2.	127
Figura 74: Coordenadas estimadas pelo INCT-GNSS-APP Post Processing.	130
Figura 75: Distâncias geodésicas em metros no RTKPOST.	133
Figura 76: Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).	134
Figura 77: Precisões obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).	135
Figura 78: Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing.	138
Figura 79: Distâncias geométricas, em metros, no RTKPOST.	140
Figura 80: Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).	142
Figura 81: Precisões obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).	143
Figura 82: Distâncias geométricas, em metros, no RTKPOST.	148
Figura 83: Detalhe das distâncias geométricas, em metros, no RTKPOST.	149

LISTA DE TABELAS

Tabela 1: Percentual de usuários Android por versão.	25
Tabela 2: Valores nominais dos principais erros no posicionamento absoluto GPS.....	48
Tabela 3: Smartphones que fornecem mensagem de navegação.....	51
Tabela 4: Smartphones que fornecem a fase da onda portadora.	52
Tabela 5: Smartphones que possuem L5.	53
Tabela 6: Smartphones que possuem mensagens de navegação, onda portadora e L5.	54
Tabela 7: Campos dos dados GNSS brutos no pacote Location.	56
Tabela 8: Ambiguidade de tempo com base no estado de rastreamento dos sinais GNSS.....	86
Tabela 9: Métodos da classe <i>GnssMeasurement</i> para calcular a fase da onda portadora.	94
Tabela 10: Especificações técnicas dos smartphones Huawei P30 Pro e Xiaomi Mi 9.	123
Tabela 11: Coordenadas de referência para a estação PRU2.	125
Tabela 12: Horários de coleta de dados com o smartphone Huawei P30 Pro.	127
Tabela 13: Horários de coleta de dados com o smartphone Xiaomi Mi 9.	128
Tabela 14: Discrepâncias em metros no INCT-GNSS-APP Post Processing.	129
Tabela 15: Precisões em metros no INCT-GNSS-APP Post Processing.	129
Tabela 16: Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.....	130
Tabela 17: Informações gerais do processamento no INCT-GNSS-APP Post Processing.	131
Tabela 18: Discrepâncias em metros no RTKPOST.	131
Tabela 19: Precisões em metros no RTKPOST.	132
Tabela 20: Distâncias geodésicas em metros no RTKPOST.....	132
Tabela 21: Informações gerais do processamento no RTKPOST.	133
Tabela 22: Huawei P30 Pro - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.	134
Tabela 23: Discrepâncias em metros no INCT-GNSS-APP Post Processing.	136
Tabela 24: Precisões em metros no INCT-GNSS-APP Post Processing.	137
Tabela 25: Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.....	137
Tabela 26: Informações gerais do processamento no INCT-GNSS-APP.	138
Tabela 27: Discrepâncias em metros no RTKPOST.	139
Tabela 28: Precisões em metros RTKPOST.	139
Tabela 29: Distâncias geodésicas em metros no RTKPOST.....	140
Tabela 30: Informações gerais do processamento no RTKPOST.	141

Tabela 31: Xiaomi Mi 9 - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.	141
Tabela 32: Huawei P30 Pro - Discrepâncias em metros no RTKPOST.....	147
Tabela 33: Huawei P30 Pro – Precisão em metros no RTKPOST.....	147
Tabela 34: Distâncias geodésicas em metros no RTKPOST.....	147
Tabela 35: Valores de RMSE para o PPP no RTKPOST.....	149

LISTA DE ABREVIATURAS E SIGLAS

ADC	- <i>Analogue-to-Digital Converter</i>
A-GNSS	- <i>Assisted-GNSS</i>
API	- <i>Application Program Interface</i>
AWS	- <i>Amazon Web Services</i>
BDS	- <i>BeiDou</i>
BNC	- <i>BKG Ntrip Client</i>
C/No	- <i>Carrier-to-Noise-density ratio</i>
CA	- <i>Coarse/Acquisition</i>
CNES	- <i>Centre National d'études Spatiales</i>
CRP	- <i>Carrier Phase Relative Positioning</i>
CSV	- <i>Comma-Separated Values</i>
DGNSS	- <i>Differential GNSS</i>
ESA	- <i>European Space Agency</i>
FCT	- <i>Faculdade de Ciências e Tecnologia</i>
GEGE	- <i>Grupo de Estudos em Geodésia Espacial</i>
GGTO	- <i>GPS to Galileo Time Offset</i>
GIS	- <i>Geographic Information System</i>
GLONASS	- <i>Globalnaya Navigatsionnaya Sputnikovaya Sistema</i>
GPS	- <i>Global Positioning System</i>
GPX	- <i>GPS eXchange Format</i>
GSD	- <i>Ground Sample Distance</i>
GSDC	- <i>Google Smartphone Decimeter Challenge</i>
HDOP	- <i>Horizontal Dilution of Precision</i>
HTML	- <i>Hypertext Markup Language</i>
HTTP	- <i>Hypertext Transfer Protocol</i>
IaaS	- <i>Infrastructure as a Service</i>
IGS	- <i>International GNSS Service</i>
IMU	- <i>Inertial Measurement Unit</i>

INCT GNSS-NavAer	- Instituto Nacional de Ciência e Tecnologia “Tecnologia GNSS no Suporte à Navegação Aérea”
IoT	- <i>Internet of Things</i>
IP	- <i>Internet Protocol</i>
ISB	- <i>InterSystemBias</i>
ISO	- <i>International Organization for Standardization</i>
JOSM	- <i>Java OpenStreetMap</i>
KML	- <i>Keyhole Markup Language</i>
MMQ	- Método dos Mínimos Quadrados
Nmea	- <i>National Marine Electronics Association</i>
Ntrip	- <i>Networked Transport of RTCM via Internet Protocol</i>
OGC	- <i>Open Geospatial Consortium</i>
OMA	- <i>Open Mobile Alliance</i>
PaaS	- <i>Platform as a Service</i>
PPP	- Posicionamento por Ponto Preciso
PPS	- Posicionamento por Ponto Simples
PRN	- <i>Pseudorandom Noise</i>
PVT	- Posição, Velocidade e Tempo
QZSS	- <i>Quasi-Zenith Satellite System</i>
RF	- <i>Radio Frequency</i>
RINEX	- <i>Receiver Independent EXchange Format</i>
RMS	- <i>Root Mean Square</i>
RMSE	- <i>Root Mean Square Error</i>
RSW	- <i>Radial, Along-track, Cross-track</i>
RTCM	- <i>Radio Technical Commission for Maritime Services</i>
RTS	- <i>Realtime IGS</i>
SaaS	- <i>Software as a Service</i>
SGBD	- Sistema Gerenciador de Banco de Dados
SGBDR	- Sistema Gerenciador de Banco de Dados Relacional
SIRGAS	- Sistema de Referência Geocêntrico para as Américas
SNR	- <i>Signal-to-Noise Ratio</i>
SQL	- <i>Structured Query Language</i>

SSR	- <i>State Space Representation</i>
sTEC	- <i>slant Total Electron Content</i>
SUPL	- <i>Secure User Plane Location</i>
SUPLCS	- <i>SUPL Configuration Services</i>
TEC	- <i>Total Electron Content</i>
TGM	- <i>Teste Global do Modelo</i>
TOW	- <i>Time of Week</i>
TTFX	- <i>Time To First Fix</i>
UNESP	- <i>Universidade Estadual Paulista “Júlio de Mesquita Filho”</i>
USB	- <i>Universal Serial Bus</i>
UTC	- <i>Universal Time Coordinated</i>
VADASE	- <i>Variometric Approach for Displacement Analysis Stand-alone Engine</i>
VARION	- <i>Variometric Approach for Real-Time Ionosphere Observations</i>
WGS 84	- <i>World Geodetic System 84</i>
WN	- <i>Week Number</i>
XML	- <i>eXtensible Markup Language</i>

SUMÁRIO

1. INTRODUÇÃO.....	22
1.1. Considerações Iniciais	22
1.2. Hipótese.....	25
1.3. Objetivos Geral e Específicos.....	26
1.4. Justificativa.....	26
1.5. Conteúdo da Dissertação	27
2. ESTADO DA ARTE	29
2.1. Artigos	29
2.2. Softwares para analisar dados GNSS brutos	32
3. PRINCÍPIOS BÁSICOS DO POSICIONAMENTO GNSS.....	43
3.1. Pseudodistância a Partir do Código	43
3.2. Fase de Batimento da Onda Portadora	45
3.3. Métodos de Posicionamento GNSS Absoluto	47
3.4. Principais Erros no Posicionamento GNSS.....	48
3.5. Método PPS Implementado no Projeto	49
4. RECURSOS GNSS A PARTIR DA VERSÃO 7.0 DO ANDROID (NOUGAT)	50
4.1. Recursos de Hardware	50
4.2. Recursos de Software	54
4.2.1. Mensagens de navegação.....	57
5. ALTERNATIVAS À AUSÊNCIA DE MENSAGENS DE NAVEGAÇÃO	58
5.1. Assisted-GNSS (A-GNSS).....	58
5.1.1. Networked Transport of RTCM via Internet Protocol (Ntrip)	60
5.1.2. Secure User Plane Location (SUPL)	61
5.2. Obtenção de Efemérides Transmitidas Através do Servidor SUPL.....	63
6. ARMAZENAMENTO DE DADOS GNSS.....	65
6.1. Armazenamento de Dados no Sistema Operacional Android	65
6.1.1. Armazenamento interno	65
6.1.2. Armazenamento externo.....	66
6.1.3. Tipos de arquivos para armazenamento de dados espaciais.....	67
6.2. Armazenamento de Dados GNSS a Partir do Android 7.0	71
6.2.1. Receiver Independent Exchange Format (RINEX).....	72

6.2.2. Bancos de dados	73
7. CÁLCULO DAS OBSERVÁVEIS DE PSEUDODISTÂNCIA E FASE DA ONDA PORTADORA EM SMARTPHONES ANDROID	79
7.1. Especificidades dos Dados GNSS em Smartphones Android	79
7.1.1. Efeito de <i>duty cycle</i>	79
7.1.2. Correção em relação ao centro de fase da antena.....	81
7.2. Cálculo de Pseudodistâncias.....	83
7.2.1. Estrutura geral de um receptor GNSS	83
7.2.2. Mensagens de navegação e estado do rastreamento de sinais GNSS	84
7.2.3. Obtenção do tempo de recepção com a técnica de tempo de recepção comum	87
7.2.4. Geração do tempo GPS	89
7.2.5. Geração das pseudodistâncias	91
7.3. Cálculo de fase da onda portadora.....	94
8. DESENVOLVIMENTO DE UM ECOSISTEMA DE PROCESSAMENTO E ANÁLISE DE DADOS GNSS A PARTIR DE SMARTPHONES ANDROID – INCT-GNSS-APP.....	96
8.1. Aplicativos para o processamento de dados GNSS brutos – INCT-GNSS-APP	96
8.1.1. Aplicativo para coleta de dados e processamento em tempo real	96
8.1.2. Aplicativo para armazenamento e compartilhamento de dados GNSS.....	105
8.2. Softwares desktop para pós processamento dos dados GNSS brutos	109
8.2.1. Software para pós processamento – INCT-GNSS-APP Post Processing.....	109
8.2.2. Software para controle de qualidade de pós processamento – INCT-GNSS-APP Post Processing Analysis.....	114
8.3. Fluxograma do ecossistema INCT-GNSS-APP	120
9. EXPERIMENTOS E RESULTADOS	122
9.1. Smartphones utilizados.....	122
9.2. Estrutura para de coleta dos dados	124
9.3. Coordenadas de referência utilizadas	125
9.4. Visualização dos arquivos	125
9.5. Coleta dos dados.....	127
9.6. Experimento 1: INCT-GNSS-APP e RTKPOST aplicando o método PPS	128
9.6.1. Dia 08/12/2021	129
9.6.2. Dia 09/12/2021	144

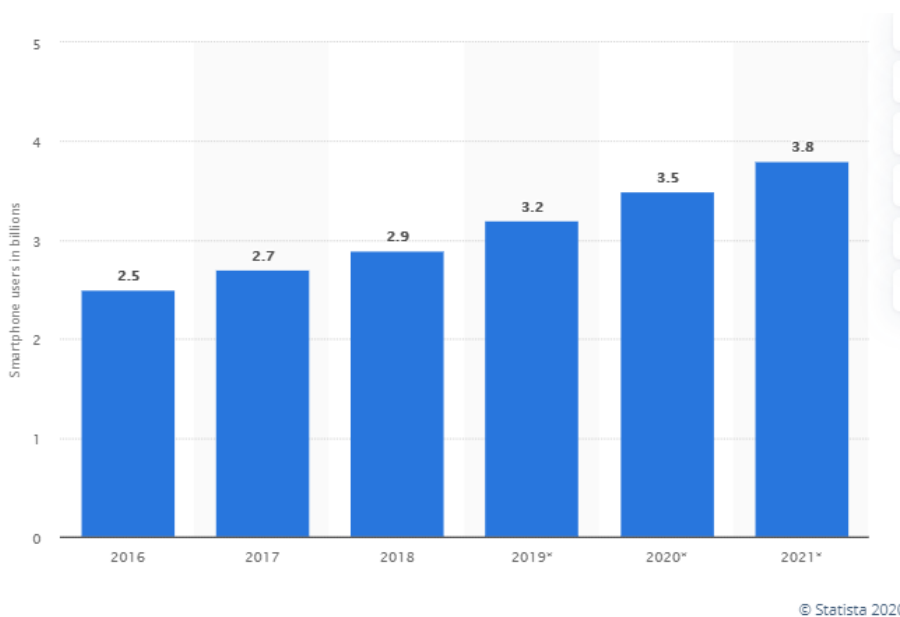
9.6.3.	Dia 10/12/2021	144
9.6.4.	Análise geral do experimento	145
9.7.	Experimento 2: RTKPOST aplicando o método PPP	146
9.7.1.	Resultados do Processamento.....	146
10.	ANÁLISES E DISCUSSÕES.....	150
11.	CONCLUSÕES E RECOMENDAÇÕES	151
	REFERÊNCIAS BIBLIOGRÁFICAS	153
	APÊNDICE A – Dados do processamento do dia 09/12/2021 no Experimento 1	156
	APÊNDICE B – Dados do processamento do dia 10/12/2021 no Experimento 1	167
	ANEXO A – Configurações aplicadas no RTKPOST no Experimento 1	178
	ANEXO B – Configurações aplicadas no RTKPOST no Experimento 2.....	182

1. INTRODUÇÃO

1.1. Considerações Iniciais

Observa-se, nos últimos anos, a crescente popularização do uso de smartphones. A Figura 1, criado pela empresa de análises estatísticas *Statista*, em 2020, mostra a evolução no número total de usuários de smartphones ao redor do mundo (STATISTA, 2022). Em 2016 eram 2,5 bilhões de usuários, já para 2021 as projeções indicavam 3,8 bilhões de usuários. Observa-se uma previsão de aumento de 40%, no número de usuários, de 2016 até 2020. Ainda segundo a empresa, em 2023 a previsão é de que existam 7,33 bilhões de usuários.

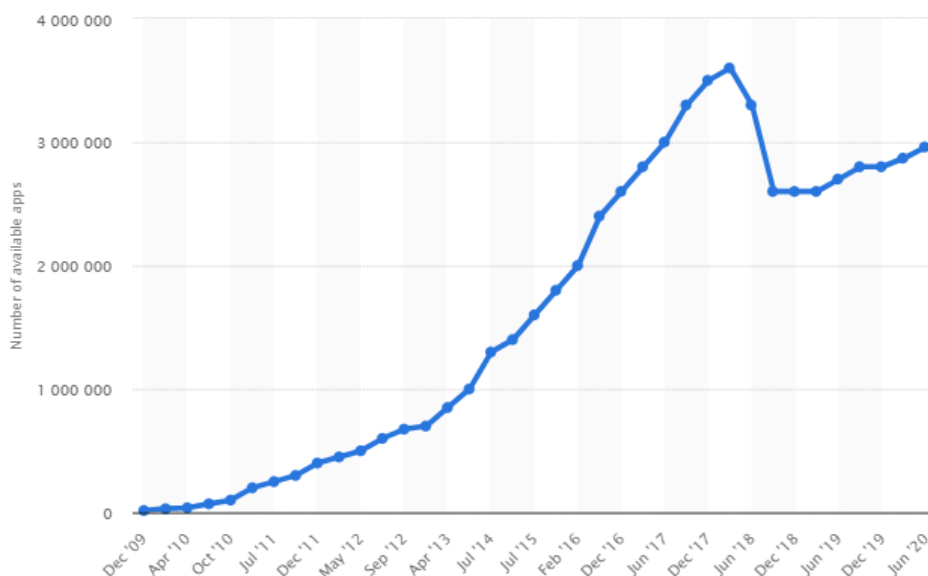
Figura 1: Número de usuários de smartphones Android em bilhões.



Fonte: (STATISTA, 2022).

Também é observado o aumento no número de aplicativos disponíveis na Google Play Store para o sistema operacional Android, conforme mostrado na Figura 2. Em 2020 foram contabilizados 2.960.000 aplicativos disponíveis para *download* (STATISTA, 2022).

Figura 2: Número de aplicativos disponíveis na Google Play Store.



Fonte: (STATISTA, 2022).

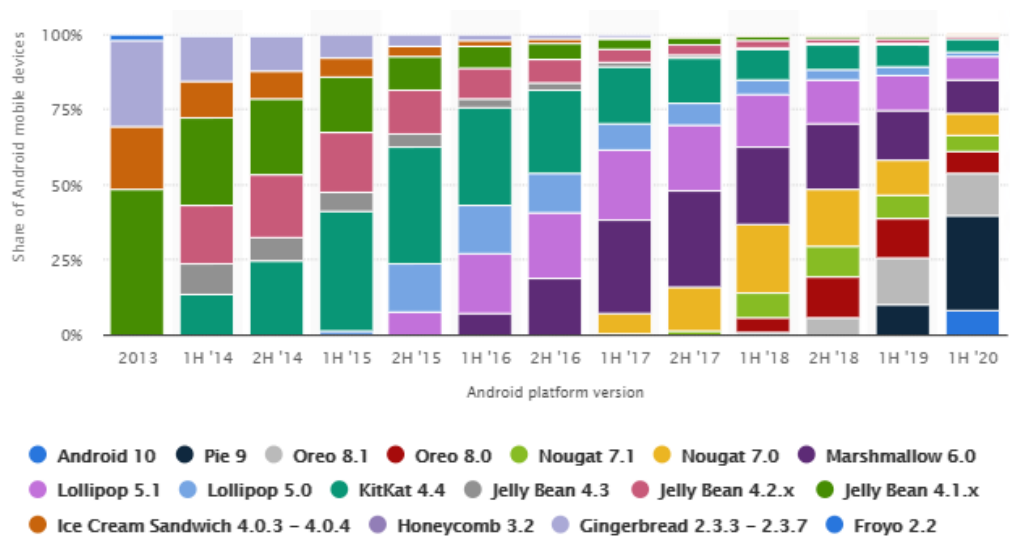
Smartphones utilizam sistemas de constelações de satélites para geolocalização, os quais são utilizados em escala global para tal finalidade. Em 1985, foram lançados os sistemas *Global Positioning System* (GPS) e *Globalnaya Navigatsionnaya Sputnikovaya Sistema* (GLONASS). O sistema GPS foi desenvolvido pelos Estados Unidos, já o GLONASS, pela ex-União Soviética. Utilizando esses sistemas foi possível desenvolver os sistemas de navegação amplamente utilizados em carros, embarcações e aviões (MONICO, 2013).

Receptores capazes de rastrear sinais de várias constelações de satélites são denominados receptores do tipo *Global Navigation Satellite System* (GNSS). Atualmente, smartphones também podem funcionar como receptores GNSS, pois os modelos mais recentes possuem sensores com essa capacidade. Um marco importante para a utilização de smartphones como receptores GNSS ocorreu em 2016. Neste ano, com o lançamento da versão 7.0 do sistema operacional Android, também conhecido como Android Nougat, os dados GNSS capturados pelos smartphones passaram a ser acessíveis de forma bruta, sem nenhum tipo de processamento (GSA, 2016). É possível obter, a partir desses smartphones, observáveis GNSS como a pseudodistância, fase da onda portadora e o *Signal-to-noise Ratio* (SNR), além das mensagens de navegação. Em versões anteriores ao Android 7.0, era possível obter, a partir dos smartphones, apenas informações como as coordenadas em latitude, longitude e a altitude geométrica do aparelho.

Com o acesso aos dados brutos GNSS tornou-se possível alcançar acurácia ao nível de metros e potencialmente centímetros, combinando métodos de posicionamento que utilizam as observáveis de pseudodistância e fase da onda portadora (BANVILLE E DIGGELEN, 2016). Explorando esses novos recursos, foram desenvolvidos aplicativos como o Geo++ RINEX Logger que cria arquivos RINEX de observação a partir dos dados coletados pelo smartphone.

Na Figura 3 são apresentadas as proporções das versões do Android dentre seus usuários ao redor do mundo. Os dados correspondem ao período desde o início de 2013 até 10 de abril de 2020 (STATISTA, 2020).

Figura 3: Porcentagem dos usuários Android por versão.



Fonte: (STATISTA, 2020).

Para o ano de 2020 os valores em porcentagem dos usuários por versão do sistema operacional Android, são apresentados na Tabela 1, obtidos do gráfico da Figura 3.

Tabela 1: Percentual de usuários Android por versão.

Versão do Android	Porcentagem de Usuários
Android 10	8,3%
Pie 9	31,3%
Oreo 8.1	14%
Oreo 8	7,3%
Nougat 7.1	5,4%
Nougat 7.0	7,5%
Marshmallow 6.0	11,2%
Lollipop 5.1	7,4%
Lollipop 5.0	1,8%
KitKat 4.4	4%
Jelly Bean 4.3	0,3%
Jelly Bean 4.2.x	0,8%
Jelly Bean 4.1.x	0,6%

Fonte: (STATISTA, 2020).

A partir da Tabela 1 é possível verificar que 73,80% dos smartphones Android possuem a versão 7.0, ou superior, do sistema operacional. Com isso, a depender do hardware desses aparelhos, até 73,80% dos smartphones Android poderiam fornecer dados GNSS brutos.

Alguns smartphone com Android 7.0 ou superior podem não fornecer dados GNSS brutos, por questões de hardware. Nesse caso é possível consultar a lista de dispositivos compatíveis, na página oficial¹ do Google para dados GNSS brutos.

1.2. Hipótese

Smartphones com o sistema operacional Android podem ser utilizados como receptores GNSS dentro de contextos específicos definidos por suas especificidades e restrições operacionais. Para isso, é preciso avaliar a qualidade dos dados GNSS provenientes desses aparelhos e aplicá-los em métodos de posicionamento. Tendo em vista que se trata de uma tecnologia ainda em desenvolvimento, também é preciso entender como acessar os dados GNSS brutos provenientes dos smartphones Android e caracterizar o respectivo processamento dos dados.

¹ <https://developer.android.com/guide/topics/sensors/gnss?hl=pt-br>

1.3. Objetivos Geral e Específicos

Essa dissertação de mestrado tem como objetivo geral avaliar as características e peculiaridades do processo de registro de dados GNSS em dispositivos móveis com Android, a partir de sua versão 7.0, verificar a qualidade do posicionamento a partir desses dados, e conceber uma estratégia para o armazenamento e distribuição desses dados.

Os objetivos específicos consistem em:

- Captura de dados GNSS em smartphone com sistema operacional Android e tratamento, em tempo real, desse fluxo de dados para processamento;
- Implementação e avaliação do método de Posicionamento por Ponto Simples (PPS) a partir dos dados capturados;
- Análise da qualidade dos dados GNSS capturados;
- Avaliação dos resultados da execução de métodos de posicionamento relativo a partir dos dados capturados; e
- Avaliar estratégias para armazenar e disponibilizar os dados coletados.

1.4. Justificativa

A partir da possibilidade da captura e processamento de dados GNSS brutos, a partir de smartphones, novas possibilidades para aplicações científicas tornaram-se viáveis. Como apontado em (LI; GENG, 2019) e (CHEM et al., 2019), houve avanços consideráveis em termos de acurácia. Aplicações em tempo real também se mostraram factíveis, como apontado em (FORTUNATO; RAVANELLI; MAZZONI, 2019).

Considerando tais avanços, ao utilizar dados GNSS brutos provenientes de smartphones e os exemplos das aplicações online para o armazenamento e visualização de dados espaciais ISMR Query Tool (VANI; SHIMABUKURO; MONICO, 2017) e OpenStreetMap, torna-se relevante a investigação quanto as possibilidades para o armazenamento e disponibilização desses dados. Torna-se relevante a existência de uma ferramenta online para que a comunidade também possa aplicar análises e utilizar aos dados armazenados.

Entretanto, a qualidade das medidas GNSS fornecidas pelos smartphones ainda precisam ser estudadas, pois esses aparelhos já apresentaram diferenças em relação aos receptores geodésicos, como apresentado em (LI; GENG, 2019). Além disso, há questões

relacionadas apenas aos receptores GNSS dos smartphones, como o efeito de *duty cycle*, abordado durante a revisão bibliográfica desse documento. Além disso, outras questões como a influência da atividade ionosférica nas observações coletadas, principalmente no Brasil, país localizado na região equatorial e de baixas latitudes, devem ser investigadas.

Além disso, a qualidade dos resultados obtidos a partir dos métodos de posicionamento GNSS utilizando dados provenientes dos smartphones também precisa ser avaliada. Atualmente os smartphones são dispositivos multi GNSS, com isso, a influência de observáveis de constelações mais recentes, como o BeiDou e Galileo, nos métodos de posicionamento precisa ser investigada. A eficácia de aplicação de modelos de correção atmosférica como nos modelos de Klobuchar e Hopfield também precisam ser investigadas.

A partir do cenário apresentado, no qual tanto a tecnologia dos smartphones quanto dos sistemas de satélites continuam em constante desenvolvimento, nessa pesquisa são realizadas a avaliação da qualidade das medidas e do posicionamento GNSS em smartphones. Também foram definidas estratégias para pós-processamento e compartilhamento desses dados.

O desenvolvimento e execução dessa pesquisa, bem como seus resultados, buscam contribuir para a compreensão das particularidades para a coleta e processamento de dados GNSS em dispositivos móveis, estendendo os estudos realizados pelo Grupo de Estudos em Geodésia Espacial (GEGE) e pelo INCT GNSS-NavAer.

1.5. Conteúdo da Dissertação

O conteúdo da dissertação está dividido em 11 capítulos. No segundo capítulo é apresentado o estado da arte, em termos de artigos e softwares disponíveis atualmente.

No terceiro, são apresentados conceitos básicos do posicionamento GNSS como as definições das observáveis e métodos de posicionamento.

No capítulo 4 são apresentados os recursos que passaram a ser disponíveis nos smartphones com o sistema Android na versão 7.0 ou superior. São discutidas questões relacionadas a software, como as mudanças nas classes da API Android disponíveis para utilização durante o desenvolvimento de aplicativos. Também são discutidas questões relacionadas ao hardware dos aparelhos, como as frequências e constelações GNSS disponíveis.

No quinto capítulo é apresentado o conceito de *Assisted-GNSS* (A-GNSS). Trata-se de uma alternativa para obtenção de dados necessários ao posicionamento GNSS e foi implementada nos softwares desenvolvidos nessa pesquisa. É abordado sobre como utilizá-lo para obter efemérides transmitidas.

Já no capítulo 6 são discutidas as alternativas para o armazenamento de dados GNSS. São apresentadas soluções mais simples, como o armazenamento através de arquivos de texto, e soluções mais complexas, como a implementação de bancos de dados em nuvem.

No capítulo 7 é discutido como obter as observáveis de pseudodistância e fase da onda portadora. Especificidades dos smartphones, que precisam ser tratadas para a obtenção de tais observáveis GNSS também são discutidas nesse capítulo.

No oitavo capítulo é descrito o ecossistema de softwares desenvolvido nessa pesquisa. Trata-se de aplicativos para Android e softwares para computadores do tipo desktop, para o pós-processamento e análise dos dados GNSS brutos provenientes dos smartphones com o sistema operacional Android.

No capítulo 9 são apresentados os experimentos e resultados obtidos nessa pesquisa. São discutidos dois experimentos que utilizam os softwares desenvolvidos neste projeto e o RTKPOST, software presente no RTKLib, para fins de comparação de resultados.

No décimo capítulo são apresentadas as análises e discussões acerca dos resultados obtidos nessa pesquisa.

Por fim, no último capítulo são apresentadas as conclusões e recomendações para futuras pesquisas.

No Apêndice A, encontram-se os resultados de processamento para o dia 09/12/2021, do 1º experimento. Já no Apêndice B, encontram-se os resultados de processamento para o dia 10/12/2021, do 1º experimento.

No Anexo A, encontram-se as configurações para o RTKPOST utilizadas no 1º experimento. Do mesmo modo, no Anexo B encontram-se as configurações para o RTKPOST utilizadas no 2º experimento.

2. ESTADO DA ARTE

Neste Capítulo são apresentados artigos acadêmicos e softwares já desenvolvidos que abordam o uso de dados GNSS brutos provenientes de smartphones com o sistema operacional Android.

2.1. Artigos

Antes mesmo que os smartphones disponibilizassem dados GNSS brutos, esses dispositivos já eram utilizados para obter dados GNSS em trabalhos acadêmicos. No projeto Mahali GNSS Logger App (KENNEDY, et al., 2016) foram utilizados smartphones para coletar dados de receptores GNSS, utilizando um adaptador Serial-para-USB. Neste trabalho, os dados dos receptores GNSS eram armazenados na memória interna do smartphone de modo temporário. Após a coleta dos dados, ao acessar uma conexão de Internet, os dados eram enviados a um diretório do Dropbox.

Durante o projeto Mahali foram realizados testes em São José dos Campos e Manaus. O objetivo do trabalho foi coletar dados relacionados ao *Total Electron Content* (TEC) para monitorar o comportamento da Ionosfera no Brasil. Através desse projeto, foi criada uma prova de conceito para demonstrar a possibilidade de utilizar smartphones para a coleta e disseminação de dados GNSS. Os smartphones utilizados foram o Samsung Galaxy S3 e aparelhos da linha Moto G, da Motorola. Esses aparelhos mostraram-se especialmente úteis em locais com infraestrutura menos desenvolvida (KENNEDY, et al., 2016).

Para fins de posicionamento GNSS já eram realizados estudos antes mesmo da liberação de dados brutos pelo Android. Pesquisadores tentavam alterar o *firmware* dos aparelhos e obtinham apenas precisão ao nível de metros. Após a liberação dos dados brutos, a partir do Android 7.0, é possível atingir precisão ao nível de centímetros ao utilizar observações de onda portadora (BANVILLE; DIGGELEN, 2016).

Em maio de 2018 foi lançado o primeiro smartphone de dupla frequência (L1/L5) disponibilizado comercialmente, trata-se do smartphone Xiaomi Mi 8. Utilizando esse aparelho, foram conduzidos estudos nas universidades de Nápoles e Roma, na Itália. Os pesquisadores analisaram a precisão das coordenadas obtidas utilizando dados GNSS do Xiaomi Mi 8. Os resultados obtidos foram de 4,57 m de precisão no posicionamento absoluto e 1,02 m no

posicionamento relativo, ao utilizar observações da constelação GNSS Galileo (CHEM, et. al., 2019).

Li e Geng (2019) avaliaram os erros de medidas multi GNSS de smartphones Android. Nesse trabalho, os autores verificaram que as observações GLONASS, quando provenientes de smartphones, possuem qualidade inferior, em comparação com as outras constelações GNSS. Essa descoberta também foi verificada nos resultados de posicionamento, uma vez que, as soluções apenas GPS obtiveram melhor resultado do que as soluções GPS+GLONASS. Os autores também verificaram que a o *Carrier-to-Noise-density ratio* (C/No), que indica a qualidade do sinal depende diretamente da qualidade da antena dos smartphones. Isso foi verificado quando smartphones no mesmo ambiente de observação apresentaram diferenças nos valores de C/No.

Como é apontado em (LI; GENG, 2019), utilizando Posicionamento por Ponto Simples (PPS) é possível obter o *Root Mean Square Error* (RMSE) horizontal e vertical entre 10-20 m. Utilizando o método *Carrier Phase Relative Positioning* (CRP) os autores obtiveram desvio padrão de 1 cm nas componentes horizontal e vertical no modo estático.

Em testes para o posicionamento cinemático, fixando os smartphones no teto de um automóvel, na cidade de Wuhan, China, os autores atingiram erro tridimensional de 0,169 m com 63% dos erros inferiores a 0,1 m e 100% dos erros inferiores a 0,5 m. Com isso, os autores verificaram que aplicações de navegação utilizando smartphones são factíveis.

Além disso, os autores apresentaram considerações importantes para o processamento de dados brutos GNSS provenientes de smartphones. Estes verificaram que o aumento do ângulo de elevação para a seleção dos satélites não resultou em aumento da razão C/No, o que seria esperado para receptores geodésicos. Por fim, os autores estudaram o impacto da correção do centro de fase da antena. A utilização de antenas externas, de centro de fase conhecido, aprimorou a precisão obtida. Como as informações de centro de fase da antena interna dos smartphones não são disponíveis, os autores sugerem, além da utilização de antenas externas, a utilização do centro geométrico do smartphone como sendo seu centro de fase.

Também em 2019, como apontado em (CHEM et. al., 2019), onde utilizando o smartphone Xiaomi Mi 8, foi avaliado o desempenho do Posicionamento por Ponto Preciso (PPP) em tempo real. Os autores obtiveram RMSE horizontal e vertical de 0,81 m e 1,65 m, respectivamente. Foram utilizados dados das constelações GPS, BeiDou e Galileo. O tempo necessário para que cada época apresentasse erro inferior a 1 m foi inferior a 30 s.

Ainda em 2019, conforme descrito em (FORTUNATO, RAVANELLI, MAZZONI, 2019), o Xiaomi Mi 8 foi utilizado em aplicações geofísicas em tempo real. Os pesquisadores utilizaram a técnica *Variometric Approach for Displacement Analysis Stand-alone Engine* (VADASE) para obter a velocidade 3D do smartphone e a técnica *Variometric Approach for Real-Time Ionosphere Observations* (VARION) para reconstruir o *slant Total Electron Content* (sTEC).

Com a técnica VADASE, os pesquisadores obtiveram erro ao nível de poucos mm/s na componente horizontal e em torno de 1cm/s na componente vertical. Segundo os autores, esses resultados confirmam a possibilidade de utilizar smartphones para a detecção de movimentos rápidos, como as oscilações periódicas causadas por terremotos.

Utilizando a técnica VARION, os pesquisadores obtiveram ruído no mesmo nível dos receptores geodésicos, demonstrando que o smartphone Xiaomi Mi 8 pode ser utilizado em aplicações de monitoramento da ionosfera em tempo real.

Além dos avanços em relação à utilização de dados GNSS brutos provenientes de *smartphones* Android, verifica-se também, avanços nas ferramentas para armazenamento e análises de dados espaciais.

Em 2017, como apontado em (VANI; SHIMABUKURO; MONICO, 2017) foi desenvolvida uma ferramenta para a integração e interpretação visual de dados de monitoramento da ionosfera, a denominada ISMR Query Tool. A ferramenta utiliza os dados das redes LISN, ICEA, CIGALA/CALIBRA e da rede do Instituto Nacional de Ciência e Tecnologia “Tecnologia GNSS no Suporte à Navegação Aérea” (INCT GNSS NavAer). Utilizando essa ferramenta é possível visualizar dados sobre cintilações ionosféricas, provenientes de receptores GNSS instalados no do território brasileiro.

Outro exemplo de ferramenta online para a análise de dados espaciais é o OpenStreetMap, que é uma ferramenta de mapeamento colaborativo e com essa ferramenta os usuários podem visualizar e compartilhar trajetórias GPS enviando arquivos no formato *GPS eXchange Format* (GPX). É possível visualizar as informações de latitude, longitude e altitude durante todo o trajeto percorrido.

Já em 2021, como apontado em (HAN, et al., 2021), foi utilizado aprendizado de máquina para aprimorar a integração entre os receptores GNSS e receptores do tipo *Inertial Measurement Unit* (IMU) de smartphones Android. Os autores utilizaram a técnica de aprendizado de máquina conhecida como *aprendizado reforçado*, para corrigir os parâmetros do filtro de Kalman. A condição para o reforço do aprendizado e correção dos valores extraídos

da IMU, era a obtenção de valores mais baixos de *Root Mean Square* (RMS) 2D através do receptor GNSS.

Desse modo, ao utilizar o aprendizado reforçado para processar dados GNSS e IMU, disponibilizados pelo Google, coletados em um veículo que percorreu um trajeto em ambiente urbano, houve melhora nos resultados. O RMS 2D diminuiu de 46,39 m para 16,99 m.

Em 2023, seguindo a tendência de utilizar a integração de sensores como estratégia para aprimorar o desempenho dos receptores GNSS, foi apresentado em (SUZUKI, 2023), foi avaliada a adoção de uma técnica de otimização em duas etapas. A técnica consiste em separar os processos de otimização de velocidade e posição. Nesse caso, a velocidade é primeiramente estimada com observações Doppler e os *outliers* são removidos. Os valores faltantes são então interpolados. No próximo passo de otimização, a velocidade otimizada na primeira etapa é utilizada como injeção e a posição é estimada com a técnica *Time-Differenced Carrier Phase* (TDCP), que é mais acurada que o Doppler.

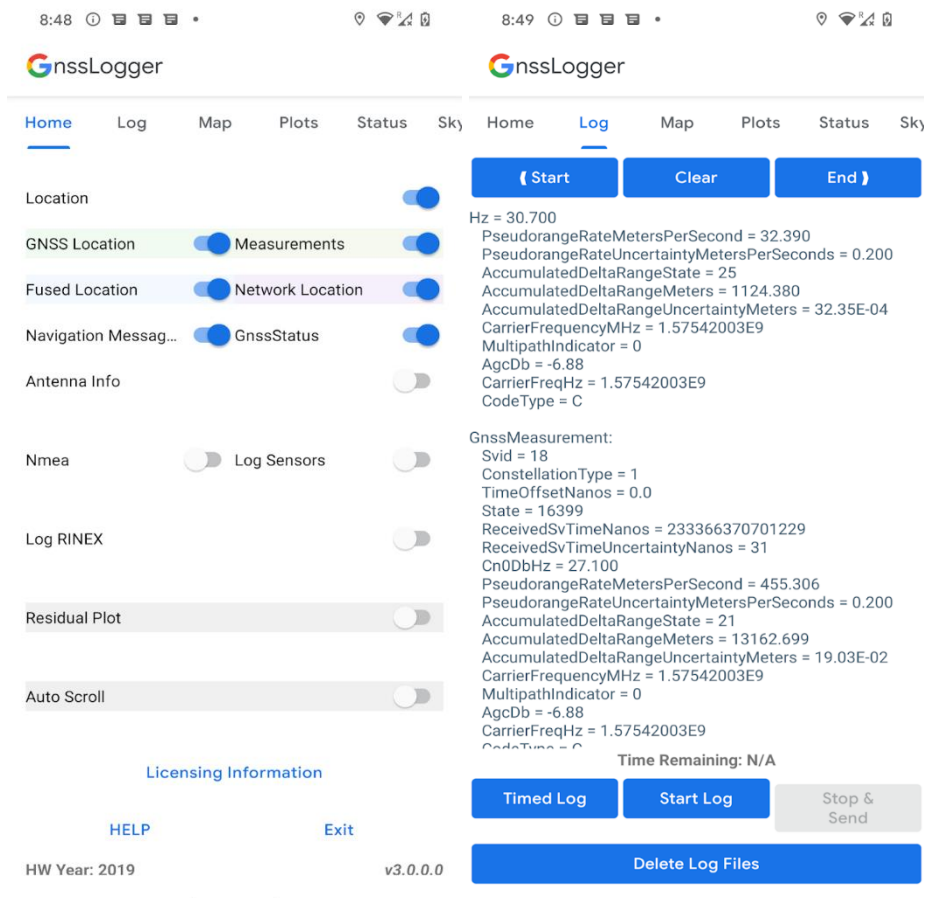
Essa técnica foi aplicada em dados disponibilizados pelo Google no *Google Smartphone Decimeter Challenge* (GSDC). O melhor valor de acurácia obtido até então era de 1,229 m. Com a técnica proposta em (SUZUKI, 2023), o valor obtido foi de 1,02 m.

2.2. Softwares para analisar dados GNSS brutos

GnssLogger

Esse aplicativo foi desenvolvido pelo Google e permite salvar um arquivo de texto com as medições GNSS capturadas pelos smartphones. Também é possível visualizar os dados coletados em tela durante a coleta dos dados. O aplicativo também cria arquivos nos formatos *Receiver Independent EXchange Format* (RINEX) e *National Marine Electronics Association* (Nmea). A tela principal do aplicativo GnssLogger é apresentada na Figura 4.

Figura 4: Tela principal do aplicativo GnsLogger.



Fonte: (GOOGLE, 2023).

O arquivo gerado pelo GnsLogger possui um cabeçalho, com a descrição dos campos presentes no arquivo e as medições GNSS contendo os dados brutos, que são representadas nas linhas com início “Raw”. Cada linha Raw é resultado da combinação dos dados GNSS capturados pelo aparelho, que transcreve os dados capturados na forma de texto em cada linha Raw. O arquivo gerado pelo Gns Logger é exemplificado na Figura 5 e pode ser analisado em softwares como o Gns Analysis App.

Figura 5: Trecho de arquivo gerado pelo GnssLogger.

```

1 # Header Description:
2 #
3 # Version: v2.0.0.1 Platform: 9 Manufacturer: motorola Model: moto x4
4 #
5 #
6 Raw, ElapsedRealtimeMillis, TimeNanos, LeapSecond, TimeUncertaintyNanos, FullBiasNanos, BiasNanos, BiasUncertaintyN
7 nd, HardwareClockDiscontinuityCount, Svid, TimeOffsetNanos, State, ReceivedSvTimeNanos, ReceivedSvTimeUncertaintyN
8 geRateUncertaintyMetersPerSecond, AccumulatedDeltaRangeState, AccumulatedDeltaRangeMeters, AccumulatedDeltaRang
9 arrierPhase, CarrierPhaseUncertainty, MultipathIndicator, SnrInDb, ConstellationType, AgcDb, CarrierFrequencyHz
10 #
11 # Fix, Provider, Latitude, Longitude, Altitude, Speed, Accuracy, (UTC) TimeInMs
12 #
13 # Nav, Svid, Type, Status, MessageId, Sub-messageId, Data (Bytes)
14 #
15 Fix, gps, -22.122525, -51.407814, 439.587402, 0.000000, 4.288000, 1545067176000
16 NMEA, $GPGSV, 3, 1, 11, 02, 02, 135, 19, 05, 29, 118, 20, 12, 42, 052, 27, 15, 03, 039, 26, 1*63
17 , 1545067174962
18 Raw, 412508830, 19247216000000, , , -1229083146220951131, 0.8184490203857422, 50.18275260226801, -1.0059338089524008
19 .7, -92.12911224365234, 0.8829999566078186, 0, 0, 0, 0, 0, , , , , 0, 3, ,
20 Raw, 412508830, 19247216000000, , , -1229083146220951131, 0.8184490203857422, 50.18275260226801, -1.0059338089524008
21 4, -630.6463623046875, 1.2220001220703125, 0, 0, 0, 0, 0, , , , , 0, 3, ,

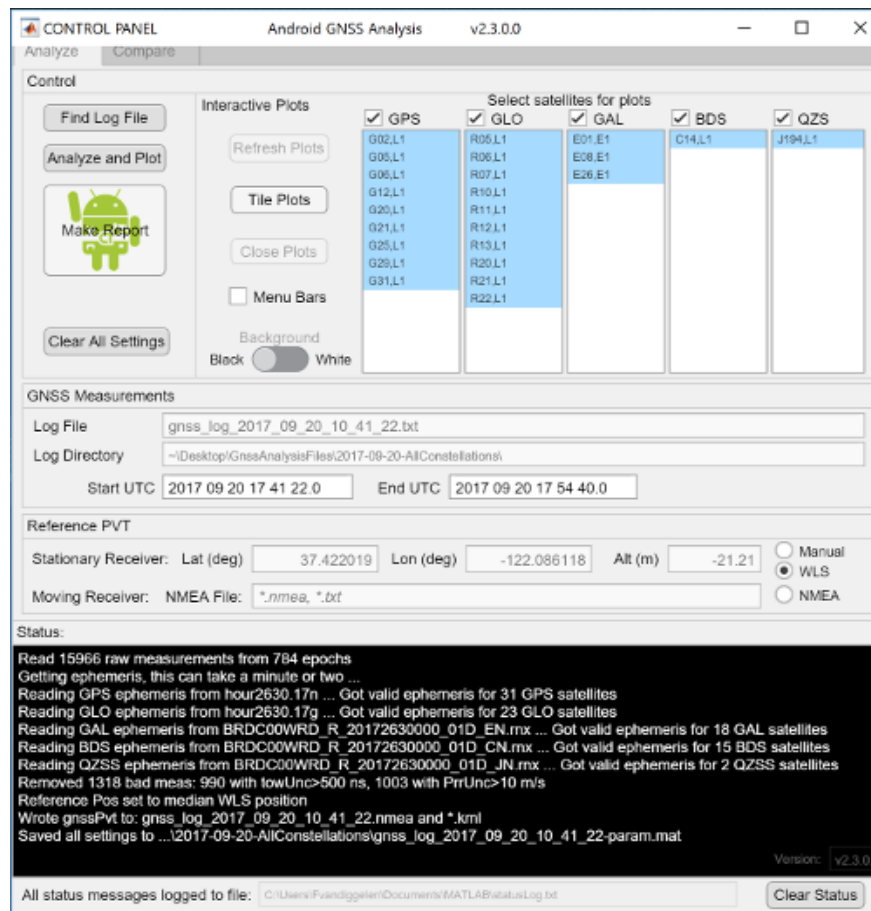
```

Fonte: (O AUTOR, 2020).

