

Pemanfaatan Algoritma Kruskal untuk Perancangan Geometri Pengamatan Baseline Moda Jaringan Survei Statik GPS

Ahmad Hakim Sufyan - 15122118¹
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
15122118@mahasiswa.itb.ac.id ahmad.hakim.sufyan@gmail.com

Abstract—Perencanaan dan persiapan pada survei statik GPS merupakan tahapan paling penting untuk menjamin keberlangsungan proses survei. Dalam proses perencanaan, geometri pengamatan menjadi sangat penting untuk dirancang. Dalam situasi tertentu, perancangan geometri pengamatan akan membutuhkan waktu yang cukup lama sehingga. Makalah ini bertujuan untuk menentukan apakah algoritma Kruskal cocok untuk digunakan dalam perancangan geometri pengamatan berupa jaringan *baseline* pengamatan survei statik GPS. Pembangunan graf *baseline* sendiri dilakukan menggunakan *script* Python dan divisualisasikan menggunakan perangkat lunak basis data PostgreSQL. Titik-titik pengamatan dipilih pada area kawasan ITB Ganesha. Hasil dari makalah ini adalah kecocokan algoritma Kruskal masih belum dapat dipastikan karena mekanisme uji coba dan penilaian yang kurang.

Keywords— Algoritma Kruskal, *baseline*, graf, jaringan, perencanaan

I. PENDAHULUAN

Sebagai salah satu tahapan pelaksanaan survei statik GPS, perencanaan dan persiapan dari survei menjadi langkah yang sangat krusial dalam keberlangsungan kegiatan survei. Tahapan ini meliputi berbagai perancangan baik dari geometri pengamatan, linimasa, personil tim, dan rancangan anggaran biaya. Masing-masing dari rencana tersebut disesuaikan dengan tujuan dari pemberi pekerja. Pada akhirnya, pemberi pekerja akan memilih rencana terbaik untuk keberlangsungan survei statik GPS.

Salah satu perencanaan yang perlu diprioritaskan adalah perancangan terhadap geometri *baseline* yang akan digunakan. *Baseline-baseline* pengamatan sendiri memiliki variasi konfigurasi yang sangat beragam dan masing-masing konfigurasi dapat bermanfaat pada kondisi yang sesuai. Pada umumnya, jumlah *baseline* yang minimal akan lebih diminati oleh pemberi pekerja karena biaya yang dikeluarkan untuk keberlangsungan survei akan relatif lebih murah. Namun, dalam penentuan *baseline* ini terdapat beberapa standar yang harus diperhatikan agar tingkat ketelitian pengukuran koordinat titik-titik pengamatan dapat terjaga

Pada praktiknya, perancangan geometri *baseline* memerlukan waktu yang cukup lama karena pertimbangan yang perlu dilibatkan pun cukup banyak. Pada umumnya, proses

perancangan akan memakan waktu yang lama apabila titik-titik pengamatan pada kegiatan survei berjumlah banyak. Hal ini yang menjadi harapan terbesar dalam penulisan makalah ini yaitu untuk mencari algoritma terbaik untuk merancang geometri *baseline* secara otomatis.

Metode yang akan digunakan untuk perancangan jaringan *baseline* survei statik GPS kali ini adalah dengan memanfaatkan algoritma Kruskal dari teori graf. Teori graf sendiri dipilih karena geometri *baseline* yang terhubung oleh titik memiliki karakteristik yang serupa selayaknya sebuah graf. Pada makalah ini, algoritma Kruskal untuk menentukan *minimum spanning tree* dari graf kombinasi *baseline* yang terbentuk akan dilakukan secara iteratif. Penggunaan algoritma ini secara iteratif akan memastikan bahwa geometri jaringan yang dihasilkan merupakan geometri tertutup.

