

E-ISSN 2962 - 6013



**Basis**  
Jurnal ilmiah matematika

VOLUME 2 NOMOR 1  
Maret 2023

PS S1 MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA & ILMU PENGETAHUAN ALAM  
UNIVERSITAS MULAWARMAN

<http://jurnal.fmipa.unmul.ac.id/index.php/Basis>



---

**Penerbit:**

Program Studi S1 Matematika  
Jurusan Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Mulawarman  
Samarinda  
Indonesia

**Maret, 2023**

**Copyright © 2023 by the Authors**

**Website:** <http://jurnal.fmipa.unmul.ac.id/index.php/Basis>

**Email:** [basis-jim@fmipa.unmul.ac.id](mailto:basis-jim@fmipa.unmul.ac.id)



## **SUSUNAN DEWAN REDAKSI**

### **Editor in Chief**

Dr. Syaripuddin, S.Si., M.Si (Universitas Mulawarman)

### **Deputy Editor in Chief**

Qonita Qurrota A'yun, S.Si, M.Sc (Universitas Mulawarman)

### **Editorial Secretary**

Hardina Sandariria, S.Si, M.Sc (Universitas Mulawarman)

### **Editor**

Prof. Nurdin, M.Si. (Universitas Hasanuddin)  
Syaiful Anam, S.Si., M.T., Ph.D. (Universitas Brawijaya)  
Dr. Muhammad Zakir, M.Si. (Universitas Hasanuddin)  
Yuki Novia Nasution, S.Si., M.Sc. (Universitas Mulawarman)  
Wasono, S.Si, M.Si. (Universitas Mulawarman)  
Brilly Maxel Salindeho, S.Si, M.Si. (Universitas Mulawarman)  
Indriasri Raming, S.Si, M.Si. (Universitas Mulawarman)  
Desi Febriani Putri, S.Si, M.Si. (Universitas Mulawarman)  
Andi Kresna Jaya, S.Si., M. Si. (Universitas Hasanuddin)  
Nurchahya Yulian Ashar, S.Si., M.Sc. (Universitas Diponegoro)

### **Web Layouting**

Fidia Deny Tisna Amijaya, S.Si, M.Si (Universitas Mulawarman)  
Sri Wigantono, S.Si, M.Sc (Universitas Mulawarman)

### **Jurnal Layouting**

Moh. Nurul Huda, S.Si, M.Si (Universitas Mulawarman)  
Ryan Rahmad Ramadhan, S.Si (Universitas Mulawarman)

**Reviewer**

Prof. Drs. Dafik, M.Sc., Ph.D. (Universitas Jember)

Dr. M. Ivan Ariful Fathoni, M.Si. (Universitas Nahdhatul Ulama Sunan Giri)

Nur Hamid, Ph.D. (Universitas Nurul Jadid)

Dr. Hadi Hermasyah, S.Si., M.Si. (Politeknik Negeri Balikpapan)

Dr. Rahmat Hidayat, M.Si. (Universitas Cokroaminoto Palopo)

Dr. Muhammad Zakir, M.Si. (Universitas Hasanuddin)

Dr. Ika Hesti Agustin, S.Si., M.Si. (Universitas Jember)

Asmaidi, S.Pd., M.Si. (Universitas Mulawarman)

Deddy Rahmadi, S.Si., M.Sc. (UIN Sunan Kalijaga)

## DAFTAR ISI

Judul Artikel	Penulis	Halaman
<i>Klasifikasi Batubara Berdasarkan Jenis Kalori dengan Menggunakan Perbandingan Jarak Euclid, Jarak Manhattan, dan Jarak Minkowski pada Algoritma Modified K-Nearest Neighbor (Studi Kasus: PT. XYZ Samarinda)</i>	Neni Rahayu, Wasono Wasono, Fidia Deny Tisna Amijaya	1-10
<i>Optimalisasi biaya dan waktu pelaksanaan pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur menggunakan Critical Path Method (CPM) dan Program Evaluation and Review Technique (PERT)</i>	Sendy Clara Aprillia, Wasono Wasono, Qonita Qurrota A'yun	11-24
<i>Penerapan Algoritma Dijkstra dan Algoritma Greedy pada Optimasi Jalur Evakuasi Banjir</i>	Muliya Wiladi, Wasono Wasono, Asmaidi Asmaidi	25-38
<i>Aplikasi Pemrograman Integer pada Masalah Penjadwalan Mata Pelajaran di SMPN 4 Tanah Grogot</i>	Ahmad Topik, Syaripuddin Syaripuddin, Qonita Qurrota A'yun	39-46
<i>Implementasi Algoritma K-Means Clustering dengan Jarak Euclidean dalam Mengelompokkan Daerah Penyebaran COVID-19 di Kabupaten Bogor</i>	Muhammad Iqbal, Syaripuddin Syaripuddin, Moh. Nurul Huda	47-56
<i>Penerapan Aljabar Max-Plus pada Pengaturan Durasi Waktu Lalu Lintas di Simpang Empat Air Putih Samarinda</i>	Leniy Eka Watiy, Syaripuddin Syaripuddin, Qonita Qurrota A'yun	57-65
<i>Penerapan Minimum Spanning Tree dalam Menentukan Rute Terpendek Distribusi Naskah Soal USBN di SMA Negeri se-Sleman</i>	Deddy Rahmadi, Hardina Sandariria	66-71