Gnss Analysis App

O arquivo gerado pelo GnssLogger pode ser analisado pelo GNSS Analysis App. Trata-se de um programa para computadores do tipo desktop desenvolvido em Matlab e disponibilizado para sistemas operacionais Windows, Linux e MacOS. Por meio desse programa é possível visualizar em detalhes os dados dos satélites observados e as medições capturadas pelo GnssLogger. Também são mostrados os satélites rastreados, bem como, a qual constelação pertencem. A tela principal do GNSS Analysis App é apresentada Figura 6.

Figura 6: Tela principal do Gnss Analysis App.

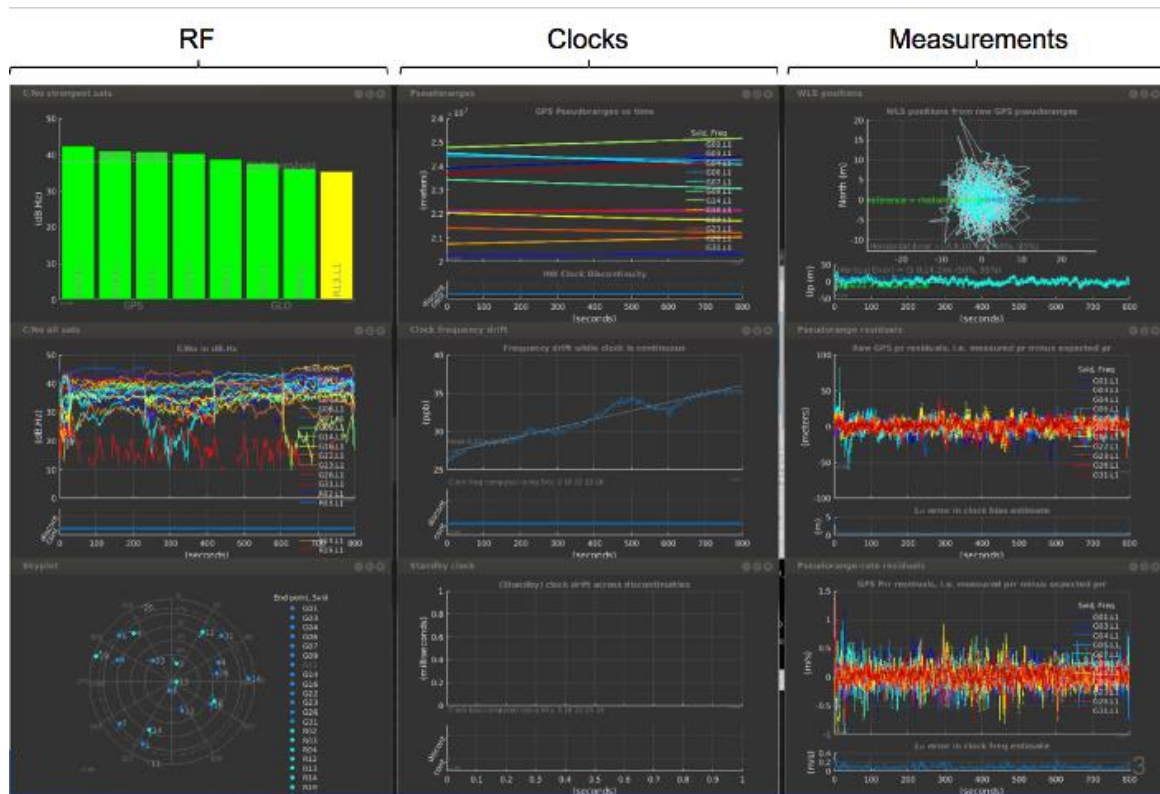


Fonte: (O AUTOR, 2020).

Além de exibir informações sobre os satélites, o programa também realiza o download, através da Internet, dos arquivos de navegação, para obter as efemérides necessárias para o processamento do arquivo gerado pelo GnssLogger e estimar a solução de posição e velocidade do smartphone no momento da captura dos dados. Após os cálculos, são gerados gráficos interativos, que são divididos em três categorias: *Radio Frequency* (RF) para indicar a qualidade dos sinais capturados; *Clock* para informações sobre os relógios como os erros para eles estimados e *Measurements* para mostrar características das medições recebidas e posições estimadas.

O conjunto de gráficos gerados pelo Gnss Analysis App é apresentado na Figura 7. O programa permite salvar os gráficos gerados e demais informações processadas em arquivos do tipo *Hypertext Markup Language* (HTML) que podem ser visualizados em navegadores web.

Figura 7: Gráficos gerados pelo GNSS Analysis App.



Fonte: (O AUTOR, 2020).

Tanto o Gns Logger quanto o Gns Analysis App possuem código fonte aberto e podem ser consultados para auxiliar no desenvolvimento de aplicações que processem dados GNSS provenientes de smartphones com o sistema operacional Android.

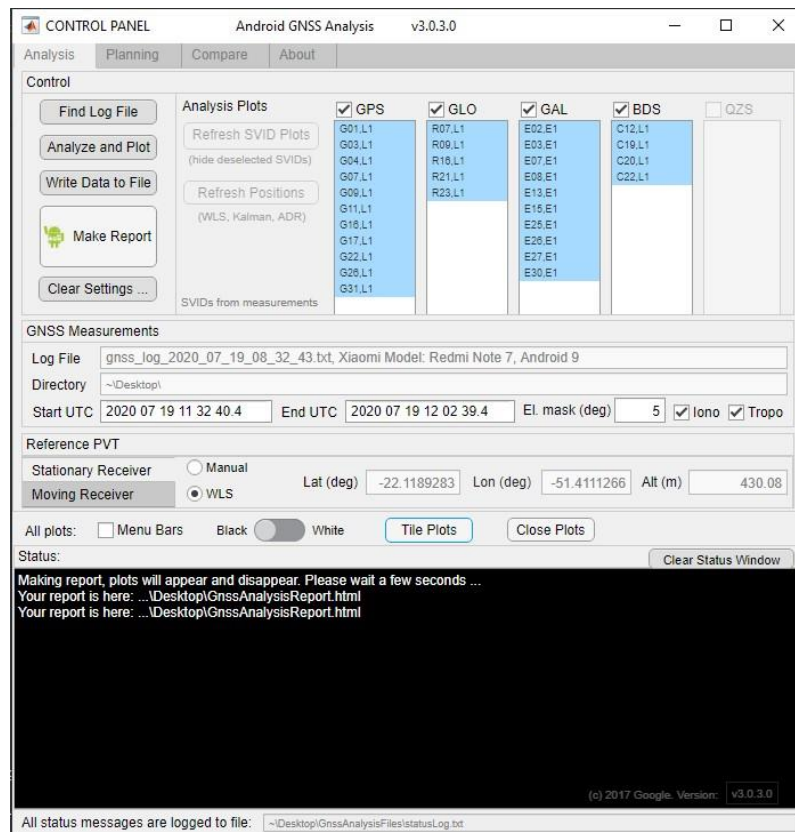
Alteração dos parâmetros de processamento no Gns Analysis App

A partir do arquivo com dados brutos, o Gns Analysis App permite alterar as configurações de processamento, sendo possível, por exemplo, reprocessar os dados selecionando quais as constelações GNSS observadas serão consideradas, gerando, por exemplo, soluções GPS+Galileo+GLONASS, ou apenas GPS, a partir do mesmo arquivo de dados brutos.

Também é possível alterar as configurações relacionadas a máscara de elevação e modelos de correção ionosférica e troposférica aplicados. Alterando as configurações e

reprocessando os dados, os gráficos interativos gerados pelo *software* são atualizados. Com isso é possível investigar a influência de determinada constelação GNSS ou modelo aplicado no que diz respeito à acurácia das soluções de posição e velocidade calculadas, como é apresentado na Figura 8.

Figura 8: Configurações do GNSS Analysis App.



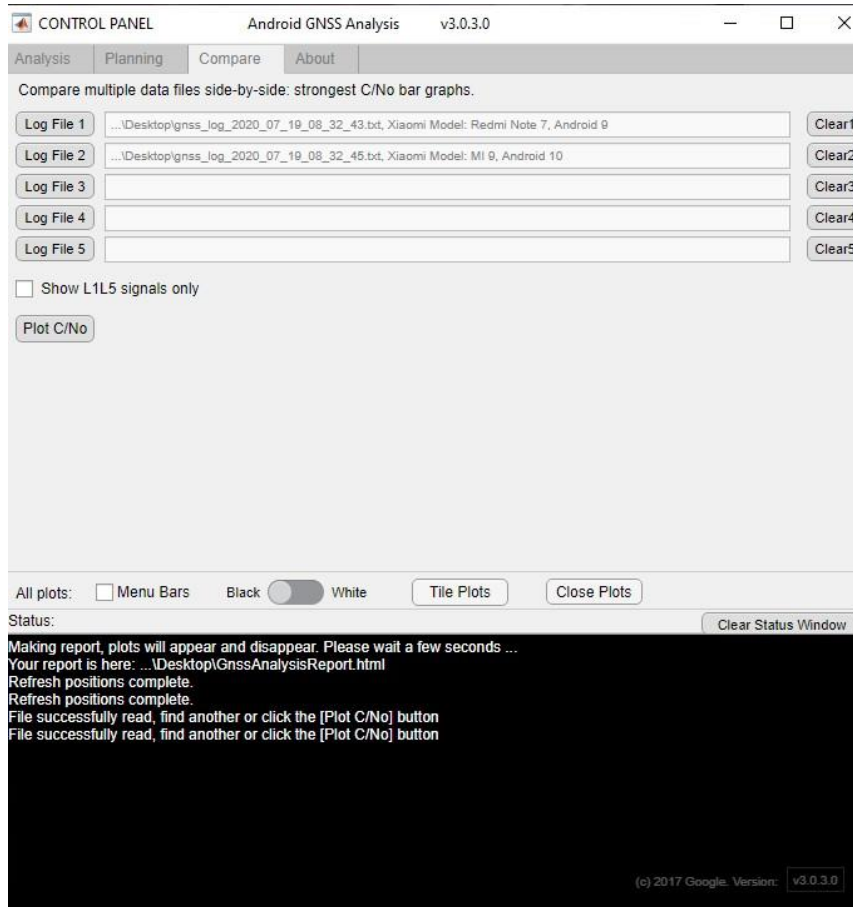
Fonte: (O AUTOR, 2020).

Comparação entre diferentes arquivos de dados brutos

O Gnss Analysis App pode ser utilizado para comparar múltiplos arquivos de dados GNSS brutos, criados pelo GnssLogger. Nos exemplos a seguir, foram comparados dois arquivos de dados brutos, coletados ao mesmo tempo e no mesmo ambiente de observação, em um bairro chamado Jardim das Rosas, próximo ao câmpus da UNESP de Presidente Prudente.

O primeiro arquivo foi coletado pelo smartphone Xiaomi Redmi Note 7 e o segundo arquivo pelo smartphone Xiaomi Mi 9. Na Figura 9 é apresentada a tela do programa para comparação dos arquivos de dados.

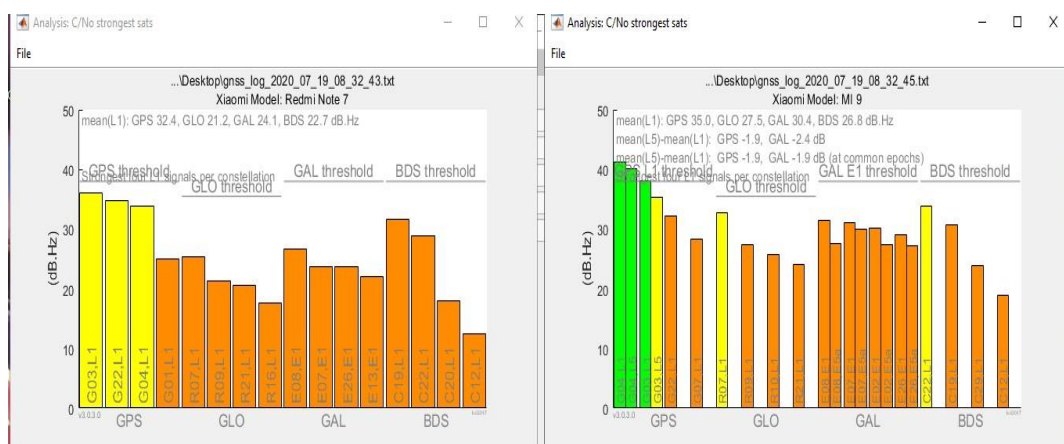
Figura 9: Tela do GNSS Analysis App para comparação de arquivos.



Fonte: (O AUTOR, 2020).

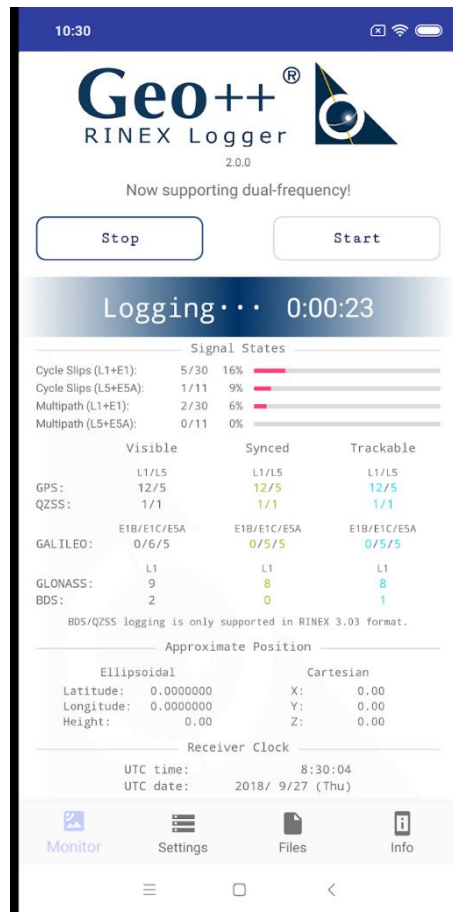
Na Figura 10 são apresentados os gráficos gerados para os dois arquivos, contendo os valores C/No médios, por constelação e satélite, para cada arquivo.

Figura 10: Gráfico com os valores de C/N0 para dois arquivos.



Fonte: (O AUTOR, 2020).

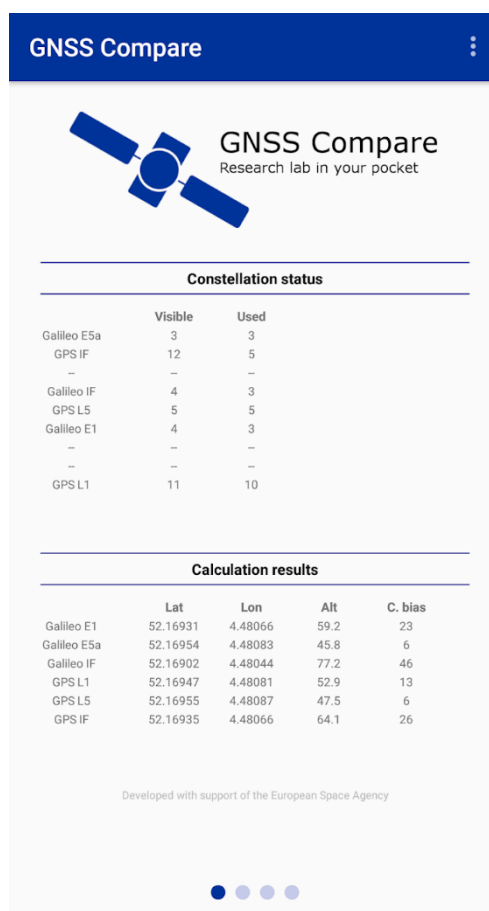
Figura 12: Tela inicial do Geo++ RINEX Logger.



Fonte: (GOOGLE, 2023).

GNSS Compare

O aplicativo para Android GNSS Compare foi o aplicativo vencedor da competição *Galileo Smartphone App Challenge*, promovida pela *European Space Agency* (ESA). O aplicativo tem por objetivo incentivar o uso do Galileo, comparando seu desempenho com o de outras constelações GNSS. Para isso, foram implementados no aplicativo, algoritmos para o cálculo de soluções de posição e velocidade do smartphone. Dentre eles, destacam-se a implementação do MMQ e Filtro de Kalman. O GNSS Compare possui código fonte aberto. Na Figura 13 é apresentada a Tela inicial do Gns Compare.

Figura 13: Tela inicial do GNSS Compare.

Fonte: (GOOGLE, 2023).

PPP WizLite

O aplicativo para Android PPP WizLite é a versão móvel do projeto PPP-Wizard desenvolvido pelo *Centre National d'études Spatiales* (CNES). Esse aplicativo executa o PPP com resolução de ambiguidade Inteira e de Zero-Diferença. Além disso, o aplicativo conecta-se com a Internet para efetuar o download de correções. Com isso, o aplicativo consegue executar o PPP e obter precisão em torno de 1 m. Na Figura 14 é apresentada a tela inicial do PPP WizLite.

Figura 14: Tela inicial do PPP WizLite.

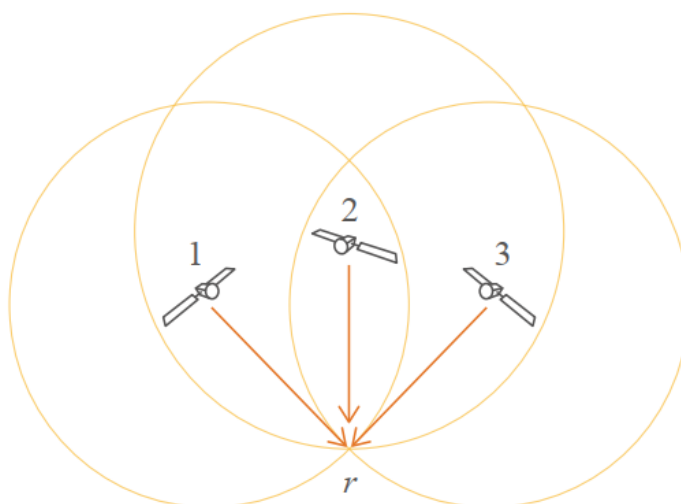


Fonte: (GOOGLE, 2023).

3. PRINCÍPIOS BÁSICOS DO POSICIONAMENTO GNSS

Para posicionar um receptor GNSS(r) em um espaço tridimensional seria necessário utilizar dados de três satélites, como ilustrado na Figura 15. O problema é resolvido ao encontrar a intersecção das três esferas formadas pelas distâncias entre o receptor e cada satélite. As intersecções das três esferas ocorrerão em dois pontos sendo um deles a posição do receptor. Essa é a ideia fundamental do posicionamento de receptores utilizando satélites. Entretanto, como os relógios dos satélites e do receptor não são sincronizados, é necessário utilizar um quarto satélite para que seja possível estimar a posição do receptor, juntamente com o erro do relógio do receptor (GSA, 2017).

Figura 15: Princípio básico de posicionamento por satélite.



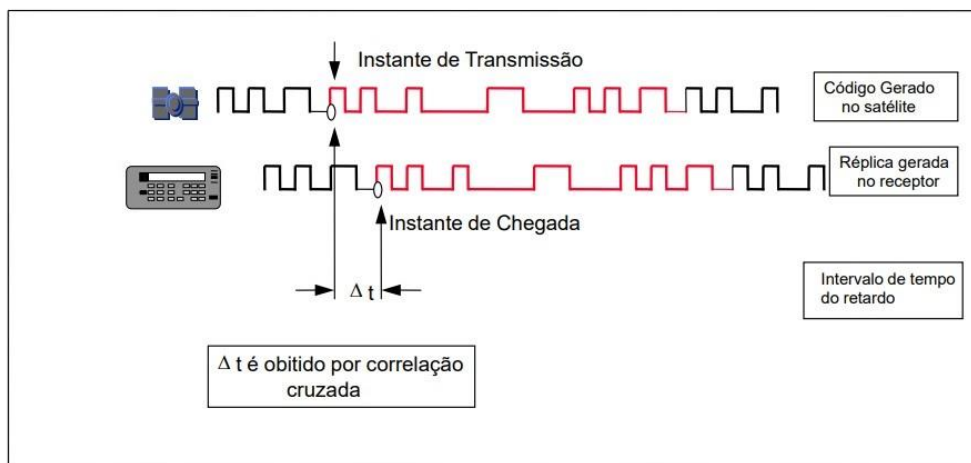
Fonte: Adaptado de (GSA, 2017).

3.1. Pseudodistância a Partir do Código

As observáveis GNSS são as medições que os receptores calculam a partir de informações fornecidas pelos satélites. Com tais medições os próprios receptores podem calcular suas respectivas posições. Para determinar a posição de um receptor, a observável básica é a pseudodistância. Outra observável que pode ser utilizada é a fase da onda portadora (MONICO, 2008).

Os satélites emitem sinais por meio de ondas eletromagnéticas como as portadoras L1 (1575,42 MHz) e L2 (1227,60 MHz) dos satélites GPS, além da mais recente onda portadora L5 (1176,45 MHz). Esses sinais contêm informações sobre o momento em que o sinal foi transmitido. As medidas de distância entre a antena do satélite e a antena do receptor são obtidas a partir dos códigos gerados no satélite e suas réplicas, geradas no receptor. O intervalo de propagação do sinal no trajeto que liga o satélite ao receptor é obtido através da técnica de correlação de código. Nesta técnica, o intervalo de propagação equivale ao retardo entre a chegada de uma transição particular do código e a réplica do código gerado no receptor (MONICO, 2008). Na Figura 16 é ilustrado o processo de correlação de código.

Figura 16: Princípio da medida de pseudodistância.



Fonte: Adaptado de (MONICO, 2008).

A distância satélite-receptor é obtida pela multiplicação do tempo de propagação do sinal, resultado da correlação de código, pela velocidade da luz. Como os relógios do satélite e do receptor não são sincronizados, essa observável é denominada pseudodistância. A pseudodistância, em metros, pode ser modelada segundo a Equação (1) (MONICO, 2008).

$$PD_R^S = \rho_r^s + c[dt_r - dt^s] + I_r^s + T_r^s + dm_r^s + \varepsilon_r^s \quad (1)$$

onde:

- PD_R^S refere-se à pseudodistância medida entre as antenas do satélite S, no instante de transmissão, e do receptor R, no instante de recepção;

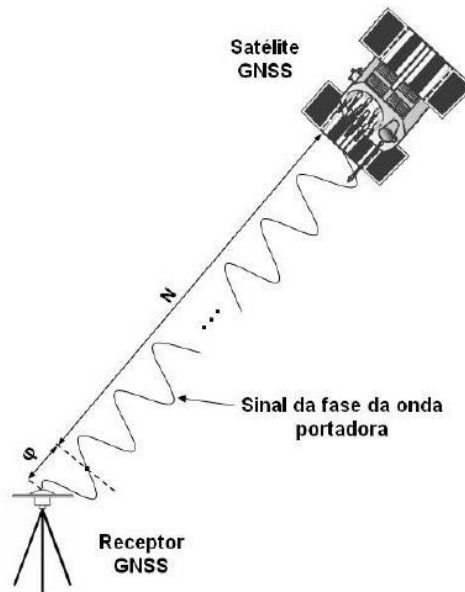
- ρ_r^s refere-se à distância geométrica entre as antenas do satélite S, no instante de transmissão, e do receptor R, no instante de recepção;
- c é a velocidade da luz no vácuo;
- dt_r É o erro do relógio do receptor r em relação ao sistema de tempo adotado por determinada constelação GNSS, como por exemplo, o erro do relógio do receptor em relação ao sistema de tempo GPS no instante de recepção do sinal;
- dt^s é o erro do relógio do satélite s em relação ao sistema de tempo adotado pela constelação GNSS ao qual o satélite pertence, como por exemplo, o erro do relógio do satélite em relação ao sistema de tempo GPS no instante de transmissão do sinal;
- T_r^s e I_r^s referem-se respectivamente aos atrasos troposférico e ionosférico;
- dm_r^s representa os efeitos de multicaminhamento e
- ε_r^s refere-se a outros efeitos não modelados e erros aleatórios.

3.2. Fase de Batimento da Onda Portadora

A observável fase de batimento da onda portadora é muito mais precisa do que a pseudodistância. Por isso, esta é a observável básica para a maioria das atividades geodésicas. Nesse texto, a observável será denominada apenas fase da onda portadora, em conformidade com a denominação mais comumente utilizada na literatura (MONICO, 2008).

Os receptores GNSS medem a parte fracionária da onda portadora e efetuam a contagem do número de ciclos que entram no receptor a partir desse momento. Desse modo, tem-se uma medida contínua. O termo N_r^s da Equação (2) corresponde ao número de ciclos inteiros, que ocorreram entre as antenas do satélite e do receptor até o momento da primeira observação. Este termo, conhecido como ambiguidade, é estimado no ajustamento, com os demais parâmetros, para, por exemplo, a execução do método PPP (MONICO, 2008). A observável fase da onda portadora é ilustrada na Figura 17.

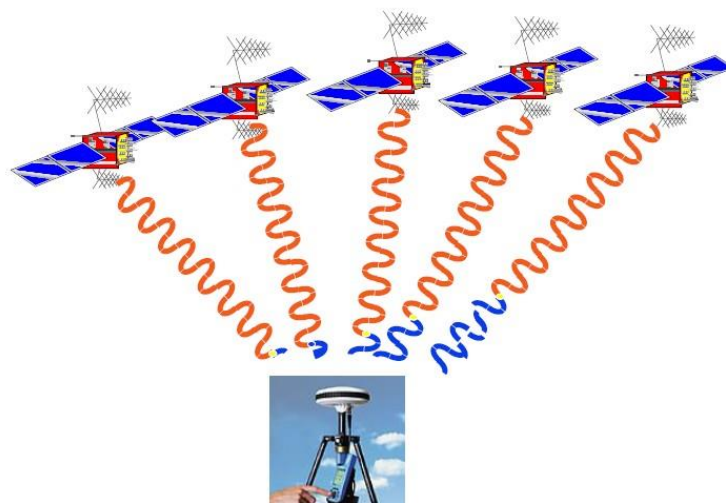
Figura 17: Ilustração da observável de fase da onda portadora.



Fonte: (MONICO, 2008).

A ambiguidade da onda portadora, entre o receptor e o satélite, permanece a mesma, até que o sinal deixe de ser rastreado ou ocorram perdas de ciclos (*cycle slips*). A ambiguidade da fase da onda portadora (representada em laranja), ao longo do tempo, é ilustrada na Figura 18 (MONICO, 2019). A porção laranja do sinal corresponde à ambiguidade.

Figura 18: Ilustração da ambiguidade da fase da onda portadora.



Fonte: (MONICO, 2019).

A Equação (2) para a observável fase da onda portadora é (Monico, Posicionamento pelo GNSS: descrição, fundamentos e, 2008):

$$\Phi_r^s(t) = f \left(\frac{\rho_r^s - I_r^s + T_r^s + dm}{c} \right) + f * [(dt_r(t) - dt^s] + [\Phi_r^s(t_0) - \Phi_r(t_0)] + N_r^s + \varepsilon_{\Phi_r}^s. \quad (2)$$

onde:

- t é o instante de recepção do sinal no receptor r ;
- $\Phi_r^s(t)$ é a fase da portadora gerada no satélite s e recebida no receptor r no instante de recepção;
- $\Phi_r(t)$ é a fase gerada no receptor no instante de recepção;
- N_r^s é a chamada ambiguidade da fase;
- $\varepsilon_{\Phi_r}^s$ é o erro da onda portadora;
- f é a frequência da onda portadora; e
- $\Phi(t_0)$ é a fase em uma época de referência t_0 e dt é o erro do relógio.

É importante destacar que, para a observável de pseudodistância o efeito da ionosfera é aditivo, enquanto, para a fase da onda portadora, o efeito é subtrativo.

3.3. Métodos de Posicionamento GNSS Absoluto

O termo posicionamento está relacionado ao processo de determinar a posição de um objeto em um determinado referencial. Caso o referencial seja o centro de massa da Terra (geocentro), o posicionamento é absoluto. Caso o referencial seja um ponto materializado de coordenadas conhecidas tem-se o posicionamento relativo. Quando o objeto está em repouso, o posicionamento é estático. Quando o objeto está em movimento tem-se o denominado posicionamento cinemático (SEGANTINE, 2005).

Os métodos de posicionamento podem também ser classificados em tempo real e pós-processados. No caso de posicionamento em tempo real, a posição do receptor é determinada quase que simultaneamente à coleta das observações. No pós-processamento, a posição do

receptor, quando a coleta das observações foi realizada, é calculada em outro momento (MONICO, 2008).

No posicionamento absoluto, também denominado Posicionamento por Ponto ou ainda Posicionamento por Ponto Simples (PPS), são utilizadas efemérides transmitidas e a observável comumente utilizada é a pseudodistância. A acurácia proporcionada por esse método é da ordem de 10 m (MONICO, 2008).

O método de Posicionamento por Ponto Preciso (PPP) utiliza como principal observável as medidas de fase da onda portadora. A modelagem de diversos erros como o atraso troposférico e ionosférico, efeitos da carga oceânica e maré terrestre, correção do centro de fase da antena, dentre outros, torna possível obter coordenadas com acurácia centimétrica ou ainda milimétrica (MONICO, 2008).

3.4. Principais Erros no Posicionamento GNSS

O posicionamento GNSS possui uma série de erros que influenciam a qualidade dos resultados obtidos. Na Tabela 2 são apresentados os principais erros no posicionamento absoluto GPS.

Tabela 2: Valores nominais dos principais erros no posicionamento absoluto GPS.

Tipo de Erro	Erro Nominal no Posicionamento Absoluto GPS
Relógio do Satélite	1 a 2 m (RMS)
Órbita do Satélite	1 a 2 m (RMS)
Atraso Ionosférico	2 a 10 m na direção zenital
Atraso Troposférico	2,3 a 2,5 m na direção zenital ao nível do mar
Multicaminho	Código: 1 a 5 m
	Fase da Portadora: 1 a 5 cm
Ruído do Receptor	Código: 0,5 m (RMS)
	Fase da Portadora: 1 a 2 mm (RMS)

RMS - *Root Mean Square*

Fonte: Adaptado de (PEREIRA, 2018).

3.5. Método PPS Implementado no Projeto

Neste projeto foram desenvolvidos softwares que implementam o Método PPS utilizando o ajustamento de observações pelo Método dos Mínimos Quadrados (MMQ) - método paramétrico, como estes são tradicionalmente encontrados na literatura. Com isso, para detalhes sobre sua formulação matemática é possível consultar (MONICO, 2008) e (SEGANTINE, 2005). Além disso, foram implementados testes para o controle de qualidade do ajustamento pelo MMQ implementado neste projeto. Para corrigir os atrasos ionosféricos e troposférico foram implementados os modelos de correção de Klobuchar (KLOBUCHAR, 1987) e Saastamoinem (SAASTAMOINEM, 1973), respectivamente.

Como primeira etapa no controle de qualidade do ajustamento das observações foi implementado o teste estatístico conhecido como Chi-quadrado (χ^2), também conhecido como Teste Global do Modelo (TGM) (SILVA, 2007).

Já como segunda etapa, é executado um método para a detecção de observações que correspondem a erros grosseiros ou *outliers*. Foi escolhido para este projeto o Método de Baarda: Data Snooping, que foi selecionado por não possui alta complexidade para implementação computacional. Outra opção seria implementar o Método de Pope: Teste Tau (MONICO, 2008). Para detalhes sobre o processo de controle de qualidade pelo método Data Snooping, é possível encontrá-los em (SILVA, 2007).

4. RECURSOS GNSS A PARTIR DA VERSÃO 7.0 DO ANDROID (NOUGAT)

Neste capítulo são discutidos os recursos de hardware e software disponíveis no sistema operacional Android, a partir da sua versão 7.0, conhecida como Android Nougat. No Capítulo 7 tais recursos serão utilizados para o desenvolvimento de softwares que processam e analisam dados GNSS brutos.

4.1. Recursos de Hardware

Nesta seção são tratadas questões ao nível do hardware dos aparelhos. São discutidas questões como as especificações de cada modelo, dentro de uma mesma linha de smartphone, além das frequências e observáveis que cada aparelho disponibiliza.

No endereço eletrônico oficial² do Google sobre medições GNSS há uma lista atualizada com as informações sobre smartphones que capturam dados GNSS e sobre os softwares desenvolvidos pela empresa para a análise desses dados, como o GnsLogger e o Gns Analysis App, apresentados na subseção 2.1.1.

Os smartphones listados no endereço eletrônico do Google possuem a frequência L1 e observável de pseudodistância da frequência L1, como recursos mínimos para estar nesta lista. Além disso, todos rastreiam ao menos a constelação GPS.

A partir da lista atualizada pelo Google, é possível verificar que, dentre todos os 58 smartphones listados:

- 15 smartphones fornecem mensagens de navegação;
- 19 smartphones fornecem a observável de fase da onda portadora;
- 11 smartphones fornecem a frequência L5 e
- 2 smartphones fornecem mensagens de navegação, fase da onda portadora e a frequência L5.

Na Tabela 3 são listados os 15 smartphones que fornecem mensagens de navegação:

² <https://developer.android.com/guide/topics/sensors/gnss>

Tabela 3: Smartphones que fornecem mensagem de navegação.

Modelo	Android	ADR (fase da operadora)	L5	Sistemas globais
Huawei Mate 20 X	9.0	sim	sim	GPS/GLO/GAL/QZS
Huawei Mate RS (Porsche Design)	8.1	sim	não	GPS/GLO/GAL
Xiaomi Mi 8	8.1	sim	sim	GPS/GLO/GAL/BDS/QZS
Huawei P20	8.1	sim	não	GPS/GLO/QZS
Samsung Galaxy S9 (Exynos)1	8.0	sim	não	GPS/GLO/GAL/QZS
Huawei Mate 10	8.0	sim	não	GPS/GLO
Huawei Mate 10 Pro	8.0	sim	não	GPS/GLO/QZS
Samsung Note 8 (Exynos)	7.1	sim	não	GPS/GLO/GAL/BDS
Huawei Honor 9	7.0	sim	não	GPS/GLO
Samsung S8 (Exynos) 2	7.0	sim	não	GPS/GLO/GAL/BDS/QZS
Huawei P10	7.0	sim	não	GPS/GLO/GAL/BDS/QZS
Huawei Honor 8	7.0	sim	não	GPS/GLO/BDS
Huawei Mate 9	7.0	sim	não	GPS/GLO/BDS
Huawei P9	7.0	sim	não	GPS/GLO/BDS
Nexus 9 (versão sem rede celular) 5	7.1	sim	não	GPS/GLO

Fonte: Adaptado de (GOOGLE, 2023).

Na Tabela 4 são listados os 19 smartphones que fornecem a fase da onda portadora.

Tabela 4: Smartphones que fornecem a fase da onda portadora.

Modelo	Android	Mensagens de navegação	L5	Sistemas globais
Huawei P30 Pro	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate 20 X	9.0	sim	sim	GPS/GLO/GAL/QZS
Huawei Mate 20 RS (Porsche Design)	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate 20 Pro	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate 20	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate RS (Porsche Design)	8.1	sim	não	GPS/GLO/GAL
Xiaomi Mi 8	8.1	sim	sim	GPS/GLO/GAL/BDS/QZS
Huawei P20	8.1	sim	não	GPS/GLO/QZS
Samsung Galaxy S9 (Exynos)1	8.0	sim	não	GPS/GLO/GAL/QZS
Huawei Mate 10	8.0	sim	não	GPS/GLO
Huawei Mate 10 Pro	8.0	sim	não	GPS/GLO/QZS
Samsung Note 8 (Exynos)	7.1	sim	não	GPS/GLO/GAL/BDS
Huawei Honor 9	7.0	sim	não	GPS/GLO
Samsung S8 (Exynos) 2	7.0	sim	não	GPS/GLO/GAL/BDS/QZS
Huawei P10	7.0	sim	não	GPS/GLO/GAL/BDS/QZS
Huawei Honor 8	7.0	sim	não	GPS/GLO/BDS
Huawei Mate 9	7.0	sim	não	GPS/GLO/BDS
Huawei P9	7.0	sim	não	GPS/GLO/BDS
Nexus 9 (sem rede celular) 5	7.1	sim	não	GPS/GLO

Fonte: Adaptado de (GOOGLE, 2023).

Na Tabela 5 são apresentados os 11 smartphones que possuem L5.

Tabela 5: Smartphones que possuem L5.

Modelo	Android	Mensagens de navegação	ADR (fase da operadora)	Sistemas globais
Xiaomi Mi 9	9.0	não	não	GPS/GLO/GAL/BDS/QZS
Samsung Galaxy Note 10+	9.0	não	não	GPS/GLO/GAL
Samsung Galaxy Note 10	9.0	não	não	GPS/GLO/GAL
Huawei P30 Pro	9.0	não	sim	GPS/GLO/GAL/BDS
One Plus 7 Pro	9.0	não	não	GPS/GLO/GAL
One Plus 7	9.0	não	não	GPS/GLO/GAL
Huawei Mate 20 X	9.0	sim	sim	GPS/GLO/GAL/QZS
Huawei Mate 20 RS (Porsche Design)	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate 20 Pro	9.0	não	sim	GPS/GLO/GAL/BDS
Huawei Mate 20	9.0	não	sim	GPS/GLO/GAL/BDS
Xiaomi Mi 8	8.1	sim	sim	GPS/GLO/GAL/BDS/QZS

Fonte: Adaptado de (GOOGLE, 2023).

Por fim, na Tabela 6 são apresentados os 2 smartphones que fornecem mensagens de navegação, fase da onda portadora e a frequência L5.

Tabela 6: Smartphones que possuem mensagens de navegação, onda portadora e L5.

Modelo	Android	Sistemas globais
Huawei Mate 20 X	9.0	GPS/GLO/GAL/QZS
Xiaomi Mi 8	8.1	GPS/GLO/GAL/BDS/QZS

Fonte: Adaptado de (GOOGLE, 2023).

A partir da análise da Tabela 6 é possível compreender o porquê de o smartphone Xiaomi Mi 8 ser utilizado em tantos trabalhos acadêmicos, como apresentado na subseção 2.1.1. Este além de ser o primeiro smartphone de dupla frequência que foi disponibilizado comercialmente (CHEM et al., 2019), continua sendo um dos únicos que disponibiliza todos os recursos GNSS que podem ser encontrados em smartphones com o sistema operacional Android.

4.2. Recursos de Software

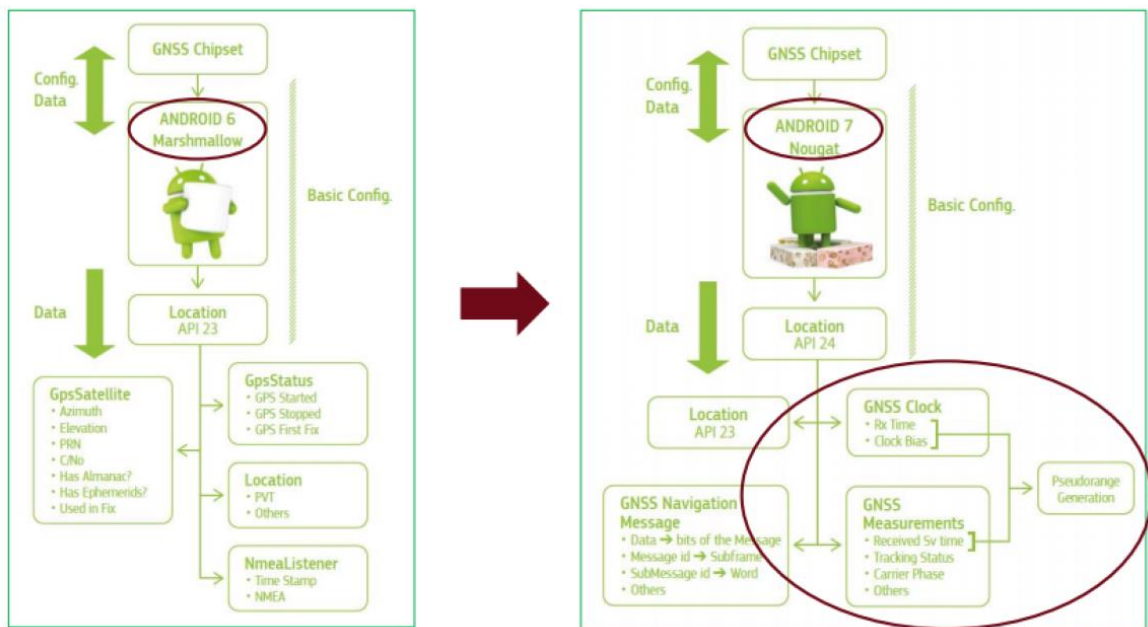
As ferramentas e recursos para o desenvolvimento de aplicativos Android são organizados segundo a definição de *Application Program Interface* (API). Cada versão de API recebe uma numeração e está associada a uma versão do sistema operacional Android. As APIs são organizadas em pacotes, sendo que cada pacote corresponde a um conjunto de classes relacionadas por funcionalidades em comum. Cada classe corresponde a uma estrutura com códigos-fonte que são utilizadas para solicitar e representar recursos ao sistema operacional Android e são referenciadas no código fonte das aplicações.

Os pacotes são organizados de modo hierárquico de acordo com sua funcionalidade. Alguns exemplos de pacotes da API Android são: *android.hardware.fingerprint* para tratar questões relacionadas à utilização do sensor de leitura de digitais e o pacote *android.bluetooth* para tratar questões relacionadas à funcionalidade de conexão *Bluetooth* do smartphone. O pacote que organiza os recursos relacionados a questões de localização, incluindo os dados GNSS brutos, é o pacote *android.location*.

Na versão 23 da API Android, as classes *GPSSatellite*, *GPSStatus* e *Location* forneciam informações básicas sobre os satélites observados e sobre a localização do

smartphone, como a última localização conhecida do smartphone. A partir da API 24, nativa da versão 7.0, é possível obter observáveis do GNSS a partir dos smartphones ou tablets que possuem hardware apropriado para isso (CAMERON, 2016). É possível acessar os dados GNSS brutos devido à adição de diversas classes ao pacote *location* como as classes *GnssClock*, *GnssMeasurements* e *GnssNavigationMessage* (ANDROID DEVELOPERS, 2023). As mudanças no pacote Location da API 23 para a API 24 do sistema operacional Android são apresentadas na Figura 19.

Figura 19: Mudanças no pacote Location da versão 23 para a versão 24 da API Android.



Fonte: (GSA, 2017).

Os principais campos das classes *GnssClock* e *GnssMeasurements* são apresentados na Tabela 7 (GSA, 2017).

Tabela 7: Campos dos dados GNSS brutos no pacote Location.