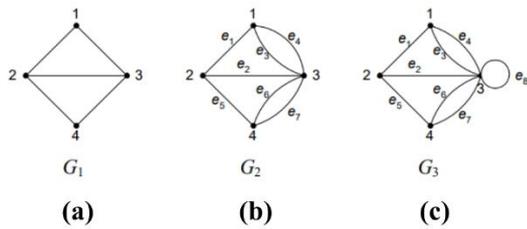
II. STUDI LITERATUR

A. Teori Graf

A.1 Definisi Graf

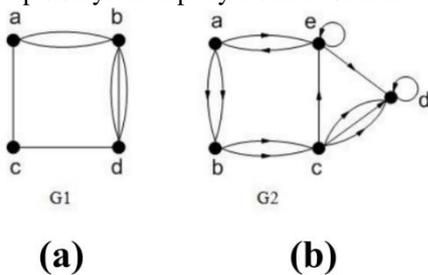
Graf merupakan representasi visual dari objek-objek diskrit beserta hubungan yang dimiliki antar objek tersebut. Secara definisi, apabila diketahui sebuah graf yaitu G , maka $G = (V, E)$ di dalam hal ini V adalah himpunan tidak-kosong dari simpul atau *vertices* yaitu $\{v_1, v_2, \dots, v_n\}$ sedangkan E adalah himpunan sisi atau *edges* yang menghubungkan satu pasang simpul yaitu $\{e_1, e_2, \dots, e_n\}$.

Graf sendiri memiliki beberapa variasi yang diklasifikasikan berdasarkan kriteria tertentu. Berdasarkan ada atau tidaknya gelang dan sisi ganda di dalam graf, maka graf dapat diklasifikasikan menjadi dua macam yaitu graf sederhana (*simple graph*) dan graf tak-sederhana (*unsimple-graph*). Graf sederhana merupakan graf yang tidak memiliki gelang maupun sisi ganda sedangkan graf tak-sederhana merupakan graf yang memiliki gelang maupun sisi ganda. Graf tak-sederhana dapat dibedakan kembali menjadi graf ganda (*multi-graph*) dan graf semu (*pseudo-graph*). Graf ganda merupakan graf yang memiliki sisi ganda sedangkan graf semu merupakan graf yang mengandung sisi gelang.



Gambar 1. Graf sederhana (a), graf ganda (b), dan graf semu (c)
 (sumber : <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf>, diakses pada 7 Januari 2025)

Kriteria selanjutnya adalah berdasarkan orientasi arah pada sisi graf. Kriteria ini dapat dibagi menjadi 2 jenis graf yaitu graf tak-berarah (*undirected graph*) dan graf berarah (*directed graph*). Graf tak-berarah merupakan graf yang sisi-sisinya tidak mempunyai orientasi arah sedangkan graf berarah merupakan graf yang setiap sisinya mempunyai orientasi arah.



Gambar 2. Graf tak-berarah (a) dan graf berarah (b)
 (sumber : <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf>, diakses pada 7 Januari 2025)

A.2 Terminologi Graf

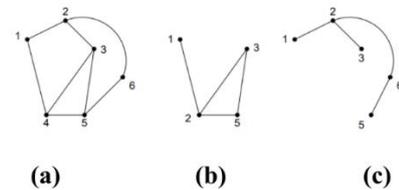
Terdapat beberapa istilah atau terminologi yang penting untuk menjelaskan karakteristik dari graf pada makalah ini. Pertama, ketetanggaan (*adjacent*) merupakan istilah yang menjelaskan bahwa apabila dua buah simpul terhubung langsung oleh suatu sisi, kedua buah simpul tersebut merupakan tetangga dari pasangan simpulnya. Selanjutnya, bersisian (*incidency*) merupakan istilah yang menjelaskan keterhubungan suatu sisi dengan simpulnya. Untuk sembarang sisi $e = (v_j, v_k)$, e bersisian dengan simpul v_j , atau e bersisian dengan simpul v_k .

Selanjutnya, simpul terpercil (*isolated vertex*) merupakan simpul yang tidak mempunyai sisi yang bersisian dengannya. Jika seluruh simpul tidak mempunyai sisi yang bersisian sama sekali atau dengan kata lain suatu graf yang himpunan sisinya adalah himpunan kosong, maka istilah yang tepat untuk menjelaskan karakteristik tersebut adalah graf kosong.

Selanjutnya, derajat (*degree*) merupakan istilah yang menjelaskan terkait jumlah sisi yang bersisian dengan suatu simpul. Terdapat beberapa hal yang dapat diingat terkait derajat dari simpul tertentu seperti derajat simpul terpercil yang selalu bernilai 0 dan derajat simpul anting-anting (*pendant vertex*) yang selalu bernilai 1. Adapun pada kondisi simpul yang bersisian dengan sisi gelang, sisi gelang terhitung selayaknya sisi ganda. Apabila seluruh simpul pada suatu graf memiliki derajat simpul yang sama, maka graf tersebut dapat disebut dengan graf teratur (*regular graph*). Apabila diketahui derajat setiap simpul adalah r pada graf teratur, maka jumlah sisi dari graf tersebut adalah $nr/2$.