<b>Judul Artikel</b>	<b>Penulis</b>	<b>Halaman</b>
<i>Hubungan antara Indeks Pembangunan Manusia dengan Garis Kemiskinan di Sumatera Barat, Sumatera Selatan dan Bengkulu</i>	Dea Alvionita Azka, Dia Cahyawati	72-77
<i>The Implementation of Minimum Spanning Tree in Finding Algebraically the Shortest Path of National-Exam-Sheet Distribution in All Senior High Schools over Bantul Regency</i>	Wakhid Fitri Albar, Deddy Rahmadi, Katya Royhana Dewi	78-82

## Klasifikasi Batubara Berdasarkan Jenis Kalori dengan Menggunakan Perbandingan Jarak *Euclid*, Jarak *Manhattan*, dan Jarak *Minkowski* pada Algoritma *Modified K-Nearest Neighbor* (Studi Kasus: PT. XYZ Samarinda)

Neni Rahayu<sup>1</sup>, Wasono<sup>2,\*</sup>, Fidia Deny Tisna Amijaya<sup>3</sup>

<sup>1,2,3</sup> *Laboratorium Matematika Komputasi Jurusan Matematika FMIPA Universitas Mulawarman*

Dikirim: September 2022; Diterima: Maret 2023; Dipublikasi: Maret 2023

Alamat Email Korespondensi: [wasono@fmipa.unmul.ac.id](mailto:wasono@fmipa.unmul.ac.id)

### Abstrak

Metode *Modified K-Nearest Neighbor* (MK-NN) merupakan perbaikan dari metode K-NN yang dapat mengatasi permasalahan *outlier*. Perhitungan nilai akurasi pada MK-NN dengan menambahkan perhitungan nilai fungsi validitas dan *weight voting* digunakan untuk meningkatkan nilai akurasi rendah pada K-NN. Penelitian ini membandingkan tiga jarak yang sering digunakan dalam K-NN, yaitu *Euclid*, *Manhattan* dan *Minkowski*. Penelitian ini bertujuan untuk mengetahui hasil pengklasifikasian batubara berdasarkan jenis kalori dan tingkat akurasi jarak terbaik dari klasifikasi pada metode MK-NN. Data yang digunakan adalah data batubara PT. XYZ Tahun 2017 dengan menggunakan 7 variabel bebas yaitu *Total Moisture* (TM), *Moisture* (M), *Ash*, *Volatile Matter* (VM), *Fixed Carbon* (FC), *Total Sulphur* (TS) dan *Gross Calorific Value* (GCV). Hasil penelitian menunjukkan bahwa dari ketiga jarak yang dibandingkan jarak *Manhattan* memiliki tingkat akurasi yang paling baik yaitu sebesar 100%, sedangkan jarak *Euclid* sebesar 92% dan jarak *Minkowski* sebesar 52%, hasil akurasi terbaik pada MK-NN tersebut diperoleh ketika *K* bernilai 1 pada proporsi 80%:20%.

### Kata Kunci:

Batubara, *Euclid*, *Manhattan*, *Minkowski*, *Modified K-Nearest Neighbor*

### PENDAHULUAN

*Data mining* merupakan proses yang mempekerjakan satu atau lebih teknik pembelajaran komputer (*machine learning*) untuk menganalisis dan mengekstraksi pengetahuan (*knowledge*) secara otomatis. *Data mining* berisi pencarian pola yang diinginkan dalam *database* besar untuk membantu pengambilan keputusan di waktu yang akan datang [1]. Klasifikasi adalah salah satu tugas dari *data mining* yang bertujuan untuk memprediksi label kategori benda yang tidak diketahui sebelumnya, dalam membedakan antara objek yang satu dengan yang lainnya berdasarkan atribut atau variabel.

Klasifikasi merupakan penggolongan atau pengelompokan fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk memperkirakan kelas dari suatu objek yang labelnya belum diketahui atau pembagian sesuatu menurut kelas-kelasnya. Metode *K-Nearest Neighbor* (K-NN) melakukan

klasifikasi terhadap objek berdasarkan data latih yang menggunakan jarak terdekat atau kemiripan terhadap objek tersebut. Pada fase pembelajaran, algoritma ini melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran [2]. Metode *Modified K- Nearest Neighbor* (MK-NN) merupakan perbaikan dari metode K-NN yang dapat mengatasi permasalahan *outlier* pada saat perhitungan nilai bobot pada K-NN dengan menambahkan perhitungan nilai validitas. Tujuan dari algoritma MK-NN adalah untuk meningkatkan akurasi dari K-NN, dengan menambahkan fungsi Validitas dan *Weight Voting* [3].

Batubara merupakan bahan galian strategis dan menempati posisi yang sangat penting dalam pembangunan nasional, maka posisi batubara sebagai bahan bakar alternatif yang sangat diharapkan dapat mengantisipasi krisis energi dengan meningkatkan pemanfaatannya untuk keperluan domestik sebagai bahan bakar pada pembangkit tenaga listrik, industri maupun untuk kepentingan ekspor [4].