Classe do Android	Campo da Classe	Descrição
GnssClock	TimeNanos	O valor do relógio interno do chipset GNSS em nanossegundos.
GnssClock	BiasNanos	A tendência do relógio em sub-nanossegundos.
GnssClock	FullBiasNanos	Diferença entre TimeNanos do receptor GPS e o verdadeiro valor do tempo GPS, contando desde o início do tempo GPS.
GnssClock	DriftNanosPerSecond	A marcha do relógio.
GnssClock	HardwareClockDiscontinuityCount	Contador de descontinuidades de hardware do receptor GNSS.
GnssClock	LeapSecond	Segundos de salto associado ao tempo do relógio.
GnssMeasurement	Constellation Type	Identificador da constelação GNSS.
GnssMeasurement	Svid	Identificador do satélite.
GnssMeasurement	State	Estado atual de sincronização para o satélite em determinada frequência.
GnssMeasurement	ReceivedSvTimeNanos	O tempo GNSS enviado pelo satélite no instante de recepção.
GnssMeasurement	AccumulatedDeltaRangeMeters	Distância acumulada desde o último reset do canal, em metros, entre o satélite e o receptor. Valores positivos indicam que o satélite está se afastando do receptor.
GnssMeasurement	Cn0DbHz	Densidade <i>Carrier-to-Noise</i>
GnssMeasurement	TimeOffsetNanos	Diferença de tempo na qual a medição foi obtida em nanossegundos.
GnssMeasurement	CarrierFrequencyHz	Frequência da onda portadora no qual os códigos e as mensagens foram modulados.
GnssMeasurement	PseudorangeRateMetersperSecond	Taxa de variação da pseudodistância em m/s naquele instante de tempo. Não está incluída a correção do erro do relógio do receptor e do satélite. Valores positivos indicam que o satélite está se afastando do receptor.

Fonte: (GSA, 2017).

4.2.1. Mensagens de navegação

As mensagens de navegação são obtidas através da classe *GnssNavigationMessage* (caso disponíveis no aparelho) no formato de bits. É possível obter a partir do Android, mensagens de navegação do tipo GPS C/A, BeiDou D1 & D2, GLONASS L1 C/A e Galileo F/NAV e I/NAV (GSA, 2017). Como são apresentadas no formato de bits, as mensagens de navegação precisam ser decodificadas e interpretadas pelos aplicativos, como é apresentado no trecho do código fonte do aplicativo Gnss Logger na Figura 20. No quadro em verde é destacada a extração, a partir da sequência de bits da mensagem de navegação, os coeficientes af_0 e af_1 , necessários para estimar o erro do relógio do satélite.

Figura 20: Decodificação de mensagem de navegação GPS no código fonte do GnssLogger.

```

245 // the navigation message contains a modulo-1023 week number
246 int week = extractBits(WEEK_INDEX, WEEK_LENGTH, rawData);
247 week = getSpsWeekWithRollover(week);
248 gpsEphemerisProto.week = week;
249
250 int uraIndex = extractBits(URA_INDEX, URA_LENGTH, rawData);
251 double svAccuracy = computeNominalSvAccuracy(uraIndex);
252 gpsEphemerisProto.svAccuracyM = svAccuracy;
253
254 int svHealth = extractBits(SV_HEALTH_INDEX, SV_HEALTH_LENGTH, rawData);
255 gpsEphemerisProto.svHealth = svHealth;
256
257 byte tgd = (byte) extractBits(TGD_INDEX, TGD_LENGTH, rawData);
258 gpsEphemerisProto.tgd = tgd * POW_2_NEG_31;
259
260 int toc = extractBits(TOC_INDEX, TOC_LENGTH, rawData);
261 double tocScaled = toc * POW_2_4;
262 gpsEphemerisProto.toc = tocScaled;
263
264 byte af2 = (byte) extractBits(AF2_INDEX, AF2_LENGTH, rawData);
265 gpsEphemerisProto.af2 = af2 * POW_2_NEG_55;
266
267 short af1 = (short) extractBits(AF1_INDEX, AF1_LENGTH, rawData);
268 gpsEphemerisProto.af1 = af1 * POW_2_NEG_43;
269
270 // a 22-bit two's complement number
271 int af0 = extractBits(AF0_INDEX, AF0_LENGTH, rawData);
272 af0 = getTwoComplement(af0, AF0_LENGTH);
273 gpsEphemerisProto.af0 = af0 * POW_2_NEG_31;
274
275 updateDecodedState(prn, SUBFRAME_1, intermediateEphemeris);

```

Fonte: (O AUTOR, 2020).

5. ALTERNATIVAS À AUSÊNCIA DE MENSAGENS DE NAVEGAÇÃO

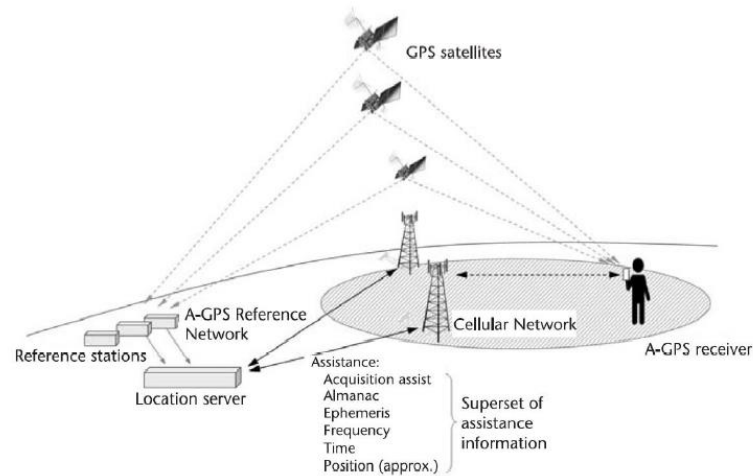
Neste projeto foram utilizados os smartphones Xiaomi Mi 9 e Huawei P30 Pro e estes não recebem diretamente mensagens de navegação, como apresentado no Capítulo 3. Portanto, nesse Capítulo é abordado o conceito de A-GNSS, que foi utilizado nesse projeto para contornar o problema da falta de efemérides durante o processamento em tempo real dos dados GNSS.

5.1. Assisted-GNSS (A-GNSS)

A técnica denominada *Assisted-GNSS* (A-GNSS) tem por objetivo aprimorar o desempenho de receptores GNSS comuns. Através do A-GNSS são enviadas aos receptores, informações iguais, ou equivalentes, às enviadas pelos satélites, através de um meio de transmissão alternativo. Mesmo recebendo essas informações de forma alternativa, o receptor GNSS ainda precisa continuar processando os sinais dos satélites. A principal vantagem da utilização do A-GNSS é diminuir o *Time To First Fix* (TTFF), ou seja, o tempo necessário para que o receptor seja capaz de determinar sua posição pela primeira vez (LIMA, 2018). Isso acontece quando o receptor obtém observáveis de pelo menos, quatro satélites.

Através do A-GNSS informações como o almanaque e efemérides transmitidas, normalmente enviadas pelos satélites GNSS a uma taxa de transmissão de 5 bits por segundo, podem ser obtidas através de redes como a Internet ou através das redes de transmissão de telefonia móvel, como é ilustrado na Figura 21. Do mesmo modo, é possível obter, através de bancos de dados com informações das torres de transmissão, uma posição aproximada do receptor GNSS que pode ser utilizada como valor inicial para o cálculo da posição do receptor (LIMA, 2018).

Figura 21: Ilustração do conceito de A-GNSS.

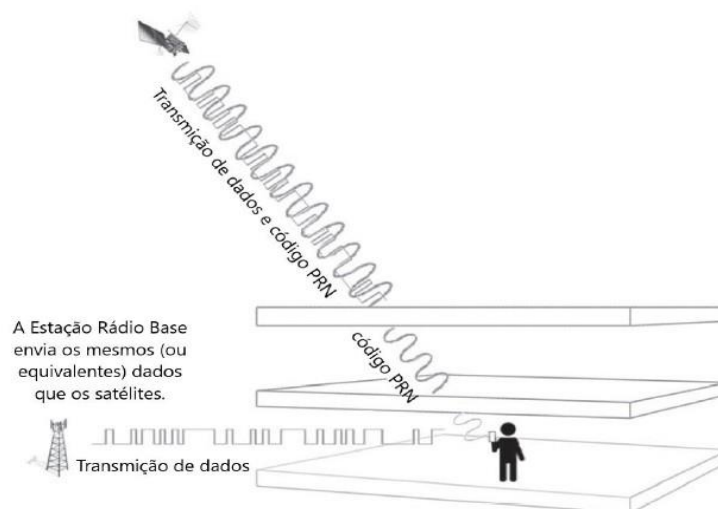


Fonte: (LIMA, 2018).

Além de fornecer informações de modo mais rápido, como apresentado no exemplo da Figura 21, a técnica de A-GNSS também pode ser utilizada para suprir a falta de determinados dados necessários ao receptor GNSS.

Um exemplo nesse sentido é quando os dados modulados na onda portadora sofrem obstrução. Como é apresentado na Figura 22, cada satélite GNSS envia um código acrescentado de um Ruído Pseudoaleatório - *Pseudorandom Noise* (PRN), além de um fluxo de dados (LIMA, 2018). Na Figura 22 o código PRN é ilustrado pelas ondas senoidais e os dados, como as efemérides transmitidas, pelas ondas quadradas.

Figura 22: Transmissão do código e dados pelo sistema A-GNSS.



Fonte: (LIMA, 2018).

Quando o sinal é obstruído este é atenuado e nesse caso, os dados podem ficar imperceptíveis ao receptor GNSS, mas o código ainda ser detectável. Com o sistema A-GNSS, as mesmas informações, ou equivalentes, podem ser acessadas através de Estações de Rádio Base, ou via SUPL (que será abordado na subseção 5.1.2). Desse modo, o receptor pode funcionar como se os dados enviados pelo satélite não tivessem sido perdidos (LIMA, 2018).

5.1.1. Networked Transport of RTCM via Internet Protocol (Ntrip)

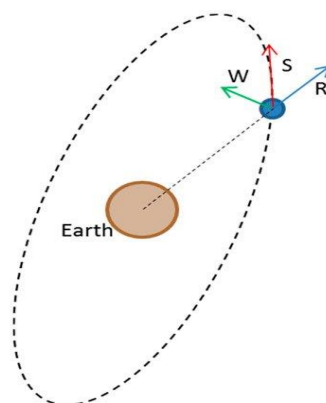
Como apresentado na Seção 3.2, grande parte dos smartphones não fornecem mensagens de navegação e, com isso, não é possível obter as efemérides transmitidas e aplicar técnicas de posicionamento em tempo real. Para contornar esse problema, é possível utilizar o protocolo Ntrip, tendo em vista que os smartphones podem conectar-se à Internet por uma conexão Wi-Fi ou 4G. Isso implica em utilizar uma estratégia do tipo A-GNSS, pois ao utilizar o Ntrip, um aplicativo pode receber os dados que não são disponibilizados pelo receptor GNSS do smartphone.

O *Networked Transport of RTCM via Internet Protocol* (Ntrip) é um protocolo implementado com base no protocolo *Hypertext Transfer Protocol* (HTTP) para a transmissão de mensagens do tipo *Radio Technical Commission for Maritime Services* (RTCM) pela Internet em tempo real. O Ntrip é o protocolo adotado pelo *International GNSS Service* (IGS) para a divulgação de seus produtos em tempo real.

Os padrões RTCM são utilizados internacionalmente para a transmissão de mensagens do tipo *Differential GNSS* (DGNSS). O *Realtime IGS* (RTS) implementa a especificação RTCM SC-104. O RTCM encontra-se atualmente na versão 3.3 (RTCM, 2019). As mensagens RTCM obtidas por meio do protocolo Ntrip compreendem as efemérides transmitidas e correções de relógio e às órbitas transmitidas. As correções são definidas segundo a especificação *State Space Representation* (SSR) e modeladas individualmente para cada satélite (BKG, 2019).

Para a correção das órbitas dos satélites são obtidos valores na forma de diferenças de coordenadas em um sistema de referência do tipo RSW (*radial, along-track, cross-track*) cuja origem é o centro de massa do próprio satélite. Para a correção dos relógios dos satélites são dados valores para corrigir os erros de seus relógios em relação a seus respectivos sistemas de tempo (NICOLINI; CAPORALI, 2018). Na Figura 23 é ilustrado um sistema de referência do tipo RSW.

Figura 23: Sistema de referência RSW.



Fonte: (NICOLINI; CAPORALI, 2018).

Portanto, utilizando o Ntrip torna-se possível obter efemérides para a aplicação de técnicas de Posicionamento por Ponto e utilizando as correções em tempo real, além de outros dados, aplicar o método de Posicionamento por Ponto Preciso em smartphones e em tempo real, mesmo que estes dispositivos não forneçam mensagens de navegação.

5.1.2. Secure User Plane Location (SUPL)

Para a obtenção de efemérides transmitidas, uma alternativa ao protocolo Ntrip, é o protocolo *Secure User Plane Location* (SUPL). O protocolo SUPL tem sua especificação mantida pela *Open Mobile Alliance (OMA) SpecWorks*.

A OMA SpecWorks é uma organização com o intuito de definir e manter especificação relacionadas a protocolos de comunicação móvel, como protocolos para smartphones e *Internet of Things* (IoT). A OMA integra empresas com interesses em comum e oferece ferramentas para a definição e manutenção de especificações, como processos bem definidos para a criação destas e de ferramentas para testes (OMA SPECWORKS, 2020).

As especificações em desenvolvimento pela OMA SpecWorks podem ser encontradas publicamente em seu *website* oficial³.

Para o protocolo SUPL existem duas especificações: *OMA Secure User Plane Location (SUPL)* e *OMA SUPL Configuration Services (SUPLCS)*.

A especificação SUPLCS é utilizada para que operadores desenvolvam serviços capazes de distribuir informações sobre configurações SUPL. As informações disponibilizadas

³ <https://www.openmobilealliance.org/wp/>

incluem, dentre outros, o nome e versão do serviço e as aplicações conectadas ao serviço. A última versão do SUPLCS é a versão 1.0, aprovada em 2014 (OMA SPECWORKS, 2014).

Já a especificação SUPL é utilizada para a obtenção de dados A-GNSS a *softwares* cliente que acessam servidores que implementam o protocolo SUPL. O SUPL pode ser utilizado em redes baseadas no *Internet Protocol* (IP), como a Internet, inclusive via WiFi e 4G. A última versão do SUPL é a versão 2.0.5, aprovada em 2019 (OMA SPECWORKS, 2019).

Na especificação SUPL são definidas as mensagens do protocolo bem como seus parâmetros. Exemplos de mensagens são SUPL INIT e SUPL RESPONSE, para respectivamente, iniciar a comunicação com o servidor e receber informações do servidor. Exemplos de parâmetros que podem ser enviados nas mensagens são: as constelações GNSS requisitadas e um valor, em horas, para a validade de efemérides solicitadas (OMA SPECWORKS, 2019). Um trecho de uma mensagem SUPL é apresentado na Figura 24.

Figura 24: Trecho de mensagem SUPL para a solicitação de dados A-GNSS.

```
GPSTime ::= SEQUENCE {
    gpsWeek      INTEGER (0..1023),
    gpstowhour   INTEGER (0..167),
    ...}

GanssRequestedCommonAssistanceDataList ::= SEQUENCE {
    ganssReferenceTime      BOOLEAN,
    ganssIonosphericModel   BOOLEAN,
    ganssAdditionalIonosphericModelForDataID00      BOOLEAN,
    ganssAdditionalIonosphericModelForDataID11      BOOLEAN,
    ganssEarthOrientationParameters                 BOOLEAN,
```

Fonte: (OMA SPECWORKS, 2019).

No primeiro trecho, correspondente a GPSTime o tempo é GPS é definido em número da semana e hora da semana GPS, já na segunda parte, correspondente a *GanssResquestedCommonAssistanceDataList* é definida uma lista do tipo booleano em relação aos parâmetros de tempo de referência, parâmetros de modelo Ionosférico e parâmetros de orientação da Terra. Essas informações podem, portanto, serem enviadas aos usuários, caso solicitadas (OMA SPECWORKS, 2019).

5.2. Obtenção de Efemérides Transmitidas Através do Servidor SUPL

Entretanto, as mensagens RTCM, transmitidas pelo Ntrip, precisam ser decodificadas bit a bit e construídas em objetos utilizáveis nas linguagens de programação. Além disso, para utilizar o Ntrip o usuário deve possuir necessariamente um cadastro prévio em algum *Caster* Ntrip que disponibilize esses dados via Internet.

Na Figura 25 são apresentadas mensagens RTCM obtidas via Ntrip, utilizando a linguagem de programação Java. As mensagens são compostas por bits e são apresentadas na imagem em formato de caracteres para identificação visual, porém a decodificação desses dados ocorre ao processar cada bit que constitui a mensagem individualmente. Uma alternativa ao protocolo Ntrip nos casos em que é necessário utilizar a Internet para obter apenas as efemérides transmitidas é o protocolo SUPL.

Figura 25: Mensagens RTCM obtidas via Ntrip em uma aplicação Java.

```
tamanho de dado : 10
a:12
Numero da mensagem : 160
0i,,i0000<~"0+Ü1â·|v0-€8<          ?šÜ]Jx'x?úí...zb #00é0ã30fç
64
a0jÃ~*ô:00@tam:768
tam:768
A!0000?i0-0çü? ;k00a!êc0I€0;†0ãY|cbçx?úí{„"-ÿ00=L0äbiYù'Ä0"0ëjÓ@A!E00±„i0>0Bie;
64
0iE`8i*Q:VF0py|bêçx€
™34e0Äê»éX^"YÄÄ
K í" {e^ó@A!Ä00(0i0ó €Dzke0Ä†0çzužâ6t00?T0ç0|000x?p?g0pÄ
64
0ç'q:éó#0 ÈÿúoÈÄ0u000@A"000'0i0?·Eë0e0
t0..Ñ*xÜ»H0l0ç 0ù].â0x€0bÄ'
ta*      ?#00ç?ci?ù€FOÜxý[Ó@A"...È010
64
```

Fonte: (O AUTOR, 2020).

O Google implementou um servidor que utiliza o protocolo SUPL no endereço *supl.google.com:7275* para, dentre outras funções, disponibilizar efemérides transmitidas, acessíveis pela Internet, e não requer cadastro prévio do usuário. Além disso, utilizando a biblioteca para acessar o servidor SUPL disponibilizada pelo Google é possível obter diretamente os objetos instanciados na linguagem de programação Java. Os objetos contêm as efemérides para cada satélite em formato de objeto Java pronto para ser utilizado pelos aplicativos Android.

Na Figura 26 é apresentado o primeiro elemento da lista retornada pelo servidor SUPL do Google com as efemérides para um satélite da constelação GPS. São destacados na imagem os campos af_0 e af_1 que já estão instanciados em variáveis do tipo *double* e podem ser imediatamente acessados e utilizados para processamento, ao contrário do que acontece com os dados obtidos via Ntrip, apresentados na Figura 26, onde são disponibilizados dados em formato de bits que precisam ser decodificados.

Figura 26: Efemérides transmitidas obtidas via servidor SUPL em uma aplicação Java.

ephResponse	#2447	EphemerisResponse
listEphemris	"size = 96"	List<GnssEphemeris>
[0]	#2343	GpsEphemeris
Herdado(a)		
af0S	-3.2918574288487434E-4	double
af1SecPerSec	-1.2505552149377763E-11	double
af2SecPerSec2	0.0	double
health	0	int
iode	45	int
keplerModel	#2473	KeplerianModel
svid	1	int
tocS	252000.0	double
tbxS	0.0	double
week	2097	int
accuracyM	0.0	double
codeL2	1	int
fitIntvFlag	false	boolean
iodc	45	int

Fonte: (O AUTOR, 2020).

Com isso, a aplicação Android desenvolvida neste projeto, solicita ao servidor SUPL do Google, efemérides transmitidas das constelações GPS, Galileo e GLONASS, além dos parâmetros dos modelos Ionosféricos de Klobuchar e NeQuick. São utilizadas efemérides GPS para o cálculo da posição dos satélites e os parâmetros *alpha* e *beta* do modelo de Klobuchar, para a aplicação deste, durante a execução do PPS em tempo real.

6. ARMAZENAMENTO DE DADOS GNSS

Nesse Capítulo são discutidas as possibilidades para o armazenamento de dados GNSS em smartphones com o sistema operacional Android. Também são abordadas quais as possibilidades dentre os diversos tipos de arquivos e banco de dados disponíveis. No fim do Capítulo, após a apresentação de tais possibilidades, é discutido sobre quais foram as opções escolhidas para o desenvolvimento dos softwares implementados neste projeto.

6.1. Armazenamento de Dados no Sistema Operacional Android

No sistema operacional Android existem diversas opções para que aplicações efetuem o salvamento de dados. Cada solução deve ser adotada considerando o tipo de dados a ser armazenado e se estes dados precisam ser privados ou acessíveis a outras aplicações e ao próprio usuário.

Existem três tipos principais de armazenamento no sistema operacional Android, a saber:

- Armazenamento interno de arquivos;
- Armazenamento de arquivos externo; e
- Banco de dados.

6.1.1. Armazenamento interno

Nesse tipo de armazenamento, os dados são gravados no espaço de armazenamento interno do smartphone. Por padrão, esses arquivos são privados e não podem ser acessados por outros aplicativos. Quando um aplicativo é instalado no sistema operacional Android, o Android cria um diretório privado no sistema de arquivos para cada aplicativo. Todos os documentos criados pela própria aplicação, como arquivos de texto no formato *Comma-Separated Values* (CSV), imagens, dentre outros, podem ser salvos nesse diretório (DESENVOLVEDORES ANDROID, 2020).

Caso o usuário desinstale o aplicativo, todos os dados salvos nesse diretório serão removidos. Portanto, essa opção de armazenamento não deve ser utilizada para dados que precisam ser mantidos independentemente do aplicativo em questão. Por exemplo, mesmo que um aplicativo que permite ao usuário capturar fotos for desinstalado, as fotos devem continuar

armazenadas no aparelho, mesmo após a desinstalação do aplicativo (Desenvolvedores Android, 2020).

6.1.2. Armazenamento externo

Todos os dispositivos com o sistema operacional Android podem utilizar um espaço de armazenamento compartilhado, conhecido como “armazenamento externo”. Esse espaço é denominado externo porque não é garantido que esteja acessível. Exemplos de dispositivos de armazenamento externo são cartões SD e pen drives que podem ser conectados ao smartphone, diretamente, via conexão do tipo *Universal Serial Bus* (USB). Esses dispositivos podem ou não estar presentes no smartphone e podem ser removidos fisicamente pelo usuário a qualquer momento, inclusive durante a execução da aplicação (DESENVOLVEDORES ANDROID, 2020).

Esse tipo de armazenamento é utilizado para dados de usuários que precisam ser disponibilizados a outros aplicativos, mesmo que o aplicativo que criou esses dados seja desinstalado. Exemplos de dados adequados ao armazenamento externo são, por exemplo, fotos capturadas ou arquivos obtidos por download. No espaço de armazenamento externo, são definidos diretórios públicos e privados (DESENVOLVEDORES ANDROID, 2020).

Os diretórios públicos servem como padrão para o armazenamento de tipos de arquivos específicos, como arquivos multimídia e obtidos por download. Os diretórios públicos não são removidos ou criados pelos aplicativos, sendo gerenciados automaticamente pelo Android (DESENVOLVEDORES ANDROID, 2020).

Os diretórios privados são criados especificamente e automaticamente, quando uma aplicação é instalada. Esses diretórios são excluídos quando a aplicação for desinstalada. Esta pode ser uma alternativa útil ao armazenamento interno, quando for requerido mais espaço, porém, não há garantias de que os dados sempre estarão disponíveis. Isso acontece porque a qualquer momento, o usuário pode remover os dispositivos de armazenamento externo (DESENVOLVEDORES ANDROID, 2020).

6.1.3. Tipos de arquivos para armazenamento de dados espaciais

Nessa Subseção são apresentados os principais tipos de arquivos para o armazenamento de dados GNSS no Android. Esses arquivos podem ser incluídos tanto no armazenamento interno quanto no armazenamento externo do smartphone.

Comma Separated Value (CSV)

O tipo de arquivo *Comma Separated Value (CSV)*, armazena os dados em texto plano e é caracterizado por organizar as informações de modo tabular. De modo geral, na primeira linha do arquivo, é definido um cabeçalho contendo a descrição dos dados. Nas linhas subsequentes, são armazenados os valores, separados por vírgula, daí o nome do arquivo (CARNES, 2019). Na Figura 27 é apresentado um exemplo de arquivo do tipo CSV. Nesse exemplo, a primeira linha representa o cabeçalho com a descrição de cada informação armazenada, seguida de cada registro em uma nova linha.

Figura 27: Exemplo de arquivo CSV.

```
name, id, favorite food
quincy, 1, hot dogs
beau, 2, cereal
abbey, 3, pizza
mrugesh, 4, ice cream
```

Fonte: (CARNES, 2019).

Arquivos CSV podem ser criados em praticamente qualquer editor de textos e comumente são utilizados como saída de programas que tratam dados. Isso ocorre, tendo em vista que arquivos CSV podem ser facilmente importados para softwares que processam dados de modo tabular, como por exemplo, o Microsoft Excel, OpenOffice Calc, ou Google Planilhas (CARNES, 2019). Na Figura 28 é apresentado como o arquivo CSV da Figura 27 ficaria em um programa de edição de planilhas como o Microsoft Excel.

Figura 28: Exemplo de arquivo CSV importado para um software de edição de planilhas.

	A	B	C
1	name	id	favorite food
2	quincy		1 hot dogs
3	beau		2 cereal
4	abbey		3 pizza
5	mrugesh		4 ice cream

Fonte: (CARNES, 2019).

GPS Exchange Format (GPX)

O Formato *GPS eXchange Format* (GPX) é um formato⁴ de arquivo em texto plano, baseado em arquivos do tipo *eXtensible Markup Language* (XML) para o armazenamento de coordenadas geodésicas na forma de pontos ou percursos.

Os arquivos GPX são utilizados para compartilhar dados GPS e disponibilizá-los a outros usuários, além da criação de mapas. Arquivos GPX são facilmente importados por diversos softwares como Google Earth, OpenStreet Map e QGIS para iniciativas de mapeamento colaborativo. Na Figura 29 é apresentado a página de um arquivo GPX compartilhado no OpenStreet Map e na Figura 30 é apresentado um trecho do respectivo arquivo (OPENSTREET MAP, 2020). Foram armazenadas as informações de latitude, longitude, elevação, *Horizontal Dilution of Precision* (HDOP) e horário de cada ponto. Essas informações podem variar conforme as *tags* do GPX utilizadas pelos softwares que geram os arquivos GPX.

⁴ Disponível em: <https://fileinfo.com/extension/gpx> Acesso em: 06 fev 2023.

Disponível em: <https://www.topografix.com/gpx.asp> Acesso em: 06 fev 2023.

Figura 29: Exemplo de arquivo GPX no OpenStreet Map.



Fonte: (OPENSTREET MAP, 2020).

Figura 30: Trecho de arquivo GPX.

```

1  <?xml version='1.0' encoding='UTF-8' standalone='yes' ?>
2  <gpx version="1.1" creator="OsmAnd 3.7.4" xmlns="http://www.topografix.com/GPX/1/1"
   "http://www.topografix.com/GPX/1/1 http://www.topografix.com/GPX/1/1/gpx.xsd">
3  <metadata>
4    <name>2020-07-23_09-34_Thu</name>
5  </metadata>
6  <trk>
7    <trkseg>
8      <trkpt lat="51.0359371" lon="-0.0277844">
9        <ele>142.509</ele>
10       <time>2020-07-23T08:34:53Z</time>
11       <hdop>3.216</hdop>
12     </trkpt>
13     <trkpt lat="51.0359624" lon="-0.027824">
14       <ele>137.263</ele>
15       <time>2020-07-23T08:35:04Z</time>
16       <hdop>3.216</hdop>
17     </trkpt>

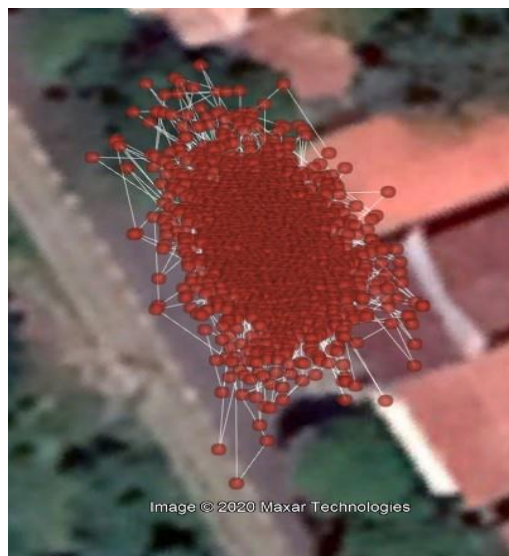
```

Fonte: (OPENSTREET MAP, 2020).

Keyhole Markup Language (KML)

O tipo de arquivo *Keyhole Markup Language (KML)* é um tipo de arquivo, baseado em XML, para tratar informações espaciais utilizado por softwares do tipo *Geographic Information System (GIS)*. Trata-se de arquivos em texto plano que contém, de modo geral, informações relacionadas a criação de marcadores em mapas, gerados a partir de coordenadas de latitude e longitude, polígonos, descrições, modelos geográficos 2D e 3D e efeitos de animação, como zoom, contidos nos arquivos KML. O tipo de arquivo KML foi adotado como

Figura 33: Arquivo KML visualizado no Google Earth.



Fonte: (O AUTOR, 2020).

6.2. Armazenamento de Dados GNSS a Partir do Android 7.0

É importante destacar que antes da versão 7.0 do Android, já existiam aplicativos para a coleta e compartilhamento de dados GNSS em smartphones, porém, uma de suas limitações era em relação aos tipos de arquivos que estes aplicativos podiam oferecer.

Essa limitação se dava ao fato de que os dados GNSS brutos não eram disponibilizados pelos smartphones. Com isso, os principais tipos de arquivos gerados por smartphones eram arquivos do tipo GPX e KML, com os resultados de posição e velocidade já calculados pelo chipset GNSS do smartphone.

Por outro lado, a partir do Android 7.0, podem ser desenvolvidos aplicativos que geram dados nos formatos CSV, para disponibilizar os dados brutos capturados pelo smartphone. Também foram desenvolvidos aplicativos que geram arquivos nos formatos RINEX de observação e navegação, nas versões 2 e 3. A partir daí, tornou-se possível o pós-processamento de dados GNSS capturados em smartphones, uma vez que as observáveis GNSS capturados por estes dispositivos podem agora ser armazenadas e compartilhadas para pós-processamento.

Uma vez que os dados GNSS brutos capturados por smartphones são acessíveis via programação, uma possibilidade para o compartilhamento desses dados é transmitindo arquivos CSV entre diferentes softwares para o processamento dos dados. Além disso, a implementação mais complexas do que utilizar arquivos de texto, como a implementa de bancos de dados

tradicionais e a utilização de serviços de armazenamento em nuvem são possibilidades para explorar esses dados, como é apresentado nessa seção.

6.2.1. Receiver Independent Exchange Format (RINEX)

A partir da versão 7.0 do Android é possível criar arquivos do tipo *Receiver Independent EXchange format* (RINEX) utilizando smartphones com hardware compatível (GSA, 2017). Esses arquivos agora podem ser armazenados no armazenamento interno ou externo dos smartphone, antes de serem compartilhados.

O tipo de arquivo RINEX foi criado para que fosse possível armazenar e transmitir dados das efemérides transmitidas e observáveis obtidos pelos receptores GNSS sem precisar conhecer o formato de arquivo de cada receptor. RINEX são arquivos de texto no formato ASCII. Os fabricantes costumam fornecer meios para a conversão dos dados no formato próprio do receptor para o formato RINEX (MONICO, 2008).

Existem as versões 1, 2 e 3 do RINEX, gerenciadas pelo IGS, que separam os dados em diferentes tipos de arquivos. Em todas as versões dos arquivos há uma seção de cabeçalho e a área de dados. Na seção de cabeçalho há a identificação do tipo de arquivo e comentários explicando o significado de cada linha de dados do arquivo (MONICO, 2008).

Os dois principais tipos de arquivos RINEX são os arquivos de observação e os de navegação. Os arquivos de observação contêm as observáveis coletadas pelos receptores GNSS, como a pseudodistância e fase da onda portadora. Nos arquivos RINEX de navegação são armazenados dados, como os elementos Keplerianos das efemérides transmitidas, que são utilizados para os cálculos de posição dos satélites em determinado instante de tempo. Também estão presentes nos arquivos de navegação parâmetros para a correção dos erros dos relógios dos satélites (MONICO, 2008) e (VANI; MONICO; SHIMABUKURO, 2014). Os parâmetros para modelos de correção ionosférica como, por exemplo, o de Klobuchar também podem ser encontrados no cabeçalho de arquivos RINEX de navegação.

Dentre os principais aplicativos para a criação de arquivos RINEX utilizando *smartphones* com Android estão: o *rinex ON* e *Geo++ RINEX Logger*. O aplicativo desenvolvido neste projeto gera RINEX de observação, para pseudodistância e onda portadora GPS no formato 3.04. Na Figura 34 é apresentado um trecho de arquivo RINEX de observação gerado pelo aplicativo Android desenvolvido nesse projeto.

Figura 34: Trecho de arquivo RINEX de observação na versão 3.

```

HUAWEI_VOG-L29_2021_12_08_08_54_51_DESKTOP_RINEX.2to C3
1      3.04      OBSERVATION DATA      M: Mixed      RINEX VERSION / TYPE
2 PPS_FCT_INCT      INCT NavAer - UNESP      2021128 115514 UTC PGM / RUN BY / DATE
3 *****COMMENT*****
4 Created by PPS-FCT-INCT      COMMENT
5 Developed by INCT GNSS-NavAer - Brazil      COMMENT
6 Contact: https://inct-gnss-NavAer.fct.unesp.br/en/index.php      COMMENT
7 *****COMMENT*****
8 INCT NavAer - UNESP      MARKER NAME
9 HUMAN      MARKER TYPE
10 INCT NavAer - UNESP INCT NavAer - UNESP      OBSERVER / AGENCY
11 unknown      null      null      REC # / TYPE / VERS
12 unknown      null      null      ANT # / TYPE
13 0000000.0000 0000000.0000 0000000.0000      APPROX POSITION XYZ
14 0.0000      0.0000      0.0000      ANTENNA: DELTA H/E/N
15 G 8 C1C L1C D1C S1C C5Q L5Q D5Q S5Q      SYS / # / OBS TYPES
16 R 4 C1C L1C D1C S1C      SYS / # / OBS TYPES
17 E 12 C1B L1B D1B S1B C1C L1C D1C S1C C5Q L5Q D5Q S5Q      SYS / # / OBS TYPES
18 C 4 C2I L2I D2I S2I      SYS / # / OBS TYPES
19 2021 12 8 11 55 14.0      GPS      TIME OF FIRST OBS
20 24 R01 1 R02 -4 R03 5 R04 6 R05 1 R06 -4 R07 5 R08 6 GLONASS SLOT / FRQ #
21 R09 -2 R10 -5 R11 0 R12 -1 R13 -2 R14 -7 R15 0 R16 -1 GLONASS SLOT / FRQ #
22 R17 4 R18 -3 R19 3 R20 2 R21 4 R22 -3 R23 3 R24 2 GLONASS SLOT / FRQ #
23 G L1C      SYS / PHASE SHIFT
24 G L5Q -0.25000      SYS / PHASE SHIFT
25 R L1C      SYS / PHASE SHIFT
26 E L1B      SYS / PHASE SHIFT
27 E L1C +0.50000      SYS / PHASE SHIFT
28 E L5Q -0.25000      SYS / PHASE SHIFT
29 C L2I      SYS / PHASE SHIFT
30 C1C 0.000 C1P 0.000 C2C 0.000 C2P 0.000      GLONASS COD/PHS/BIS
31 *****END OF HEADER*****
32 > 2021 12 8 11 55 15.0 0 8
33 G02 20641122.648 110299901.148 -1043.325 37.000
34 G05 25035364.178 133781395.572 -1691.414 37.000
35 G06 22654636.721 121059509.868 2117.597 38.000
36 G12 22022401.409 117681036.048 -1205.189 27.000
37 G20 23946277.837 127961648.37 -1949.246 36.000
38 G24 20074861.463 107273973.06 -150.864 38.000

```

Fonte: (O AUTOR, 2023).

6.2.2. Bancos de dados

Tipos de Banco de dados

Um banco de dados é um conjunto de dados, relativos a determinado domínio ou ambiente, por exemplo, uma empresa ou universidade, armazenado em um ou vários computadores. Esses dados guardam algum tipo de relacionamento entre si (FÁTIMA, 2019).

Os *softwares* responsáveis por gerenciar o armazenamento e acesso aos dados são conhecidos como softwares do tipo Sistema Gerenciador de Banco de Dados (SGBD). Existem diversas classificações para bancos de dados, considerando como os dados são estruturados e manipulados pelos SGBDs, bem como as infraestruturas de hardware e software utilizadas nesses sistemas (FÁTIMA, 2019).

Os bancos de dados são divididos em dois principais tipos, sendo eles: bancos de dados relacionais e banco de dados não relacionais (FÁTIMA, 2019).

Bancos de dados relacionais

Bancos de dados Relacionais utilizam SGBDS do tipo Sistema Gerenciador de Banco de Dados Relacional (SGBDR). A principal característica desse tipo de banco de dados é a utilização de esquemas para definir a estrutura dos dados que serão armazenados no banco de dados (FÁTIMA, 2019).

A estrutura dos dados que serão armazenados é previamente definida, na forma de tabelas. Os dados são armazenados na forma de linhas, representando cada registro armazenado, e colunas, que representam os atributos de cada registro. A junção das linhas e colunas definem as tabelas, o que representam as relações entre as informações (FÁTIMA, 2019).

Em bancos de dados relacionais é utilizada a linguagem denominada *Structured Query Language* (SQL) para a definição do esquema do banco de dados, criando as tabelas que armazenarão os dados. A linguagem SQL se tornou um padrão da *International Organization for Standardization* (ISO) em 1987. Utilizando a linguagem SQL é possível, além de definir o esquema do banco de dados, manipular os dados armazenados. É possível inserir, alterar, ou excluir dados, além de criar consultas para extrair informações detalhadas, a partir de diversas tabelas, utilizando SQL (SAM, 2020). Os comandos SQL podem ser embutidos em códigos de outras linguagens de programação, como C ou Java, ou ser utilizados diretamente nos SGBDRs.

Cada tabela em um banco de dados relacional possui um campo de identificação único para cada registro. Com isso, a alteração dos dados de uma tabela implica em alterações em todos os dados associados a essa tabela. Por exemplo, é possível criar uma tabela para clientes, com um identificador para o cliente, além de demais informações como nome e endereço e uma tabela de produtos, com informações como um identificador e o preço do produto. Com essas duas tabelas, uma tabela de vendas pode conter apenas os identificadores do cliente e do produto. Assim, alterações nas informações do cliente, como a atualização de seu endereço, serão automaticamente atualizadas para usuários, que, por exemplo, visualizem a informação de determinada compra daquele cliente. Qualquer alteração em qualquer tabela afetará as tabelas associadas de maneira previsível e sistemática (FÁTIMA, 2019), (SAM, 2020).

A integridade dos dados é garantida pelas restrições impostas pelo SGBDR. Através do SGBDR é garantido que os dados armazenados são precisos e confiáveis. Exemplos de bancos de dados relacionais são: o MySQL, que possui código fonte aberto, PostgreSQL, Microsoft SQL Server e Oracle (FÁTIMA, 2019).

As principais vantagens dos bancos de dados relacionais é sua previsibilidade. Como são criados a partir de esquemas estritos, cada novo registro deve se ajustar a um modelo pré-estabelecido. Com isso, os dados são facilmente avaliáveis. Como esse tipo de banco de dados é muito bem estruturado, é pequeno o espaço para a ocorrência de erros (FÁTIMA, 2019).

Como desvantagem, os bancos de dados relacionais, devidos a seus esquemas altamente restritivos, apresentam problemas para escalar em quantidade de dados como é observado na Internet atualmente. Além disso, devido às restrições de esquema, é difícil a migração de dados entre diferentes SGBDRs (FÁTIMA, 2019).

Banco de dados não relacionais

Os bancos de dados não relacionais, ao contrário do apresentado anteriormente, não possuem estrutura⁵ rígida. Em vez de tabelas, são definidas coleções. Cada coleção pode possuir vários documentos e os documentos podem ou não seguir um padrão específico. Com isso, os bancos de dados não relacionais também são conhecidos como bancos de dados Não-SQL (*NoSQL*) (FÁTIMA, 2019).

Uma coleção pode, por exemplo, possuir um documento com nome e endereço e outro documento, com apenas o nome. Isso é possível porque não há um esquema pré-definido para os documentos (FÁTIMA, 2019).

Os principais tipos de bancos de dados não relacionais são:

- Armazenamento chave-valor;
- Banco de dados orientado a documentos;
- Bancos de dados baseados em grafos;
- Motores de busca; e
- Bancos de dados baseados em nuvem.

Os bancos de dados do tipo armazenamento chave-valor, orientados a grafos e motores de busca possuem uso específico e não foram implementados neste projeto. Com isso tem-se o foco dessa Subseção nos bancos de dados orientados a documentos e em nuvem.

⁵ Disponível: Acesso em: 09 jan 2023.

Banco de dados orientado a documentos

Este tipo de banco de dados é caracterizado por armazenar documentos no formato JSON. Com isso, os dados armazenados podem ter uma grande variedade de tipos e valores e estes valores podem ser organizados hierarquicamente. Alguns exemplos são o Couchbase e MongoDB (FÁTIMA, 2019).

Banco de dados baseados em nuvem

Esse tipo de banco de dados foi desenvolvido para oferecer soluções em ambientes de computação em nuvem, seja em redes privadas ou públicas, como a Internet.

Utilizando bancos de dados em nuvem é possível construir aplicações que funcionam, aproveitando-se de recursos disponíveis graças à utilização da computação em nuvem⁶, como *Software as a Service* (SaaS) e *Infrastructure as a Service* (IaaS).

Aplicações SaaS são caracterizadas por disponibilizarem softwares como serviço ao invés de como um produto. Isso quer dizer que ao invés de efetuar o download de um software e instalá-lo em seu computador, o usuário da aplicação pode apenas acessá-lo através de uma rede como a Internet. Com isso, toda a complexidade da aplicação e questões relacionadas à infraestrutura necessária para a sua execução não são visíveis ao usuário (FABIANO, 2021).

Exemplos de SaaS são serviços de transmissão de conteúdo multimídia, sob demanda, através de *streaming*. Nesse tipo de serviço o usuário não precisa instalar nem tampouco conhecer os detalhes sobre o software da empresa responsável pelo serviço. O usuário apenas precisa acessar o serviço, no caso, o software de transmissão de conteúdo multimídia para visualização do conteúdo através da Internet, pagando por isso uma mensalidade, ou seja, adquirindo um software como um serviço (FABIANO, 2021).

Aplicações IaaS são caracterizadas por disponibilizar a infraestrutura, necessária à implementação e execução de aplicações em nuvem, como um serviço. Com isso, os servidores e recursos como espaço de armazenamento e capacidade de processamento podem ser solicitados, sob demanda, pelos desenvolvedores da aplicação. É possível, por exemplo, que um sistema de vendas online solicite temporariamente, em períodos de maior atividade de vendas, como na *Black Friday*, a implantação de servidores adicionais em sua aplicação. Com

⁶ Disponível em: <https://www.javatpoint.com/types-of-databases> Acesso em: 12 jan 2023.

isso o fluxo adicional de informações e processamento pode ser utilizado, e tarifado, somente durante o período necessário. O sistema de vendas online não precisa se preocupar quanto à instalação de componentes físicos ou de software destes servidores como seus discos rígidos e instalação/configuração dos sistemas operacionais e demais softwares instalados nesses servidores etc. (FABIANO, 2021).

Por fim, aplicações que utilizam de SaaS e IaaS podem ser implementadas devido a existência de infraestruturas do tipo *Platform as a Service* (PaaS). Esses são ambientes de desenvolvimento que permitem a implantação e configuração de aplicações que funcionam como SaaS, uma vez que dispõem de uma estrutura de IaaS para que possam atender de modo adequado seus usuários, conforme a demanda da aplicação. Na Figura 35 é ilustrada a relação entre os conceitos de SaaS, PaaS e IaaS (MICROSOFT AZURE).

Figura 35: Relação entre os conceitos de SaaS, PaaS e IaaS.



Fonte: (MICROSOFT AZURE).

Alguns exemplos de plataformas para o desenvolvimento de aplicações e bancos de dados em nuvem⁷ são:

- Amazon Web Services (AWS);
- Microsoft Azure;
- Kamatera;
- PhonixNAP;
- ScienceSoft; e
- Google Cloud Platform.

⁷ Disponível em: <https://www.javatpoint.com/types-of-databases> Acesso em: 14 jan 2023.