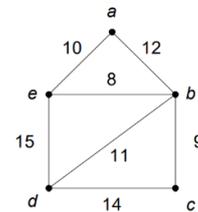
Selanjutnya upagraf (*subgraph*) merupakan istilah yang

menjelaskan “potongan” graf dari suatu graf lainnya. Secara definisi, apabila $G = (V, E)$ adalah sebuah graf, maka $G_1 = (V_1, E_1)$ adalah upagraf dari G jika $V_1 \subseteq V$ dan $E_1 \subseteq E$. Selain itu, komplemen merupakan istilah yang menjelaskan graf yang “terpotong” upagrafnya. Secara definisi, komplemen dari upagraf G_1 terhadap graf G adalah graf $G_2 = (V_2, E_2)$ sedemikian sehingga $E_2 = E - E_1$ dan V_2 adalah himpunan simpul yang anggota-anggota E_2 bersisian dengannya. Sebuah upagraf $G_1 = (V_1, E_1)$ dari graf $G = (V, E)$ dikatakan sebagai upagraf merentang (*spanning subgraph*) apabila $V_1 = V$ atau dengan kata lain upagraf G_1 memiliki semua simpul dari G .



Gambar 3. Graf G (a) yang memiliki upagraf (b) dan komplemen dari upagrafnya (c)
 (sumber : <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf>, diakses pada 7 Januari 2025)

Terakhir, graf berbobot (*weighted graph*) merupakan graf yang setiap sisinya diberi sebuah harga (bobot). Terdapat istilah lain yang seringkali digunakan untuk mendeskripsikan graf ini yaitu graf berlabel [1]. Namun, istilah tersebut kurang cocok digunakan karena secara definisi, label berupa angka tidak hanya diberikan kepada sisi saja, melainkan kepada simpul juga.



Gambar 4. Graf berbobot
 (sumber : <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf>, diakses pada 7 Januari 2025)

A.3 Pohon

Pohon merupakan graf tak-berarah yang terhubung dan tidak mengandung sirkuit. Sirkuit (*circuit*) atau siklus (*cycle*) sendiri merupakan lintasan yang berawal dan berakhir pada simpul yang sama. Graf terhubung juga merupakan karakteristik suatu graf jika untuk setiap pasang simpul v_i dan v_j dalam himpunan V terdapat lintasan dari v_i ke v_j . Apabila kondisi tersebut tidak terpenuhi, maka graf disebut sebagai graf tak-terhubung (*disconnected graph*) dan graf tersebut bukanlah sebuah pohon.

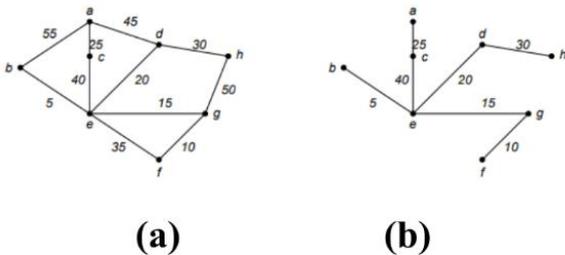
Terdapat beberapa sifat-sifat atau properti dari pohon. Sebuah teorema menyatakan bahwa apabila terdapat suatu graf tak-berarah sederhana $G = (V, E)$ dan jumlah simpulnya n , maka semua pernyataan di bawah ini adalah ekuivalen:

- G adalah pohon.
- Setiap pasang simpul di dalam G terhubung dengan lintasan tunggal.
- G terhubung dan memiliki $m = n - 1$ sisi.
- G tidak mengandung sirkuit dan memiliki $m = n - 1$ buah sisi.

- G tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat satu sirkuit saja.
- G terhubung dan semua sisinya adalah jembatan

A.4 Minimum Spanning Tree

Pohon merentang minimum atau *Minimum Spanning Tree* adalah sebuah jenis pohon merentang (*spanning tree*) yang memiliki jumlah bobot minimum dari suatu graf. Pohon merentang sendiri merupakan upagraf merentang (*spanning subgraph*) dari sebuah graf terhubung. Upagraf merentang tentunya memiliki seluruh simpul dari suatu graf terhubung. Adapun dalam menentukan *minimum spanning tree*, terdapat beberapa algoritma seperti algoritma Prim dan algoritma Kruskal. Apabila terdapat n buah simpul pada suatu graf, maka jumlah sisi dari *minimum spanning tree* adalah $n - 1$.