Berdasarkan latar belakang di atas, penelitian ini mengklasifikasikan batubara dengan mengambil studi kasus klasifikasi batubara di PT XYZ Samarinda yang mana sebelumnya penelitian serupa telah dilakukan oleh [5] menggunakan metode *Genetic Modified K- Nearest Neighbor*. Pengembangan metode dilakukan penulis dengan menambahkan 3 jarak sehingga didapatkan judul "Klasifikasi Batubara Berdasarkan Jenis Kalori dengan Menggunakan Perbandingan Jarak *Euclid*, Jarak *Manhattan*, dan Jarak *Minkowski* pada Algoritma *Modified K-Nearest Neighbor* (Studi Kasus: PT. XYZ Samarinda)".

## **METODE PENELITIAN**

Pada penelitian ini, populasi yang digunakan yaitu seluruh data hasil pemeriksaan laboratorium batubara pada PT. XYZ Samarinda tahun 2017, sedangkan sampel pada penelitian ini adalah 60 data hasil pemeriksaan laboratorium batubara yang setiap klasifikasinya masing-masing memiliki 20 data di PT. XYZ Samarinda tahun 2017. Teknik sampling yang digunakan adalah *Total Sampling* atau *Sampling Jenuh*. Variabel pada penelitian ini adalah nomor sampel batubara, hasil pemeriksaan laboratorium batubara PT. XYZ Samarinda, dan klasifikasi batubara berdasarkan jenis kalori.

Adapun langkah-langkah analisis algoritma MK-NN sebagai berikut :

1. Analisis statistika deskriptif
2. Normalisasi data
3. Pengacakan data
4. Pembagian data *training* dan data *testing*
5. Perhitungan jarak *Euclid*, *Manhattan* dan *Minkowski*

## **HASIL DAN PEMBAHASAN**

1. Analisis Statistika Deskriptif

Pada analisis deskriptif ini, akan dilakukan gambaran umum dari hasil pengumpulan data batubara berdasarkan jenis kalori di PT. XYZ Samarinda. Berikut adalah hasil analisis deskriptif yang dapat dilihat pada Tabel 1.

Tabel 1. Perbandingan Statistika Deskriptif

Kategori		TM (%)	M (%)	ASH (%)	VM (%)	FC (%)	TS (%)	GCV (Kcal/Kg)
Lignite	Rata-Rata	34,784	18,962	13,775	35,859	31,455	0,908	4442
	Variansi	107,958	27,775	54,032	4,963	2,594	1,749	15968,316
	Nilai Minimum	10,39	5,45	5,27	30,09	28,7	0,15	4178
	Nilai Maksimum	51,99	25,57	33,94	39,03	34,06	5,7	4598
Sub-Bituminous	Rata-Rata	26,356	14,694	5,162	40,711	39,435	0,818	5525,35
	Variansi	1,271	1,605	3,833	1,019	2,644	0,155	23297,713
	Nilai Minimum	24,37	12,59	3,31	38,59	35,42	0,44	5235
	Nilai Maksimum	28,98	17,06	10,73	42,31	42,24	2,06	5817
Bituminous	Rata-Rata	16,773	10,804	3,355	40,455	45,389	0,667	6303,55
	Variansi	20,755	5,467	2,667	3,340	3,894	0,621	86895,839
	Nilai Minimum	6,63	5,69	0,75	37,54	41,39	0,16	5974
	Nilai Maksimum	21,9	13,34	7,75	43,78	49,27	2,87	6974

Berdasarkan Tabel 1 dapat dilihat bahwa nilai rata-rata dan variansi sangat bervariasi, beberapa dari data di atas juga memiliki nilai rata-rata lebih besar daripada nilai variansi.

## 2. Normalisasi Data

Adapun perhitungan untuk normalisasi data adalah sebagai berikut.

Menghitung rata-rata:

$$\begin{aligned}\bar{x}_1 &= \frac{1}{n} \times \sum_{i=1}^n x_{i1} \\ &= \frac{1}{60} \times (20,73 + 29,63 + 44,85 + \dots + 20,67) \\ &= \frac{1}{60} \times 1558,25 \\ &= 25,97\end{aligned}$$

Menghitung varian dan simpangan baku:

$$\begin{aligned}\sigma_1^2 &= \frac{1}{n-1} \times \sum_{i=1}^n (x_{i1} - \bar{x}_1)^2 \\ &= \frac{1}{60-1} \times [(20,73 - 25,97)^2 + (29,63 - 25,97)^2 + \dots + (20,67 - 25,97)^2] \\ &= \frac{1}{59} \times 5717,91 \\ &= 96,914 \\ \sigma_1 &= \sqrt{96,914} \\ &= 9,8445\end{aligned}$$

Adapun nilai normalisasi data pada variabel *Total Moisture* (TM) untuk data ke-1, 2 dan 3 adalah:

$$\begin{aligned}\hat{x}_{11} &= \frac{x_{11} - \bar{x}_1}{\sigma_1} \\ &= \frac{20,73 - 25,97}{9,8445} \\ &= -0,5323 \\ \hat{x}_{21} &= \frac{x_{21} - \bar{x}_1}{\sigma_1} \\ &= \frac{29,63 - 25,97}{9,8445} \\ &= 0,3716 \\ \hat{x}_{31} &= \frac{x_{31} - \bar{x}_1}{\sigma_1} \\ &= \frac{44,85 - 25,97}{9,8445} \\ &= 1,9177\end{aligned}$$

Perhitungan yang sama dilakukan seterusnya dari data ke-4 sampai dengan data ke-60 dan dilanjutkan untuk semua variabel. Berikut adalah hasil normalisasi data yang diperoleh dapat dilihat pada Tabel 2.