Firestore

O Firestore é uma plataforma do Google que fornece recursos de computação em nuvem para o desenvolvimento de aplicativos para dispositivos móveis. Dentre os diversos produtos disponibilizados está o Cloud Firestore, que é um banco de dados baseado em documentos e em nuvem. Também existe o serviço Storage do Firestore, para o armazenamento de arquivos. Utilizando o Firestore, o usuário utiliza de modo simplificado, recursos da Google Cloud Platform, como o recurso de Bucket, que no ecossistema Firestore corresponde ao Storage (FIRESTORE).

O aplicativo Android desenvolvido nesse projeto utiliza o Firestore como principal serviço para o armazenamento e compartilhamento de dados GNSS brutos.

7. CÁLCULO DAS OBSERVÁVEIS DE PSEUDODISTÂNCIA E FASE DA ONDA PORTADORA EM SMARTPHONES ANDROID

Neste Capítulo, depois de apresentados os recursos necessários para obter os dados GNSS brutos e as opções de armazenamento/compartilhando desses dados, é discutido como obter as observáveis GNSS de pseudodistância e fase da onda portadora, como estas são comumente descritas na literatura e disponibilizadas em arquivos RINEX, para serem processadas por softwares como *BKG Ntrip Client* (BNC), RTKLib, PPP Wizard, dentre outros.

7.1. Especificidades dos Dados GNSS em Smartphones Android

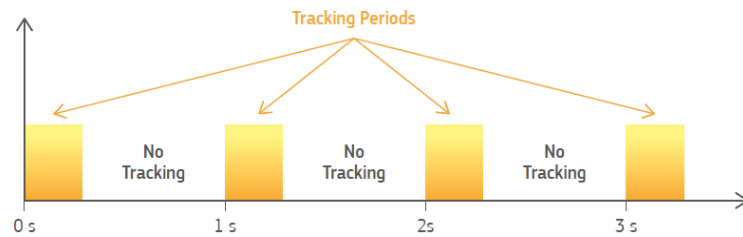
Nesta seção são discutidos procedimentos para o tratamento de dados GNSS provenientes de smartphones, que, de modo geral, não precisam ser executados em outros tipos de receptores GNSS. Dentre estes destacam-se o efeito de *duty cycle* e o método para cálculo de pseudodistâncias, apresentados nas próximas subseções.

7.1.1. Efeito de *duty cycle*

Como é apontado em (ROBUSTELLI; BAIOCCHI; PUGLIANO, 2019) existem questões importantes a serem tratadas ao processar os dados GNSS provenientes dos smartphones, como o comportamento de *duty cycle*. Este comportamento é observado em alguns aparelhos que disponibilizam a observável fase da onda portadora.

O comportamento de *duty cycle* é uma estratégia adotada pelo sistema operacional Android para não comprometer o consumo de bateria do aparelho. Nessa estratégia, o receptor GNSS do smartphone coleta dados por 200 ms e é desligado por 800 ms, automaticamente. Isso pode gerar perdas de ciclos e degradar a qualidade da observável fase da onda portadora, inviabilizando o seu uso (ROBUSTELLI; BAIOCCHI; PUGLIANO, 2019). Na Figura 36 é ilustrado o comportamento de *duty cycle*.

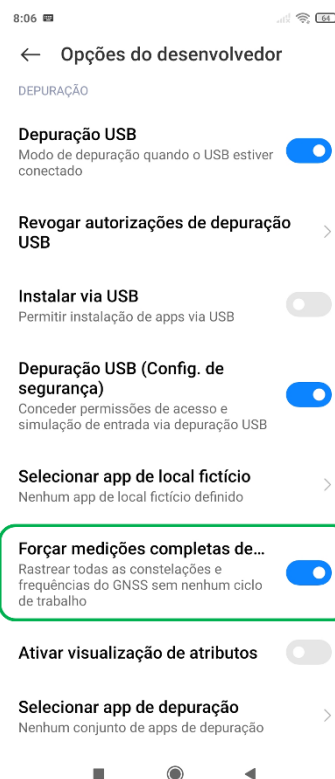
Figura 36: Efeito de *duty cycle* durante o tempo.



Fonte: (GSA, 2017).

A partir da versão 9.0 do sistema operacional Android, conhecido como Android Pie, o comportamento de *duty cycle* pode ser desativado via software como é apresentado na Figura 37. É destacada na imagem a opção para desligar o efeito de *duty cycle*. Entretanto, essa seção de configurações só pode ser acessada depois da ativação do modo desenvolvedor no smartphone e, ainda assim, cada aparelho deve ser analisado para verificar se suas observáveis de onda portadora sofrem de *duty cycle*, como apontado em (ROBUSTELLI; BAIOCCHI; PUGLIANO, 2019).

Figura 37: Opção para desativar o efeito de *duty cycle* via software no Android 9.0.



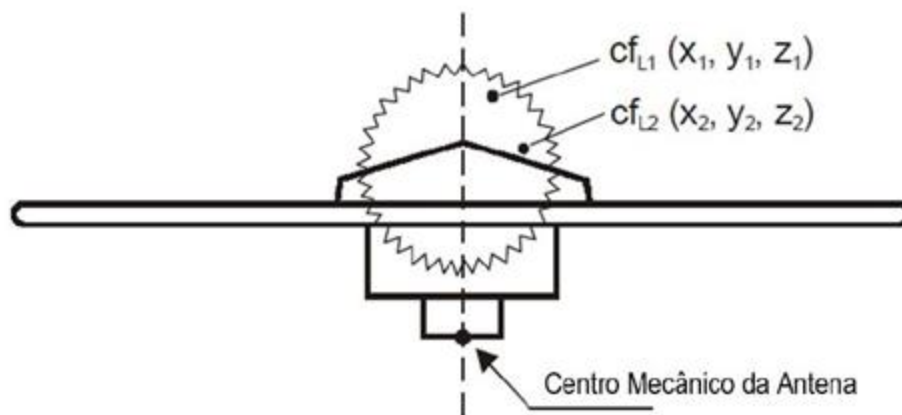
Fonte: (O AUTOR, 2023).

7.1.2. Correção em relação ao centro de fase da antena

O centro de fase eletrônico (ou simplesmente centro de fase) é o ponto observado em uma antena GNSS, onde todas as radiações parecem convergir. Trata-se do ponto onde os campos eletromagnéticos de todas as irradiações incidentes parecem se juntar na fase. Portanto, as soluções de posicionamento são calculadas neste ponto no espaço (LIMA, 2018).

Em aplicações de alta precisão, as posições do centro de fase de todas as antenas utilizadas no projeto devem ser conhecidas e referenciadas a um ponto identificado fisicamente na antena. Este deslocamento, também denominado *offset*, representa a distância entre o centro de fase da antena e o seu centro mecânico. Essas informações devem ser fornecidas pelo fabricante da antena. Os centros de fase entre diferentes ondas portadoras, como L1 e L2, podem apresentar diferentes *offsets*, como é apresentado na Figura 38 (LIMA, 2018).

Figura 38: Representação dos centros de fase em L1 e L2.



Fonte: (LIMA, 2018).

De modo ideal, o centro de fase da antena não deveria depender da direção na qual o sinal chega à antena. Entretanto, verifica-se na prática que pequenas variações ocorrem em função dos ângulos de elevação e azimute dos satélites. Dependendo das características da antena, os efeitos de variações no centro de fase podem atingir alguns milímetros ou até centímetros (LIMA, 2018).

Como apontado em (CHEM et al., 2019) e (ROBUSTELLI; BAIOCCHI; PUGLIANO, 2019) um dos principais problemas que envolviam o Posicionamento por Ponto Preciso em smartphones era justamente a incapacidade de corrigir os *offsets* do centro de fase

das antenas internas dos smartphones. Para contornar esse problema foram realizados estudos buscando alternativas como utilizar antenas externas, considerar o centro geométrico do smartphone como sendo o centro de fase, ou efetuar posicionamentos relativos em busca de melhores resultados em termos de precisão, devido a incapacidade de corrigir este erro.

Um marco importante nesse sentido ocorreu em julho de 2020. Neste período, foi lançada a versão 11 do sistema operacional Android, o que corresponde a versão 30 da API Android. Nessa versão da API, foram incluídas informações, até então inexistentes, sobre o centro de fase nos smartphones.

Foram acrescentadas no pacote *Location* da API 30 do Android as seguintes classes:

- *GnssCapabilities* e
- *GnssAntennaInfo*.

A classe *GnssCapabilities* é utilizada para verificar se é possível obter as informações sobre a antena GNSS interna do smartphone. Caso essas informações, estejam disponíveis, são acessadas através da classe *GnssAntennaInfo*.

A classe *GnssAntennaInfo* possui duas subclasses:

- *GnssAntennaInfo.PhaseCenterOffset* e
- *GnssAntennaInfo.SphericalCorrections*.

A classe *GnssAntennaInfo.PhaseCenterOffset* é utilizada para obter os valores de *offset* em relação ao centro de fase da antena GNSS interna do smartphone. A classe *GnssAntennaInfo.SphericalCorrections* possibilita obter correções, através de um mapeamento esférico, para as variações no centro de fase da antena. São computadas correções esféricas utilizando um vetor de números decimais bidimensional, onde as linhas e colunas representam ângulos calculados a partir do azimute e elevação dos satélites, respectivamente.

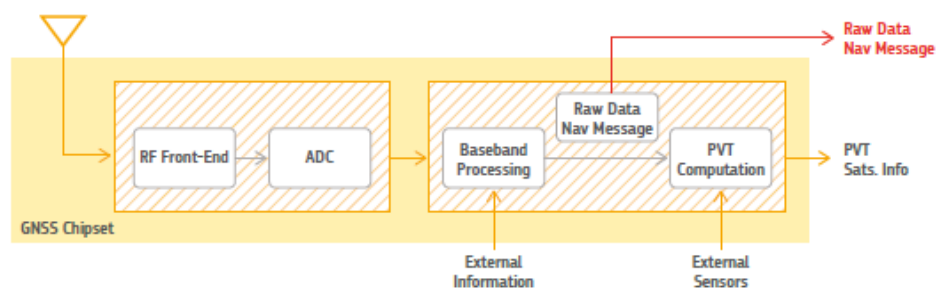
7.2. Cálculo de Pseudodistâncias

Em smartphones com o sistema operacional Android, a observável de pseudodistância não pode ser obtida diretamente. Essa observável deve ser calculada utilizando os parâmetros fornecidos pelo Android (GSA, 2017). Para a compreensão do processo de obtenção de pseudodistâncias são abordados, a seguir, a arquitetura de receptores GNSS e o processamento dos sinais GNSS nesses receptores.

7.2.1. Estrutura geral de um receptor GNSS

Um receptor GNSS processa sinais para prover aos usuários estimativas de Posição, Velocidade e Tempo (PVT). Essas estimativas são obtidas através de medidas de pseudodistâncias, informações na forma de mensagens de navegação e, opcionalmente, dados auxiliares de fontes externas, como outros sensores (GSA, 2017). Um diagrama de blocos genérico que representa a arquitetura de um receptor GNSS é apresentado na Figura 39.

Figura 39: Diagrama de blocos genérico para representar um receptor GNSS.



Fonte: (GSA, 2017).

O bloco RF (parte esquerda do diagrama) inclui a antena e um dispositivo *front-end*. Ambos são necessários para o processamento de sinais analógicos. Podem ser incluídos filtros no bloco RF, amplificadores de ruído ou filtros. O elemento final do bloco é um conversor do tipo *Analogue-to-Digital Converter* (ADC) (GSA, 2017).

Em smartphones, os blocos de processamento de banda base e processamento de PVT (lado direito do diagrama) são unidades de processamento de sinal baseadas em software. Essas unidades são baseadas em softwares porque foram construídas para operar sobre hardware de propósito geral, tendo em vista que, o sistema operacional Android é utilizado em uma ampla

variedade de smartphones, com diferentes recursos de hardware e diferentes *chipsets* GNSS (GSA, 2017).

Com isso, tarefas, comuns a outros tipos de receptores GNSS, devem ser executadas via software pelos desenvolvedores de aplicações Android. A unidade de processamento de banda base é responsável pela aquisição e rastreamento dos sinais GNSS, além de decodificar as mensagens de navegação. Dados auxiliares podem ser utilizados para diminuir o TTFF, como apresentado no Capítulo 5 (GSA, 2017).

7.2.2. Mensagens de navegação e estado do rastreamento de sinais GNSS

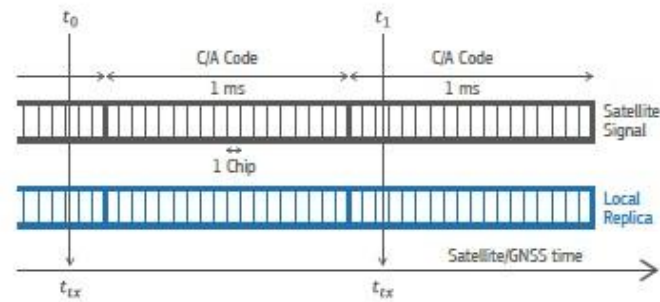
O receptor GNSS precisa sincronizar com o sinal transmitido pelo satélite, neste projeto o código processado é o código *Coarse/Acquisition* (C/A).

O processamento do código C/A é o mais utilizado em dispositivos de baixo custo, como smartphones. O código C/A possui 1 ms de comprimento. A réplica do código, gerada pelo receptor, precisa ser sincronizada com o código do sinal enviado pelo satélite. Para isso a cópia gerada no receptor precisa ser gerada considerando o tempo de propagação do sinal e o efeito Doppler que afeta o sinal emitido pelo satélite. O efeito Doppler ocorre, devido ao movimento do satélite, em relação ao receptor, da ordem de 3 km/s. Com isso, cada satélite possui uma frequência aparente (devido ao efeito Doppler), no receptor, diferente e pode ser identificado a partir de seu PRN (GSA, 2017) e (LIMA, 2018).

Nos próximos parágrafos será explicado como um receptor GNSS obtém o *tempo de transmissão* de um sinal, utilizando como exemplo o sistema GPS. Para isso, é importante destacar que o Tempo GPS é transmitido utilizando dois parâmetros: O número da semana GPS - *Week Number* (WN) e o Tempo da Semana - *Time of Week* (TOW). O número da semana conta quantas semanas ocorreram desde o início do tempo GPS, na meia noite (0h) na escala de tempo UTC (*Universal Time Coordinated*) entre 5 e 6 de janeiro de 1980. Uma semana tem início na meia noite no domingo. O TOW conta quantos segundos ocorreram dentro da semana atual (GSA, 2017).

Na Figura 40 é apresentada uma sincronização perfeita entre o código periódico, transmitido pelo satélite (repetido a cada 1 ms), e sua réplica local, gerada pelo receptor GNSS.

Figura 40: Sincronização com o código C/A.



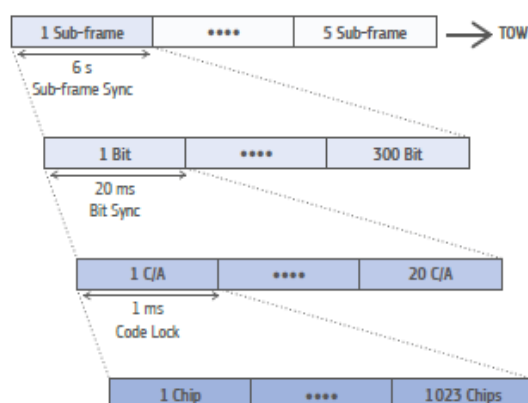
Fonte: (GSA, 2017).

Nesse momento, o receptor não é capaz de obter o tempo completo de transmissão do sinal. Em t_0 o receptor consegue obter apenas o atraso relativo ao início do período do código C/A. Do mesmo modo, em t_1 , o receptor consegue obter apenas o atraso em relação ao início do código C/A atual. O número de códigos C/A inteiros entre t_0 e t_1 é desconhecido.

Se o receptor está sincronizado apenas com o código C/A, o intervalo de validade do tempo de transmissão está entre 0 e 1 ms, fazendo com que o cálculo do intervalo de propagação seja ambíguo (GSA, 2017).

Para resolver essa ambiguidade de tempo, o receptor GNSS deve explorar a estrutura da mensagem de navegação. No caso do sistema GPS, 5 subquadros consecutivos constituem um quadro. Uma mensagem de navegação é formada por 25 quadros. Na Figura 41 é apresentada a estrutura de mensagens de navegação GPS na frequência L1 e código C/A (GSA, 2017).

Figura 41: Estrutura da mensagem de navegação GPS L1 C/A.



Fonte: (GSA, 2017).

Estruturas similares ao do sistema GPS são utilizadas nos sistemas Galileo e BeiDou. O sistema GLONASS se diferencia dos demais sistemas, considerando os intervalos de validade dos tempos de transmissão em dias e não em semanas.

Tomando como exemplo o sistema GPS, existem os seguintes estados de sincronização para obter o tempo do satélite (GSA, 2017):

- **Code Lock:** O receptor está travado no código C/A. O intervalo de validade é 0-1 ms.
- **Bit Sync:** O receptor está sincronizado com ao nível de bits. O intervalo de validade é de 0-20 ms.
- **Subframe Sync:** O receptor está sincronizado com os subquadros. O intervalo de validade é 0-6 s.
- **Tow Decoded:** Todos os subquadros contém o TOW. Portanto, uma vez que o TOW é decodificado, o intervalo de validade é de 0-1 semana.

Existem na literatura métodos mais avançados capazes de calcular soluções de PVT usando apenas pseudodistâncias fracionárias. Essas soluções de navegação são conhecidas como problema de navegação de tempo grosseiro (*Coarse-time Navigation Problem*). Além disso, receptores A-GNSS podem também obter o TOW através de sistemas externos e não precisam decodificar as mensagens de navegação (GSA, 2017).

Na Tabela 8 são apresentadas as ambiguidades de tempo de transmissão com base no estado de rastreamento das quatro principais constelações GNSS (GSA, 2017).

Tabela 8: Ambiguidade de tempo com base no estado de rastreamento dos sinais GNSS.

GPS		GALILEO		GLONASS		BeiDou	
Estado	Tempo	Estado	Tempo	Estado	Tempo	Estado	Tempo
C/A code	1 ms	E1BC code	4 ms	C/A code	1 ms	C/A code	1 ms
Bit	20 ms	E1C 2nd code	100 ms	Bit	20 ms	Bit	20 ms
Subframe sync	6 s	E1B page	2 s	String	2 s	Subframe sync	6 s
TOW	1 semana	TOW	1 semana	Time of Day	1 dia	TOW	1 semana

Fonte: (GSA, 2017).

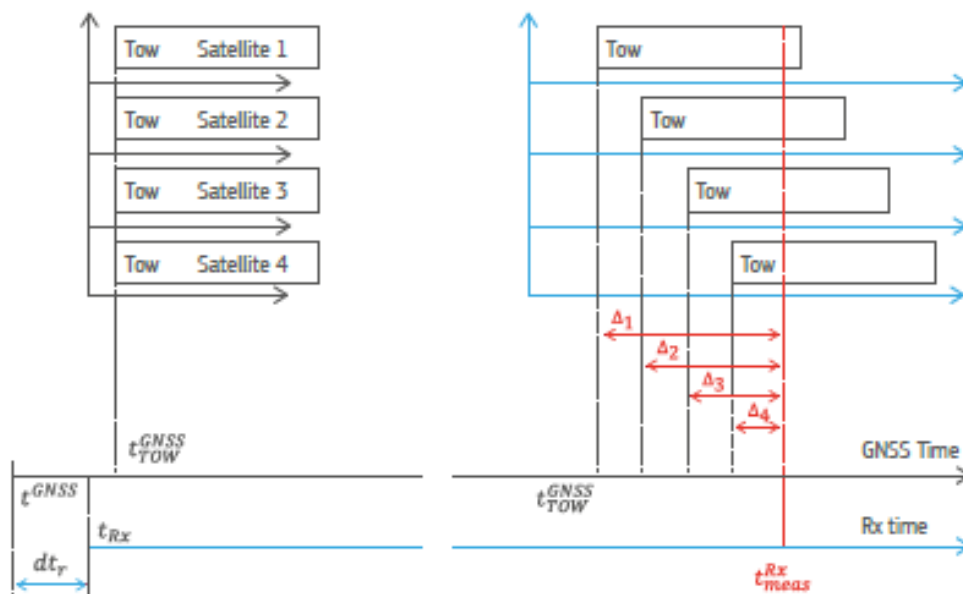
7.2.3. Obtenção do tempo recepção com a técnica de tempo de recepção comum

Como abordado no Capítulo 3, o cálculo da pseudodistância é obtido a partir do intervalo de propagação do sinal GNSS. O intervalo de propagação do sinal, por sua vez, é obtido a partir de medições de tempo de transmissão e recepção. Na subseção anterior, foi definido como obter, utilizando smartphones Android, o tempo de transmissão dos sinais enviados pelos satélites. Nesta seção será definido como obter o tempo de recepção dos sinais GNSS na antena do smartphone.

A geração das pseudodistâncias pode ser implementada através de dois métodos diferentes: tempo de recepção comum ou tempo de transmissão comum. Nos próximos parágrafos a geração das pseudodistâncias é definida utilizando o método de tempo de recepção comum.

O tempo de recepção comum é também conhecido como tempo de medição (t_{meas}^{Rx}). Por exemplo, quatro satélites emitem sinais ao mesmo tempo, com o mesmo TOW sendo transmitido por todos os satélites no mesmo instante de tempo (Parte esquerda da Figura 42).

Figura 42: Cálculo de pseudodistâncias através do método de tempo de recepção comum.



Fonte: (GSA, 2017).

Como os sinais possuem diferentes caminhos de propagação, os quatro sinais chegam ao receptor com diferentes atrasos. O receptor então calcula o deslocamento de tempo entre o

TOW e o instante de tempo atual. Esse deslocamento de tempo é o tempo de medição (t_{meas}^{RX}). Com isso, o tempo de transmissão dos satélites (disponibilizados no sistema de tempo GNSS da respectiva constelação) no tempo de medição, pode ser expresso como (GSA, 2017):

$$\begin{aligned} t_{Tx,Sat1} &= t_{TOW}^{GNSS} + \Delta_1, \\ t_{Tx,Sat2} &= t_{TOW}^{GNSS} + \Delta_2, \\ t_{Tx,Sat3} &= t_{TOW}^{GNSS} + \Delta_3, \\ t_{Tx,Sat4} &= t_{TOW}^{GNSS} + \Delta_4 \end{aligned} \quad (13)$$

onde:

- t_{TOW}^{GNSS} é o TOW transmitido e
- Δ_i é o atraso entre o TOW e o tempo de medição para o i -ésimo satélite.

É importante destacar que este procedimento só é realizado quando o receptor está sincronizado com o sinal GNSS ao nível de *TOW Decoded*, como é apresentado na subseção anterior.

Antes de computar a primeira solução PVT e decodificar o TOW de pelo menos um satélite, o receptor não tem informação sobre os sistemas de tempo GNSS. Portanto, o receptor precisa assumir premissas para ser capaz de gerar o tempo Rx e calcular o primeiro conjunto de pseudodistâncias. O primeiro sinal de satélite recebido é utilizado como referência. O tempo de recepção (t_{meas}^{RX}) é o tempo transmitido mais um tempo de propagação de referência (t_{ref}^{path}). Um valor padrão entre 65 e 85 ms é geralmente assumido (GSA, 2017). No aplicativo Gns Logger, é encontrado em seu código fonte, o valor de 75 ms para o cálculo das pseudodistâncias GPS.

Portanto, o primeiro tempo de recepção (de um único satélite) pode ser computado como:

$$t_{meas}^{RX}[1] = t_{Tx,Sat1}[1] + t_{ref}^{path} \quad (3)$$

Todas as outras pseudodistâncias são geradas em relação à primeira. É importante ressaltar que um erro comum a todos os satélites é introduzido porque t_{ref}^{path} foi fixado, não estimado. Portanto, uma tendência (*bias*) entre o tempo GNSS e o tempo de recepção é

introduzido (dt_r). O tempo de recepção, no sistema de tempo GNSS de referência, pode ser expresso como (GSA, 2017):

$$t_{meas}^{RX,GNSS}[k] = t_{meas}^{RX}[k] - dt_r [1] \quad (4)$$

onde:

- k é o k -ésimo tempo de recebimento e
- dt_r é o erro do relógio do receptor em relação ao tempo GNSS de referência.

Uma vez que o primeiro conjunto de pseudodistâncias foi computado e a primeira solução PVT foi obtida, $dt_r [1]$ é determinado e utilizado por todos os conjuntos subsequentes de pseudodistâncias. Entretanto, o receptor computa e atualiza o valor do erro do relógio em todos os conjuntos de solução PVT. Portanto, os conjuntos de pseudodistâncias subsequentes podem ser computados como (GSA, 2017):

$$\begin{aligned} \rho_1 [k] &= (t_{meas}^{RX}[k] - t_{Tx,Sat1}[k] + dt_r [1]) * c, \\ \rho_2 [k] &= (t_{meas}^{RX}[k] - t_{Tx,Sat2}[k] + dt_r [1]) * c, \\ \rho_3 [k] &= (t_{meas}^{RX}[k] - t_{Tx,Sat3}[k] + dt_r [1]) * c, \\ \rho_4 [k] &= (t_{meas}^{RX}[k] - t_{Tx,Sat4}[k] + dt_r [1]) * c \end{aligned} \quad (5)$$

onde:

- c é a velocidade da luz no vácuo.

Alguns receptores podem prover o valor de $t_{meas}^{RX}[k]$ como a soma do tempo de recepção e um valor de deslocamento (GSA, 2017).

7.2.4. Geração do tempo GPS

O sistema operacional Android não provê o tempo GNSS diretamente. É fornecido a marcação de tempo do hardware interno do *chipset* GNSS e um valor de tendência ao real tempo GPS (ambos em nanossegundos). O tempo GPS pode ser computado, a partir do relógio interno do receptor GNSS do smartphone por (GSA, 2017):

$$GpsTime = TimeNanos - (FullBiasNanos + BiasNanos)[ns] \quad (6)$$

onde:

- *FullBiasNanos* é o valor de tendência entre o relógio do receptor e o tempo GPS; e
- *BiasNanos* é o valor de tendência em sub-nanosegundos.

Caso o receptor tenha estimado o tempo GPS utilizando uma constelação não GPS, o tempo GPS estimado pode ser calculado como (GSA, 2017):

$$GpsTime = TimeNanos - (FullBiasNanos + BiasNanos) - InterSystemBias \quad (7)$$

onde:

- *InterSystemBias* (ISB) é o deslocamento entre o tempo GPS e o tempo GNSS utilizado na estimação.

Por exemplo, se o tempo foi estimado usando o Sistema de Tempo Galileo, o valor de *InterSystemBias* correspondente ao GGTO. O GGTO (*GPS to Galileo Time Offset*) é utilizado para converter medidas de tempo no sistema de tempo do Galileo para o sistema de tempo do GPS. Além disso, graças à interoperabilidade entre os sistemas GPS e Galileo, o GGTO é transmitido nas mensagens de navegação (GSA, 2017).

Para o ajustamento envolvendo observações dos sistemas GPS, Galileo, GLONASS e BeiDou, o ISB também pode ser estimado no ajustamento, como uma incógnita adicional. Adotando o sistema GPS como referência, os ISBs para converter os tempos GLONASS, Galileo e BeiDou para o sistema de tempo GPS devem ser determinados, por exemplo, utilizando o MMQ. Com isso, o seguinte vetor de incógnitas será estimado (GSA, 2017):

$$\begin{pmatrix} dx \\ dy \\ dz \\ c * dt_r \\ c * dt_{GPS/GLONASS} \\ c * dt_{GPS/Galileo} \\ c * dt_{GPS/BeiDou} \end{pmatrix}$$

7.2.5. Geração das pseudodistâncias

Como foi discutido neste capítulo, o sistema operacional Android não provê pseudodistâncias. Ao invés disso, ele provê todos os parâmetros necessários ao seu cálculo. Como foi apresentado nas subseções anteriores, os tempos de transmissão e recepção dos sinais GNSS também podem ser obtidos, em seus respectivos sistemas de tempo. Por fim, a partir dos tempos de transmissão e recepção é possível calcular a pseudodistância como (GSA, 2017):

$$\rho = \frac{(t_{RX} - t_{TX})}{1E9} * c \quad (8)$$

onde:

- t_{TX} é o tempo de transmissão do sinal, no tempo GNSS de referência e
- t_{RX} é o tempo de recepção do sinal, no tempo GNSS de referência.

O valor de t_{TX} é definido como:

$$t_{TX} = ReceivedSvTimeNanos[ns]$$

O intervalo de validade de t_{TX} devido à ambiguidade de tempo, discutida na subseção anterior, deve ser considerado segundo o estado de rastreamento do sinal GNSS.

Para obter pseudodistâncias completas (sem ambiguidade de tempo), são requeridos os seguintes estados de rastreamento (GSA, 2017):

- GPS: *TOW Decoded*;
- BeiDou: *Tow Decoded*;
- GLONASS: *Time of Day*; e
- Galileo:
 - *TOW Decoded*
 - *EIC 2nd Code* (intervalo de validade de 0-100 ms)

O tempo de recepção, no sistema de tempo GNSS completo (desde o início da contagem do sistema), pode ser reconstruído como (GSA, 2017):

$$t_{RX_{GNSS}} = TimeNanos + TimeOffsetNanos - (FullBiasNanos(1) + BiasNanos(1))[ns] \quad (9)$$

$t_{RX_{GNSS}}$ é provido apenas no sistema GNSS de referência que o receptor utiliza para fornecer os tempos de recepção, geralmente, o sistema de tempo GPS. Entretanto, t_{TX} é provido para cada sistema GNSS, isto é, no Tempo GPS para medições GPS, no Tempo GLONASS para medições GLONASS etc. Portanto, t_{TX} precisa ser convertido para o mesmo sistema de tempo de referência que $t_{RX_{GNSS}}$ e vice-versa (GSA, 2017).

$t_{RX_{GNSS}}$ está em tempo GNSS completo (marcação de tempo desde o início do tempo GNSS correspondente), enquanto que t_{TX} depende do estado de rastreamento do sinal, sendo uma contagem de tempo relativa à semana, ou dia, no caso do GLONASS. Quando a pseudodistância é calculada, ambos devem estar no mesmo intervalo (GSA, 2017).

A seguir será apresentado como calcular pseudodistâncias GPS com base nos valores de $t_{RX_{GNSS}}$ e t_{TX} , garantindo que (GSA, 2017):

1. Ambas as medições estão no mesmo sistema de tempo GNSS e
2. Ambas as medições possuem o mesmo intervalo de validade.

Com estado de rastreamento de *TOW Decoded* o tempo de recepção, no tempo GPS é (GSA, 2017):

$$t_{RX} = t_{RX_{GNSS}} - weekNumberNanos[ns] \quad (10)$$

WeekNumberNanos é o número de nanosegundos que ocorreram desde o início do tempo GPS até a semana GPS (WN) atual. Com isso, obtém-se o tempo de recepção do sinal, também no formato TOW, assim como é obtido para o tempo de transmissão t_{TX} . *WeekNumberNanos* pode ser calculado como (GSA, 2017):

$$weekNumberNanos = floor\left(\frac{-FullBiasNanos}{NumberNanoSecondsWeek}\right) * NumberNanoSecondsWeek [ns] \quad (11)$$

onde:

- *NumberNanoSeconds* é o número de nanosegundos dentro de uma semana, ou seja, $NumberNanoSeconds = 604800e9$.

Processo similar pode ser aplicado para as demais constelações GNSS, considerando seus respectivos intervalos de validade para cada estado de rastreamento dos sinais GNSS.

Possuindo os valores de T_{RX} e T_{TX} a construção das pseudodistâncias pode ser executada conforme a Equação (8).

Exemplo de cálculo de pseudodistância

O trecho de código em Matlab, apresentado a seguir demonstra como calcular as pseudodistâncias para medições Galileo, GPS e BeiDou quando o estado de rastreamento do sinal GNSS é *TOW Decoded* (GSA, 2017).

Figura 43: Exemplo em código Matlab para o cálculo de pseudodistâncias a partir de dados GNSS brutos capturados por smartphone Android.

```

% Select GPS + GAL TOW decoded (state bit 3 enabled)
pos = find( (gnss.Const == 1 | gnss.Const == 6) & bitand(gnss.State,2^3);
% Generate the measured time in full GNSS time
tRx_GNSS = gnss.timeNano(pos) - (gnss.FullBiasNano(1) + gnss.BiasNano(1));
% Change the valid range from full GNSS to TOW
tRx = mod(tRx_GNSS(pos),WEEKSEC*1e9);
% Generate the satellite time
tTx = gnss.ReceivedSvTime(pos) + gnss.TimeOffsetNano(pos);
% Generate the pseudorange
prMilliseconds = (tRx - tTx);
pr = prMilliseconds *Constant.C*1e-9;

```

Fonte: (GSA, 2017).

Na primeira linha, é verificado se o satélite pertence a constelação GPS ou Galileo, através do atributo `gnss.Const`. Através da função *bitand* é feita a verificação do estado de rastreamento do sinal, verificando se este corresponde a *TOW Decoded*, a fim de evitar a ambiguidade de tempo, como é discutido na subseção 7.2.2.

Na segunda linha, é obtido o tempo do relógio do receptor através do atributo `gnss.timeNano`. É importante ressaltar que o valor do relógio é obtido para cada satélite processado, uma vez que a variável *pos* é utilizada. Entretanto, os valores de `FullBiasNano` e

BiasNanos, que representam o erro do relógio do receptor em relação ao sistema de tempo GPS, são sempre os mesmos, uma vez que esses valores são acessados pela primeira vez, o que é indicado no código através do índice 1, ao acessar esses dois atributos. Isso acontece porque, está sendo aplicada a técnica de recebimento de tempo comum, e por isso o sinal do primeiro satélite rastreado é utilizado como referência.

Na terceira linha o tempo de recepção é convertido para segundos.

Na quarta linha, é obtido o tempo de transmissão do sinal, cujo valor corresponde à aplicação da Equação 3, da Seção 6.2.3. Entretanto, neste exemplo soma-se ao tempo do satélite um valor de deslocamento, através da propriedade `gnss.TimeOffsetNano`, ao invés de utilizar um valor padrão de deslocamento para o caminho percorrido, como é adotado na Equação 3.

Por fim, nas duas últimas linhas de código, possuindo os tempos de transmissão e recepção do sinal, o intervalo de propagação do sinal é obtido e a pseudodistância é calculada

7.3. Cálculo de fase da onda portadora

Ao contrário do que acontece para as pseudodistâncias, a observável de fase da onda portadora pode ser calculada para cada satélite, independentemente. Para isso, é preciso utilizar os seguintes métodos da classe *GnssMeasurement*, como é apresentado na Tabela 9 (GSA, 2017):

Tabela 9: Métodos da classe *GnssMeasurement* para calcular a fase da onda portadora.

Classe do Android	Campo da Classe	Descrição
GnssMeasurement	State	Estado atual de sincronização para o satélite em determinada frequência.
GnssMeasurement	AccumulatedDeltaRangeMeters	Distância acumulada desde o último <i>reset</i> do canal, em metros, entre o satélite e o receptor. Valores positivos indicam que o satélite está se afastando do receptor.
GnssMeasurement	CarrierFrequencyHz	Frequência da onda portadora no qual os códigos e as mensagens foram modulados.

Fonte: (GSA, 2017).

O método *AccumulatedDeltaRangeMeters* correspondente a onda portadora em metros, desde que o valor de *State* seja igual a *TOW Decoded*. Com isso, basta fazer uma operação de conversão com os campos, para obter a onda portadora em ciclos, como esta costuma ser utilizadas nos arquivos RINEX e softwares de processamento de dados GNSS.

$$\begin{aligned} \text{wave_length_meters} &= \frac{c}{\text{CarrierFrequencyHz}} \\ \text{carrier_phase_cycles} &= \text{AccumulatedDeltaRangeMeters} / \text{wave_length} \end{aligned} \quad (12)$$

onde:

- c é a velocidade da luz no vácuo, em metros por segundo.

8. DESENVOLVIMENTO DE UM ECOSSISTEMA DE PROCESSAMENTO E ANÁLISE DE DADOS GNSS A PARTIR DE SMARTPHONES ANDROID – INCT-GNSS-APP

Para poder analisar a qualidade das medidas e dos processamentos dos dados GNSS brutos provenientes de smartphones com o sistema operacional Android, foram desenvolvidos softwares para processamento em tempo real e para pós-processamento. Assim sendo, foi implementado um ecossistema de softwares denominado INCT-GNSS-APP.

O objetivo ao desenvolver esses softwares foi ter a capacidade de acompanhar todo o fluxo dos dados GNSS capturados pelo smartphone. Através destes é possível processar os dados GNSS brutos logo após a sua coleta, além de transmiti-los e armazená-los para reprocessamentos, que por sua vez também são avaliados estatisticamente.

Na Seção 8.1 é apresentado o aplicativo Android desenvolvido, para a coleta e processamento de dados em tempo real. Além disso, é apresentado também o aplicativo responsável por enviar os dados coletados para um banco de dados de armazenamento em nuvem.

Na Seção 8.2 são apresentados os softwares desenvolvidos para sistemas operacionais do tipo desktop (Windows, MacOS e Linux) que efetuam o pós-processamento e análise da qualidade dos dados provenientes dos aplicativos para Android.

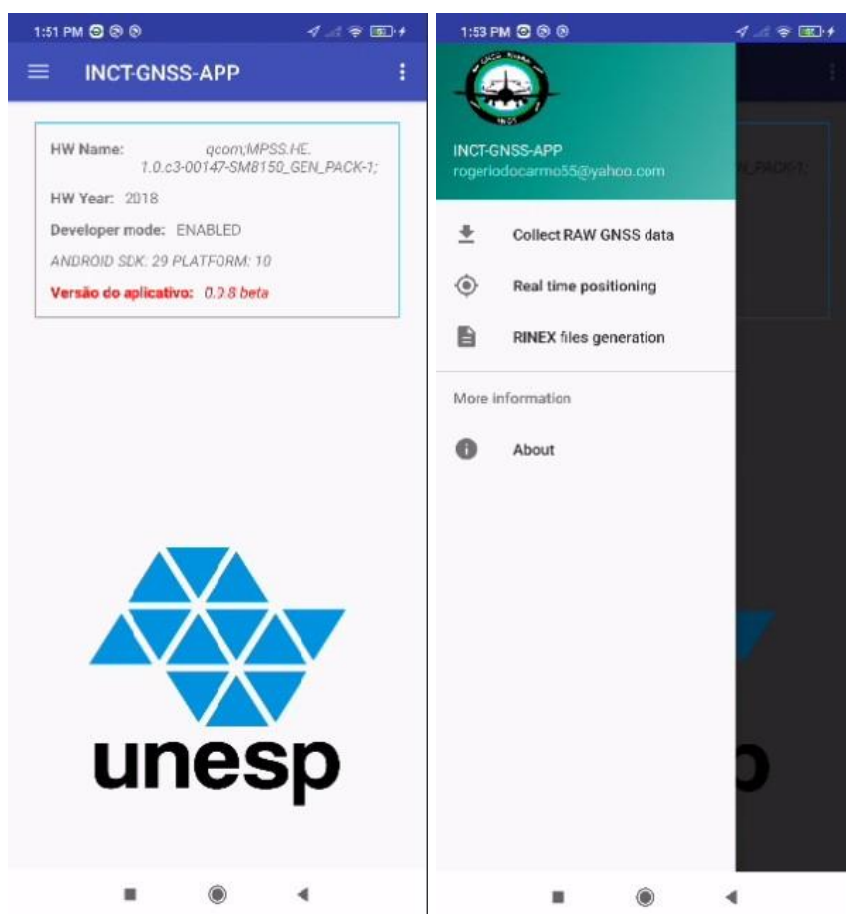
8.1. Aplicativos para o processamento de dados GNSS brutos – INCT-GNSS-APP

Nesta seção são abordados os aplicativos Android desenvolvidos nessa pesquisa. Foram desenvolvidos dois aplicativos, como é apresentado nas próximas subseções.

8.1.1. Aplicativo para coleta de dados e processamento em tempo real

O aplicativo para coleta e processamento de dados em tempo real foi implementado utilizando a linguagem de programação Java, sendo nomeado como INCT-GNSS-APP. Nesse aplicativo são coletados dados no formato esperado pelo software Gns Analysis App, um arquivo RINEX na versão 3.04 além de outros arquivos discutidos nessa subseção.

A tela principal do INCT-GNSS-APP é apresentada na Figura 44.

Figura 44: Tela Principal do INCT-GNSS-APP.

Fonte: (O AUTOR, 2020).

Coleta de dados brutos

A aplicação cria arquivos de texto com os dados GNSS brutos capturados pelo smartphone. Este arquivo tem a mesma estrutura definida pelo *Gnss Logger*. Com isso, os arquivos de dados brutos gerados pelo INCT-GNSS-APP podem ser analisados no *Gnss Analysis App*. Na Figura 45 é apresentada a interface do aplicativo ao coletar os dados brutos e na Figura 46 é apresentado um trecho de um arquivo de dados brutos criado pelo aplicativo.

Figura 45: Tela do INCT-GNSS-APP coletando dados brutos.



Fonte: (O AUTOR, 2020).

Figura 46: Trecho de arquivo de dados brutos gerado pelo INCT-GNSS-APP.

```

cppfunctions.conf | gnss_log_2020_07_01_16_56_27.txt
1 #
2 # Header Description:
3 #
4 # Version: v2.0.0.1 Platform: 9 Manufacturer: Xiaomi Model: Redmi Note 7
5 #
6 #
7 Raw, ElapsedRealtimeMillis, TimeNanos, LeapSecond, TimeUncertaintyNanos, FullBiasNanos, BiasNanos, BiasUncertaintyNanos, DriftInuityCount, Svid, TimeOffsetNanos, State, ReceivedSvTimeNanos, ReceivedSvTimeUncertaintyNanos, Cn0DbHz, PseudorangeRateMeterltaRangeState, AccumulatedDeltaRangeMeters, AccumulatedDeltaRangeUncertaintyMeters, CarrierFrequencyHz, CarrierCycles, CarrllationType, AgcDb, CarrierFrequencyHz
8 #
9 # Fix, Provider, Latitude, Longitude, Altitude, Speed, Accuracy, (UTC) TimeInMs
10 # Nav, Svid, Type, Status, MessageId, Sub-messageId, Data (Bytes)
11 #
12 Raw, 290694019, 315754637000000, , -1277352853108416526, -0.5858173370361328, 111995.19038200378, 36.132813617513015, 240.02294, 1.314500093460083, 0, 0, 0, 0, 0, 1.57542003E9, , , , 0, 1, 0.33, 1.57542003E9
13 Raw, 290694024, 315754637000000, , -1277352853108416526, -0.5858173370361328, 111995.19038200378, 36.132813617513015, 240.022, 0.9769999980926514, 0, 0, 0, 0, 1.57542003E9, , , , 0, 1, 0.33, 1.57542003E9
14 Raw, 290694824, 315755337000000, , -1277352853108406759, -0.2322826385498047, 743.2185229845345, 0.7885030009914966, 43.62578409000039100647, 0, 0, 0, 0, 1.60256256E9, , , , 0, 3, 1.74, 1.60256256E9
15 Raw, 290694829, 315755337000000, , -1277352853108406759, -0.2322826385498047, 743.2185229845345, 0.7885030009914966, 43.62578, 1.244499921798706, 0, 0, 0, 0, 1.57542003E9, , , , 0, 1, 2.88, 1.57542003E9

```

Fonte: (O AUTOR, 2020).

Geração de arquivo RINEX de observação

O INCT-GNSS-APP gera em tempo real, um arquivo RINEX de observação na versão 3.04. São inseridas neste arquivo as observáveis de pseudodistância e fase da onda portadora na frequência L1. Na Figura 47 é apresentada a interface do aplicativo ao gerar um arquivo RINEX de observação. Além disso, na Figura 48 é apresentado um trecho do arquivo RINEX gerado.

Figura 47: Tela do INCT-GNSS-APP durante criação de arquivo RINEX na versão 2.11.



Fonte: (O AUTOR, 2020).