Gambar 5. graf berbobot (a) beserta minimum spanning tree-nya (b) (sumber : <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/23-Pohon-Bag1-2024.pdf>, diakses pada 7 Januari 2025)

A.5 Algoritma Kruskal

Algoritma Kruskal merupakan salah satu cara untuk menentukan *minimum spanning tree* dari suatu graf terhubung [2]. Algoritma ini bekerja dengan empat langkah yaitu:

1. Urutkan sisi-sisi dari graf dari sisi dengan bobot terkecil hingga bobot terbesar.
2. Siapkan T yang masih kosong.
3. Pilih sisi (u, v) dengan bobot minimum yang tidak membentuk sirkuit di T . Tambahkan (u, v) ke dalam T .
4. Ulangi langkah 3 sebanyak $n - 1$ kali.

B. Survei Statik GPS

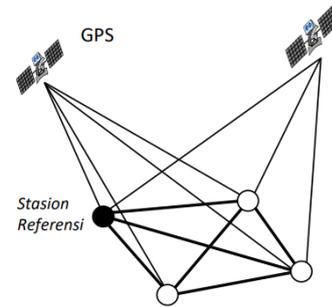
B.1 Definisi Survei Statik GPS

Survei Statik GPS merupakan salah satu metode pengamatan koordinat titik di permukaan bumi dengan bantuan sistem satelit GNSS berupa GPS. Pada umumnya, metode pengamatan dengan sistem satelit bekerja dengan mengukur jarak tempuh sinyal elektromagnetik dari satelit ke alat pengamat (*receiver*). Koordinat titik pengamat kemudian dapat dihitung dengan mempertimbangkan jarak tempuh sinyal beserta koordinat satelit.

Survei Statik GPS sendiri merupakan metode pengamatan statik yang menyebabkan durasi pengamatan pada satu sesi pengamatan umumnya memerlukan waktu yang relatif lebih lama dari metode kinematik. Terlepas lamanya durasi pengamatan, data yang diperoleh akan semakin banyak sehingga metode ini menawarkan tingkat ketelitian yang cukup tinggi.

Selain itu, survei ini pula merupakan metode diferensial yang menyebabkan persamaan pengamatan mengalami pengurangan

(*differencing*) pada beberapa parameter persamaan jarak [3]. Parameter yang dikurangi, baik dieliminasi maupun direduksi, merupakan parameter bias dan kesalahan. Pada umumnya, nilai parameter bias dan kesalahan ini perlu dimodelkan atau diestimasi. Proses penentuan nilai bias dan kesalahan sendiri yang akurat sangat sulit untuk diimplementasikan sehingga pengurangan parameter bias dan kesalahan dapat meningkatkan tingkat ketelitian posisi koordinat titik pengamatan. Metode diferensial juga memerlukan setidaknya dua *receiver* yang bekerja secara simultan untuk melakukan pengamatan pada epok (waktu) yang sama.



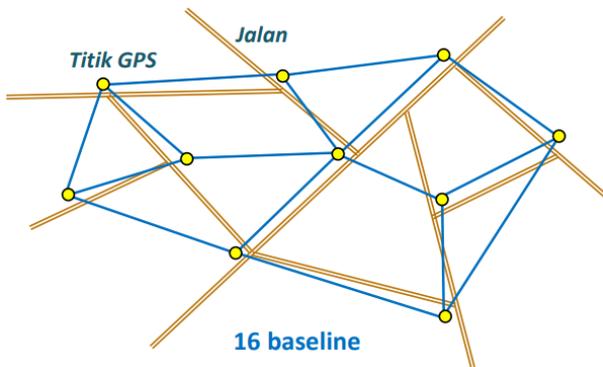
Gambar 6. Ilustrasi sederhana metode pengamatan survei statik GPS (sumber : https://cdn-edunex.itb.ac.id/61908-GNSS-Surveying/296538-Perencanaan-dan-Persiapan-Survei-Statik-GPS-Peralatan-dan-Geometri-Pengamatan/1730588135783_Survei-GNSS-8---Perencanaan-dan-Persiapan-Survei-Statik-GPS---1---2024.pdf, diakses pada 8 Januari 2025)

B.2 Geometri Survei Statik GPS

Sebagai salah satu tahapan pelaksanaan survei statik GPS, tahap perencanaan dan persiapan survei menjadi tahap paling krusial dalam menjamin kualitas keberlangsungan pelaksanaan survei. Salah satu hal yang harus direncanakan dengan baik adalah geometri dari titik-titik pengamatan. Dalam survei ini, penggunaan *baseline* menjadi penting untuk keperluan pengolahan data.