Tabel 2. Data Normalisasi

NO	Kelas	TM (%)	M (%)	ASH (%)	VM (%)	FC (%)	TS (%)	GCV (Kcal/Kg)
1	1	-0,5323	-1,0748	3,3228	-3,1419	-1,1902	-0,3608	-1,5063
2	1	0,3717	0,3289	1,7266	-1,0139	-1,6147	0,5099	-1,5653
3	1	1,9177	2,2363	0,0817	-1,3979	-1,1952	-0,5371	-1,3253
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
58	3	-0,766	-0,4489	-0,7589	0,1029	1,1072	2,2843	0,7444
59	3	-1,2962	-0,6786	-0,7133	0,5257	1,0406	-0,4930	1,3162
60	3	-0,5385	-0,3119	-0,7369	1,2409	0,4379	1,9647	1,0058

Selanjutnya akan dilakukan pengacakan data untuk dijadikan sebagai data *training* dan data *testing*.

### 3. Pengacakan Data

Pengacakan data pada penelitian ini bertujuan agar semua data mendapat kesempatan yang sama menjadi data *training* dan data *testing*. Berikut adalah hasil pengacakan data yang diperoleh dapat dilihat pada Tabel 3.

Tabel 3. Pengacakan Data

NO	Kelas	TM (%)	M (%)	ASH (%)	VM (%)	FC (%)	TS (%)	GCV (Kcal/Kg)
1	1	-1,5827	-1,9748	4,16496	-2,9975	-1,4399	1,90957	-1,1054
2	2	-0,1626	0,0085	-0,6473	1,08942	0,16163	-0,1844	0,22038
3	2	0,18479	-0,3414	0,10524	0,50105	-0,0814	-0,3938	-0,0586
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
58	2	-0,1108	0,05276	0,04082	-0,1472	-0,0182	-0,1954	-0,1101
59	3	-0,4328	-0,3329	-1,0495	0,49753	1,14053	-0,504	0,9329
60	1	-0,3434	0,52487	1,42967	-1,1372	-1,395	5,40323	-1,2273

Selanjutnya akan dilakukan pembagian data *training* dan data *testing*.

### 4. Pembagian Data *Training* dan Data *Testing*

Adapun perhitungan pembagian data *training* dan data *testing* sebagai berikut.

Jumlah data *training* = Proporsi data *training* × n

$$= \frac{80}{80+20} \times 60$$

$$= 48$$

Jumlah data *testing* = n – Jumlah data *training*

$$= 60 - 48$$

$$= 12$$

Berdasarkan hasil perhitungan dengan proporsi 80%:20% diperoleh bahwa 48 data pertama hasil randomisasi data untuk masing-masing variabel penelitian berfungsi sebagai data *training* yang akan digunakan pada metode MK-NN. Berikut adalah data *training* yang diperoleh dapat dilihat pada Tabel 4.

Tabel 4. Data *Training*

NO	Kelas	TM (%)	M (%)	ASH (%)	VM (%)	FC (%)	TS (%)	GCV (Kcal/Kg)
1	1	-1,5827	-1,9748	4,16496	-2,9975	-1,4399	1,90957	-1,1054
2	2	-0,1626	0,0085	-0,6473	1,08942	0,16163	-0,1844	0,22038
3	2	0,18479	-0,3414	0,10524	0,50105	-0,0814	-0,3938	-0,0586
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
46	2	0,09134	0,3668	-0,3865	0,57856	-0,1564	0,05804	-0,0687
47	2	-0,0184	-0,2866	-0,4808	0,53981	0,47794	-0,0081	0,28698
48	1	0,3717	0,32886	1,72661	-1,0139	-1,6147	0,50991	-1,5653

Berikut adalah data *testing* yang diperoleh dapat dilihat pada Tabel 5.

Tabel 5. Data *Testing*

NO	Kelas	TM (%)	M (%)	ASH (%)	VM (%)	FC (%)	TS (%)	GCV (Kcal/Kg)
1	1	0,86436	0,40895	0,92692	-0,5876	-1,0304	-0,7024	-1,2122
2	3	-1,7158	-1,9242	0,05025	-0,5171	1,70823	-0,3718	1,68815
3	2	0,30567	-0,0927	-0,4211	1,16341	-0,0332	0,43276	0,25054
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10	2	-0,1108	0,05276	0,04082	-0,1472	-0,0182	-0,1954	-0,1101
11	3	-0,4328	-0,3329	-1,0495	0,49753	1,14053	-0,504	0,9329
12	1	-0,3434	0,52487	1,42967	-1,1372	-1,395	5,40323	-1,2273

Selanjutnya dilakukan perhitungan dari ketiga jarak yang digunakan.