Figura 48: Trecho de arquivo RINEX gerado pelo INCT-GNSS-APP.

```

HUAWEI_VOG-L29_2021_12_08_08_54_51_DESKTOP_RINEX.21o [3]
  1      3 04      OBSERVATION DATA      M. Mixed      RINEX VERSION / TYPE
  2 PPS_FCT_INCT      INCT NavAer - UNESP      2021128      115514 UTC      PRN / RUN BY / DATE
  3 *****COMMENT*****
  4 Created by PPS-FCT-INCT      COMMENT
  5 Developed by INCT-GNSS-NavAer - Brazil      COMMENT
  6 Contact: https://inct-gnss-navaer.fct.unesp.br/en/index.php      COMMENT
  7 *****COMMENT*****
  8 INCT NavAer - UNESP      MARKER NAME
  9 HUMAN      MARKER TYPE
 10 INCT NavAer - UNESP INCT NavAer - UNESP      OBSERVER / AGENCY
 11 unknown      null      null      REC # / TYPE / VERS
 12 unknown      null      ANT # / TYPE
 13 0000000.0000      0000000.0000      0000000.0000      APPROX POSITION XYZ
 14 0.0000      0.0000      0.0000      ANTENNA: DELTA H/E/N
 15 G 8 C1C L1C D1C S1C C5Q L5Q D5Q S5Q      SYS / # / OBS TYPES
 16 R 4 C1C L1C D1C S1C      SYS / # / OBS TYPES
 17 E 12 C1B L1B D1B S1B C1C L1C D1C S1C C5Q L5Q D5Q S5Q      SYS / # / OBS TYPES
 18 C 4 C2I L2I D2I S2I      SYS / # / OBS TYPES
 19 2021 12 8 11 55 14.0      GPS      TIME OF FIRST OBS
 20 24 R01 1 R02 -4 R03 5 R04 6 R05 1 R06 -4 R07 5 R08 6      GLONASS SLOT / FRQ #
 21 R09 -2 R10 -5 R11 0 R12 -1 R13 -2 R14 -7 R15 0 R16 -1      GLONASS SLOT / FRQ #
 22 R17 4 R18 -3 R19 3 R20 2 R21 4 R22 -3 R23 3 R24 2      GLONASS SLOT / FRQ #
 23 G L1C      SYS / PHASE SHIFT
 24 G L5Q -0.25000      SYS / PHASE SHIFT
 25 R L1C      SYS / PHASE SHIFT
 26 E L1B      SYS / PHASE SHIFT
 27 E L1C -0.50000      SYS / PHASE SHIFT
 28 E L5Q -0.25000      SYS / PHASE SHIFT
 29 C L2I      SYS / PHASE SHIFT
 30 C1C 0.000 C1P 0.000 C2C 0.000 C2P 0.000      GLONASS COD/PHS/BIS
 31 *****END OF HEADER*****
 32 > 2021 12 8 11 55 15.0      0 8
 33 G02 20641122.648      110299901.148      -1043.325      37.000
 34 G05 25035364.178      133781395.572      -1691.414      37.000
 35 G06 22654636.721      121059509.868      2117.597      38.000
 36 G12 22022401.409      117681036.048      -1205.189      27.000
 37 G20 23946277.897      107961648.37      -1949.246      36.000
 38 G24 20074861.863      107239793.06      -150.964      36.000
 39 G29 22959516.358      122688694.206      -1986.634      36.000
 40 G25 23801508.658      127188045.774      -2054.563      25.000
 41
 42 > 2021 12 8 11 55 16.0      0 8

```

Fonte: (O AUTOR, 2023).

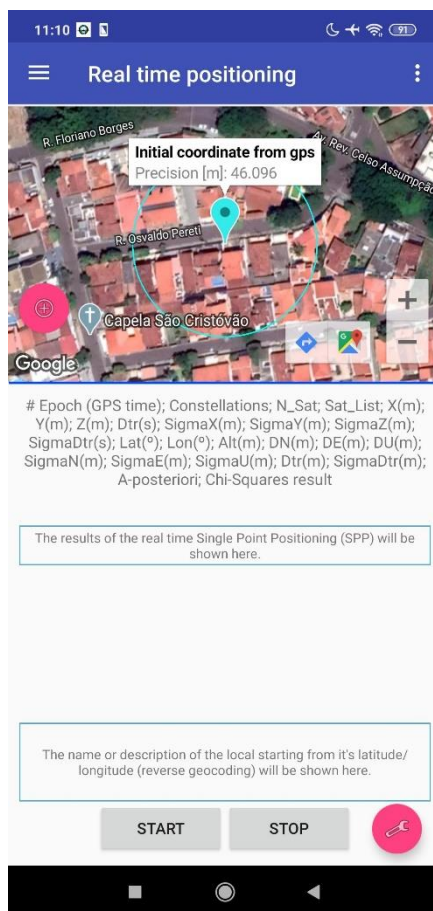
Posicionamento por Ponto Simples em tempo real

Na aplicação desenvolvida é empregado o método de Posicionamento por Ponto Simples em tempo real. Para isso é utilizado o método de recepção comum para o cálculo das pseudodistâncias e o MMQ para o ajustamento. São utilizadas como observações as pseudodistâncias na frequência L1 da constelação GPS.

A tela de posicionamento em tempo real é apresentada na Figura 49. É apresentada na tela, através de um mapa gerado pelo Google Maps, o marcador de posição inicial (em azul). Este marcador é resultado da solicitação de uma coordenada de referência inicial para ser utilizada pela aplicação: quando a aplicação é iniciada, é solicitada a última localização conhecida do dispositivo, na forma de latitude, longitude e altitude; que pode ser fornecida pelo *chipset* GNSS ou ainda através da rede de telefonia móvel ou WiFi, sendo fornecida pelo Android qual delas possuir a maior acurácia no momento.

Essa solicitação é feita utilizando métodos da API 23 do Android, para que se obtenha uma localização direta, sem considerar os dados GNSS brutos. A coordenada inicial, caso encontrada, é marcada no centro do mapa com um círculo azul, de raio igual à precisão horizontal associada àquela coordenada (Figura 49).

Figura 49: Tela do INCT-GNSS-APP para a execução do PPS em tempo real.

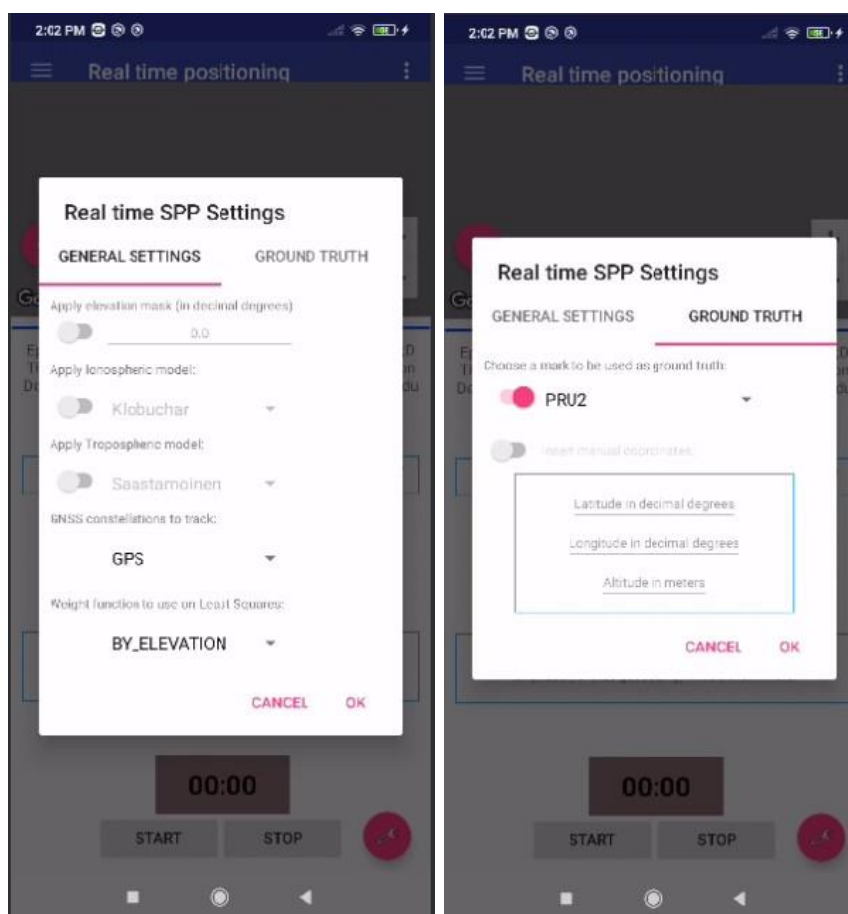


Fonte: (O AUTOR, 2020).

A coordenada inicial obtida é então convertida pelo app para coordenadas cartesianas. As coordenadas cartesianas são utilizadas para preencher o cabeçalho dos arquivos RINEX. Estas também são utilizadas como valores iniciais no ajustamento pelo MMQ e cálculo de azimute e elevação dos satélites observados. Por fim, a coordenada inicial é enviada via Internet ao servidor SUPL do Google, para a obtenção das efemérides transmitidas.

Para o processamento em tempo real é possível alterar configurações como o ângulo mínimo de elevação dos satélites. Também é possível escolher modelos de correção ionosférica e troposférica, além de definir uma coordenada de referência para o processamento, como é apresentado na segunda aba da tela de configurações. A tela de configurações do INCT-GNSS-APP é apresentada na Figura 50.

Figura 50: Telas de configurações do PPS em tempo real.



Fonte: (O AUTOR, 2020).

Após o *download* das efemérides transmitidas, via Internet, do servidor SUPL do Google, é iniciado o ajustamento via MMQ. O resultado do processamento de cada época de dados é apresentado na tela em formato textual e através de um marcador vermelho no mapa. Além disso, também é apresentado em tela o endereço associado à cada coordenada estimada, utilizando um processo denominado Geocoding, através de um serviço online do Google.

Figura 51: Tela do INCT-GNSS-APP executando o PPS em tempo real.



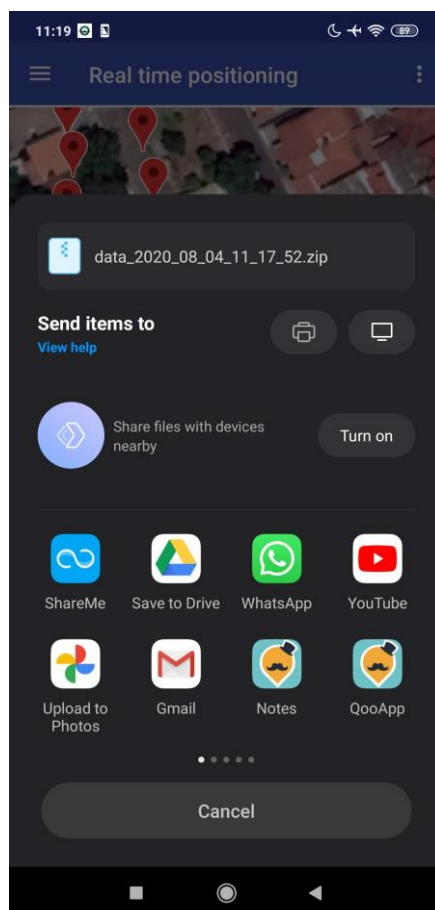
Fonte: (O AUTOR, 2020).

Quando o processamento do PPS em tempo real é finalizado é então criado um arquivo compactado, no formato ZIP, contendo os arquivos gerados durante o processamento, sendo estes:

- Arquivos de texto dos dados GNSS brutos no formato do Gns Analysis App;
- Arquivo RINEX de observação na versão 3.04, para a constelação GPS (L1);
- Arquivo de texto com os resultados do processamento do PPS;
- Arquivo de texto com as coordenadas dos satélites utilizados no PPS;
- Arquivo de texto com os resíduos do ajustamento;
- Arquivo KML com as coordenadas extraídas do resultado do processamento;
- Arquivo binário com as efemérides do SUPL utilizadas;
- Arquivo binário com as configurações definidas para a execução do PPS; e
- Arquivo binário com os dados GNSS brutos, de todas as constelações e frequências disponíveis.

O arquivo ZIP pode ser compartilhado diretamente pela interface nativa no Android, como é apresentado na Figura 52. Além disso, o arquivo também pode ser enviado para um banco de dados em nuvem conforme é mostrado na subseção 6.2.2.

Figura 52: Arquivo gerado pelo INCT-GNSS-APP para compartilhamento.



Fonte: (O AUTOR, 2020).

Arquivo CSV com os resultados da execução do PPS em tempo real

O INCT-GNSS-APP gera um arquivo de texto no formato CSV contendo o resultado da execução do PPS em tempo real. O arquivo é dividido em suas seções: cabeçalho e área de dados. Na seção de cabeçalho as linhas começam com o caractere “#” e contém informações sobre as configurações selecionadas para o processamento. Já na área de dados estão os resultados propriamente ditos. Cada linha na área de dados é o resultado do processamento de uma época de dados. O caractere separador escolhido para o arquivo foi o caractere “;”. Um trecho de um arquivo de resultados gerado pelo INCT-GNSS-APP é apresentado na Figura 53.

Figura 53: Trecho de arquivo CSV de resultados, gerado pelo INCT-GNSS-APP.

```

1 |
2 | # Header Description:
3 | #
4 | # Version: 1 Platform: 9 Manufacturer: Xiaomi Model: Redmi Note 7
5 | #
6 | # Epoch (GPS time); Constellations: N_Sat; Sat_List; X(m); Y(m); Z(m); Dtr(s); SigmaX(m); SigmaY(m); SigmaZ(m); SigmaDtr(s); Lat(ou) Lon(ou) Alt(m); DH(m); DE(m); DO(m); SigmaD(m);
  SigmaZ(m); SigmaU(m); Dtr(m); SigmaDtr(m); A-posteriori; Chi-Squares result
7 | #
8 | # GNSS Constellations used: GALILEO
9 | # Ionospheric Model used: None
10 | # Tropospheric Model used: None
11 | # Elevation Mask Applied: None
12 | # Weight Function Applied: Identity
13 | # Statistical Test Applied: Chi-Squares (95%) - See the app documentation for details.
14 | #
15 | # Starting reference location:
16 | # Provider: gps
17 | # Lat(ou) Lon(ou) Alt(m): -22.11897159; -51.41107025; 420.60565185546975;
18 | # Horizontal Accuracy: 27.872002
19 | # Vertical Accuracy: 27.872002
20 | #
21 | 2020-07-07 13:25:45.442; GALILEO; 4; E9 E11 E24 E31 ; 3687426.0415929034; -4620946.702692455; -2394771.899011785; -1.543544143529639E-6; Infinity; Infinity; Infinity; Infinity;
  -22.11903473008564; -51.41093118418965; 382.7870705351233; -3.7370235724753353; -6.994624119987565; -43.131529754911596; NaN; NaN; NaN; -462.8680329600393; Infinity; Infinity; false
22 | 2020-07-07 13:25:50.442; GALILEO; 5; E4 E9 E11 E24 E31 ; 3687438.80965007; -4620957.84488821; -2394784.1629683743; -1.50693341401559E-6; 20.087547089754666; 14.740483490697957;
  6.165940310266121; 5.1295496737323146E-6; -22.11901720794666; -51.411022714332264; 417.32846611272544; -2.5160481346173913; -14.8788477105717; -8.205769490334752;
  24.37969323301305; -7.443767439322208; 6.139753866969654; -451.7672223006236; 15.37788503666507; 65.69470531054793; false
23 | 2020-07-07 13:25:51.442; GALILEO; 5; E4 E9 E11 E24 E31 ; 3687423.458167822; -4620987.116301922; -2398777.268330638; -1.54689999009936E-6; 16.042962371288396; 13.253045398315008;
  7.342522019023543; 4.612328293083944E-6; -22.119020599945035; -51.41107415382019; 398.0932218795642; 7.043354244074434; -10.094421968165646; -25.08942927963715; 21.821510599865112;
  -6.69182212951597; 5.5204289151064705; -463.74959478030017; 13.82742627123585; 53.12030405215251; false
24 | 2020-07-07 13:25:52.442; GALILEO; 5; E4 E9 E11 E24 E31 ; 3687447.476594253; -4620987.026194127; -2398773.8716789215; -1.5002830397224724E-6; 28.262945461360903; 20.798164618783137;
  11.509379292971216; 7.22415204281377E-6; -22.11904149048302; -51.4105918784995; 410.8263763330306; -14.8746621100569494; -5.4710089702289025; -15.749274497957172; 34.3369894478069;
  -18.485007579194621; 8.639172859190520; -449.77354017411164; 21.857503059569672; 130.33971264969664; false
25 | 2020-07-07 13:25:53.442; GALILEO; 5; E4 E9 E11 E24 E31 ; 3687450.878145148; -4620995.06522914; -2398769.599329484; -1.50464011232349E-6; 27.301176842487105; 20.02591264744311;
  11.0566621433332; 6.97063161304168E-6; -22.119037426342; -51.410854039629164; 409.5643557119896; -18.7847523242424002; -1.281378939931416; -16.68055514614392; 33.1324444781014;
  -10.11779495820065; 8.330439113365464; -451.0797976365423; 20.897427864948935; 121.3563889234027; false

```

Fonte: (O AUTOR, 2020).

Na seção de cabeçalho são apresentadas informações sobre: modelos de correção ionosférica e troposférica (se aplicados); máscara de elevação (se aplicada); função de ponderação utilizada no ajustamento e informações referente ao teste Chi-Quadrado aplicado, conforme é discutido na Seção 3.5.

Na seção de dados, são apresentadas para cada época: O tempo GPS, número de satélites utilizados e uma lista com tais satélites; as coordenadas geocêntricas cartesianas e seus respectivos desvio-padrão. Também são apresentadas as coordenadas geodésicas de latitude, longitude e altitude, além do erro do relógio do receptor em segundos e em metros. As coordenadas são dadas no *World Geodetic System 84* (WGS 84). O valor a posteriori do ajustamento e o resultado do teste Chi-Quadrado também são apresentados para cada época processada.

8.1.2. Aplicativo para armazenamento e compartilhamento de dados GNSS

Com o intuito de armazenar e gerenciar o acesso aos dados gerados pelo INCT-GNSS-APP foi implementado um segundo aplicativo denominado: INGSS-GNSS-APP Uploader.

O INCT-GNSS-APP Uploader foi desenvolvido como um aplicativo a parte para que fosse possível utilizar as tecnologias mais recentes disponíveis no contexto de desenvolvimento de aplicativos móveis. Com isso o INGSS-GNSS-APP Uploader foi desenvolvido utilizando o framework de desenvolvimento híbrido React Native.

O React Native⁸ é um framework criado e mantido pela empresa Meta, responsável pelo Facebook. Por tratar-se de um framework de desenvolvimento híbrido, o React Native é capaz de gerar código nativo para o Android (Java) e para o iOS (Objective C) a partir de código escrito na linguagem de programação Javascript. Com isso ganha-se agilidade no processo de desenvolvimento.

Apesar de gerar código para o iOS, que no escopo desse projeto não é utilizado, ainda assim é viável a utilização do React Native. Utilizando esse framework é possível ter acesso à diversos componentes visuais customizáveis e diversos plugins disponibilizados por diversos desenvolvedores ao redor do mundo. Dentre esses plugins existe o plugin para acesso ao Firestore⁹, serviço do Firebase para armazenamento em nuvem.

Com isso, diminui-se a complexidade e tempo de desenvolvimento para criar um banco de dados em nuvem para armazenar e compartilhar os dados GNSS brutos. Desenvolvê-lo no nível nativo, utilizando Java traria complexidade adicional ao projeto. A tela principal do INCT-GNSS-APP Uploader é apresentada na Figura 54.

Figura 54: Tela inicial do INCT-GNSS-APP Uploader.



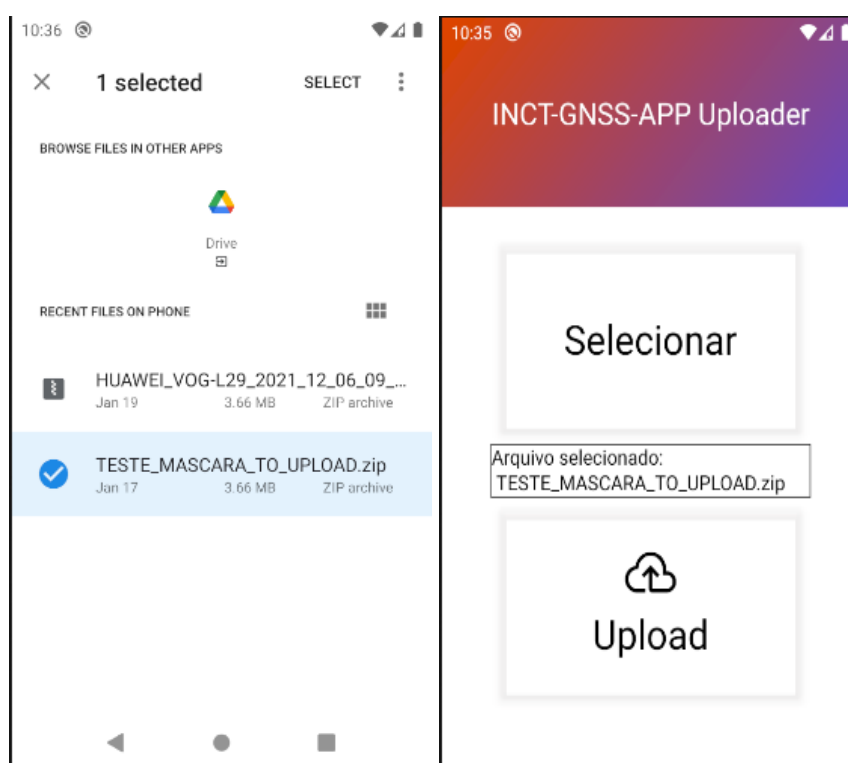
Fonte: (O AUTOR, 2023).

⁸ <https://reactnative.dev/>

⁹ <https://www.npmjs.com/package/@react-native-firebase/firestore>

Esse aplicativo funciona ao selecionar os arquivos do tipo criados pelo INCT-GNSS-APP e enviá-los ao banco de dados de armazenamento em nuvem. Ao clicar no botão “Selecionar”, o usuário seleciona um arquivo para ser enviado, conforme é apresentado na Figura 55.

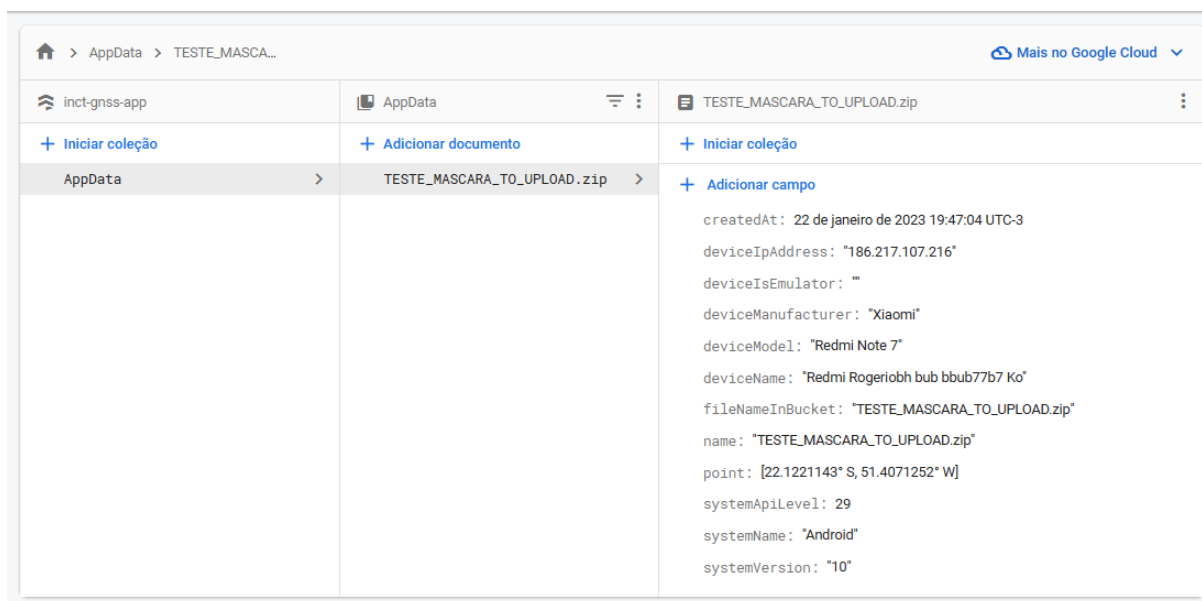
Figura 55: Seleção de arquivo para upload no INCT-GNSS-APP Uploader.



Fonte: (O AUTOR, 2023).

Por fim, ao clicar no botão “Upload” o arquivo é enviado e listado na página do Firebase correspondente ao banco de dados do aplicativo, como é apresentado na Figura 56.

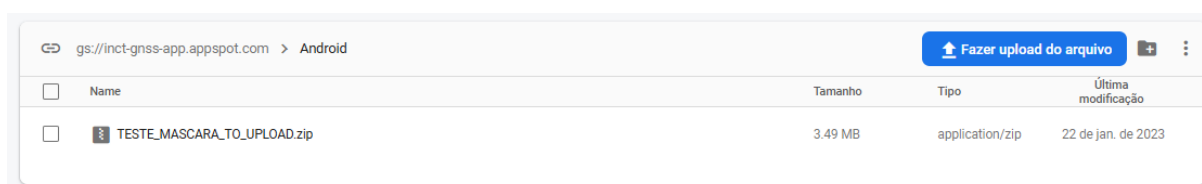
Figura 56: Arquivo armazenado no banco de dados em nuvem do Firestore.



Fonte: (O AUTOR, 2023).

Nesse banco de dados são armazenadas informações sobre o dispositivo e versão do Android utilizado no smartphone que efetuou o envio do arquivo. O campo *fileNameInBucket* representa o nome do arquivo zip, que fica armazenado em um segundo banco de dados dedicado apenas ao armazenamento de arquivos binários, como é apresentado na Figura 57. O segundo banco de dados foi implementado utilizando o plugin para o Firebase Storage¹⁰.

Figura 57: Arquivo .zip gerado pelo INCT-GNSS-APP e armazenado no banco de dados em nuvem.



Fonte: (O AUTOR, 2023).

¹⁰ <https://www.npmjs.com/package/@react-native-firebase/storage>

8.2. Softwares desktop para pós processamento dos dados GNSS brutos

Com o intuito de aprimorar e encontrar possíveis erros na implementação do aplicativo INCT-GNSS-APP e executar testes e validações da qualidade dos resultados, foram implementados dois softwares: O INCT-GNSS-APP Post Processing e o INCT-GNSS-APP Post Processing Analysis.

8.2.1. Software para pós processamento – INCT-GNSS-APP Post Processing

Para o pós-processamento dos dados GNSS coletados foi desenvolvido um software de uso desktop, utilizando a linguagem de programação Java e a o pacote de componentes gráficos Java Swing¹¹. Com isso, o software pode ser executado nos principais sistemas operacionais para computadores do tipo desktop: Windows, Linux e MacOS. Este software recebeu o nome de INCT-GNSS-APP Post Processing e sua interface gráfica é apresentada na Figura 58.

Figura 58: Interface gráfica do INCT-GNSS-APP Post Processing.

The screenshot shows a Java Swing window titled "INCT-GNSS-APP Post Processing". The window contains a form with the following elements:

- Three input fields labeled "Dados Brutos", "Efemérides", and "Configurações", each followed by a "Selecionar" button.
- An "Opções" section containing:
 - Three checkboxes under "Ionosfera": "Klobuchar", "Hopfield", and "Saastamoinen".
 - A "Função Peso" dropdown menu currently showing "IDENTITY".
 - A "Máscara de Elevação (*)" checkbox and an associated input field.
- Two input fields: "Diretório de trabalho:" and "Prefixo" (which contains the text "_DESKTOP_").
- A large "Processar" button centered below the "Prefixo" field.
- A "Saídas" section at the bottom with a large empty text area for output.

Fonte: (2022, O AUTOR).

¹¹ <https://docs.oracle.com/en/java/javase/15/docs/api/index.html>

Com este software é possível selecionar os três arquivos binários gerados pelo INCT-GNSS-APP, que correspondem aos dados GNSS brutos, às efemérides transmitidas obtidas pelo SUPL e o arquivo de configurações do PPS, gerados durante o processamento em tempo real. A partir daí, é possível escolher entre aplicar, caso necessários, o modelo de Klobuchar ou escolher dentre os modelos de Hopfield e Saastamoinem. Também é possível escolher utilizar a matriz de identidade como a matriz de pesos do ajustamento ou aplicar uma função para considerar o ângulo de elevação dos satélites. Por fim, é possível aplicar uma máscara de elevação mínima para a seleção dos satélites a serem processados.

O código fonte do INCT-GNSS-APP Post Processing foi criado a partir do código fonte do aplicativo INCT-GNSS-APP original. Isso foi possível porque o Android utiliza como linguagem nativa a linguagem Java. Assim sendo, as melhorias de código no processamento do PPS pós processado, executadas a partir de testes com o INCT-GNSS-APP Post Processing, poderão ser inseridas novamente no aplicativo INCT-GNSS-APP original, e assim aprimorar o posicionamento em tempo real.

Arquivos gerados pelo INCT-GNSS-APP Post Processing

Assim como no aplicativo Android, o INCT-GNSS-APP Post Processing cria diversos arquivos durante seu processamento. Nesta Seção é decorrido sobre tais arquivos, sendo eles:

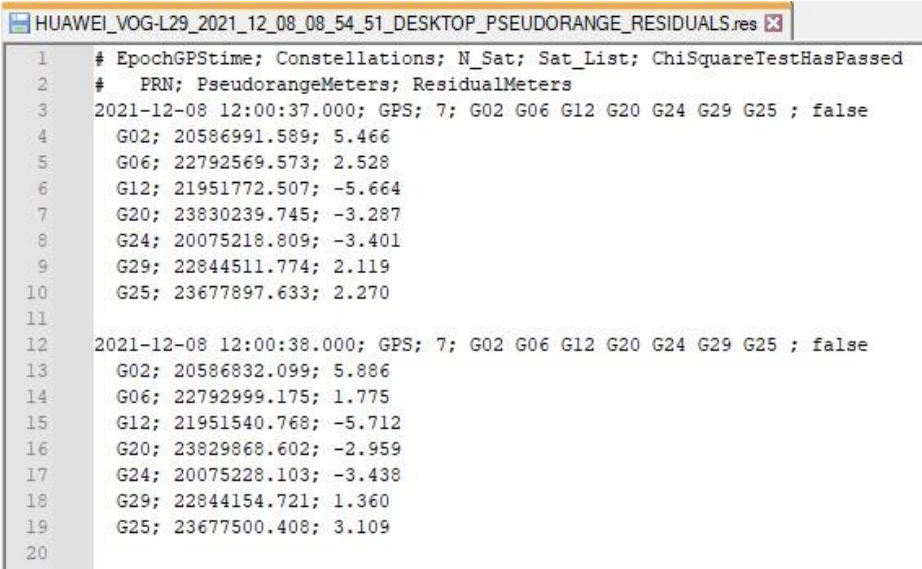
- Arquivo RINEX de observação na versão 3.04;
- Arquivo de resultados do PPS pós processado;
- Arquivo com os resíduos do ajustamento;
- Arquivo com as coordenadas dos satélites utilizados no ajustamento;
- Arquivo GPX com as coordenadas de resultado do processamento e
- Arquivo KML com as coordenadas de resultado do processamento.

Os arquivos KML, de coordenadas dos satélites e de resíduos são semelhantes aos arquivos criados pelo aplicativo Android. As principais diferenças estão nos acréscimos ao arquivo de resultados e a criação do arquivo GPX.

Arquivo com os resíduos do ajustamento

O INCT-GNSS-APP Post Processing cria um arquivo com os resíduos para cada observável de pseudodistância utilizada no MMQ. No arquivo são mostradas as próprias pseudodistâncias e a quais satélites correspondem, como é apresentado na Figura 59. Esse arquivo é utilizado para gerar o gráfico de resíduos apresentado na próxima subseção.

Figura 59: Trecho de arquivo de resíduos gerado pelo INCT-GNSS-APP Post Processing.



```

1 # EpochGPStime; Constellations; N_Sat; Sat_List; ChiSquareTestHasPassed
2 # PRN; PseudorangeMeters; ResidualMeters
3 2021-12-08 12:00:37.000; GPS; 7; G02 G06 G12 G20 G24 G29 G25 ; false
4 G02; 20586991.589; 5.466
5 G06; 22792569.573; 2.528
6 G12; 21951772.507; -5.664
7 G20; 23830239.745; -3.287
8 G24; 20075218.809; -3.401
9 G29; 22844511.774; 2.119
10 G25; 23677897.633; 2.270
11
12 2021-12-08 12:00:38.000; GPS; 7; G02 G06 G12 G20 G24 G29 G25 ; false
13 G02; 20586832.099; 5.886
14 G06; 22792999.175; 1.775
15 G12; 21951540.768; -5.712
16 G20; 23829868.602; -2.959
17 G24; 20075228.103; -3.438
18 G29; 22844154.721; 1.360
19 G25; 23677500.408; 3.109
20

```

Fonte: (O AUTOR, 2023).

Arquivo com as coordenadas dos satélites utilizados no ajustamento

O INCT-GNSS-APP Post Processing também cria um arquivo com os as coordenadas estimadas durante a execução do PPS. No arquivo são apresentados as coordenadas cartesianas e os ângulos de azimute e elevação de cada satélite. Esse arquivo pode ser utilizado para criar gráficos do tipo *skyplot* por exemplo, caso seja necessário. Um trecho de arquivo de coordenadas é apresentado na Figura 60.

Figura 60: Trecho de arquivo de coordenadas gerado pelo INCT-GNSS-APP Post Processing.

```

HUAWEI_VOG-L29_2021_12_08_08_54_51_DESKTOP_SAT_COORDS.sat
1 # EpochGPStime; Constellations; N_Sat; Sat_List; ChiSquareTestHasPassed
2 # PRN; AzimuthDeg; ElevationDeg; Xm; Ym; Zm; DTSS
3 2021-12-08 12:00:39.000; GPS; 7; G02 G06 G12 G20 G24 G29 G25 ; false
4 G02; 72.56; 68.07; 20834078.555; -14458873.747; -7397004.488; -0.000644892214
5 G06; 132.55; 29.2; 18436078.109; 438918.385; -19069578.135; 0.000137358507
6 G12; 195.41; 44.66; 5821661.129; -13932251.927; -22104859.428; -0.000137701277
7 G20; 50.08; 15.54; 24735683.736; -2580696.960; 8930432.908; 0.000517755321
8 G24; 303.09; 78.03; 12881057.223; -21754886.735; -7698803.899; 0.000273479427
9 G29; 287.69; 30.61; -2790039.390; -26355354.942; -1238598.996; -0.000451165242
10 G25; 226.95; 22.38; -7153540.665; -16768816.801; -19681203.585; 0.000252588145
11
12 2021-12-08 12:00:40.000; GPS; 7; G02 G06 G12 G20 G24 G29 G25 ; true
13 G02; 72.58; 68.08; 20833559.291; -14457951.734; -7399992.185; -0.000644892218
14 G06; 132.56; 29.2; 18434337.619; 440613.653; -19071209.758; 0.000137358517
15 G12; 195.4; 44.66; 5824200.708; -13931217.517; -22104828.909; -0.000137701280
16 G20; 50.09; 15.54; 24736762.514; -2580183.582; 8927571.486; 0.000517755319
17 G24; 303.12; 78.03; 12881983.395; -21755323.189; -7695872.683; 0.000273479427
18 G29; 287.68; 30.61; -2789802.761; -26355233.638; -1241820.123; -0.000451165248
19 G25; 226.95; 22.38; -7151190.311; -16768180.301; -19682592.016; 0.000252588155
20

```

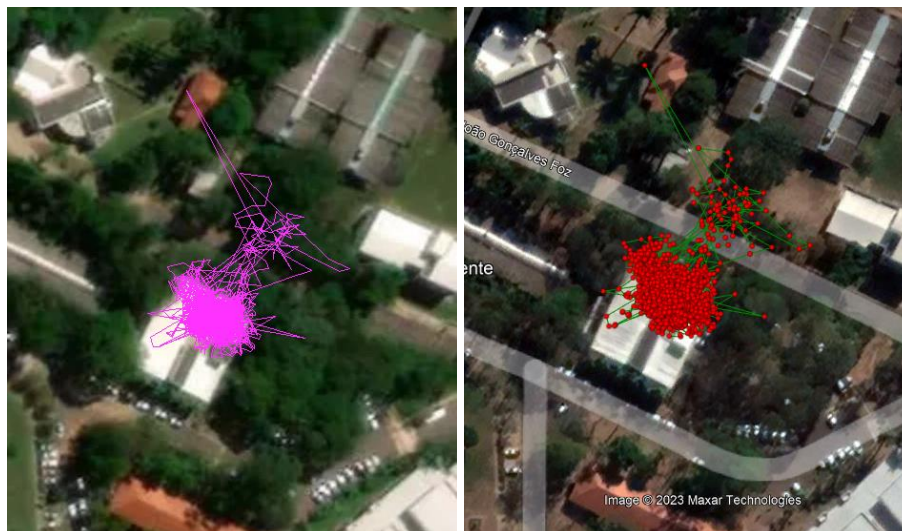
Fonte: (O AUTOR, 2023).

Arquivos GPX e KML

Conforme apresentado na subseção 5.1.3, os arquivos GPX são utilizados para armazenar dados de trajetórias. Uma outra vantagem deste tipo de arquivo é que este também pode ser utilizado no Java OpenStreetMap (JOSM¹²) que é a versão desktop do OpenStreetMap, em alternativa ao uso do Google Earth. Um exemplo de arquivo GPX gerado pelo INCT-GNSS-APP e seu correspondente no formato KML foram inseridos no OpenStreetMap e Google Earth, sendo apresentados na Figura 61.

¹² <https://josm.openstreetmap.de/>

Figura 61: Visualização dos arquivos GPX e KML gerados pelo INCT-GNSS-APP Post Processing.



Fonte: (O AUTOR, 2023).

Arquivo CSV de resultados do processamento

O arquivo CSV de resultados criado pelo INCT-GNSS-APP Post Processing é semelhante àquele criado originalmente pelo aplicativo Android. A principal diferença é que na versão pós processada o arquivo contém mais informações.

Na versão pós-processada foi implementado o controle de qualidade pelo método de Data Snooping. Com isso, foram acrescentadas nas informações de resultado de cada época uma lista com os satélites excluídos pelo Data Snooping. Também foram adicionadas informações de discrepância em relação à coordenada de referência adotada, bem como o erro 3D obtido em cada época. As coordenadas obtidas são dadas no WGS 84. Um trecho do arquivo de resultados gerado pelo INCT-GNSS-APP Post Processing, após a execução do PPS pós processado é apresentado na Figura 62.

Figura 62: Trecho de arquivo de resultado de pós processamento gerado pelo INCT-GNSS-APP Post Processing.

```

HUAWEI_VOG-L29_2021_12_08_08_54_51_DESKTOP_PROCESSED_RESULT.coord L3
1 #
2 # Header Description:
3 #
4 # Version: 1 Platform: null Manufacturer: null Model: null
5 #
6 #
7 # GNSS Constellations used: GPS
8 # Ionospheric Model used: Klobuchar
9 # Tropospheric Model used: Saastamoinen
10 # Elevation Mask Applied: 15.0°
11 # Weight Function Applied: By Satellite Elevation Angle [°]
12 # Statistical Tests Applied:
13 # General Adjustment p-value Chi-Squared test (95%)
14 # Data Snooping Chi-Squared test (99%)
15 # Starting reference location:
16 # Provider:
17 # Lat(°); Lon(°); Alt(m): 0.0; 0.0; 0.0;
18 # Horizontal Accuracy: 0.0
19 # Vertical Accuracy: 0.0
20 #
21 # Ground truth reference location: None
22 EpochGPStime; Constellations; N_Sat; Sat_List; X[m]; Y[m]; Z[m]; Dtr[s]; SigmaX[m]; SigmaY[m]; SigmaZ[m]; SigmaDtr[s]; Lat[Deg]; Lon[Deg]; Alt[m]; DX[m]; DY[m]; DZ[m];
3D_ERROR_XYZ[m]; dn[m]; de[m]; du[m]; Sigma_dn[m]; Sigma_de[m]; Sigma_du[m]; 3D_ERROR_enu[m]; Dtr[m]; SigmaDtr[m]; DegreesOfFreedom; AprioriValue; ChiSquareTestCriticalValue;
ChiSquareTestHasFailed; DataSnoopingWtMaxValue; DataSnoopingRemovedSats_List
23 2021-12-08 11:55:15.000; GPS; 6; G02 G06 G12 G24 G29 G25 ; 3687710.064; -4620674.026; -2387113.901; 0.000000069025; 6.596; 11.504; 11.156; 0.000000038197; -22.122060945;
-51.407010364; 463.764; -17.648; 10.754; 10.512; 23.186; 10.718; 20.502; 1.547; 5.400; 2.021; 16.342; 23.186; 20.393; 11.451; 2; 3.961; 2.996; false; 0.000;
24 2021-12-08 11:55:16.000; GPS; 6; G02 G06 G12 G24 G29 G25 ; 3687709.610; -4620673.993; -2387114.563; 0.000000035428; 7.674; 13.384; 12.982; 0.000000044445; -22.122067565;
-51.407013541; 463.719; -17.194; 10.711; 11.174; 23.135; 11.238; 20.120; 2.027; 6.284; 2.351; 19.014; 23.135; 10.621; 13.324; 2; 5.361; 2.996; false; 0.000;
25 2021-12-08 11:55:17.000; GPS; 6; G02 G06 G12 G24 G29 G25 ; 3687711.357; -4620677.856; -2387115.831; 0.000000015508; 6.420; 11.198; 10.863; 0.000000037188; -22.122064169;
-51.407023724; 468.010; -18.941; 14.584; 12.442; 26.949; 11.682; 23.901; 4.300; 5.259; 1.967; 15.909; 26.949; 4.649; 11.149; 2; 3.752; 2.996; false; 0.000;
26 2021-12-08 11:55:18.000; GPS; 6; G02 G06 G12 G24 G29 G25 ; 3687709.454; -4620674.972; -2387114.508; -0.000000019538; 5.599; 9.765; 9.475; 0.000000032436; -22.122064806;
-51.407020699; 464.325; -17.038; 11.700; 11.119; 23.470; 10.859; 20.616; 2.813; 4.588; 1.715; 13.975; 23.470; -5.857; 9.724; 2; 2.853; 2.996; true; 0.000;

```

Fonte: (O AUTOR, 2023).

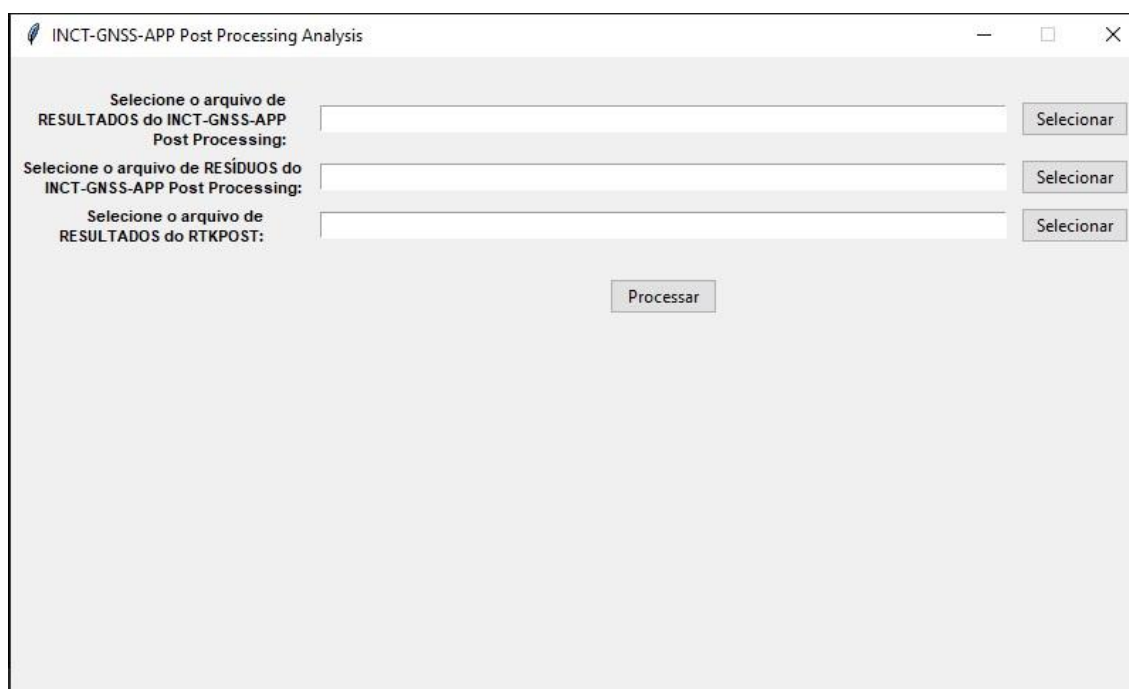
8.2.2. Software para controle de qualidade de pós processamento – INCT-GNSS-APP Post Processing Analysis

Para efetuar a análise estatística numérica e gráfica dos resultados de pós processamento executados pelo INCT-GNSS-APP Post Processing foi criado o INCT-GNSS-APP Post Processing Analysis.