Baseline merupakan garis vektor tiga-dimensi yang menghubungkan sepasang titik pengamatan. Perancangan *baseline* sendiri memiliki variasi yang sangat luas sehingga perancang survei diberi fleksibilitas terkait konfigurasi *baseline* yang diinginkan sesuai dengan keperluan tingkat ketelitian pengamatan. Pada umumnya terdapat beberapa persyaratan yang harus dipenuhi untuk merancang *baseline* yang baik yaitu:

- Jarak *baseline* tidak boleh terlalu jauh (10 km merupakan jarak tipikal yang digunakan).
- Kumpulan *baseline* yang membentuk suatu *loop* tidak boleh terdiri dari empat *baseline*.
- Masing-masing titik harus terhubung oleh setidaknya dua *baseline*, dengan kata lain suatu titik harus terhubung dengan dua titik atau lebih.
- Kumpulan-kumpulan *baseline* sebaiknya membentuk moda jaringan untuk memberikan tingkat kualitas data yang lebih baik dari moda radial.



Gambar 7. Contoh konfigurasi baseline moda jaringan
 (sumber : https://cdn-edunex.itb.ac.id/61908-GNSS-Surveying/296538-Perencanaan-dan-Persiapan-Survei-Statik-GPS-Peralatan-dan-Geometri-Pengamatan/1730588135783_Survei-GNSS-8---Perencanaan-dan-Persiapan-Survei-Statik-GPS---1---2024.pdf, diakses pada 8 Januari 2025)

Adapun detail terkait rancangan *baseline* yang baik dapat dilihat pada SNI 19-6724-2002 yang merupakan standar perencanaan pembuatan jaring kontrol horizontal menggunakan survei statik GPS [4].

Dalam konteks makalah ini, geometri yang dibentuk dari kumpulan *baseline* dan titik pengamatan memiliki kemiripan terhadap karakteristik graf pada teori graf. Titik pengamatan dapat dianggap sebagai simpul sedangkan *baseline* dapat dianggap sebagai sisi dari suatu graf.

III. METODE

A. Persiapan Graf

Pada makalah ini, titik-titik *benchmark* yang tersebar di sekitar kawasan ITB Ganesha akan digunakan sebagai titik pengamatan survei statik GPS. Titik-titik pengamatan ini dipilih sedemikian rupa agar terdistribusi dengan baik di wilayah ITB Ganesha. Adapun nama titik beserta koordinat *easting* dan *northing* dari titik-titik pengamatan adalah sebagai berikut:

- ITB 138;788479.631;9237761.325
- ITB048;788719.123;9237758.280
- ITB 136;788481.584;9237629.600
- ITB010;788333.590;9237760.367
- ITB044;788611.529;9237854.398
- ITB017;788247.142;9237500.949
- ITB037;788258.448;9237321.078
- ITB086;788334.674;9237332.637
- ITB052;788687.386;9237469.146
- ITB026;788473.182;9237412.200
- ITB039;788620.612;9237561.314
- ITB 132;788344.492;9237609.119
- ITB008;788407.108;9237515.512
- GPS3;788610.973;9237323.445
- ITB041;788607.761;9237681.509



Gambar 8. Pesebaran titik-titik pengamatan survei statik GPS
 (sumber : arsip pribadi)

Data titik-titik tersebut kemudian disimpan dalam format .csv dengan delimiter ‘;’ dengan penamaan file “xytitik.txt”. Kali ini, pembuatan graf berupa kemungkinan seluruh *baseline* dari 15 titik pengamatan akan dilalukan dengan perangkat lunak Visual Studio Code menggunakan *script* Python berikut.

```

11 matrix_data = []
12
13 with open("xytitik.txt") as f:
14     for line in f:
15         line_data = line.strip().split(',')
16
17         row = [str(value) for value in line_data]
18
19         matrix_data.append(row)
20
21 listtitik = np.array(matrix_data)
22 ntitik = len(listtitik)
23
24 # TITIK GENERATOR
25 with open("titik.txt", "w") as f:
26     for i in range(ntitik):
27         name = listtitik[i, 0]
28         x = float(listtitik[i, 1])
29         y = float(listtitik[i, 2])
30         f.write("%s;%s;%s\n" % (i + 1, name, x, y))
31
32 # BASELINE GENERATOR
33 id = 0
34 with open("baseline.txt", "w") as f:
35     for i in range(ntitik):
36         x1 = float(listtitik[i, 1])
37         y1 = float(listtitik[i, 2])
38         for j in range(i + 1, ntitik):
39             name = ("%s-%s" % (listtitik[i, 0], listtitik[j, 0]))
40             x2 = float(listtitik[j, 1])
41             y2 = float(listtitik[j, 2])
42             id += 1
43             length = length(x1, x2, y1, y2)
44
45             f.write("%s;%s;%s;%s;%s\n" % (id, name, length, x1, y1, x2, y2))
46
47