## 5. Perhitungan Jarak sebagai berikut.

### a. Jarak *Euclid*

#### 1) Perhitungan nilai validitas

Adapun perhitungan validitas adalah sebagai berikut.

Perhitungan validitas pada data *training* ke-1

$$\text{validitas}(a) = \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a))$$

$$\begin{aligned} \text{validitas}(1) &= \frac{1}{1} [s((x_1). (n_1(x_{36})))] \\ &= \frac{1}{1} \times 1 \\ &= 1 \end{aligned}$$

Perhitungan validitas pada data *training* ke-2

$$\text{validitas}(a) = \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a))$$

$$\begin{aligned} \text{validitas}(2) &= \frac{1}{1} [s((x_1). (n_1(x_5)))] \\ &= \frac{1}{1} \times 1 \\ &= 1 \end{aligned}$$

Perhitungan di atas dilakukan sampai dengan data *training* ke-48.

#### 2) Perhitungan *weighted voting*

Adapun perhitungan *weight voting* adalah sebagai berikut.

- Perhitungan *weight voting* data *testing* 1

$$\begin{aligned} w(x_{27}, y_1) &= \text{validitas (27)} \times \frac{1}{d(x_{27}, y_1) + 0,5} \\ &= 1 \times \frac{1}{0,6537+0,5} \\ &= 0,86675 \end{aligned}$$

- Perhitungan *weight voting* data *testing* 2

$$\begin{aligned} w(x_8, y_2) &= \text{validitas (8)} \times \frac{1}{d(x_8, y_2) + 0,5} \\ &= 1 \times \frac{1}{1,6469+0,5} \\ &= 0,46579 \end{aligned}$$

⋮

- Perhitungan *weight voting* data *testing* 11

$$\begin{aligned} w(x_{15}, y_{11}) &= \text{validitas (15)} \times \frac{1}{d(x_{15}, y_{11}) + 0,5} \\ &= 1 \times \frac{1}{0,3384+0,5} \\ &= 1,19274 \end{aligned}$$

- Perhitungan *weight voting* data *testing* 12

$$\begin{aligned} w(x_{19}, y_{12}) &= \text{validitas (19)} \times \frac{1}{d(x_{19}, y_{12}) + 0,5} \\ &= 1 \times \frac{1}{4,6048+0,5} \\ &= 0,19589 \end{aligned}$$

Adapun perhitungan *voting* diperoleh perbandingan kelas prediksi data *testing* dan kelas data *testing* asli dapat dilihat pada Tabel 6.

Tabel 6. Hasil Perbandingan Kelas Prediksi Data *Testing* dan Kelas Aslinya Jarak *Euclid*

Data <i>Testing</i>	Kelas Prediksi	Kelas Data Asli	Keterangan
1	1	1	Benar
2	3	3	Benar
3	2	2	Benar
4	1	1	Benar
5	1	1	Benar
6	3	3	Benar
7	1	1	Benar
8	2	2	Benar
9	3	3	Benar
10	2	2	Benar
11	2	3	Salah
12	2	2	Benar

Selanjutnya dilakukan perhitungan untuk menentukan nilai akurasi.

### 3) Perhitungan Nilai Akurasi

Berikut perhitungan akurasi prediksi tepat dari metode MK-NN yaitu sebagai berikut:

$$\begin{aligned}
 \text{akurasi} &= \frac{\text{jumlah data yang diprediksi benar}}{\text{jumlah prediksi yang dilakukan}} \times 100\% \\
 &= \frac{11}{12} \times 100\% \\
 &= 92\%
 \end{aligned}$$

b. Jarak *Manhattan*

1) Perhitungan nilai validitas

Adapun perhitungan validitas adalah sebagai berikut.

Perhitungan validitas pada data *training* ke-1

$$\begin{aligned}
 \text{validitas}(a) &= \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a)) \\
 \text{validitas}(1) &= \frac{1}{1} [s((x_1). (n_1(x_{36})))]) \\
 &= \frac{1}{1} \times 1 \\
 &= 1
 \end{aligned}$$

Perhitungan validitas pada data *training* ke-2

$$\begin{aligned}
 \text{validitas}(a) &= \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a)) \\
 \text{validitas}(2) &= \frac{1}{1} [s((x_1). (n_1(x_5)))] \\
 &= \frac{1}{1} \times 1 \\
 &= 1
 \end{aligned}$$

Perhitungan di atas dilakukan sampai dengan data *training* ke-48.

2) Perhitungan *weighted voting*

Adapun perhitungan *weight voting* adalah sebagai berikut.