O INCT-GNSS-APP Post Processing Analysis foi implementado utilizando o framework para criação de interfaces gráficas do Python denominado tkinter¹³. O tkinter, assim como a linguagem de programação Python, são multiplataforma. Com isso, o INCT-GNSS-APP Post Processing Analysis pode ser executado nos principais sistemas operacionais para computadores do tipo desktop: Windows, MacOS e Linux. A interface gráfica do INCT-GNSS-APP Post Processing Analysis é apresentada na Figura 63.

¹³ <https://docs.python.org/3/library/tkinter.html>

Figura 63: Interface gráfica do INCT-GNSS-APP Post Processing Analysis.



Fonte: (2023, O AUTOR).

O INCT-GNSS-APP Post Processing Analysis gera dados estatísticos como média e desvio padrão para as discrepâncias e precisões estimadas no ajustamento. Estes valores são então armazenados em arquivos no formato do Microsoft Excel (.xlsx). Além disso, também é salvo um arquivo de texto contendo os valores de *Root Mean Squared Error* (RMSE) 2D e 3D, como é apresentado na Figura 64.

Figura 64: Arquivos criados pelo FCT-INCT-GNSS Post Processing Analysis.

Nome	Data de modificação	Tipo	Tamanho
PPS-FCT-INCT - Delta DX[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
PPS-FCT-INCT - Delta DY[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
PPS-FCT-INCT - Delta DZ[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
PPS-FCT-INCT - Sigma SigmaX[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
PPS-FCT-INCT - Sigma SigmaY[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
PPS-FCT-INCT - Sigma SigmaZ[m].xlsx	02/02/2023 15:12	Planilha do Micro...	6 KB
RMSE.csv	02/02/2023 15:13	Arquivo CSV	1 KB
RTKLib - Delta DX[m].xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB
RTKLib - Delta DY[m].xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB
RTKLib - Delta DZ[m].xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB
RTKLib - Sigma sdx(m).xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB
RTKLib - Sigma sdy(m).xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB
RTKLib - Sigma sdz(m).xlsx	02/02/2023 15:13	Planilha do Micro...	6 KB

Fonte: (O AUTOR, 2023).

Os arquivos para o Excel possuem valores de média e desvio padrão além de valor mínimo e valores máximos para vários percentis. Um exemplo para a discrepância da coordenada cartesiana X em metros, que no software é denominada DX[m]. É apresentado na Figura 65.

Figura 65: Trecho de arquivo criado pelo FCT-INCT-GNSS Post Processing Analysis.

	A	B	C
1		DX[m]	
2	count	3532	
3	mean	-10.3744	
4	std	14.90956	
5	min	-139.619	
6	25%	-16.4593	
7	50%	-9.7815	
8	75%	-3.054	
9	max	107.712	
10			

Fonte: (2023, O AUTOR).

O arquivo de texto contém os RMSE 2D e 3D calculados para os pós processamentos, em metros. Os valores de RMSE são calculados pelo INCT-GNSS-APP Post Processing Analysis segundo as Equação 13 e Equação 14 (Monico, Posicionamento pelo GNSS: descrição, fundamentos e, 2008).

$$RMSE\ 2D = \sqrt{\begin{matrix} (X_{est} - X_{ref})^2 + (\sigma_{X_{est}} - \sigma_{X_{ref}})^2 + \\ (Y_{est} - Y_{ref})^2 + (\sigma_{Y_{est}} - \sigma_{Y_{ref}})^2 \end{matrix}} \quad (13)$$

$$RMSE\ 3D = \sqrt{\begin{matrix} (X_{est} - X_{ref})^2 + (\sigma_{X_{est}} - \sigma_{X_{ref}})^2 + \\ (Y_{est} - Y_{ref})^2 + (\sigma_{Y_{est}} - \sigma_{Y_{ref}})^2 + \\ (Z_{est} - Z_{ref})^2 + (\sigma_Z - \sigma_{Z_{ref}})^2 \end{matrix}} \quad (14)$$

Onde: X, Y e Z são as coordenadas cartesianas geocêntricas;

- *est* refere-se aos valores estimados no ajustamento e;

- *ref* refere-se aos valores de referência adotadas para a coordenada conhecida.

Um exemplo de arquivo com os valores de RMSE calculados pelo INCT-GNSS-APP Post Processing Analysis é apresentado na Figura 66.

Figura 66: Arquivo de RMSEs criado pelo INCT-GNSS-APP Post Processing Analysis.

```

RMSE.csv
1 Software;RMSE_2D;RMSE_3D
2 PPS-FCT-INCT;25.566548294102745;26.87215208140438
3 RTKLib;18.202609702630586;19.220113715653483
4

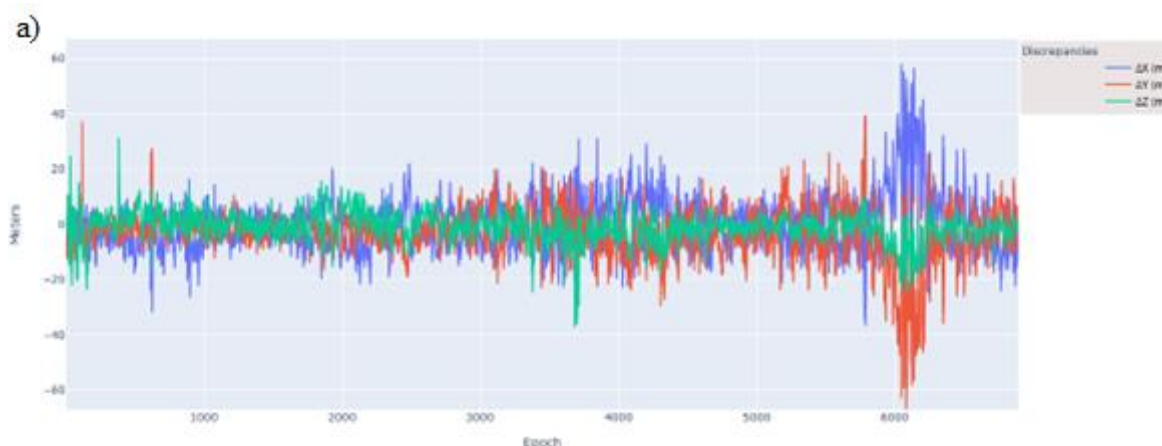
```

Fonte: (2023, O AUTOR).

Gráficos gerados pelo INCT-GNSS-APP Post Processing

Além dos arquivos com dados numéricos apresentados nesta Seção, o INCT-GNSS-APP Post Processing Analysis também gera gráficos para serem utilizados na análise de qualidade dos processamentos. Os gráficos criados pelo software foram implementados utilizando a biblioteca Plotly¹⁴ e por isso são interativos, conforme é apresentado no gráfico de discrepâncias da Figura 67.

Figura 67: Gráfico de discrepâncias interativo criado pelo INCT-GNSS-APP Post Processing Analysis, a) gráfico estático, b) gráfico com análise exploratória através do mouse e c) gráfico com seleção de área para aplicação de zoom.



¹⁴ <https://plotly.com/>



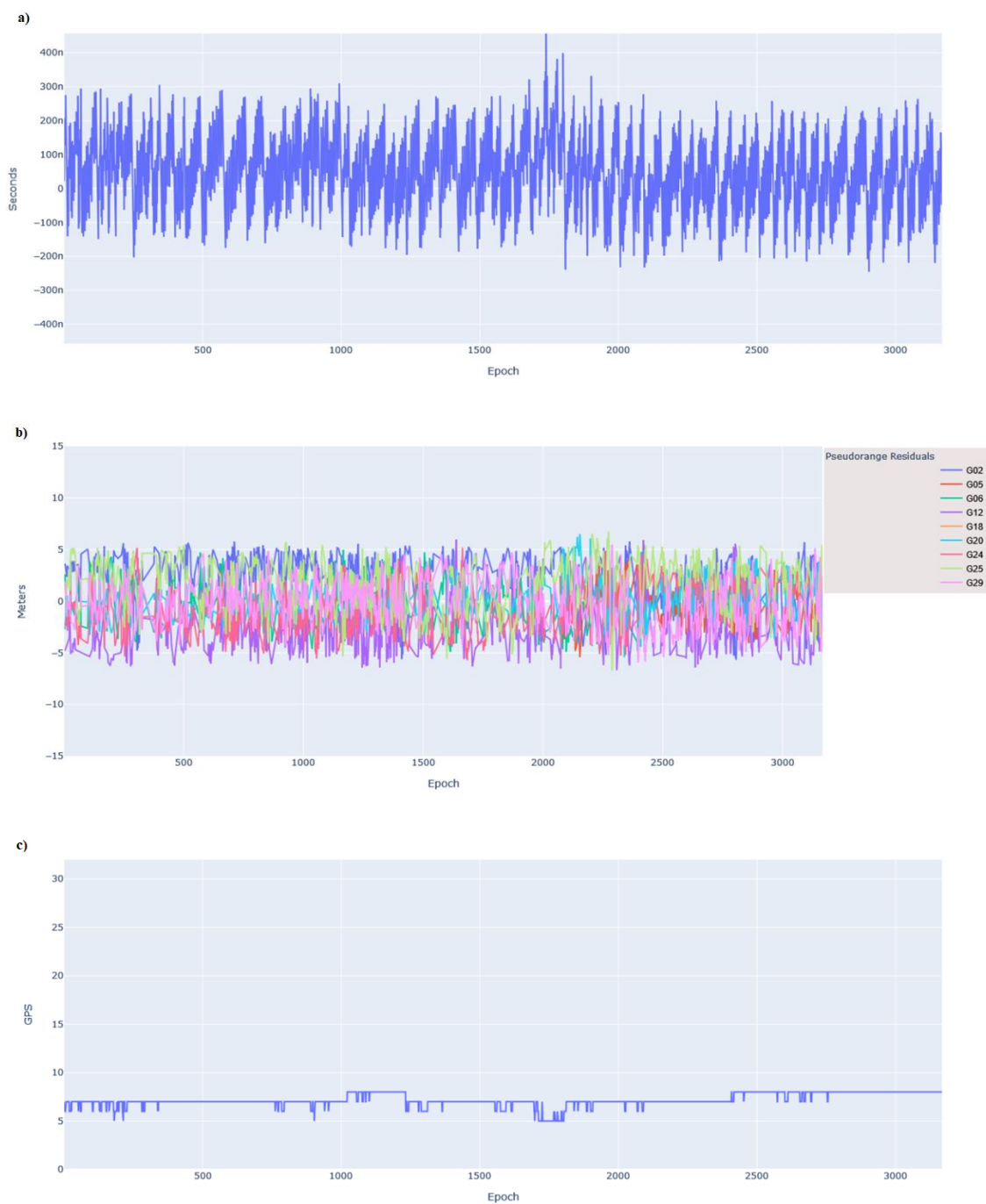
Fonte: (O AUTOR, 2023).

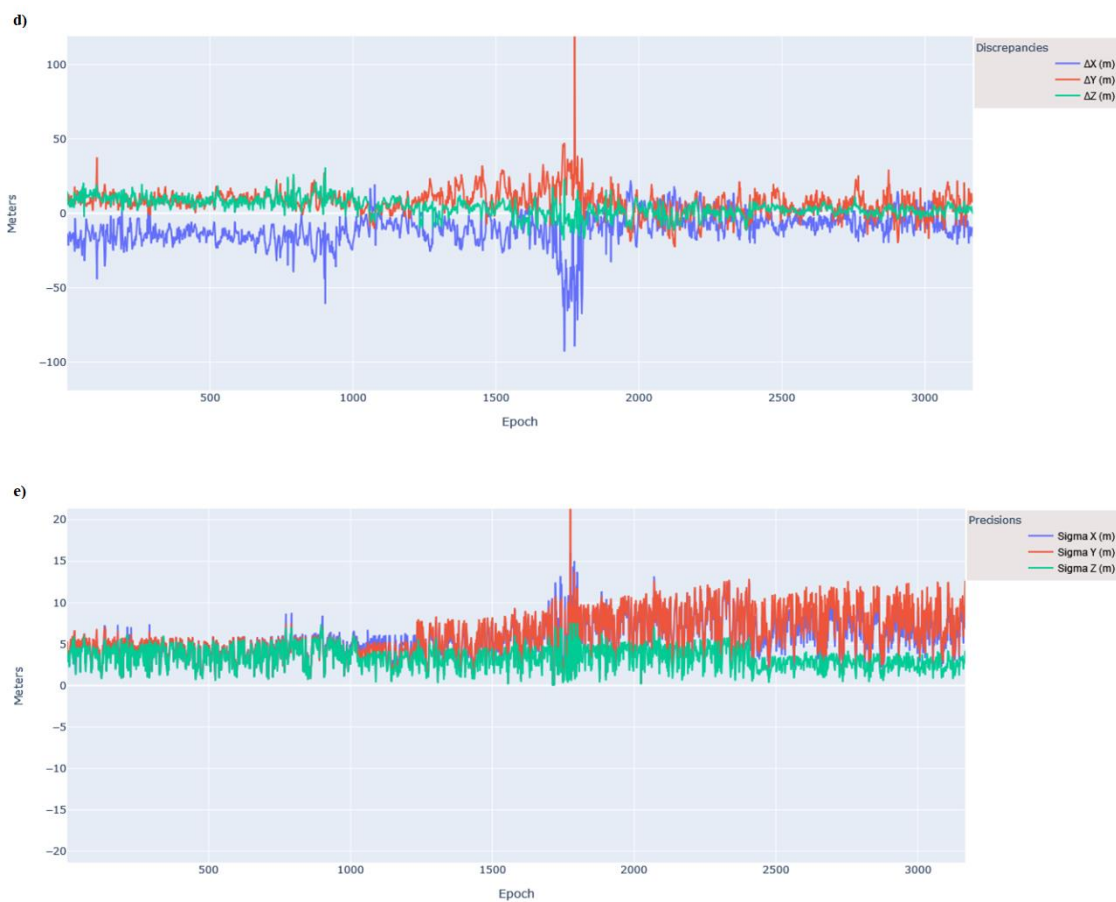
O INCT-GNSS-APP Post Processing Analysis gera cinco tipos de gráficos, sendo eles:

- Gráfico de erro do relógio do receptor, em relação ao sistema de tempo GPS;
- Gráfico de resíduos do ajustamento pelo MMQ;
- Gráfico do número de satélites utilizados no processamento;
- Gráfico de discrepâncias em relação à coordenada de referência; e
- Gráfico de precisão das coordenadas estimadas.

Na Figura 68 são apresentados exemplos de gráficos gerados pelo INCT-GNSS-APP Post Processing Analysis.

Figura 68: Gráficos gerados pelo INCT-GNSS-APP Post Processing Analysis, a) gráfico do erro do relógio do receptor, b) gráfico de resíduos do ajustamento, c) gráfico do números de satélites utilizados, d) gráfico de discrepâncias e e) gráfico de precisões.





Fonte: (O AUTOR, 2023).

8.3. Fluxograma do ecossistema INCT-GNSS-APP

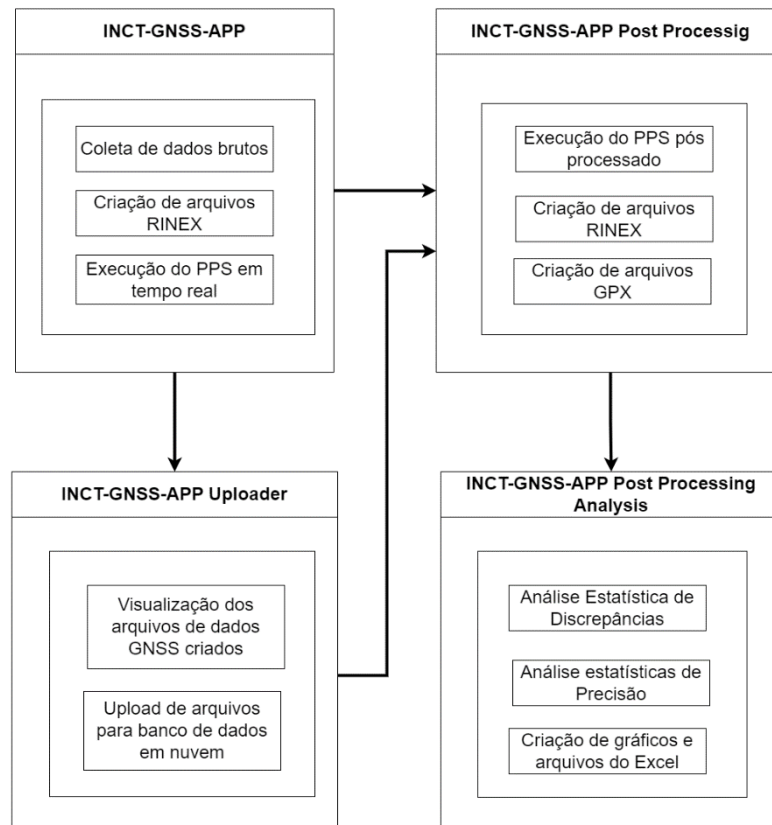
Na Figura 69 é apresentado o fluxograma que envolve o ecossistema de softwares desenvolvido nessa pesquisa. O início do fluxo acontece no aplicativo para Android INCT-GNSS-APP. Nesse aplicativo o usuário pode coletar dados GNSS brutos, arquivos RINEX ou executar o PPS em tempo real. Após isso, o usuário pode transferir os arquivos criados para um computador, de modo direto, utilizando uma conexão USB, por exemplo, ou utilizando o aplicativo para Android INCT-GNSS-APP Uploader. Nesse caso, os arquivos gerados pelo INCT-GNSS-App, ao executar o PPS, são enviados a um banco de dados em nuvem, implementado via Firebase. Após isso, o usuário pode baixar os arquivos para o computador através da Internet.

Neste ponto, com os dados acessíveis ao INCT-GNSS-APP Post Processing, é possível executar o pós processamento dos dados GNSS brutos originalmente capturados pelo

smartphone. Como resultado do pós processamento, o arquivo CSV de resultados e o arquivo RINEX são gerados.

Por fim, o INCT-GNSS-APP Post Processing Analysis é responsável por fornecer informações da qualidade do pós-processamento. São criadas tabelas com os valores de discrepância e precisão estimadas, bem como seus respectivos gráficos.

Figura 69: Fluxograma da interação entre os softwares do ecossistema INCT-GNSS-APP.



Fonte: (O AUTOR, 2023).

9. EXPERIMENTOS E RESULTADOS

Neste capítulo são apresentados os resultados ao testar o processamento de dados GNSS coletados por smartphones com o sistema operacional Android, utilizando diferentes métodos de processamento e softwares.

Através dos experimentos busca-se identificar possíveis diferenças entre os resultados a partir de dados coletados simultaneamente em smartphones diferentes. Ao processar os dados no INCT-GNSS-APP Post Processing é possível validar a implementação do MMQ, bem como da implementação do cálculo de pseudodistâncias, implementado considerando as especificidades para smartphones Android, discutidas nesse projeto. Além disso, processando os dados no RTKLib, a partir de arquivos RINEX gerados pelo INCT-GNSS-APP Post Processing, é possível validar o cálculo das observáveis de fase da onda portadora.

Caso os resultados do RTKPOST, correspondam a valores próximos àqueles esperados conforme a literatura, verifica-se a boa qualidade dos dados GNSS brutos capturados pelos smartphones Android.

9.1. Smartphones utilizados

Para a execução dos experimentos foram utilizados dois smartphones, sendo eles:

1. Huawei P30 Pro Dual SIM VOG-L29 - Global market e
2. Xiaomi Mi 9

Em relação ao nome do Huawei P30, o termo Dual SIM, significa que este possui dois *slots* do tipo *SIM* para a inserção de chips de operadores de celular. Além disso, o termo VOG-L29 – Global market especifica a variação do modelo, dentre os três disponíveis no mercado, sendo eles: VOG-L04 – Global market; VOG-L09 – Global market e VOG-L29 – Global market (GMS Arena, 2023).

Portanto, desse ponto em diante no texto quando utilizado o termo Huawei P30 Pro, fica definido que é uma referência ao smartphone Huawei P30 Pro Dual SIM VOG-L29 - Global market.

Na Figura 70 são apresentadas imagens dos aparelhos Huawei P30 Pro e Xiaomi Mi 9, respectivamente.

Figura 70: Imagens dos aparelhos Huawei P30 Pro a) e Xiaomi Mi 9 b).

Fonte: (GMS ARENA, 2023).

As especificações técnicas dos dois smartphones são apresentadas na Tabela 10.

Tabela 10: Especificações técnicas dos smartphones Huawei P30 Pro e Xiaomi Mi 9.

Característica	Huawei P30 Pro	Xiaomi Mi 9
Sistema operacional	Android 9.0 (Pie), atualizável para Android 10	Android 9.0 (Pie), atualizável para Android 10
Interface gráfica	EMUI 10	MIUI 12.5
Chipset	Kirin 980 (7 nm)	Qualcomm SM8150 Snapdragon 855 (7 nm)
CPU	Octa-core (2x2.6 GHz Cortex-A76 & 2x1.92 GHz Cortex-A76 & 4x1.8 GHz Cortex-A55)	Octa-core (1x2.84 GHz Kryo 485 & 3x2.42 GHz Kryo 485 & 4x1.78 GHz Kryo 485)
GPU	Mali-G76 MP10	Adreno 640
GNSS	GPS (L1+L5), GLONASS (L1), BDS (B1), GALILEO (E1+E5a), QZSS (L1+L5)	GPS (L1+L5), GLONASS (L1), BDS (B1), GALILEO (E1+E5a), QZSS (L1+L5)
Onda portadora	Sim	Não

Fonte: Adaptado de (GMS ARENA, 2023) e (ANDROID DEVELOPERS, 2023).

A respeito das especificações dos dois smartphones é importante destacar que apesar de possuírem a mesma versão do Android, possuem interfaces gráficas diferentes, dado que são do mesmo fabricante. Em termos de hardware os aparelhos possuem Chipset, CPU e GPU diferentes, ao mesmo em que, possuem as mesmas constelações GNSS e frequências.

Um detalhe importante é que o Xiaomi Mi 9 não apresenta a observável de fase da onda portadora, o que é diferente da versão anterior desse smartphone: o Xiaomi Mi 8, que foi objeto de estudo de (CHEM et al., 2019).

Este detalhe em que a versão mais antiga de um aparelho apresenta mais recursos GNSS do que a versão mais atual pode parecer contraditória a primeiro momento. Entretanto, deve-se lembrar que smartphones são desenvolvidos para o mercado tradicional, então o detrimento no fornecimento de uma observável GNSS, para melhorar aspectos do uso comum aos usuários, como melhorias na tela ou câmera, faz sentido no contexto geral de smartphones.

9.2. Estrutura para de coleta dos dados

Para a coleta de dados foi instalada uma estrutura próxima a estação de monitoramento ionosférico PRU2, que faz parte da rede INCT NavAer. A PRU2 fica sobre o terraço da central de laboratórios do curso de Graduação em Engenharia Cartográfica e de Agrimensura, da Faculdade de Ciências e Tecnologia (FCT) da Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), câmpus de Presidente Prudente/SP. A estrutura montada para a execução dos experimentos, próxima a PRU2, é apresentada na Figura 71.

Figura 71: Estrutura montada próxima a PRU2 para a coleta de dados.



Fonte: (O AUTOR, 2021).

Essa estrutura foi definida para evitar multicaminho, deixando os smartphones o mais alto possível. Caso necessário, é possível processar os dados da PRU2 coletados durante o mesmo período do que os smartphones.

9.3. Coordenadas de referência utilizadas

Para a execução do PPS pós processado e análise gráfica e numérica de seus resultados, foram utilizadas as coordenadas da estação PRU2 que são apresentadas na Tabela 11. Essas coordenadas foram obtidas, a partir da métodos de posicionamento de alta precisão, por (MACHADO et al., 2022). As coordenadas são dadas no *Datum* denominado Sistema de Referência Geocêntrico para as Américas (SIRGAS) 2000 e na época 2000.4.

Tabela 11: Coordenadas de referência para a estação PRU2.

Coordenada	Valor (m)	Precisão (m)
X	3687692,416	0,001
Y	-4620663,272	0,001
Z	-2387103,389	0,001

Fonte: Adaptado de (MACHADO et. al., 2022).

9.4. Visualização dos arquivos

Para a interpretação visual do processamento dos arquivos serão utilizados arquivos do tipo GPX, gerados pelo INCT-GNSS-APP Post Processing e pelo INCT-GNSS-APP Post Processing Analysis. A fim de melhor fidelidade das informações, os arquivos serão visualizados utilizando um ortofotomosaico de *Ground Sample Distance* (GSD) de 5 cm. O ortofotomosaico é apresentado na Figura 72. É destacado na imagem o local da central de laboratórios de cartografia, onde os dados foram coletados.

Figura 72: Ortofotomosaico utilizado para visualização de resultados.



Fonte: T2R Soluções Tecnológicas.

Para a melhor identificação visual do erro em relação à coordenada de referência da PRU2, foram criados círculos concêntricos e centrados na coordenada de referência da PRU2, como é apresentado na Figura 73. O círculo menor contém 5 metros de raio. Os círculos exteriores contêm 10, 20 e 30 metros de raio, respectivamente.

Figura 73: Círculos centrados na coordenada de referência da estação PRU2.



Fonte: (O AUTOR, 2020).

9.5. Coleta dos dados

Os dados coletados para os experimentos foram coletados entre o dia 06/12/2021 e 10/12/2021 como é apresentado na Tabela 12 e na Tabela 13. Os horários de início e fim das coletas são apresentadas em horário local (UTC – 3h). Os dados correspondem a semana GPS 2187 e aos dias do ano de 340 a 344, do ano de 2021.

Tabela 12: Horários de coleta de dados com o smartphone Huawei P30 Pro.

Data	Horário de Início	Horário de Término	Duração	Dia da Semana	Dia do Ano
06/12/2021	09:08:45	10:03:19	00:54:34	Segunda-feira	340
07/12/2021	09:13:09	10:57:07	01:43:58	Terça-feira	341
08/12/2021	08:55:15	10:50:35	01:55:20	Quarta-feira	342
09/12/2021	08:56:52	10:52:36	01:55:44	Quinta-feira	343
10/12/2021	08:42:23	10:35:56	01:53:33	Sexta-feira	344

Fonte: (O AUTOR, 2023).

Tabela 13: Horários de coleta de dados com o smartphone Xiaomi Mi 9.

Data	Horário de Início	Horário de Término	Duração	Dia da Semana	Dia do Ano
06/12/2021	-	-	-	Segunda-feira	340
07/12/2021	-	-	-	Terça-feira	341
08/12/2021	08:54:41	10:50:51	01:56:10	Quarta-feira	342
09/12/2021	08:57:18	11:19:54	02:22:36	Quinta-feira	343
10/12/2021	08:42:54	10:58:43	02:15:49	Sexta-feira	344

Fonte: (O AUTOR, 2023).

Como é apresentado na Tabela 13, os dados coletados nos dias 06/12/2021 e 07/12/2021 com o smartphone Xiaomi Mi 9 foram inválidos e não puderam ser processados. É possível que tenha acontecido algum problema com o smartphones nesse caso, como superaquecimento ou outro problema de hardware.

Com isso, os experimentos aqui apresentados correspondem aos dias 08/12/2021, 09/12/2021 e 10/12/2021 no período das 09:00 às 10:30, horário local. Assim sendo, foram processados 1 hora e 30 minutos de dados GNSS brutos, capturados em 3 dias consecutivos e com 2 smartphones diferentes. Os dados são coletados a cada 1 segundo, totalizando 5400 épocas de dados.

9.6. Experimento 1: INCT-GNSS-APP e RTKPOST aplicando o método PPS

Neste experimento o arquivo RINEX de observação, gerado durante o pós processamento, pelo INCT-GNSS-APP Post Processing foi processado no RTKPOST 2.4.2. Foi aplicado um PPS com os modelos de Klobuchar e Saastamoinem, além de máscara de elevação de 15°. As mesmas configurações foram aplicadas ao INCT-GNSS-APP Post Processing. Detalhes sobre as configurações do RTKPOST podem ser encontradas no Anexo A. É possível colocar o conteúdo de texto do Anexo A em um arquivo de texto no formato .conf e carregá-lo no RTKPOST, para aplicar tais configurações ao software.

9.6.1. Dia 08/12/2021

Smartphone Huawei P30 Pro

Processamento no INCT-GNSS-APP Post Processing

Na Tabela 14 são apresentadas informações relacionadas às discrepâncias obtidas para cada época processada pelo INCT-GNSS-APP Post Processing e que passaram no Teste Geral do Ajustamento.

Nesta tabela, o campo $Min_{100\%}$ representa o menor valor encontrado no primeiro quartil. O campo $Max_{25\%}$ representa o maior valor encontrado no segundo quartil. O campo $Max_{75\%}$ representa o maior valor no terceiro quartil e $Max_{100\%}$ representa o maior valor da amostra, no quarto quartil.

Tabela 14: Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	$Min_{100\%}$	$Max_{25\%}$	$Max_{50\%}$	$Max_{75\%}$	$Max_{100\%}$
ΔX	-10,303	10,69	-92,991	-15,473	-9,743	-3,9245	22,249
ΔY	7,406	8,41	-22,814	2,684	7,361	11,6255	118,986
ΔZ	4,04	5,81	-18,32	0,6185	3,923	7,9545	30,904

Fonte: (O AUTOR, 2023).

Os valores da Tabela 14 apontam que a componente Y obteve máximo de 118,986 metros. As precisões para as coordenadas estimadas são apresentadas na Tabela 15.

Tabela 15: Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	$Min_{100\%}$	$Max_{25\%}$	$Max_{50\%}$	$Max_{75\%}$	$Max_{100\%}$
σ_X	5,613	2,388	0,002	3,969	5,302	7,2135	16,112
σ_Y	5,95	2,683	0,002	4,012	5,49	7,905	21,346
σ_Z	3,332	1,24	0,001	2,421	3,338	4,253	7,593

Fonte: (O AUTOR, 2023).

Os valores da Tabela 15 apontam que a componente Y obteve os piores valores em termos de precisão. As distâncias geodésicas entre as coordenadas estimadas e a coordenada de referência da PRU2 são apresentados na Tabela 16.

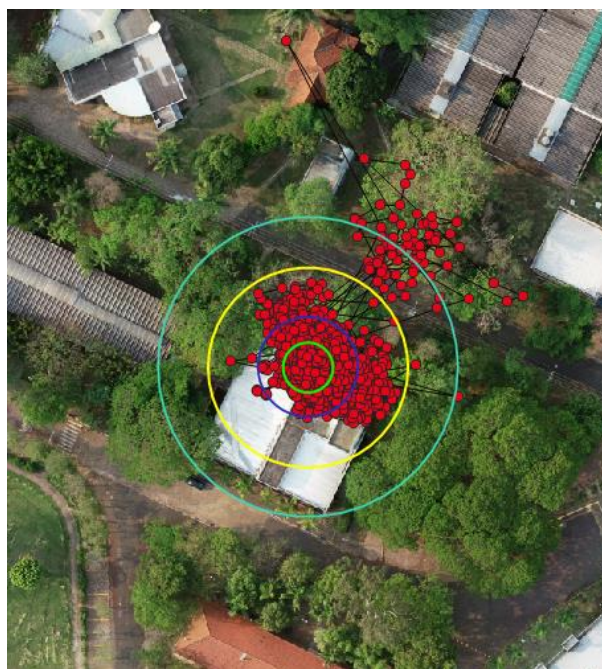
Tabela 16: Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	1511	47,71%
≤ 10 m	2766	87,34%
≤ 20 m	3087	97,47%
≤ 30 m	3118	98,45%
> 30 m	49	1,55%

Fonte: (O AUTOR, 2023).

Os valores da Tabela 16 apontam que 97% das épocas apresentam distância inferior a 20 metros. Na Figura 74, as informações da Tabela 16 são ilustradas a partir da exibição dos pontos equivalentes às coordenadas estimadas para cada época.

Figura 74: Coordenadas estimadas pelo INCT-GNSS-APP Post Processing.



Fonte: (O AUTOR, 2023).

As informações gerais do processamento no INCT-GNSS-APP Post Processing são apresentadas na Tabela 17.

Tabela 17: Informações gerais do processamento no INCT-GNSS-APP Post Processing.

Nº total de épocas:	5400
Nº de épocas processadas:	3167 (58%)
RMSE 2D:	20,630 m
RMSE 3D:	22,099 m

Fonte: (O AUTOR, 2023).

Os valores da Tabela 17 apontam que os valores de RMSE foram ambos superior a 20 metros.

Processamento no RTKPOST

Na Tabela 18 são apresentadas as informações relacionadas às discrepâncias obtidas.

Tabela 18: Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min_{100%}	Max_{25%}	Max_{50%}	Max_{75%}	Max_{100%}
ΔX	-0,807	7,558	-31,888	-5,836	-1,117	3,856	31,350
ΔY	-2,291	6,581	-29,652	-6,062	-2,338	1,372	27,700
ΔZ	-0,486	4,903	-37,208	-3,201	-0,443	2,348	31,356

Fonte: (O AUTOR, 2023).

Os valores da Tabela 18 apontam que as discrepâncias tiveram valor máximo de 37,20 metros. Além disso, a componente Y apresentou o menor valor máximo. Esses resultados são o contrário do que aponta os dados do processamento no INCT-GNSS-APP Post Processing, que apresentou maiores discrepância em Y.

Na Tabela 19 são apresentadas as informações relacionadas à precisão das coordenadas estimadas.

Tabela 19: Precisões em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	5,696	1,560	3,583	4,303	5,310	7,226	9,110
σ_Y	6,177	1,941	4,119	4,322	5,693	8,129	10,573
σ_Z	3,699	0,422	2,846	3,409	3,701	4,009	6,679

Fonte: (O AUTOR, 2023).

Os valores da Tabela 19 apontam que a coordenada Y obteve pior valor de precisão, assim como no processamento executado no INCT-GNSS-APP Post Processing, entretanto, para o processamento no RTKPOST a diferença de precisão entre as três coordenadas foi menor.

Na Tabela 20 são apresentadas as distâncias geodésicas entre as coordenadas estimadas e a coordenada de referência da PRU2.

Tabela 20: Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	3184	59,64%
≤ 10 m	4981	93,29%
≤ 20 m	5323	99,70%
≤ 30 m	5338	99,98%
> 30 m	1	0,02%

Fonte: (O AUTOR, 2023).

Os valores da Tabela 20 apontam que 99% das épocas apresentam distância inferior a 20 m. Esse valor foi superior ao obtido pelo INCT-GNSS-APP Post Processing.

As informações da Tabela 20 são ilustradas na Figura 75, a partir da exibição dos pontos equivalentes às coordenadas estimadas para cada época

Figura 75: Distâncias geodésicas em metros no RTKPOST.



Fonte: (O AUTOR, 2023).

As informações gerais do processamento no RTKPOST são apresentadas na Tabela 21.

Tabela 21: Informações gerais do processamento no RTKPOST.

Nº total de épocas:	5400
Nº de épocas processadas:	5339 (98%)
RMSE 2D:	13,531 m
RMSE 3D:	14,872 m

Fonte: (O AUTOR, 2023).

Os valores da Tabela 21 apontam que ambos os valores de RMSE foram inferiores a 15 metros.

Comparação Huawei P30 Pro: INCT-GNSS-APP Post Processing e RTKPOST

Na Tabela 22 é apresentada a comparação entre o processamento dos dois softwares.

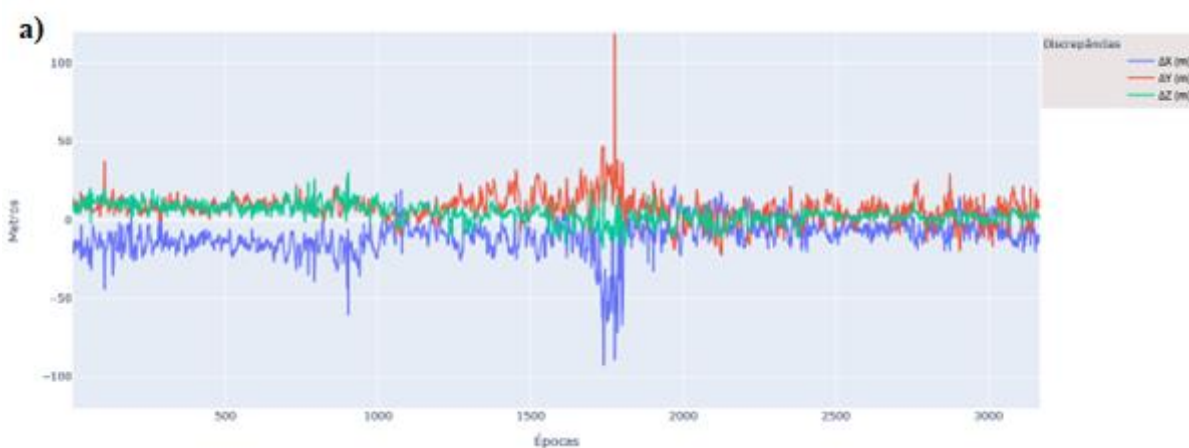
Tabela 22: Huawei P30 Pro - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

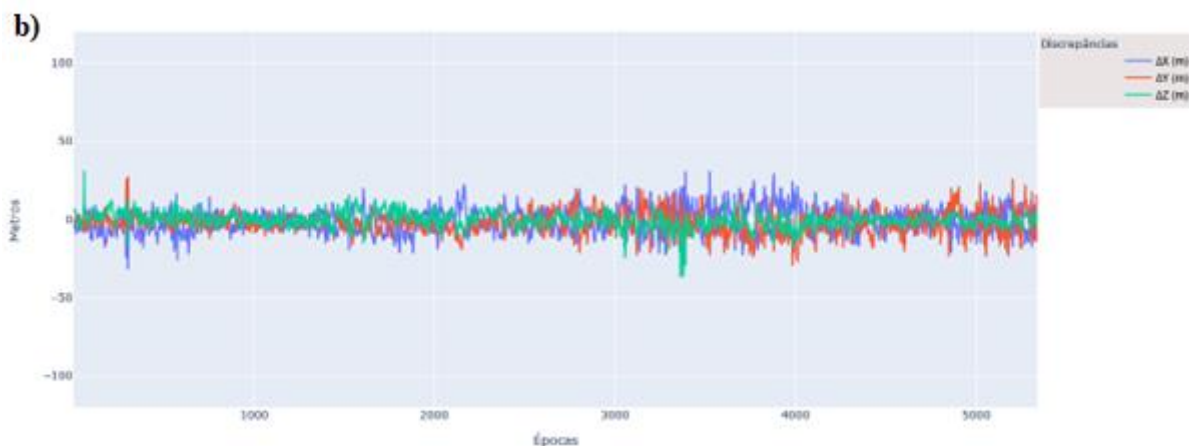
Software:	INCT-GNSS-APP Post Processing	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	3168 (58%)	5339 (98%)
RMSE 2D:	20,631 m	13,531 m
RMSE 3D:	22,100 m	14,872 m

Fonte: (O AUTOR, 2023).

Com o intuito de auxiliar na comparação, além dos dados da Tabela 22, são apresentados na Figura 76 os gráficos de discrepâncias e na Figura 77 os gráficos de precisão correspondentes aos dois softwares.

Figura 76: Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).

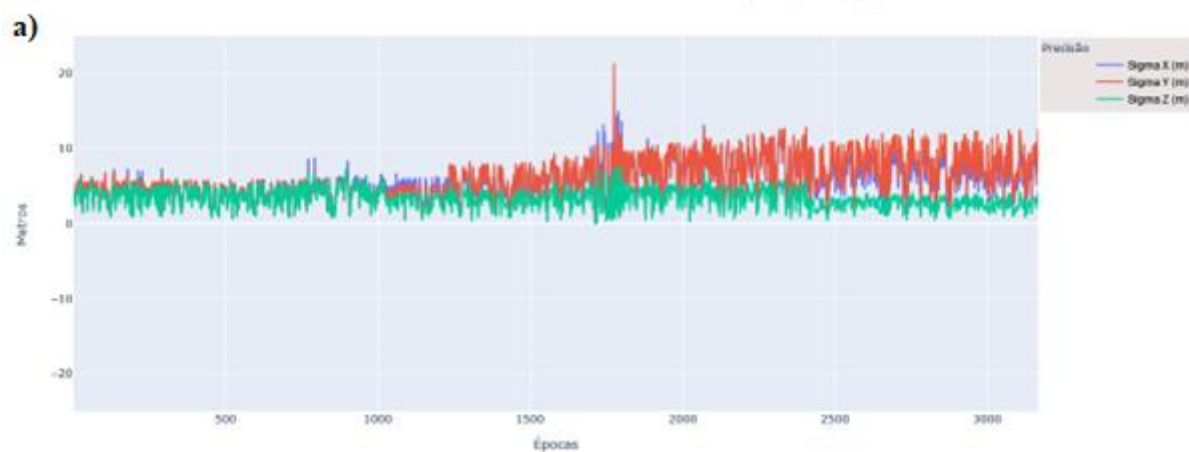


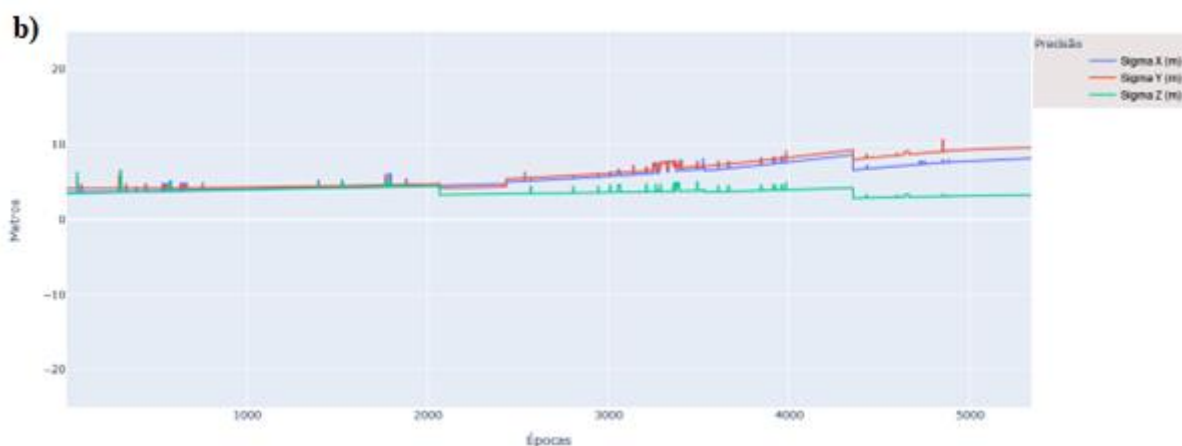


Fonte: (O AUTOR, 2023).

É possível verificar que não houve época com discrepância superior a 50 metros no processamento do RTKPOST, enquanto que para o INCT-GNSS-Post Processing obteve discrepâncias acima dos 50 metros para algumas épocas e uma época com discrepância maior do que 100 metros em Y.

Figura 77: Precisões obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).





Fonte: (O AUTOR, 2023).

É possível perceber que para o processamento do RTKPOST as precisões ficaram com valor de no máximo 10 metros, enquanto que para o INCT-GNSS-APP Post Processing apresentou valores superiores a 10 metros para algumas épocas, sendo que uma época teve precisão superior a 20 metros em Y.

De modo geral, o processamento no RTKPOST obteve melhores resultados.

Smartphone Xiaomi Mi 9

Processamento no INCT-GNSS-APP Post Processing

Na Tabela 23 são apresentados os valores de discrepâncias obtidos no processamento do INCT-GNSS-APP Post Processing.

Tabela 23: Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-8,781	13,897	-97,725	-15,983	-8,200	-0,647	58,981
ΔY	6,850	15,264	-84,553	-1,151	6,187	13,571	110,151
ΔZ	3,051	9,988	-59,681	-2,569	2,950	8,494	59,636

Fonte: (O AUTOR, 2023).

Os valores da Tabela 23 apontam que as componentes X e Y apresentaram valores, em módulo, de 97,72 e 110,15 metros. Já a componente Z obteve valor máximo de 59,68 metros, valor inferior ao das demais. Na Tabela 24 são apresentados os valores de precisão obtidos no processamento do INCT-GNSS-APP Post Processing.