```

Gambar 9. Script pembuatan baseline
 (sumber : arsip pribadi)

Perlu digarisbawahi bahwa sistem koordinat dari titik pengamatan ini berada pada sistem koordinat proyeksi UTM Zona 48S. Hal ini menyebabkan perhitungan bobot berupa jarak *baseline* dapat dilakukan dengan persamaan jarak garis miring sederhana. Geometri dari *baseline* berupa *linestring* yang dibuat pun ditulis dalam format *Well Known Text* (WKT) dengan tujuan untuk proses visualisasi graf pada perangkat lunak PostgreSQL.

```

4 # FUNGSI JARAK GARIS MIRING
5 def lengthB(a, b, c, d):
6     deltaX = a - b
7     deltaY = c - d
8     l = math.sqrt(deltaX**2 + deltaY**2)
9     return l
10

```

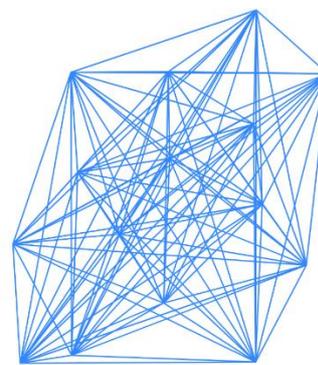
Gambar 10. Script fungsi jarak garis miring untuk perhitungan bobot sisi
 (sumber : arsip pribadi)

Setelah kombinasi dari seluruh *baseline* telah dibuat, *baseline-baseline* tersebut kemudian akan diurutkan dari bobot

terkecil hingga bobot terbesar. Berikut adalah hasil pembuatan seluruh kombinasi *baseline* dari titik-titik pengamatan survei statik GPS beserta visualisasi dari graf-nya menggunakan *geometry viewer* dari perangkat lunak PostgreSQL yang telah terinstalasi ekstensi PostGIS.