• Perhitungan *weight voting* data *testing* 1

$$\begin{aligned}
 w(x_{27}, y_1) &= \text{validitas}(27) \times \frac{1}{d(x_{27}, y_1) + 0,5} \\
 &= 1 \times \frac{1}{1,3+0,5} \\
 &= 0,55435
 \end{aligned}$$

• Perhitungan *weight voting* data *testing* 2

$$\begin{aligned}
 w(x_{21}, y_2) &= \text{validitas}(21) \times \frac{1}{d(x_{21}, y_2) + 0,5} \\
 &= 1 \times \frac{1}{3,55+0,5} \\
 &= 0,24697
 \end{aligned}$$

⋮

• Perhitungan *weight voting* data *testing* 11

$$\begin{aligned}
 w(x_{15}, y_{11}) &= \text{validitas}(15) \times \frac{1}{d(x_{15}, y_{11}) + 0,5} \\
 &= 1 \times \frac{1}{0,77+0,5} \\
 &= 0,78703
 \end{aligned}$$

• Perhitungan *weight voting* data *testing* 12

$$\begin{aligned}
 w(x_{48}, y_{12}) &= \text{validitas}(48) \times \frac{1}{d(x_{19}, y_{12}) + 0,5} \\
 &= 1 \times \frac{1}{6,78+0,5} \\
 &= 0,13732
 \end{aligned}$$

Adapun perhitungan *voting* diperoleh perbandingan kelas prediksi data *testing* dan kelas data *testing* asli dapat dilihat pada Tabel 7.

Tabel 7. Hasil Perbandingan Kelas Prediksi Data *Testing* dan Kelas Aslinya Jarak *Manhattan*

Data <i>Testing</i>	Kelas Prediksi	Kelas Data Asli	Keterangan
1	1	1	Benar
2	3	3	Benar
3	2	2	Benar
4	1	1	Benar
5	1	1	Benar
6	3	3	Benar
7	1	1	Benar
8	2	2	Benar
9	3	3	Benar
10	2	2	Benar
11	3	3	Benar
12	1	1	Benar

Selanjutnya dilakukan perhitungan untuk menentukan nilai akurasi.

### 3) Perhitungan Nilai Akurasi

Berikut perhitungan akurasi prediksi tepat dari metode MK-NN yaitu sebagai berikut:

$$\begin{aligned}
 \text{akurasi} &= \frac{\text{jumlah data yang diprediksi benar}}{\text{jumlah prediksi yang dilakukan}} \times 100\% \\
 &= \frac{12}{12} \times 100\% \\
 &= 100\%
 \end{aligned}$$

### c. Jarak *Minkowski*

#### 1) Perhitungan nilai validitas

Adapun perhitungan validitas adalah sebagai berikut.

Perhitungan validitas pada data *training* ke-1

$$\begin{aligned}
 \text{validitas}(a) &= \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a)) \\
 \text{validitas}(1) &= \frac{1}{1} [s((x_1). (n_1(x_{36})))] \\
 &= \frac{1}{1} \times 1 \\
 &= 1
 \end{aligned}$$

Perhitungan validitas pada data *training* ke-2

$$\begin{aligned}
 \text{validitas}(a) &= \frac{1}{K} \sum_{i=1}^k s(\text{lbl}(x_a)) \\
 \text{validitas}(2) &= \frac{1}{1} [s((x_1). (n_1(x_5)))] \\
 &= \frac{1}{1} \times 1 \\
 &= 1
 \end{aligned}$$

Perhitungan di atas dilakukan sampai dengan data *training* ke-48.

#### 2) Perhitungan *weighted voting*

Adapun perhitungan *weight voting* adalah sebagai berikut.

- Perhitungan *weight voting* data *testing* 1

$$\begin{aligned} w(x_{27}, y_1) &= \text{validitas (27)} \times \frac{1}{d(x_{27}, y_1) + 0,5} \\ &= 1 \times \frac{1}{0,0537+0,5} \\ &= 1,80597 \end{aligned}$$

- Perhitungan *weight voting* data *testing* 2

$$\begin{aligned} w(x_8, y_2) &= \text{validitas (8)} \times \frac{1}{d(x_8, y_2) + 0,5} \\ &= 1 \times \frac{1}{0,666+0,5} \\ &= 0,85796 \end{aligned}$$

⋮

- Perhitungan *weight voting* data *testing* 11

$$\begin{aligned} w(x_{15}, y_{11}) &= \text{validitas (15)} \times \frac{1}{d(x_{15}, y_{11}) + 0,5} \\ &= 1 \times \frac{1}{0,006+0,5} \\ &= 1,975 \end{aligned}$$

- Perhitungan *weight voting* data *testing* 12

$$\begin{aligned} w(x_{19}, y_{12}) &= \text{validitas (19)} \times \frac{1}{d(x_{19}, y_{12}) + 0,5} \\ &= 1 \times \frac{1}{23,37+0,5} \\ &= 0,0419 \end{aligned}$$

Adapun perhitungan *voting* diperoleh perbandingan kelas prediksi data *testing* dan kelas data *testing* asli dapat dilihat pada Tabel 8.

Tabel 8. Hasil Perbandingan Kelas Prediksi Data *Testing* dan Kelas Aslinya Jarak *Minkowski*

Data Testing	Kelas Prediksi	Kelas Data Asli	Keterangan
1	1	1	Benar
2	3	3	Benar
3	3	2	Salah
4	1	1	Benar
5	1	1	Benar
6	3	3	Benar
7	1	1	Benar
8	3	2	Salah
9	2	3	Salah
10	3	2	Salah
11	2	3	Salah
12	2	2	Benar

Selanjutnya dilakukan perhitungan untuk menentukan nilai akurasi.