Tabela 24: Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	5,947	2,876	0,115	3,897	5,494	7,729	16,980
σ_Y	6,329	3,321	0,147	4,012	5,654	8,111	20,347
σ_Z	3,713	1,482	0,047	2,712	3,733	4,706	10,099

Fonte: (O AUTOR, 2023).

Os valores da Tabela 24 apontam que os valores de precisão obtidos estão no intervalo de no mínimo 0,047 e no máximo 20,034 metros.

Na Tabela 25 são apresentadas as distâncias geodésicas obtidas no processamento do INCT-GNSS-APP Post Processing.

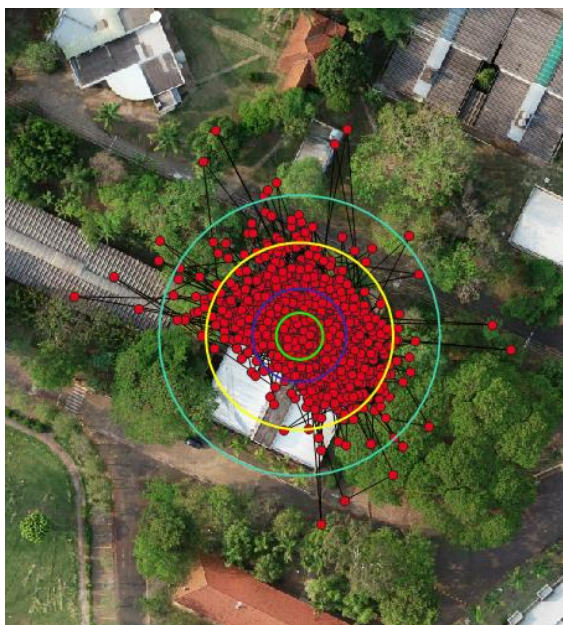
Tabela 25: Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	548	23,91%
≤ 10 m	1529	66,71%
≤ 20 m	2191	95,59%
≤ 30 m	2272	99,13%
> 30 m	20	0,87%

Fonte: (O AUTOR, 2023).

Os valores da Tabela 25 apontam que 95,59% das épocas tiveram distância menor ou igual a 20 metros. Neste caso, menos de 1% das épocas obtiveram distância superior a 20 metros. As informações da Tabela 25 são ilustradas na Figura 78.

Figura 78: Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing.



Fonte: (O AUTOR, 2023).

Tabela 26: Informações gerais do processamento no INCT-GNSS-APP.

Nº total de épocas:	5400
Nº de épocas processadas:	2992 (55%)
RMSE 2D:	25,390 m
RMSE 3D:	27,743 m

Fonte: (O AUTOR, 2023).

Os valores da Tabela 26 apontam que ambos os valores de RMSE foram superiores a 25 metros.

Processamento no RTKPOST

Na Tabela 27 são apresentados os valores de discrepâncias obtidos no processamento do RTKPOST.

Tabela 27: Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	4,658	21,370	-346,829	-3,678	3,781	11,964	330,778
ΔY	-3,913	25,372	-431,993	-10,800	-3,330	3,583	385,034
ΔZ	-4,168	12,433	-121,655	-9,600	-3,828	1,829	191,592

Fonte: (O AUTOR, 2023).

Os valores da Tabela 27 apontam que para as três componentes, a média ficou entre 3 e 5 metros, enquanto que os desvios padrão ficaram no intervalo de 12 a 25 metros. Para as componentes X e Y os valores máximos chegaram a mais de 300 metros, enquanto que na componente Z os valores chegaram a 191,59 metros. As precisões obtidas no processamento do RTKPOST são apresentadas na Tabela 28.

Tabela 28: Precisões em metros RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	6,075	2,099	3,583	4,418	5,718	7,258	32,637
σ_Y	6,698	2,893	4,119	4,452	5,904	8,138	47,928
σ_Z	4,044	1,035	2,847	3,570	3,861	4,273	31,758

Fonte: (O AUTOR, 2023).

Os valores da Tabela 28 apontam que as precisões foram inferiores a 10 metros para 75% das épocas processadas. Entretanto, tiveram valor máximo superior a 30 metros para as componentes X e Z e de 47,92 metros para a componente Y.

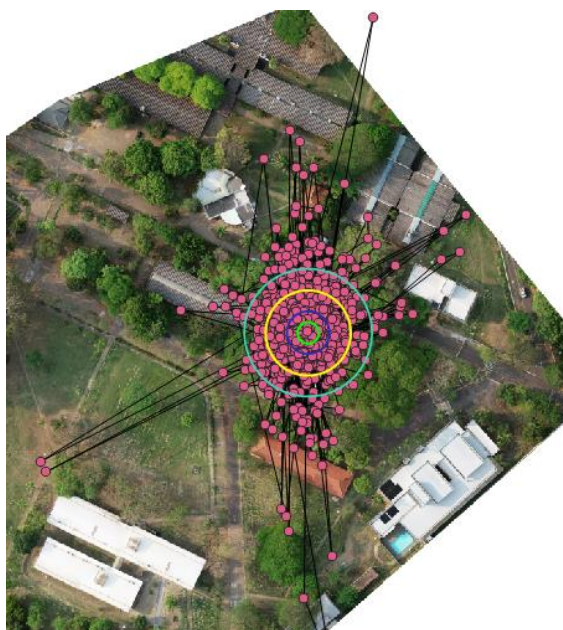
As distâncias geodésicas para as coordenadas estimadas e a coordenadas de referência da PRU2 são apresentadas na Tabela 29.

Tabela 29: Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	1677	32,93%
≤ 10 m	3588	70,46%
≤ 20 m	4728	92,85%
≤ 30 m	4943	97,07%
> 30 m	149	2,93%

Fonte: (O AUTOR, 2023).

Os valores da Tabela 29 apontam que 92,85% das épocas tiveram distância inferior a 20 metros. Os valores da Tabela 29 são ilustrados na Figura 79.

Figura 79: Distâncias geométricas, em metros, no RTKPOST.

Fonte: (O AUTOR, 2023).

As informações gerais do processamento no RTKPOST são apresentadas na Tabela 30.

Tabela 30: Informações gerais do processamento no RTKPOST.

Nº total de épocas:	5400
Nº de épocas processadas:	5092 (94%)
RMSE 2D:	35,096 m
RMSE 3D:	37,697 m

Fonte: (O AUTOR, 2023).

Os valores da Tabela 30 apontam que ambos os valores de RMSE foram superiores a 35 metros.

Comparação Xiaomi Mi 9: INCT-GNSS-APP e RTKPOST

Na Tabela 31 é apresentada a comparação para o processamento dos dois softwares.

Tabela 31: Xiaomi Mi 9 - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

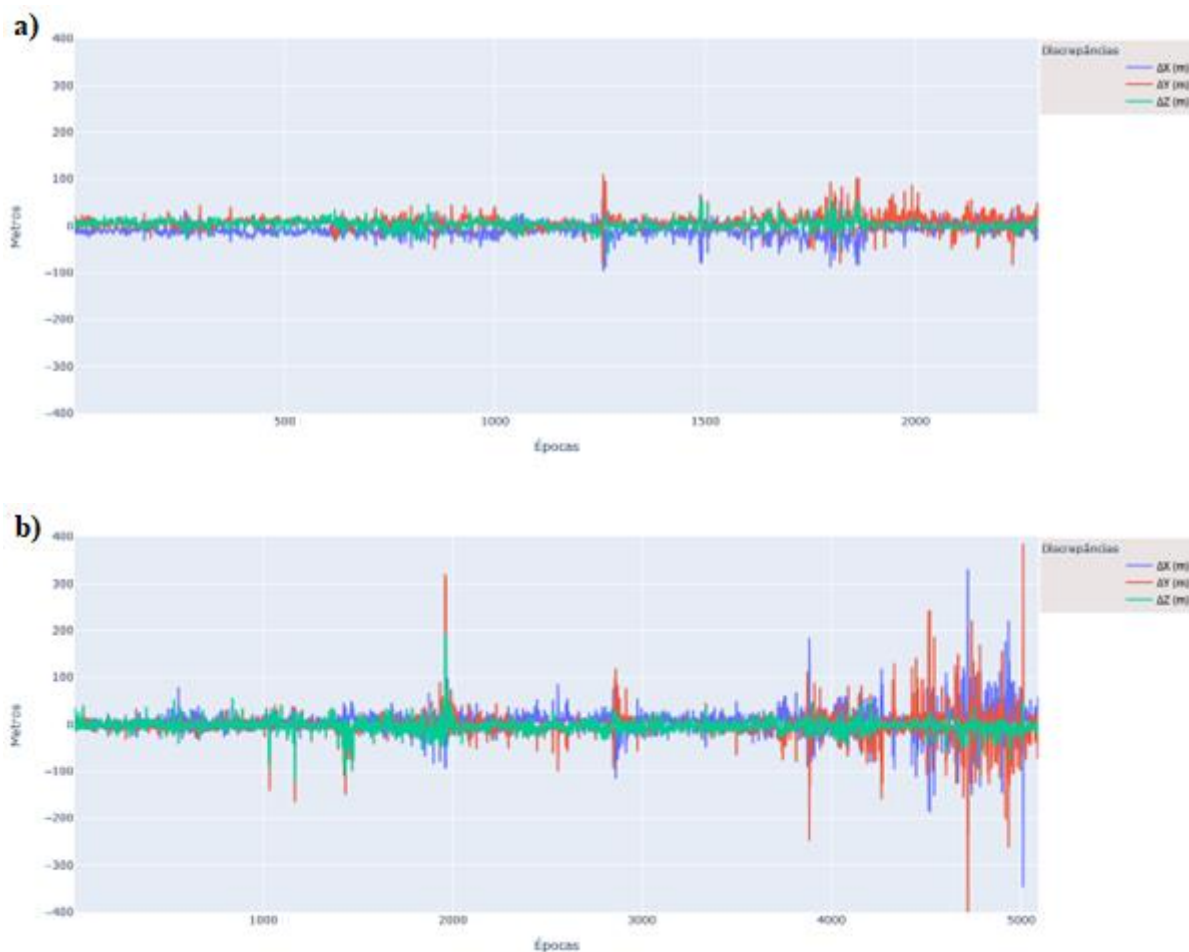
Software:	INCT-GNSS-APP Post Processing	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	2992 (55%)	5092 (94%)
RMSE 2D:	25,390 m	35,096 m
RMSE 3D:	27,743 m	37,697 m

Fonte: (O AUTOR, 2023).

É possível verificar que os valores de RMSE para o RTKPOST foram superiores aos obtidos pelo INCT-GNSS-APP Post Processing, no valor de 10 metros.

Para auxiliar na comparação entre os dois resultados são apresentados nas Figuras 80 e 81 os gráficos de discrepância e precisão.

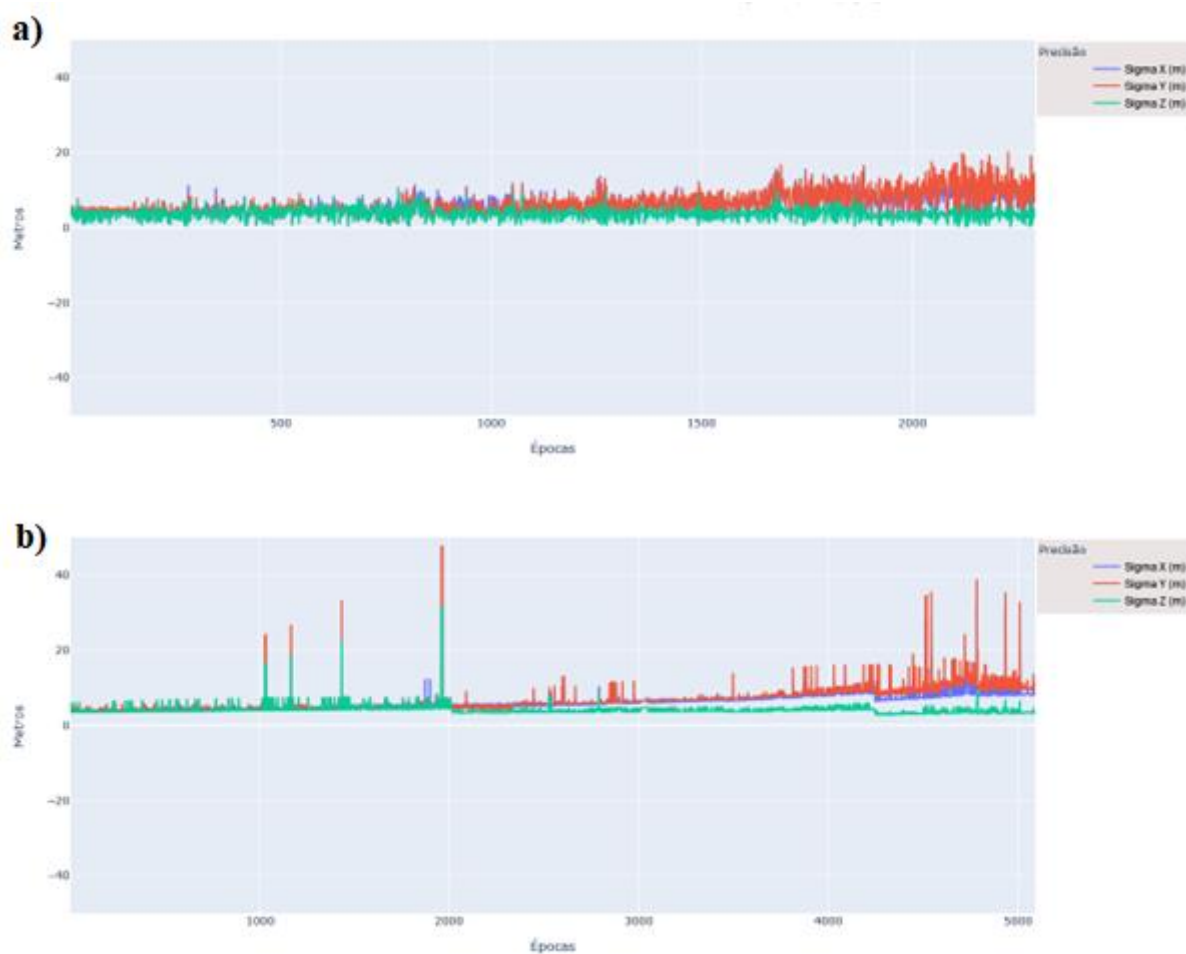
Figura 80: Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).



Fonte: (O AUTOR, 2023).

É possível perceber que em ambos os casos há um aumento expressivo nas discrepâncias das épocas finais, em comparação com as épocas iniciais. Isso é percebido principalmente na componente Y. Neste caso, o INCT-GNSS-APP Post Processing apresentou discrepâncias máximas na componente Y, em torno de 100 metros. Já o RTKPOST apresentou valores superior a 400 metros.

Figura 81: Precisões obtidas pelo INCT-GNSS-APP Post Processing a) e pelo RTKPOST b).



Fonte: (O AUTOR, 2023).

É possível perceber que para ambos os casos, há uma piora da precisão ao longo do tempo, principalmente para a componente Y. O INCT-GNSS-APP Post Processing apresentou valores de no máximo 20 metros para a componente Y, enquanto que o RTKPOST apresentou valores superiores a 30 metros. No processamento do RTKPOST algumas épocas obtiveram altos valores de precisão para Y e Z, em algumas épocas antes da metade do processamento, o que aparentam serem *outliers*.

Comparação para o dia 08/12/2021: Huawei P30 Pro e Xiaomi Mi 9

De modo geral, o smartphone Huawei P30 Pro apresentou melhores resultados. Além disso, dentre os dois softwares utilizados o RTKPOST apresentou os melhores resultados, exceto para o processamento dos dados do smartphone Xiaomi Mi 9.

9.6.2. Dia 09/12/2021

De modo semelhante ao que foi apresentado para o dia 08/12/2021, foram obtidos os valores estatísticos de discrepâncias e precisão para o smartphone Huawei P30 Pro e Xiaomi Mi 9, além de sua visualização no ortofotomosaico e através de gráficos. Tanto os valores para o processamento no INCT-GNSS-APP Post Processing, quanto para o RTKPOST foram calculados. Tais valores podem ser encontrados no Apêndice A.

Comparação para o dia 09/12/2021: Huawei P30 Pro e Xiaomi Mi 9

De modo geral, o smartphone Huawei P30 Pro apresentou melhores resultados. No processamento no INCT-GNSS-APP Post Processing os dados do Huawei P30 Pro resultaram em RMSE com 4 metros a mais do que o processamento no RTKPOST. Já para o Xiaomi Mi 9, o processamento no INCT-GNSS-APP Post Processing apresentou RMSEs de 4 metros a mais do que os RMSEs do Huawei P30 Pro, no mesmo software. Diferentemente dos processamentos com dados do Huawei P30 Pro, com os dados do Xiaomi Mi 9 o RTKPOST apresentou piores resultados ao comparar o processamento com os mesmos dados no INCT-GNSS-APP Post Processing com aumento nos RMSEs em torno de 12 metros a mais nos processamentos do RTKPOST.

9.6.3. Dia 10/12/2021

Assim como foi feito para os dados do dia 09/12/2021, os valores numéricos e Figuras do dia 10/12/2021 encontram-se no Apêndice B.

Comparação para o dia 10/12/2021: Huawei P30 Pro e Xiaomi Mi 9

De modo geral, o smartphone Huawei P30 Pro apresentou melhores resultados. No processamento no INCT-GNSS-APP Post Processing os dados do Huawei P30 Pro resultaram em RMSE com cerca de 4 metros a mais do que o processamento no RTKPOST. Já para o Xiaomi Mi 9, o processamento no INCT-GNSS-APP Post Processing apresentou RMSEs de 4 metros a mais do que os RMSEs do Huawei P30 Pro, no mesmo software. Diferentemente dos processamentos com dados do Huawei P30 Pro, com os dados do Xiaomi Mi 9 o RTKPOST apresentou piores resultados ao comparar o processamento com os mesmos dados no INCT-GNSS-APP Post Processing com aumento nos RMSEs em torno de 10 metros a mais nos processamentos do RTKPOST.

9.6.4. Análise geral do experimento

De modo geral observa-se nos processamentos o aumento ao longo do tempo das discrepâncias e precisões para as coordenadas estimadas. Esse aumento ocorre principalmente na componente Y. Dentre os dois smartphones, o aumento em Y ocorre em maior magnitude para os dados provenientes do Xiaomi Mi 9. Além disso, o RTKPOST sempre processou mais épocas do que INCT-GNS-APP Post Processing. O RTKPOST processou o maior número de épocas no dia 08, processando 5,339 épocas dentre as 5,400 disponíveis para processamento. Já o INC-GNSS-APP Post Processing processou no máximo 3168 épocas, também no dia 08. Esses valores referem-se ao processado com dados do Huawei P30 Pro.

Para o smartphone Huawei P30 Pro, os valores de RMSE 2D e 3D variavam entre 20 e 22 metros nos processamentos do INCT-GNSS-APP Post Processing. Já nos processamentos do RTKPOST, os dados do smartphone Huawei P30 Pro os valores variaram entre 13 e 17 metros.

Já para os smartphone Xiaomi Mi 9, os valores de RMSE 2D e 3D variaram entre 24 e 27 metros nos processamentos do INCT-GNSS-APP Post Processing. Já nos processamentos do RTKPOST, os dados do Xiaomi Mi 9 variaram entre 33 e 40 metros.

Para o processamento no Xiaomi Mi 9, houve épocas com altas discrepâncias como apresentado na Figura 78, para o dia 08. Nesses casos o RTKPOST apresentou valores de RMSE maiores do que aqueles apresentados pelo INCT-GNSS-APP Post Processing. Esse comportamento também ocorreu no dia 09, como pode ser verificado no Apêndice A.

9.7. Experimento 2: RTKPOST aplicando o método PPP

Neste experimento, foram utilizados os mesmos dados coletados para o Experimento 1, apresentado na Seção 8.5. Para esse caso foram utilizados apenas os dados do smartphone Huawei P30 Pro. Isso foi definido porque este é o único entre os dois aparelhos que disponibiliza a observável de fase da onda portadora.

O intervalo de processamento manteve-se o mesmo: de 08/12/2021 a 10/12/2021 no período das 09:00 às 10:30, horário local. Assim sendo, foram processados 1 hora e 30 minutos de dados GNSS brutos, capturados em 3 dias consecutivos, com um smartphone. Os dados são coletados a cada 1 segundo, totalizando 5,400 épocas de dados.

Para o processamento dos dados foi utilizado o método PPP do RTKPOST. Apesar de selecionar o método PPP no software, não é possível definir o processamento executado como PPP. Isso acontece porque os dados processados correspondem apenas as observáveis de fase da onda portadora e pseudodistância na frequência L1. Com isso, não foi possível aplicar combinações *ion-free*. Também não são aplicadas correções em relação ao centro de fase da antena. As efemérides utilizadas são as efemérides precisas finais do IGS e máscara de elevação aplicada foi de 15°. Trata-se portando, de algo como um “PPS aprimorado”. Detalhes sobre a configuração do RTKPOST para esse experimento podem ser encontradas no Anexo B.

Além disso, não foi possível desativar o efeito de *duty cycle*, porque a EMUI 10, interface gráfica do aparelho, não mostra a opção para desativar o *duty cycle* via software, mesmo que o sistema operacional Android possibilite essa alteração, como foi apresentado na subseção 7.1.1.

9.7.1. Resultados do Processamento

Os dados foram processados no RTKPOST e as 5400 épocas de dados, foram resumidos em uma coordenada estimada final, já que a opção *Single* para a configuração *Solution for Static Mode* foi selecionada, e o método de posicionamento aplicado foi o *PPP Static*. Com isso, na Tabela 32 são apresentadas as discrepâncias para as três componentes estimadas.

Tabela 32: Huawei P30 Pro - Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-0,048	0,841	-0,756	-0,513	-0,270	0,306	0,882
ΔY	-2,774	0,683	-3,435	-3,125	-2,816	-2,443	-2,070
ΔZ	-1,229	0,729	-2,026	-1,546	-1,067	-0,831	-0,595

Fonte: (O AUTOR, 2023).

Os valores da Tabela 32 apontam que a componente Y apresentou maior discrepância, com valor máximo de 3,43 metros, em módulo. A componente Z apresentou os segundos maiores valores, com valor máximo de 2,02 metros. Já a componente X teve valores inferiores a 1 metro para todos os percentis analisados. A precisão para as coordenadas estimadas é apresentada na Tabela 33.

Tabela 33: Huawei P30 Pro – Precisão em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	0,046	0,003	0,043	0,044	0,046	0,047	0,049
σ_Y	0,052	0,002	0,050	0,052	0,053	0,053	0,054
σ_Z	0,034	0,001	0,033	0,033	0,034	0,034	0,035

Fonte: (O AUTOR, 2023).

Os valores da Tabela 33 apontam que as três componentes apresentaram precisão decimétrica.

Na Tabela 34 são apresentadas as distâncias geodésicas em relação à coordenada de referência da PRU2.

Tabela 34: Distâncias geodésicas em metros no RTKPOST.

Data	Distância (m)
08/12/2021	1,885
09/12/2021	1,980
10/12/2021	1,595

Fonte: (O AUTOR, 2023).

Os valores da Tabela 34 apontam que todas as distâncias geodésicas foram inferiores a 2 metros. Na Figura 82, os valores da Tabela 34 podem ser visualizados no ortofotomosaico.

Figura 82: Distâncias geométricas, em metros, no RTKPOST.



Fonte: (O AUTOR, 2023).

A partir da análise da Figura 82 é possível perceber que todas as épocas tiveram distâncias inferiores a 5 metros. Com isso, na Figura 83 é apresentado apenas o círculo mais interno, com 5 metros de raio. Também é mostrado o ponto central do círculo, que é a coordenada de referência da PRU2.

Figura 83: Detalhe das distâncias geométricas, em metros, no RTKPOST.



Fonte: (O AUTOR, 2023).

Por fim, os valores de RMSE para o processamento são apresentados na Tabela 35.

Tabela 35: Valores de RMSE para o PPP no RTKPOST.

RMSE 2D:	2,913 m
RMSE 3D:	3,217 m

Fonte: (O AUTOR, 2023).

10. ANÁLISES E DISCUSSÕES

Os resultados obtidos apontam que os dados brutos GNSS podem ser utilizados como receptores GNSS comuns em ambientes sem multicaminho. No experimento 1, utilizando o smartphone Huawei P30 Pro e o método PPS no RTKPOST, foi possível obter valores de RMSE entre 13 e 17 metros, o que está de acordo com o esperado pela literatura.

No experimento 2, ao utilizar os dados do Huawei P30 Pro no método PPP foi possível obter valores de RMSE inferiores a 5 metros. Esse valor não corresponde ao que esperado ao executar o método PPP, como é comumente retratado na literatura. Entretanto, como foram utilizados dados de apenas uma frequência, além da utilização de modelos de correção ionosférica e troposférica, ao invés de utilizar combinações *ion free* ou mapas ionosféricos, este também é um resultado positivo para o projeto, demonstrando boa qualidade ao utilizar métodos de posicionamento GNSS com dados GNSS brutos provenientes de smartphones Android. Além disso, o efeito de *duty cycle*, não pode ser desativado, já que a interface gráfica do aparelho não apresenta essa configuração ao usuário.

Os processamentos executados no RTKPOST, foram todos efetuados utilizando arquivos RINEX criados a partir dos smartphones utilizados nos experimentos, e com isso, para o smartphone Huawei P30 Pro, é possível também afirmar a boa qualidade das medidas capturadas por estes smartphones, que foram utilizadas para obter as observáveis GNSS de pseudodistância e fase da onda portadora e aplicá-las em arquivos RINEX, o que resultou em resultados do PPS compatíveis com o que é descrito na literatura para receptores GNSS tradicionais. Os problemas apresentados no processamento dos dados do Xiaomi Mi 9, apresentaram piores resultados devido a pior qualidade de suas medidas, já que os dados foram coletados sob as mesmas condições e processados nos mesmos softwares.

11. CONCLUSÕES E RECOMENDAÇÕES

A pesquisa realizada proporcionou, além de uma revisão bibliográfica sobre o processo de captura e processamento de dados GNSS brutos provenientes de smartphones com o sistema operacional Android, bem como suas peculiaridades, 1) investigações e análises da aplicação de métodos, já estabelecidos na literatura, como o PPS, PPP, MMQ e Data Snooping, 2) a utilização de recursos disponibilizados pelo Android para o armazenamento e compartilhamento dos dados GNSS brutos, como acesso ao armazenamento interno e externo do aparelho, bem como o acesso à aplicações de computação em nuvem, como o Firebase, 3) aplicativos Android para a coleta, processamento em tempo real, e armazenamento dos dados GNSS brutos, 4) dois softwares multiplataforma para o pós processamento e controle de qualidade dos dados capturados pelos aplicativos para Android desenvolvidos. Produtos desses softwares, como os arquivos GPX e gráficos gerados também puderam ser utilizados para análise exploratória dos resultados e sua posterior inserção no texto desta dissertação, após as validações dos resultados.

Em ambos os experimentos, utilizou-se o ecossistema de softwares desenvolvidos nessa pesquisa e o RTKPOST, software de referência para o processamento de dados GNSS.

A respeito das diferenças obtidas ao processar dados do smartphone Huawei P30 Pro e Xiaomi Mi 9, fica evidente que os melhores resultados foram obtidos ao utilizar o Huawei P30.

Em se tratando da comparação entre o software INCT-GNSS-APP Post Processing e RTKLib, fica evidente que os resultados do RTKPOST ficaram melhores para a maioria dos processamentos executados, com cerca de 5 metros a menos nos valores de RMSE. A exceção para isso foram execuções no RTKPOST utilizando dados do Xiaomi Mi 9.

Com a experiência adquirida durante o desenvolvimento desta pesquisa, a partir dos resultados apresentados e considerando o fato de que há constante aumento nos recursos disponibilizados pelos smartphones Android, no contexto dos dados GNSS brutos, futuros trabalhos podem seguir as seguintes recomendações:

- Utilizar aparelhos cuja interface gráfica possibilite o desligamento do comportamento de *duty cycle*;
- Executar métodos de posicionamento DGNSS e posicionamento relativo;
- Executar os processamentos considerando ambas as frequências L1 e L5;

- Testar as correções para o centro de fase da antena, fornecidas a partir da versão 11 do Android;
- Investigar a configuração interna dos *chipsets* GNSS para poder estimar com precisão o porquê das diferenças entre resultados provenientes de smartphones diferentes, o que provavelmente é resultado da qualidade das medidas de cada aparelho;
- Investigar a integração com a IMU e possivelmente outros sensores dos smartphones; e
- Estimar métricas ionosféricas como o TEC.

REFERÊNCIAS BIBLIOGRÁFICAS

- ANDROID DEVELOPERS. Medições GNSS brutas. Disponível em: <<https://developer.android.com/guide/topics/sensors/gnss?hl=pt-br>>. Acesso em: 6 fev 2023.
- BANVILLE, S.; DIGGELEN, F. V. Precise GNSS for Everyone: Precise Positioning Using Raw GPS Measurements from Android Smartphones. *GPS World*, 2016. pp 42-48.
- BKG. RTCM-3 Message Types. BKG. Disponível em: <<https://igs.bkg.bund.de/ntrip/rtcmmessageatypes>>. Acesso em: 29 Fevereiro 2020.
- CAMERON, A. Google opens up GNSS pseudoranges. Disponível em: <<https://www.gpsworld.com/google-opens-up-gnss-pseudoranges/>>. Acesso em: 15 jun 2017.
- CARNES, B. What is a CSV File and How to Open the CSV File Format. Disponível em: <<https://www.freecodecamp.org/news/what-is-a-csv-file-and-how-to-open-the-csv-file-format/>>. Acesso em: 20 jul 2020.
- CHEM, B.; GAO, C.; LIU, Y.; SUN, P. Real-time Precise Point Positioning with a Xiaomi MI 8 Android Smartphone. *Sensors*, 2019. no. 12: 2835.
- DESENVOLVEDORES ANDROID. Visão geral do armazenamento de dados e arquivos. Desenvolvedores Android. Disponível em: <<https://developer.android.com/guide/topics/data/data-storage?hl=pt-br>>. Acesso em: 28 jul 2020.
- DESENVOLVEDORES GOOGLE. Referência do KML. Disponível em: <<https://developers.google.com/kml/documentation/kmlreference?hl=pt-br>>. Acesso em: 26 jul 2020.
- FABIANO, C. SaaS: o que é, exemplos, benefícios e métricas. Disponível em: <<https://deskmanager.com.br/blog/o-que-e-saas/>>. Acesso em: 29 jul 2020.
- FÁTIMA, N. Uma rápida visão geral dos diferentes tipos de bancos de dados. Astera, 2019. Disponível em: <<https://www.astera.com/pt/type/blog/a-quick-overview-of-different-types-of-databases/>>. Acesso em: 28 jul 2020.
- FIREBASE. Firebase. Disponível em: <<https://firebase.google.com/?hl=pt>>. Acesso em: 8 fev 2023.
- FORTUNATO, M.; RAVANELLI, M.; MAZZONI, A. Real-Time Geophysical Applications with Android GNSS Raw Measurements. *Remote Sensing*, 2019. no. 18: 2113.
- GMS ARENA. Compare specs. Disponível em: <https://www.gsmarena.com/compare.php3?idPhone1=9635&idPhone2=9507#vog-129,*,*>. Acesso em: 6 jan 2023.
- GSA. GNSS mobile apps: using Nougat to access raw GNSS measurements. 2016. Disponível em: <<https://www.euspa.europa.eu/newsroom/news/gnss-mobile-apps-using-nougat-access-raw-gnss-measurements>>. Acesso em: 5 jul 2017.
- GSA. Using Gns Raw Measurements on Android Devices. 2017. Disponível em: <https://www.gsa.europa.eu/system/files/reports/gnss_raw_measurement_web_0.pdf>. Acesso em: 5 jul 2020.
- HAN, K.; LEE, S.; SONG, YOUNG-JIN.; LEE, HAK-BEOM.; PARK, DONG-HYUK.; WON, JONG-HOON. Precise Positioning with Machine Learning based Kalman Filter using GNSS/IMU Measurements from Android Smartphone. *In: ION GNSS+*, 2021. St. Louis, Missouri (USA) e Online. **Anais** [...]. St Louis: Institute of Navigation, 2021.

- KENNEDY, A.; KINGSBURY, R.; COSTER, A.; PANKRATIUS, V.; ERICKSON, P. FAGUNDES, P.; PAULA, E.; CAHOY, K.; VIERINEN, J. Using a smartphone for GNSS ionospheric data. 2016. Disponível em: <<https://www.gpsworld.com/innovation-theres-an-app-for-that/>>. Acesso em: 27 fev 2016.
- KLOBUCHAR, J. A. Ionospheric time-delay algorithm for single-frequency GPS users. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23, 1987. n. 3, pp. 325–331.
- LI, G.; GENG, J. Characteristics of raw multi-GNSS measurement error from Google Android smart devices. *GPS Solutions*, 2019. 23, 90.
- LIMA, E. C. D. *Proposta de Metodologia para melhora do posicionamento obtido através de receptores GPS de baixo custo*. Tese de Doutorado em Ciências, Universidade de São Paulo, São Paulo, 2018, 200 p.
- MACHADO, B. L. et al. Estimativa das coordenadas das estações da rede GNSS. In: XII Colóquio Brasileiro de Ciências Geodésicas, 2022. Curitiba. *Anais* [...]. Curitiba: Universidade Federal do Paraná, 2022.
- MICROSOFT AZURE. O que é IaaS? Disponível em: <<https://azure.microsoft.com/pt-br/overview/what-is-iaas/>>. Acesso em: 6 ago 2020.
- MONICO, J. F. G. *Posicionamento pelo GNSS: descrição, fundamentos e aplicações*. 2ª ed. São Paulo: Ed. da UNESP, 2008. 476 p.
- MONICO, J. F. G. Do GPS e Glonass ao GNSS. *Mundo Geo*. 2013. Disponível em: <<http://mundogeo.com/blog/2013/06/05/do-gps-e-glonass-ao-gnss>>. Acesso em: 1 ago 2017.
- MONICO, J. F. G. *Fundamentos de Geodésica e Levantamentos - Notas de Aula*. Presidente Prudente: Departamento de Cartografia - UNESP, 2019.
- NICOLINI, L.; CAPORALI, A. Investigation on Reference Frames and Time Systems in Multi-GNSS. *Remote Sensing*, 2018. no. 1: 80.
- OMA SPECWORKS. SUPL Configuration Services - Approved Version 1.0. OMA SpecWorks. 2014. Disponível em: <https://www.openmobilealliance.org/release/SUPLCS/V1_0-20140318-A/>. Acesso em: 2 ago 2020.
- OMA SPECWORKS. *OMA SpecWorks. UserPlane Location Protocol - Approved Version 2.0.5*. 2019. Disponível em: <https://www.openmobilealliance.org/release/SUPL/V2_0_5-20191028-A/>. Acesso em: 02 ago 2020.
- OMA SPECWORKS. *OMA SpecWorks FAQ*. OMA SpecWorks. Disponível em: <<https://omaspecworks.org/about/oma-specworks-faq/>>. Acesso em: 2 ago 2020.
- OPENSTREET MAP. *Visualizando trilha 2020_07_23_09_34_Thu_Sue_Horsted_Keynes_Highbrook_Ardingly.gpx*. OpenStreet Map, 2020. Disponível em: <<https://www.openstreetmap.org/user/PeterMapit/traces/3362866>>. Acesso em: 26 jul 2020.
- PEREIRA, V. A. S. *Investigação da usabilidade do GBAS no Brasil*. Tese de Doutorado em Ciências Cartográficas, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Presidente Prudente, 2018, 305p.
- ROBUSTELLI, U.; BAIOCCHI, V.; PUGLIANO, G. Assessment of Dual Frequency GNSS Observations from a Xiaomi Mi 8 Android Smartphone and Positioning Performance Analysis. *Electronic*, 2019, no. 1: 91.
- RTCM. ABOUT RTCM. 2019. Disponível em: <<https://www.rtcn.org/about.html>>. Acesso em: 1 ago 2019.

- SAASTAMOINEM, J. Contribution to the Theory of Atmospheric Refraction. *Bulletin Geodésique*, 1973, vol. 105, set 1972, vol. 106, dez 1972, vol. 107, mar 1973.
- SAM, S. *Types of databases*. 2020. Disponível em: <<https://www.tutorialspoint.com/Types-of-databases>>. Acesso em: 28 jul 2020.
- SEGANTINE, P. C. L. *GPS: Sistema de Posicionamento Global*. São Carlos: EESC/USP, 2005. 381 p.
- SILVA, H. A. D. *Adequação de softwares comerciais às exigências da lei 10.267/2001*. Relatório de Iniciação Científica, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Presidente Prudente, 2007, 58p.
- STATISTA. Android operating system share worldwide by OS version. 2020. Disponível em: <www.statista.com/statistics/271774/share-of-android-platforms-on-mobiledevices-with-android-os>. Acesso em: 7 ago 2020.
- STATISTA. Number of available applications in the Google Play Store from December 2009 to September 2022. 2022. Disponível em: <<https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store>>. Acesso em: 25 jan 2022.
- STATISTA. *Number of smartphone subscriptions worldwide from 2016 to 2021, with forecasts from 2022 to 2027*. 2022. Disponível em: <<https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>>. Acesso em: 25 jan 2023.
- SUZUKI, T. Precise Position Estimation Using Smartphone Raw GNSS Data Based on Two-Step Optimization. *Sensors*, 2023, no. 3: 1205.
- VANI, B. C.; MONICO, J. F. G.; SHIMABUKURO, M. H. Fundamentos e Aspectos Computacionais para Posicionamento por Ponto GPS. *Revista Brasileira de Geomática*, vol. 2, n. 1, 2014. pp. 8-19.
- VANI, B. C.; SHIMABUKURO, M. H.; MONICO, J. F. G. Visual exploration and analysis of ionospheric scintillation monitoring data: The ISMR Query Tool. *Computers & Geosciences*, vol. 104, 2017. pp 125-134.

**APÊNDICE A – Dados do processamento do dia 09/12/2021 no
Experimento 1**

SMARTPHONE HUAWEI P30 PRO

Processamento no INCT-GNSS-APP Post Processing

- Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-5,432	10,667	-128,815	-9,833	-5,152	-0,167	47,678
ΔY	5,300	12,330	-71,321	-0,119	4,512	9,994	163,824
ΔZ	2,134	5,553	-35,643	-0,555	2,147	4,680	43,136

- Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	5,903	2,438	0,057	4,158	5,640	7,540	14,627
σ_Y	6,335	2,935	0,067	4,199	5,809	8,353	22,093
σ_Z	3,422	1,266	0,065	2,534	3,392	4,248	8,931

- Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	1666	59,46%
≤ 10 m	2580	92,08%
≤ 20 m	2736	97,64%
≤ 30 m	2769	98,82%
> 30 m	33	1,18%

- Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing.



- Informações gerais do processamento no INCT-GNSS-APP Post Processing.

Nº total de épocas:	5400
Nº de épocas processadas:	2802 (51%)
RMSE 2D:	20,318 m
RMSE 3D:	21,482 m

Processamento no RTKPOST

- Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-0,159	9,279	-92,876	-5,116	0,100	4,922	56,892
ΔY	-2,320	9,824	-70,313	-7,492	-2,704	2,919	77,251
ΔZ	-0,662	4,883	-22,124	-3,485	-1,016	1,836	47,858

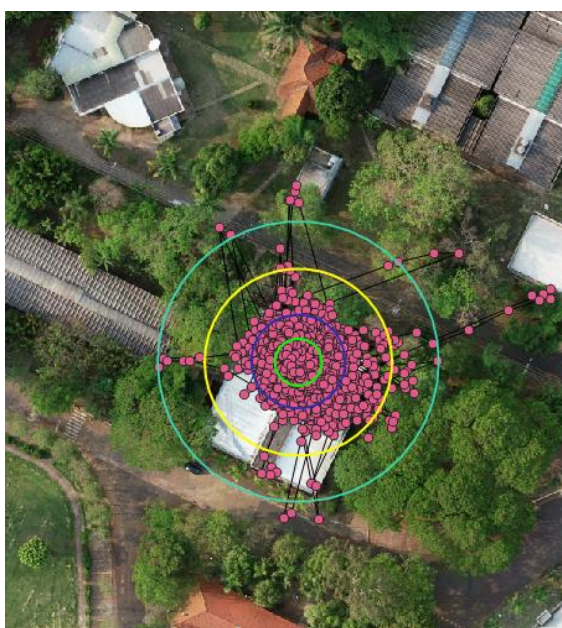
- Precisões em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	6,032	1,726	3,658	4,509	5,664	7,579	12,088
σ_Y	6,593	2,300	4,111	4,376	6,013	8,558	16,253
σ_Z	3,767	0,525	2,981	3,337	3,695	4,101	7,470

- Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
<= 5 m	2888	54,14%
<= 10 m	4894	91,75%
<= 20 m	5271	98,82%
<= 30 m	5313	99,61%
> 30 m	21	0,39%

- Distâncias geodésicas em metros no RTKPOST.



- Informações gerais do processamento no RTKPOST.