Tabel 1. Daftar *baseline-baseline* yang dapat dibentuk

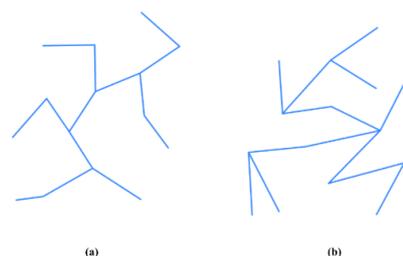
NAMA BASELINE	BOBOT JARAK (m)	NAMA BASELINE	BOBOT JARAK (m)	NAMA BASELINE	BOBOT JARAK (m)
ITB037-ITB086	77	ITB037-ITB008	244	ITB048-ITB008	395
ITB132-ITB008	112	ITB010-ITB008	255	ITB044-ITB008	395
ITB052-ITB039	113	ITB138-ITB008	256	ITB048-ITB132	403
ITB039-ITB041	120	ITB136-ITB044	259	ITB017-ITB041	403
ITB026-ITB008	122	ITB136-ITB052	260	ITB017-GPS3	404
ITB138-ITB136	131	ITB008-ITB041	260	ITB048-ITB026	424
ITB048-ITB041	135	ITB136-ITB017	267	ITB010-ITB086	427
ITB136-ITB008	136	ITB048-ITB136	270	ITB037-ITB039	434
ITB136-ITB041	136	ITB010-ITB017	273	ITB017-ITB052	441
ITB136-ITB132	138	ITB132-ITB041	273	ITB086-ITB041	443
ITB048-ITB044	144	ITB086-ITB132	276	ITB010-ITB037	445
ITB017-ITB132	145	ITB086-GPS3	276	ITB048-GPS3	448
ITB138-ITB010	146	ITB039-ITB132	280	ITB138-ITB086	452
ITB138-ITB041	150	ITB008-GPS3	280	ITB037-ITB052	453
ITB010-ITB132	151	ITB052-ITB008	284	ITB138-GPS3	457
ITB136-ITB039	154	ITB010-ITB041	285	ITB010-ITB052	458
ITB086-ITB026	159	ITB048-ITB052	290	ITB044-ITB026	463
ITB017-ITB008	160	ITB010-ITB044	293	ITB138-ITB037	492
ITB138-ITB044	161	ITB044-ITB039	293	ITB037-ITB041	501
ITB026-GPS3	163	ITB037-ITB132	300	ITB044-ITB017	507
ITB052-GPS3	164	ITB026-ITB041	301	ITB010-GPS3	517
ITB044-ITB041	172	ITB136-ITB086	331	ITB044-GPS3	530
ITB017-ITB037	180	ITB136-GPS3	332	ITB048-ITB017	537
ITB017-ITB086	189	ITB138-ITB017	349	ITB048-ITB086	573
ITB086-ITB008	196	ITB138-ITB026	349	ITB044-ITB086	590
ITB136-ITB010	197	ITB010-ITB039	349	ITB048-ITB037	635
ITB138-ITB132	203	ITB037-GPS3	352	ITB044-ITB037	639
ITB026-ITB039	209	ITB138-ITB052	358		
ITB136-ITB026	217	GPS3-ITB041	358		
ITB039-ITB008	218	ITB044-ITB132	362		
ITB048-ITB039	220	ITB086-ITB039	366		
ITB052-ITB026	221	ITB052-ITB132	370		
ITB052-ITB041	226	ITB010-ITB026	375		
ITB037-ITB026	233	ITB017-ITB039	378		
ITB026-ITB132	235	ITB086-ITB052	378		
ITB039-GPS3	238	ITB136-ITB037	380		
ITB138-ITB048	239	ITB048-ITB010	385		
ITB017-ITB026	242	ITB132-GPS3	390		
ITB138-ITB039	244	ITB044-ITB052	392		



Gambar 11. Visualisasi graf dari kombinasi *baseline* titik-titik pengamatan (sumber : arsip pribadi)

B. Algoritma Kruskal Iteratif

Setelah daftar *baseline* telah tersusud dari bobot terkecil hingga terbesar, algoritma Kruskal dapat digunakan untuk menentukan *minimum spanning tree* dari graf *baseline* tersebut. Mengingat bahwa pohon merentang setidaknya akan menghasilkan beberapa simpul anting-anting (simpul daun), penentuan *minimum spanning tree* akan dilakukan sebanyak dua kali. Setelah menentukan pohon merentang pertama, komplemen dari *minimum spanning tree* terhadap graf *baseline* harus ditentukan untuk menentukan *minimum spanning tree* iterasi ke-dua. Hal ini ditujukan agar sisi graf yang telah digunakan pada penentuan *minimum spanning tree* pertama tidak terpakai lagi sehingga apabila kedua *minimum spanning tree* ini digabung, maka akan menghasilkan geometri jaringan *baseline* tertutup. Detail dari sisi mana saja yang digunakan untuk membentuk kedua *minimum spanning tree* dapat dilihat kembali di Tabel 1 (tabel berwarna jingga untuk hasil iterasi pertama sedangkan tabel berwarna biru untuk hasil iterasi kedua). Berikut merupakan hasil iteratif penentuan *minimum spanning tree*.

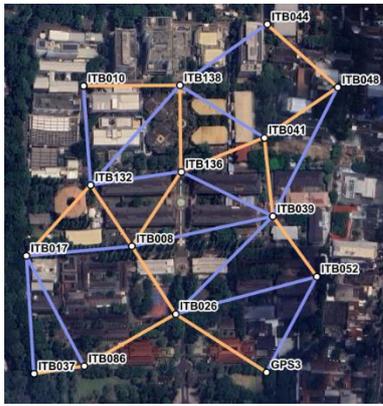


Gambar 12. *Minimum spanning tree* iterasi pertama (a) dan iterasi kedua (b) (sumber : arsip pribadi)

IV. HASIL DAN PEMBAHASAN

A. Hasil

Hasil dari makalah kali ini adalah visualisasi jaringan *baseline* survei statik GPS pada wilayah ITB Ganesha dalam bentuk peta citra satelit. Visualisasi ini ditujukan untuk mempermudah interpretasi lokasi dari rancangan geometri survei. Setelah kedua *minimum spanning tree* telah dibuat, perangkat lunak QGIS digunakan untuk keperluan visualisasi. Basis data *baseline* yang telah dibuat melalui perangkat lunak PostgreSQL dapat dihubungkan dengan perangkat lunak QGIS melalui fitur *Add PostGIS Layers*.