### 3) Perhitungan Nilai Akurasi

Berikut perhitungan akurasi prediksi tepat dari metode MK-NN yaitu sebagai berikut:

$$\begin{aligned} \text{akurasi} &= \frac{\text{jumlah data yang diprediksi benar}}{\text{jumlah prediksi yang dilakukan}} \times 100\% \\ &= \frac{8}{12} \times 100\% \\ &= 58\% \end{aligned}$$

## 6. Perbandingan Nilai Akurasi

Perbandingan nilai akurasi untuk mengetahui nilai akurasi terbaik diantara jarak *Euclid*, jarak *Manhattan* dan jarak *Minkowski*. Adapun hasil perbandingan nilai akurasi dapat dilihat pada Tabel 9.

Tabel 9. Hasil Perbandingan akurasi antar tiga jarak

Proporsi	Jarak <i>Euclid</i>	Jarak <i>Manhattan</i>	Jarak <i>Minkowski</i>
80%:20%	92%	100%	52%

Pada perhitungan menggunakan proporsi 80%:20% didapatkan nilai akurasi jarak *Euclid* sebanyak 92%, jarak *Manhattan* sebanyak 100%, dan jarak *Minkowski* sebanyak 52%.

## PENUTUP

Berdasarkan hasil penelitian dan pembahasan, maka kesimpulan yang dapat diambil dalam pengklasifikasian dengan metode MK-NN dalam klasifikasi batubara di PT. XYZ Samarinda berdasarkan kelas prediksi dengan menggunakan Jarak *Euclid* yang termasuk klasifikasi *Lignite* adalah data *testing* 1, 4, 5 dan 7, klasifikasi *Sub-Bituminous* adalah data *testing* 3, 8, 10, 11 dan 12, klasifikasi *Bituminous* adalah data *testing* 2, 6 dan 9. Jarak *Manhattan* yang termasuk klasifikasi *Lignite* adalah data *testing* 1, 4, 5, 7 dan 12, klasifikasi *Sub-Bituminous* adalah data *testing* 3, 8, dan 10, klasifikasi *Bituminous* adalah data *testing* 2, 6, 9 dan 11. Jarak *Minkowski* yang termasuk klasifikasi *Lignite* adalah data *testing* 1, 4, 5 dan 7, klasifikasi *Sub-Bituminous* adalah data *testing* 9, 11 dan 12, klasifikasi *Bituminous* adalah data *testing* 2, 3, 6, 8 dan 10.

Perbandingan akurasi metode pengukuran jarak *Euclid*, jarak *manhattan*, dan jarak *minkowski* untuk pengklasifikasian batubara di PT. XYZ telah dilakukan dan memberikan nilai atau tingkat akurasi yang berbeda-beda, yaitu untuk jarak *Euclid* sebanyak 92%, jarak *manhattan* sebanyak 100%, dan jarak *minkowski* sebanyak 52%. Dengan demikian, dapat disimpulkan bahwa jarak *manhattan* merupakan jarak dengan persentasi akurasi terbaik yang dapat diterapkan dalam algoritma *Modified K-Nearest Neighbor* untuk menghitung k lasifikasi batubara berdasarkan jenis kalori dengan  $K=1$  dan proporsi 80%:20%.

Berdasarkan penelitian yang telah dilakukan, pada penelitian selanjutnya dapat melakukan perbandingan jarak dengan metode klasifikasi lainnya seperti *Support Vector Machine (SVM)*.

## DAFTAR PUSTAKA

- [1] Hermawati, F. A. (2013). *Data Mining*. Yogyakarta: Andi.
- [2] Feldman, R., dan Sanger, J. (2007). *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge University Press.
- [3] Parvin, H., Alizadeh, H dan Minae-Bidgoli, B. (2008). *MKNN: Modification on K-Nearest Neighbor Classification*. San Francisco. USA.
- [4] Sukandarrumidi. (2017). *Batubara dan Pemanfaatannya: Pengantar Teknologi Batubara Menuju Lingkungan Bersih*. Yogyakarta: UGM Press.
- [5] Wahyudi, N., Wahyuningsih, S., dan Amijaya, F. (2020). Optimasi Klasifikasi Batubara Berdasarkan Jenis Kalori dengan menggunakan *Genetic Modified K-Nearest Neighbor (GMK-NN)*. *EKSPONENSIAL*, 10(2), 103-112

## Optimalisasi Biaya dan Waktu Pelaksanaan Pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur menggunakan *Critical Path Method (CPM)* dan *Program Evaluation and Review Technique (PERT)*

Sendy Clara Aprillia<sup>1</sup>, Wasono<sup>2,\*</sup>, Qonita Qurrota A'yun<sup>3</sup>

<sup>1,2,3</sup> *Laboratorium Matematika Komputasi Program Studi Matematika Jurusan Matematika FMIPA Universitas Mulawarman*

Dikirim: Oktober 2022;

Diterima: Maret 2023;

Dipublikasi: Maret 2023

Alamat Email Korespondensi: [wasono@fmipa.unmul.ac.id](mailto:wasono@fmipa.unmul.ac.id)