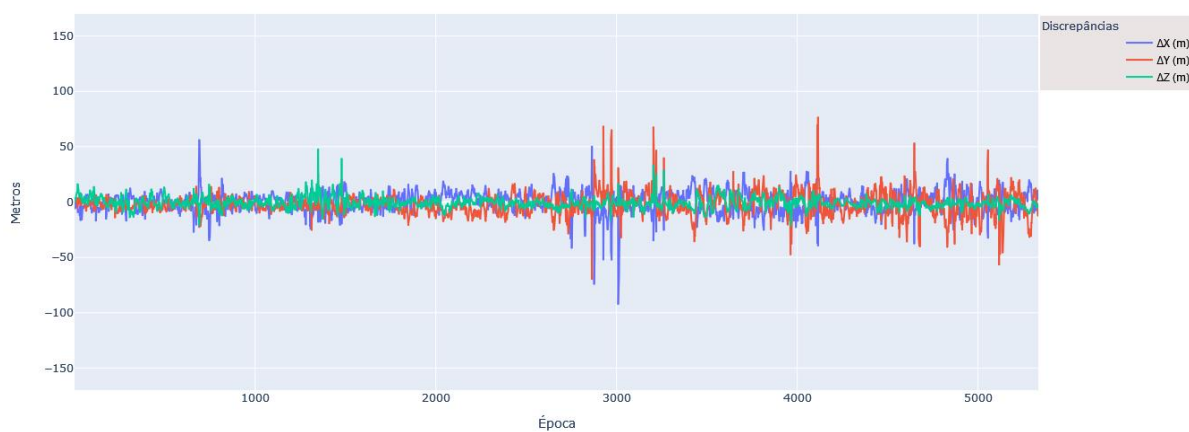
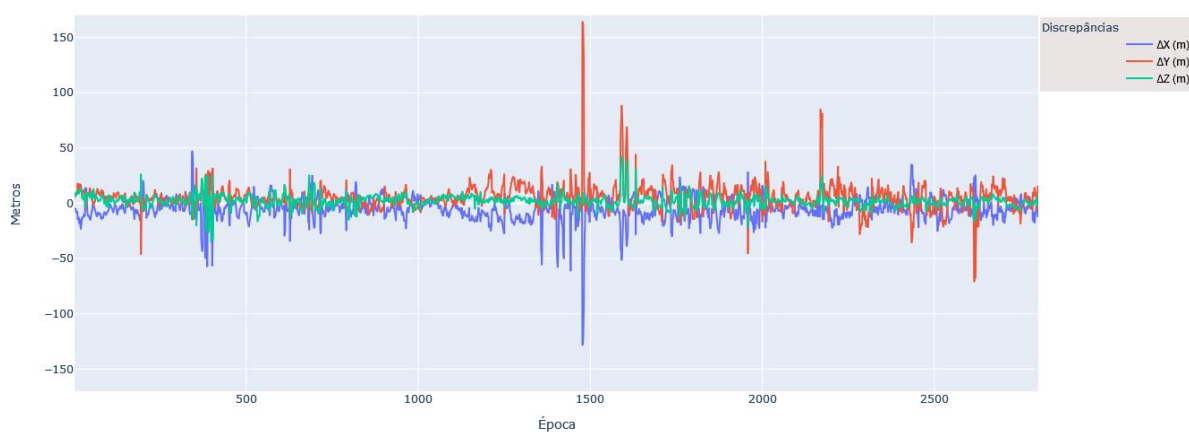
Nº total de épocas:	5400
Nº de épocas processadas:	5334 (98%)
RMSE 2D:	16,616 m
RMSE 3D:	17,743 m

Comparação Huawei P30 Pro: INCT-GNSS-APP Post Processing e RTKPOST

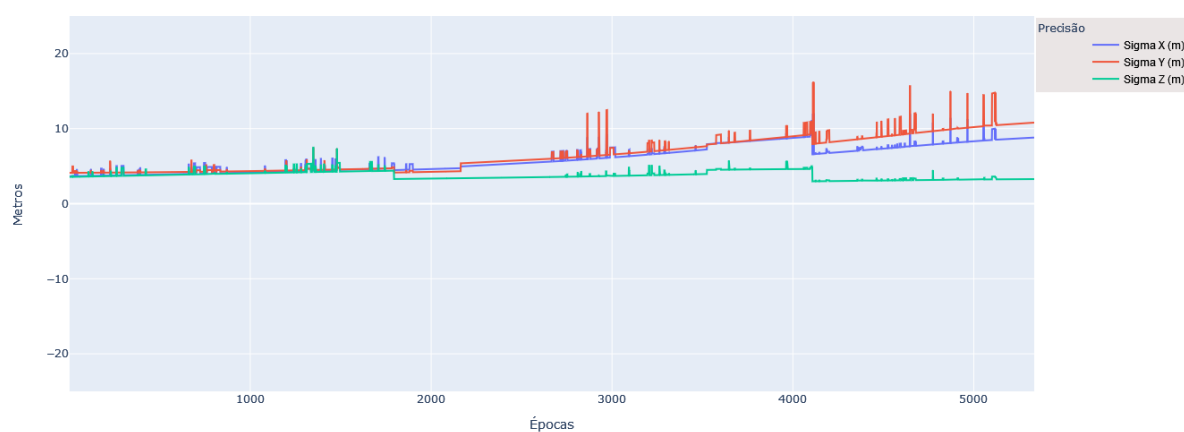
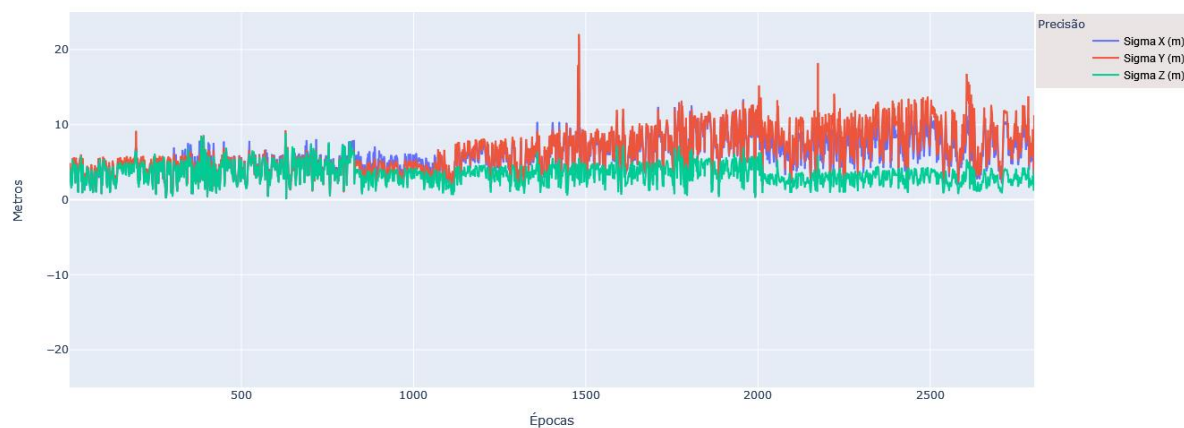
- Huawei P30 Pro - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

Software:	INCT-GNSS-APP	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	2802 (51%)	5334 (98%)
RMSE 2D:	20,318 m	16,616 m
RMSE 3D:	21,482 m	17,743 m

- Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



- Precisões obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



SMARTPHONE XIAOMI MI 9

Processamento no INCT-GNSS-APP Post Processing

- Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-7,745	13,906	-99,158	-14,943	-6,988	0,505	86,595
ΔY	5,676	14,541	-75,902	-2,385	5,078	12,496	117,776
ΔZ	2,642	9,837	-46,770	-3,314	2,267	7,928	55,595

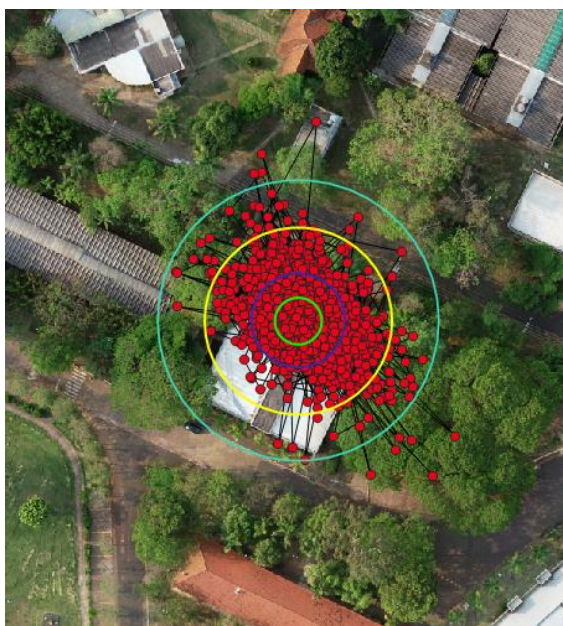
- Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_x	6,138	2,962	0,001	4,091	5,755	7,893	17,992
σ_y	6,558	3,311	0,001	4,317	6,038	8,364	20,306
σ_z	3,741	1,559	0,000	2,658	3,739	4,740	13,054

- Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	631	26,32%
≤ 10 m	1662	69,34%
≤ 20 m	2314	96,54%
≤ 30 m	2387	99,58%
> 30 m	10	0,42%

- Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing.



- Informações gerais do processamento no INCT-GNSS-APP.

Nº total de épocas:	5400
Nº de épocas processadas:	2397 (44%)
RMSE 2D:	24,438 m
RMSE 3D:	26,783 m

Processamento no RTKPOST

- Discrepâncias em metros no RTKPOST

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	5,146	20,445	-147,991	-2,841	4,973	12,905	286,579
ΔY	-4,088	27,435	-363,223	-11,769	-4,302	3,027	476,990
ΔZ	-3,665	15,996	-223,706	-9,887	-3,734	2,250	333,487

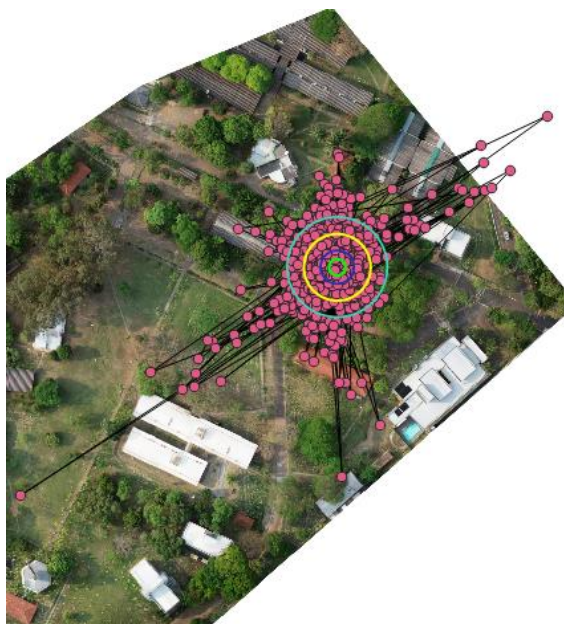
- Precisões em metros RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	6,435	2,332	3,659	4,643	6,042	8,036	32,608
σ_Y	7,269	3,577	4,111	5,029	6,416	8,684	46,806
σ_Z	4,168	1,856	2,981	3,466	3,879	4,426	31,180

- Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	1673	32,08%
≤ 10 m	3583	68,71%
≤ 20 m	4792	91,89%
≤ 30 m	5020	96,26%
> 30 m	195	3,74%

- Distâncias geométricas, em metros, no RTKPOST.



- Informações gerais do processamento no RTKPOST.

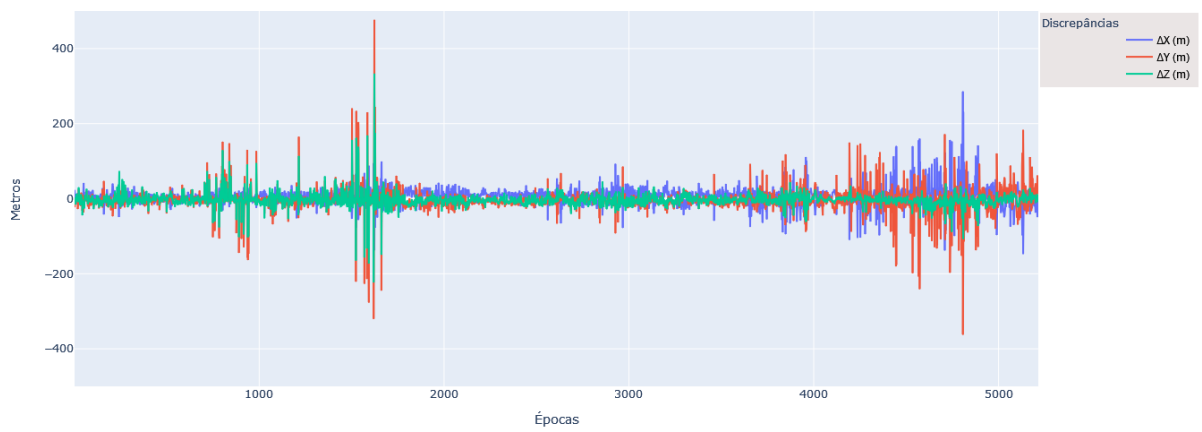
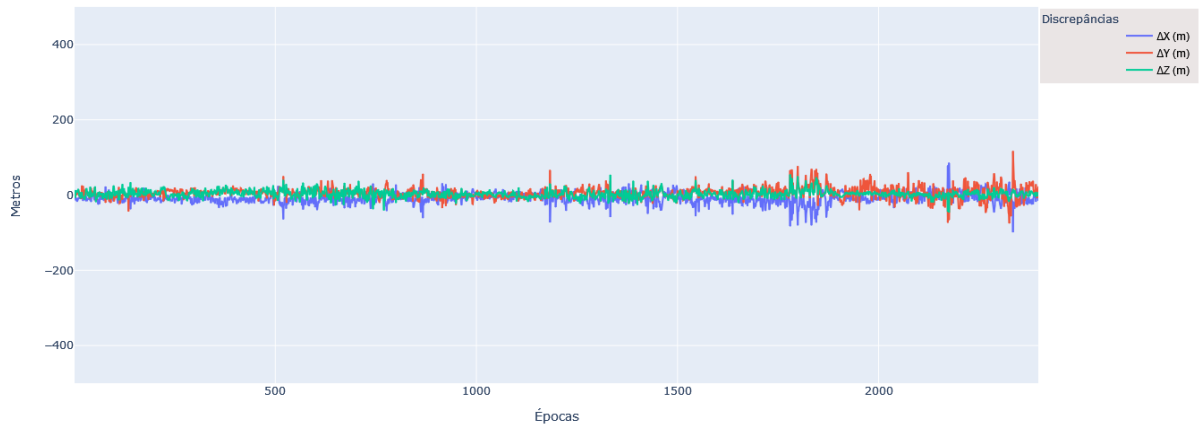
Nº total de épocas:	5400
Nº de épocas processadas:	5215 (96%)
RMSE 2D:	36,415 m
RMSE 3D:	40,201 m

Comparação Xiaomi Mi 9: INCT-GNSS-APP Post Processing e RTKPOST

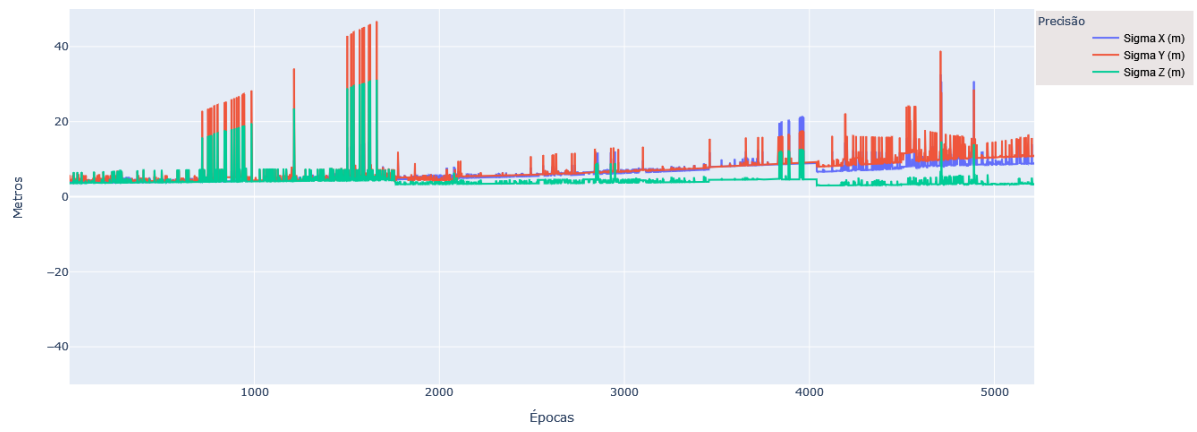
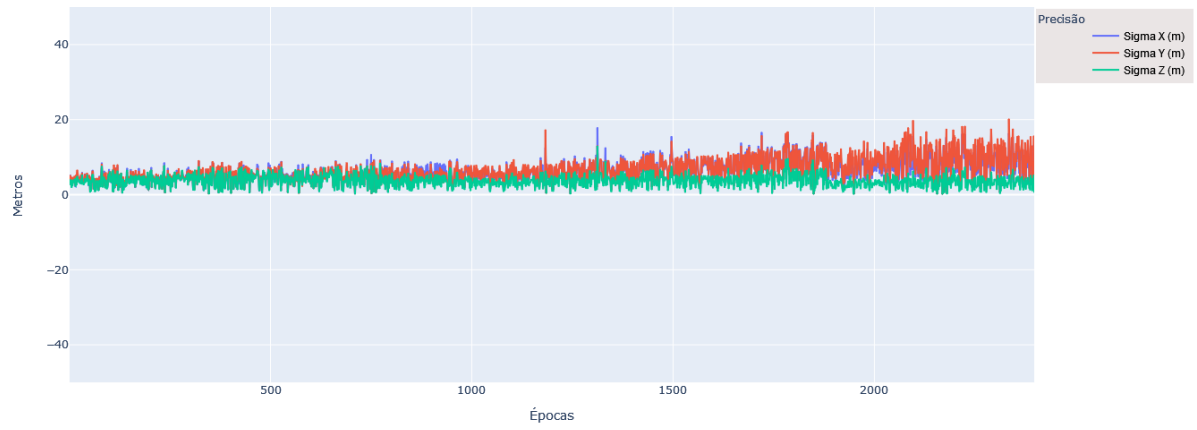
- Xiaomi Mi 9 - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

Software:	INCT-GNSS-APP Post Processing	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	2397 (44%)	5215 (96%)
RMSE 2D:	24,438 m	36,415 m
RMSE 3D:	26,783 m	40,201 m

- Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



- Precisões obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



**APÊNDICE B – Dados do processamento do dia 10/12/2021 no
Experimento 1**

SMARTPHONE HUAWEI P30 PRO

Processamento no INCT-GNSS-APP Post Processing

- Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-6,560	8,914	-71,259	-11,316	-6,230	-1,327	25,037
ΔY	8,665	9,687	-24,711	2,843	8,414	14,026	56,685
ΔZ	5,142	6,161	-16,520	0,706	4,539	8,807	48,095

- Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_x	5,993	2,452	0,379	4,238	5,734	7,627	13,453
σ_y	6,426	2,811	0,391	4,314	5,907	8,437	14,313
σ_z	3,336	1,256	0,235	2,410	3,326	4,196	7,972

- Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	1524	52,34%
≤ 10 m	2642	90,73%
≤ 20 m	2893	99,35%
≤ 30 m	2905	99,76%
> 30 m	7	0,24%

- Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing



- Informações gerais do processamento no INCT-GNSS-APP Post Processing.

Nº total de épocas:	5400
Nº de épocas processadas:	2912 (53%)
RMSE 2D:	19,556 m
RMSE 3D:	21,436 m

Processamento no RTKPOST

- Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	0,288	9,373	-81,665	-4,704	0,394	5,576	47,138
ΔY	-2,163	8,546	-47,817	-6,750	-2,637	1,779	106,372
ΔZ	-1,476	5,295	-34,190	-4,524	-1,847	1,111	37,517

- Precisões em metros no RTKPOST

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	6,428	1,381	4,175	5,149	6,605	7,760	12,022
σ_Y	6,729	1,974	4,272	4,640	6,599	8,647	16,070
σ_Z	3,838	0,549	2,809	3,264	3,923	4,194	7,139

- Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
<= 5 m	3080	57,46%
<= 10 m	4981	92,93%
<= 20 m	5302	98,92%
<= 30 m	5346	99,74%
> 30 m	14	0,26%

- Distâncias geodésicas em metros no RTKPOST;



- Informações gerais do processamento no RTKPOST.

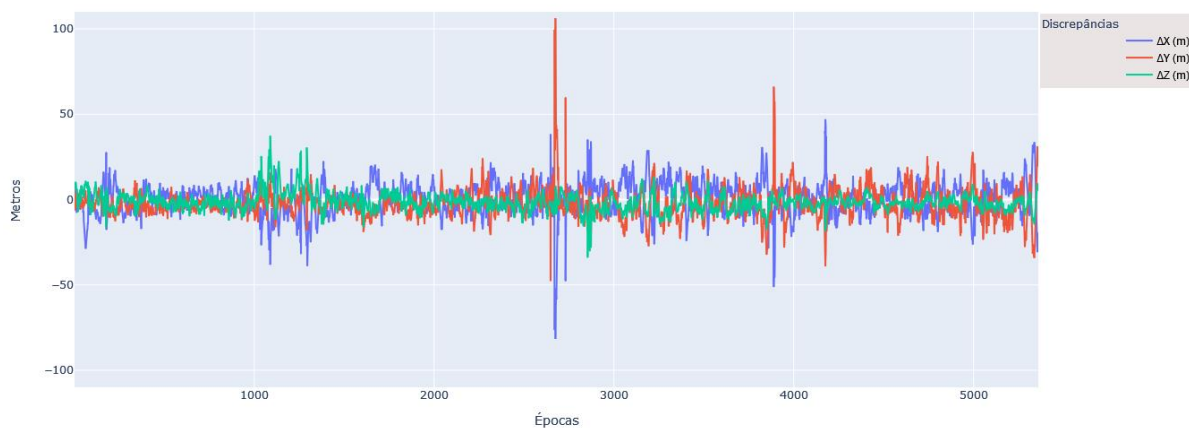
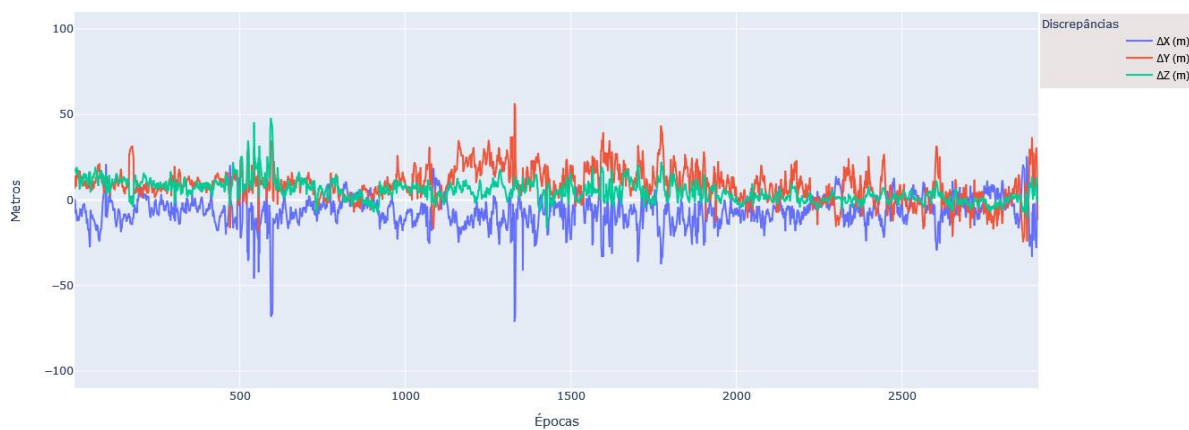
Nº total de épocas:	5400
Nº de épocas processadas:	5360 (99%)
RMSE 2D:	16,062 m
RMSE 3D:	17,413 m

Comparação Huawei P30 Pro: INCT-GNSS-APP Post Processing e RTKPOST

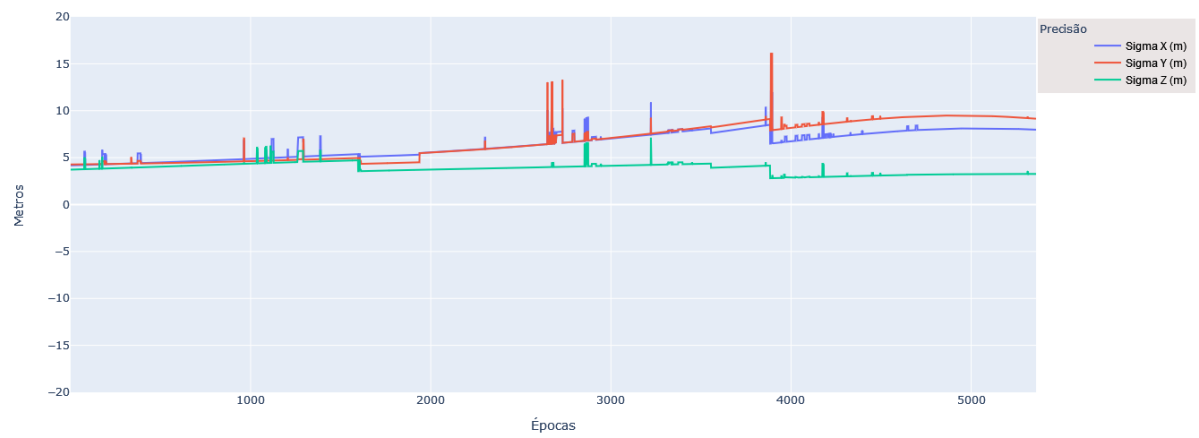
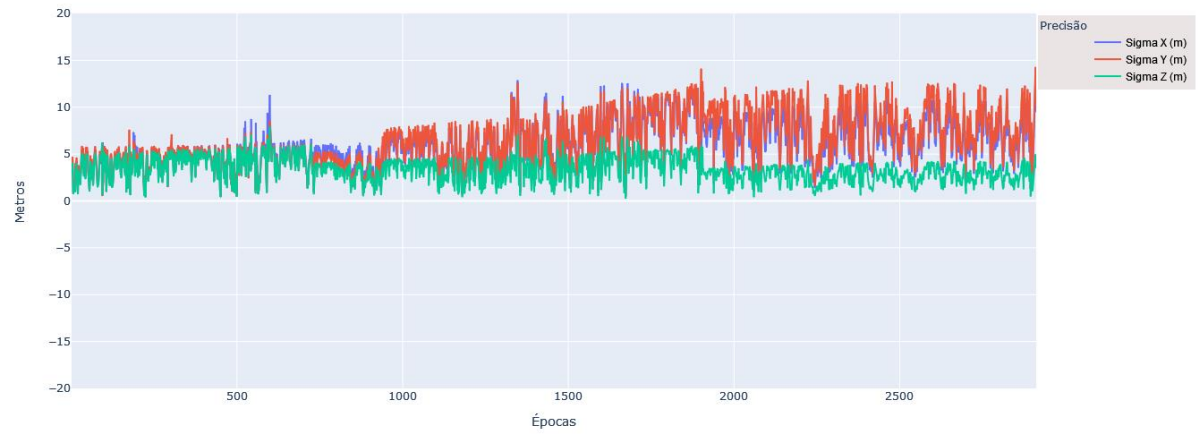
- Huawei P30 Pro - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

Software:	INCT-GNSS-APP Post Processing	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	2912 (53%)	5360 (99%)
RMSE 2D:	19,556 m	16,062 m
RMSE 3D:	21,436 m	17,413 m

- Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



- Precisões obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



SMARTPHONE XIAOMI MI 9

Processamento no INCT-GNSS-APP Post Processing

- Discrepâncias em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	-7,679	13,352	-97,675	-16,138	-8,060	-0,042	71,530
ΔY	5,615	13,795	-79,388	-1,899	6,273	13,131	109,689
ΔZ	2,510	9,413	-38,526	-3,208	2,476	7,912	48,544

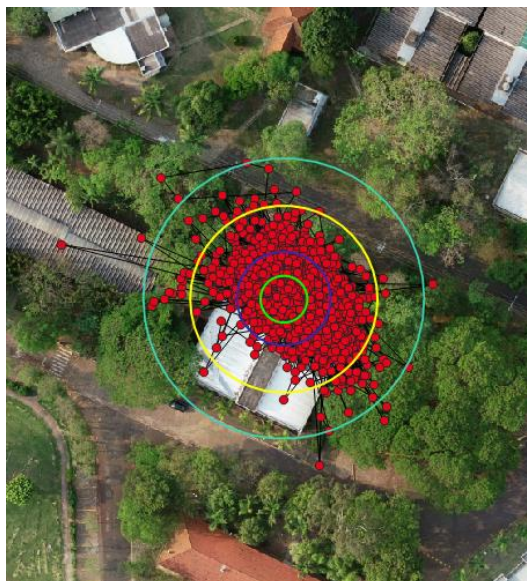
- Precisões em metros no INCT-GNSS-APP Post Processing.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_X	6,727	3,282	0,068	4,306	6,678	8,880	19,472
σ_Y	6,599	3,576	0,050	4,029	6,212	8,775	19,556
σ_Z	3,992	1,918	0,052	2,577	3,945	5,308	13,851

- Distâncias geodésicas em metros no INCT-GNSS-APP Post Processing.

Limiar do erro 2D	Número de épocas	Porcentagem
≤ 5 m	668	25,99%
≤ 10 m	1821	70,86%
≤ 20 m	2495	97,08%
≤ 30 m	2560	99,61%
> 30 m	10	0,39%

- Distâncias geométricas, em metros, no INCT-GNSS-APP Post Processing.



- Informações gerais do processamento no INCT-GNSS-APP.

Nº total de épocas:	5400
Nº de épocas processadas:	2570 (47%)
RMSE 2D:	23,901 m
RMSE 3D:	26,186 m

Processamento no RTKPOST

- Discrepâncias em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
ΔX	5,470	20,223	-134,483	-3,957	4,682	13,217	201,903
ΔY	-4,578	23,165	-252,746	-11,824	-3,743	4,058	176,487
ΔZ	-4,011	12,766	-106,743	-10,096	-3,960	2,177	123,352

- Precisões em metros no RTKPOST.

Coordenada	Média (μ)	Desvio padrão (σ)	Min _{100%}	Max _{25%}	Max _{50%}	Max _{75%}	Max _{100%}
σ_x	7,193	2,054	4,425	5,609	6,948	8,418	30,690
σ_y	7,516	2,748	4,365	5,162	7,133	9,063	28,429
σ_z	4,357	1,194	2,963	3,785	4,234	4,566	14,662

- Distâncias geodésicas em metros no RTKPOST.

Limiar do erro 2D	Número de épocas	Porcentagem
<= 5 m	1750	33,13%
<= 10 m	3731	70,64%
<= 20 m	4880	92,39%
<= 30 m	5132	97,16%
> 30 m	150	2,84%

- Distâncias geométricas, em metros, no RTKPOST.



- Informações gerais do processamento no RTKPOST.

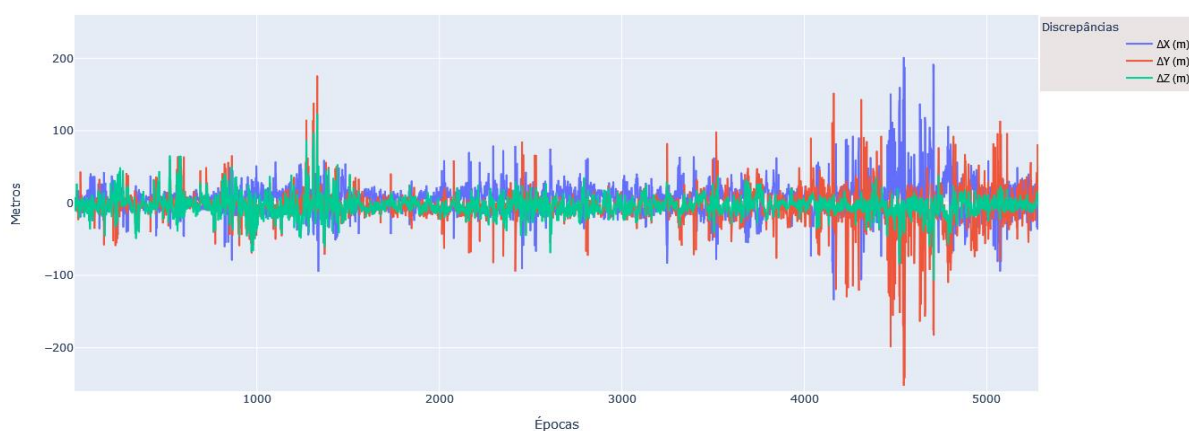
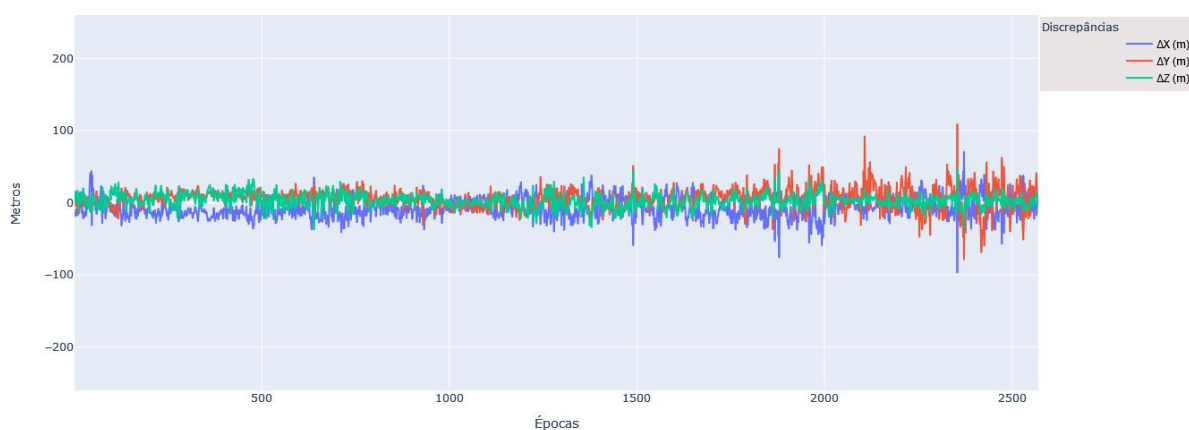
Nº total de épocas:	5400
Nº de épocas processadas:	5282 (97%)
RMSE 2D:	33,411 m
RMSE 3D:	36,272 m

Comparação Xiaomi Mi 9: INCT-GNSS-APP Post Processing e RTKPOST

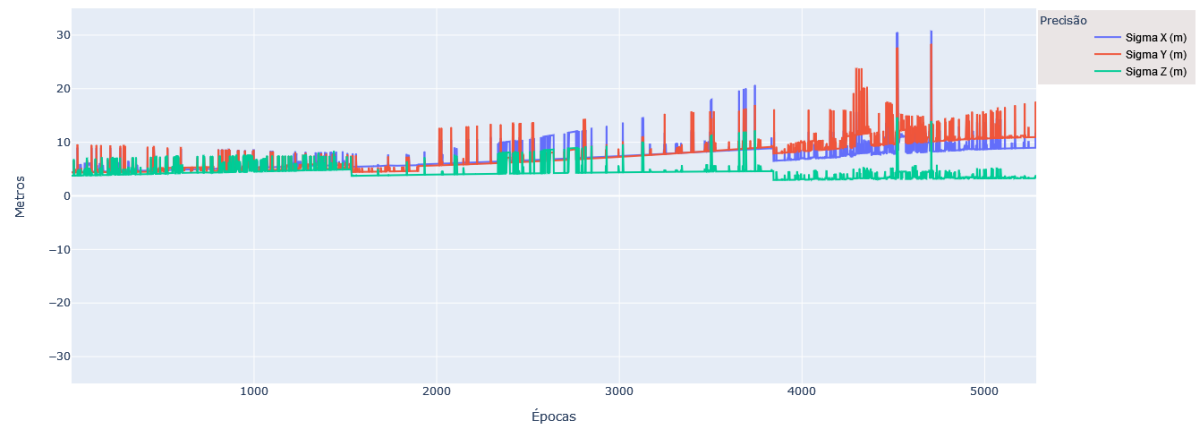
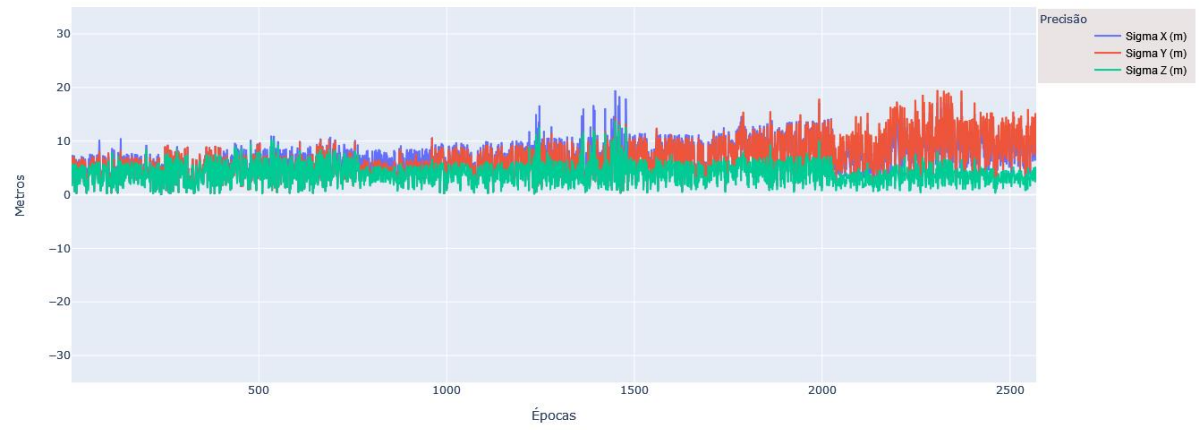
- Xiaomi Mi 9 - Comparação entre INCT-GNSS-APP Post Processing e RTKPOST.

Software:	INCT-GNSS-APP Post Processing	RTKPOST
Nº total de épocas:	5400	5400
Nº de épocas processadas:	2570 (47%)	5282 (97%)
RMSE 2D:	23,901 m	33,411 m
RMSE 3D:	26,186 m	36,272 m

- Discrepâncias obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



- Precisões obtidas pelo INCT-GNSS-APP Post Processing e o RTKPOST.



ANEXO A – Configurações aplicadas no RTKPOST no Experimento 1

```
# rtkpost options (2023/02/10 17:48:18, v.2.4.2)
```

```
pos1-posmode                =single                #
(0:single,1:dgps,2:kinematic,3:static,4:movingbase,5:fixed,6:ppp-kine,7:ppp-static,8:ppp-
fixed)
pos1-frequency              =11                # (1:11,2:11+12,3:11+12+15,4:11+12+15+16,5:11+12+15+16+17)
pos1-soltype                =forward          # (0:forward,1:backward,2:combined)
pos1-elmask                 =15                # (deg)
pos1-snrmask_r              =off              # (0:off,1:on)
pos1-snrmask_b              =off              # (0:off,1:on)
pos1-snrmask_L1             =0,0,0,0,0,0,0,0
pos1-snrmask_L2             =0,0,0,0,0,0,0,0
pos1-snrmask_L5             =0,0,0,0,0,0,0,0
pos1-dynamics               =off              # (0:off,1:on)
pos1-tidecorr               =off              # (0:off,1:on,2:otl)
pos1-ionoopt                =brdc             # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,6:qzs-
brdc,7:qzs-lex,8:vtec_sf,9:vtec_ef,10:gtec)
pos1-tropopt                =saas             # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad)
pos1-sateph                 =brdc             # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
pos1-posopt1                =on               # (0:off,1:on)
pos1-posopt2                =on               # (0:off,1:on)
pos1-posopt3                =on               # (0:off,1:on)
pos1-posopt4                =on               # (0:off,1:on)
pos1-posopt5                =on               # (0:off,1:on)
pos1-exclsats               =                  # (prn ...)
pos1-navsys                 =1                # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode                 =continuous      # (0:off,1:continuous,2:instantaneous,3:fix-and-hold)
pos2-gloarmode              =on               # (0:off,1:on,2:autocal)
pos2-bdsarmode              =on               # (0:off,1:on)
pos2-arthres                =3
pos2-arlockent              =0
pos2-arelmask               =0                # (deg)
pos2-arminfix               =10
pos2-elmaskhold             =0                # (deg)
pos2-aroutcnt               =5
pos2-maxage                 =30              # (s)
pos2-syncsol                =off             # (0:off,1:on)
pos2-slipthres              =0.05            # (m)
pos2-rejionno               =30              # (m)
pos2-rejgdop                =30
pos2-niter                  =1
pos2-baselen                =0                # (m)
pos2-basesig                =0                # (m)
out-solformat               =xyz             # (0:llh,1:xyz,2:enu,3:nmea)
out-outthead                =on               # (0:off,1:on)
out-outopt                  =on               # (0:off,1:on)
out-timesys                 =gpst            # (0:gpst,1:utc,2:jst)
out-timeform                =hms             # (0:tow,1:hms)
out-timendec                =3
```

```

out-degform      =deg      # (0:deg,1:dms)
out-fieldsep     =,
out-height       =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid        =internal # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic    =all      # (0:all,1:single)
out-nmeaintv1   =0        # (s)
out-nmeaintv2   =0        # (s)
out-outstat      =off      # (0:off,1:state,2:residual)
stats-eratio1    =100
stats-eratio2    =100
stats-errphase   =0.003    # (m)
stats-errphaseel =0.003    # (m)
stats-errphasebl =0        # (m/10km)
stats-errdoppler =10      # (Hz)
stats-stdbias    =30      # (m)
stats-stdiono    =0.03     # (m)
stats-stdtrop    =0.3     # (m)
stats-prnaccelh  =10      # (m/s^2)
stats-prnaccelv  =10      # (m/s^2)
stats-prnbias    =0.0001   # (m)
stats-prniono    =0.001    # (m)
stats-prntrop    =0.0001   # (m)
stats-clkstabil  =5e-12    # (s/s)
ant1-postype     =llh      # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant1-pos1        =90      # (deg|m)
ant1-pos2        =0        # (deg|m)
ant1-pos3        =-6335367.6285 # (m|m)
ant1-anttype     =
ant1-antdele     =0        # (m)
ant1-antdeln     =0        # (m)
ant1-antdelu     =0        # (m)
ant2-postype     =rinexhead # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant2-pos1        =0        # (deg|m)
ant2-pos2        =0        # (deg|m)
ant2-pos3        =0        # (m|m)
ant2-anttype     =
ant2-antdele     =0        # (m)
ant2-antdeln     =0        # (m)
ant2-antdelu     =0        # (m)
misc-timeinterp  =off      # (0:off,1:on)
misc-sbasatsel   =0        # (0:all)
misc-rnxopt1     =
misc-rnxopt2     =
file-satantfile  =
file-rcvantfile  =
file-staposfile  =
file-geoidfile   =
file-ionofile    =
file-dcbfile     =

```

file-eopfile =
file-blqfile =
file-tempdir =
file-geexefile =
file-solstatfile =
file-tracefile =

ANEXO B – Configurações aplicadas no RTKPOST no Experimento 2

rtkpost options (2023/02/15 13:12:43, v.2.4.2)

```

pos1-posmode                                =ppp-static      #
(0:single,1:dgps,2:kinematic,3:static,4:movingbase,5:fixed,6:ppp-kine,7:ppp-static,8:ppp-
fixed)
pos1-frequency    =11+12+15 # (1:11,2:11+12,3:11+12+15,4:11+12+15+16,5:11+12+15+16+17)
pos1-soltype      =forward # (0:forward,1:backward,2:combined)
pos1-elmask       =15      # (deg)
pos1-snrmask_r    =on      # (0:off,1:on)
pos1-snrmask_b    =off     # (0:off,1:on)
pos1-snrmask_L1   =25,25,25,25,25,25,25,25,25
pos1-snrmask_L2   =0,0,0,0,0,0,0,0,0
pos1-snrmask_L5   =0,0,0,0,0,0,0,0,0
pos1-dynamics     =on      # (0:off,1:on)
pos1-tidecorr     =otl     # (0:off,1:on,2:otl)
pos1-ionoopt      =brdc    # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,6:qzs-
brdc,7:qzs-lex,8:vtec_sf,9:vtec_ef,10:gtec)
pos1-tropopt      =saas    # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad)
pos1-sateph       =precise # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
pos1-posopt1      =on      # (0:off,1:on)
pos1-posopt2      =on      # (0:off,1:on)
pos1-posopt3      =on      # (0:off,1:on)
pos1-posopt4      =on      # (0:off,1:on)
pos1-posopt5      =on      # (0:off,1:on)
pos1-exclsats     =        # (prn ...)
pos1-navsys       =1       # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode       =continuous # (0:off,1:continuous,2:instantaneous,3:fix-and-hold)
pos2-gloarmode    =on      # (0:off,1:on,2:autocal)
pos2-bdsarmode    =on      # (0:off,1:on)
pos2-arthres      =3
pos2-arlockent    =0
pos2-arelmask     =0       # (deg)

```

```

pos2-arminfix    =10
pos2-elmaskhold  =0      # (deg)
pos2-aroutcnt    =5
pos2-maxage      =30     # (s)
pos2-syncsol     =off    # (0:off,1:on)
pos2-slipthres   =0.05   # (m)
pos2-rejionno    =30     # (m)
pos2-rejgdop     =30
pos2-niter       =1
pos2-baselen     =0      # (m)
pos2-basesig     =0      # (m)
out-solformat    =xyz     # (0:llh,1:xyz,2:enu,3:nmea)
out-outhead      =on     # (0:off,1:on)
out-outopt       =on     # (0:off,1:on)
out-timesys      =gpst   # (0:gpst,1:utc,2:jst)
out-timeform     =hms    # (0:tow,1:hms)
out-timendec     =3
out-degform      =deg    # (0:deg,1:dms)
out-fieldsep     =,
out-height       =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid        =internal # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic    =single  # (0:all,1:single)
out-nmeaintv1    =0      # (s)
out-nmeaintv2    =0      # (s)
out-outstat      =off    # (0:off,1:state,2:residual)
stats-eratio1    =100
stats-eratio2    =100
stats-errphase   =0.003   # (m)
stats-errphaseel =0.003   # (m)
stats-errphasebl =0      # (m/10km)
stats-errdoppler =10     # (Hz)

```

```

stats-stdbias    =30      # (m)
stats-stdiono    =0.03    # (m)
stats-stdtrop    =0.3     # (m)
stats-prnaccelh  =10     # (m/s^2)
stats-prnaccelv  =10     # (m/s^2)
stats-prnbias    =0.0001  # (m)
stats-prniono    =0.001   # (m)
stats-prntrop    =0.0001  # (m)
stats-clkstab    =5e-12   # (s/s)
ant1-postype     =llh     # (0:llh,1:xyz,2:single,3:posfile,4:rinxhead,5:rtcm)
ant1-pos1        =90     # (deg|m)
ant1-pos2        =0      # (deg|m)
ant1-pos3        =-6335367.6285 # (m|m)
ant1-anttype     =
ant1-antdele     =0      # (m)
ant1-antdeln     =0      # (m)
ant1-antdelu     =0      # (m)
ant2-postype     =rinxhead # (0:llh,1:xyz,2:single,3:posfile,4:rinxhead,5:rtcm)
ant2-pos1        =0      # (deg|m)
ant2-pos2        =0      # (deg|m)
ant2-pos3        =0      # (m|m)
ant2-anttype     =
ant2-antdele     =0      # (m)
ant2-antdeln     =0      # (m)
ant2-antdelu     =0      # (m)
misc-timeinterp  =off    # (0:off,1:on)
misc-sbasatsel   =0      # (0:all)
misc-rnxopt1     =
misc-rnxopt2     =
file-satantfile  =
file-rcvantfile  =

```

file-staposfile =
file-geoidfile =
file-ionofile =
file-dcbfile =
file-eopfile =
file-blqfile =
file-tempdir =
file-geexefile =
file-solstatfile =
file-tracefile =