Gambar 13. Hasil jaringan *baseline* yang dibentuk dari dua minimum spanning tree (sumber : arsip pribadi)

B. Pembahasan

Jaringan yang telah dibuat memiliki beberapa kelebihan dan kekurangan. Mengacu pada SNI 19-6724-2002 terkait standar survei statik GPS orde-3, terdapat beberapa hal telah memasuki kualifikasi. Dengan analisis subjektif dari penulis, jarak yang dipilih sudah cukup baik mengingat bahwa beberapa jarak *baseline* yang paling minimum diantara yang lainnya telah digunakan. Salah satu karakteristik dari jaringan tersebut yang telah memasuki kualifikasi adalah jumlah *baseline* yang tidak lebih dari empat *baseline* untuk satu *loop baseline*. Hanya saja, makalah ini tidak memiliki intensi untuk merancang geometri jaringan yang mempertimbangkan aspek *loop baseline* sehingga kelebihan ini bisa saja hanya sebatas keberuntungan penggunaan algoritma.

Dilihat dari kekurangannya, dapat dilihat secara visual bahwa terdapat 2 titik dengan konektivitas lemah (hanya mengikat ke dua titik lainnya saja).

V. KESIMPULAN

Kualitas dari hasil jaringan *baseline* yang dibuat masih belum bisa dinyatakan dengan pasti. Kecocokan dari penggunaan algoritma Kruskal sendiri masih belum bisa dinyatakan karena kurangnya uji coba yang dilakukan dengan kondisi yang beragam. Dengan demikian, perlu adanya penelitian lebih lanjut terkait kecocokan algoritma Kruskal terhadap perancangan geometri jaringan *baseline* survei statik GPS.

VI. SARAN

Perlu adanya mekanisme penilaian terhadap jaringan yang dibuat. Hal ini tentunya menjadi tantangan tersendiri dikarenakan geometri jaringan yang sangat bervariasi menyebabkan parameter penilaian tidak dapat ditetapkan secara umum. Selain kurangnya mekanisme penilaian, karakteristik lainnya selain jarak yang minimum sebaiknya dipertimbangkan. Hal ini ditujukan agar pemilihan algoritma dapat sesuai dengan keperluan dari perancangan jaringan *baseline* survei statik GPS berdasarkan standar yang memang telah ditetapkan dengan mempertimbangkan banyak hal. Terakhir, sebaiknya algoritma yang digunakan telah dibuat dalam bentuk *script*. Hal ini ditujukan agar penggunaan algoritma tertentu dapat diujicobakan di kondisi yang beragam. Dengan banyaknya pertimbangan terhadap uji coba algoritma tertentu akan

memperjelas kecocokan algoritma tersebut terhadap kasus perancangan jaringan *baseline* survei statik GPS.

VII. UCAPAN TERIMA KASIH

Saya ucapkan rasa terima kasih tertinggi saya kepada Tuhan Yang Maha Esa karena telah memberi segala berkahnya dalam proses penulisan makalah ini. Saya juga ingin ucapkan terima kasih kepada dosen pengampu kelas matakuliah Matematika Diskrit saya karena telah senantiasanya mengajar saya yang berasal dari Prodi Teknik Geodesi dan Geomatika. Terakhir, saya sangat berharap agar ilmu yang saya peroleh dari matakuliah Prodi Informatika ini dapat bermanfaat untuk perkembangan ilmu dan teknologi di Prodi Teknik Geodesi dan Geomatika.

DAFTAR PUSTAKA

- [1] R. Munir, Matematika Diskrit, Revisi Keenam, Bandung: Informatika Bandung, 2016.
- [2] J. Kleinberg dan É. Tardos, Algorithm Design, Boston: Pearson Education, Inc, 2006.
- [3] H. Z. Abidin, PENENTUAN POSISI DENGAN GPS DAN APLIKASINYA, Bandung: ITB Press, 2021.
- [4] Badan Standardisasi Nasional, "SNI 19-7149-2005 : Jaring Kontrol Horizontal," [Online]. Available: <https://www.geomasi.com/gallery/SNI%2019-6724-2002%20Pengukuran%20JKHN%20GPS.pdf>. [Diakses 8 Januari 2025].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Januari 2025

Ahmad Hakim Sufyan - 15122118