### Abstrak

Pelaksanaan proyek-proyek berskala kecil ataupun berskala besar membutuhkan perencanaan, penjadwalan serta pengkoordinasian yang baik. Dalam riset operasi terdapat teknik penjadwalan proyek untuk memecahkan persoalan pengelolaan suatu proyek. Di antara metode untuk menyelesaikan masalah penjadwalan proyek adalah *Critical Path Method (CPM)* dan *Program Evaluation And Review Technique (PERT)*. Metode CPM digunakan untuk mengoptimalkan biaya proyek melalui pengurangan waktu penyelesaian proyek. Sedangkan, metode PERT lebih menekankan pengoptimalan waktu penyelesaian proyek dengan tiga taksiran waktu yaitu waktu optimistik, waktu realistik, dan waktu pesimistik. Penelitian ini bertujuan untuk memperoleh pengoptimalan biaya dan waktu pelaksanaan pembangunan rumah tinggal di kecamatan Rantau Pulung Kutai Timur. Data yang digunakan adalah jenis pekerjaan dan waktu penyelesaian proyek serta biaya upah pekerja yang perhitungannya dibantu *Software Lingo*. Hasil analisis menunjukkan bahwa durasi minimum penyelesaian proyek rumah tinggal di Kecamatan Rantau Pulung Kutai Timur dengan metode PERT adalah selama 40 hari. Adapun hasil optimalisasi dengan metode CPM adalah selama 39 hari yang berarti dapat dipercepat 14 hari dengan biaya upah pekerja sebesar Rp34.930.000,- atau bertambah sebesar Rp805.000,- dari biaya upah pekerja pada waktu normal.

### Kata Kunci:

*Critical Path Method (CPM), Penjadwalan Proyek, Program Evaluation and Review Technique (PERT).*

### PENDAHULUAN

Indonesia adalah negara berkembang yang memiliki banyak penduduk dengan pertumbuhan yang sangat pesat. Hal ini menyebabkan kebutuhan akan rumah tinggal semakin meningkat. Untuk memenuhi kebutuhan akan rumah tinggal tersebut, sangat diperlukan perencanaan pelaksanaan proyek pembangunan rumah. Perencanaan proyek dapat berjalan dengan baik menggunakan suatu manajemen perencanaan agar sumber daya dapat dimanfaatkan dengan optimal.

Riset operasi adalah salah satu bidang ilmu manajemen yang menggunakan perangkat dan matematika untuk memecahkan suatu persoalan manajemen dalam mengambil keputusan terbaik [1]. Riset operasi memiliki metode yang bisa digunakan

dalam perencanaan proyek di antaranya ialah *Critical Path Method (CPM)* dan *Program Evaluation and Review Technique (PERT)* [2].

Metode CPM digunakan untuk menganalisis jaringan kerja sehingga dapat mengoptimalkan biaya proyek melalui pengurangan waktu penyelesaian proyek. Sedangkan, metode PERT lebih menekankan analisis jaringan kerja sehingga dapat mengoptimalkan waktu penyelesaian proyek. Meskipun PERT dan CPM adalah serupa dalam pendekatan dasarnya, namun berbeda pada taksiran waktu yang diperlukan dalam melaksanakan aktivitas. Bagi setiap aktivitas, metode PERT menggunakan tiga taksiran (waktu optimistik, waktu realistik dan waktu pesimistik) untuk menentukan waktu penyelesaian aktivitas yang diharapkan. Sedangkan, metode CPM menggunakan estimasi tunggal waktu aktivitas dengan asumsi waktu aktivitas diketahui dengan pasti [3]. Penelitian ini menggunakan metode CPM dan PERT untuk menentukan optimalisasi biaya dan waktu pelaksanaan pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur.

## **LANDASAN TEORI**

### **1). Riset Operasi**

Menurut [4], riset operasi berkaitan dengan pengambilan keputusan yang optimal dalam pemodelan deterministik dan sistem probabilistik yang berasal dari kehidupan nyata. Riset operasi memiliki beberapa tahapan dalam proses penyelesaian masalah, berikut beberapa tahapan dalam riset operasi:

- a. Merumuskan masalah
- b. Membentuk model matematis
- c. Mencari penyelesaian masalah
- d. Menguji model
- e. Melaksanakan keputusan

### **2). Manajemen Proyek**

Manajemen proyek ialah proses merencanakan suatu kegiatan agar berjalan optimal dengan mengatur berbagai sumber daya yang tersedia sebagai pendukung kegiatan tersebut [5]. Perencanaan dalam manajemen suatu proyek perlu mempertimbangkan segala aspek seperti anggaran biaya, keuangan, manajemen sumber daya, manajemen produksi, harga, efektivitas, efisiensi, pemasaran, mutu, waktu dan lingkungan dapat terorganisir dengan baik agar hasil dari proyek sesuai dengan sasaran dan tujuan yang direncanakan [6].

Kegiatan proyek dapat dibagi menjadi dua tahap, yaitu tahap perencanaan dan tahap pelaksanaan. Tahap perencanaan adalah mengidentifikasi sebuah gagasan ke dalam bentuk perancangan yang lebih jelas dengan mengamati pokok-pokok ide tersebut. Selanjutnya mengimplementasikan perencanaan yang dibutuhkan pada pelaksanaan konstruksi [7].

### **3). *Critical Path Method (CPM)***

Menurut [8] CPM adalah model manajemen proyek yang mengutamakan biaya sebagai objek yang dianalisis. Persoalan pokok yang menjadi perhatian model CPM adalah berapa besar biaya untuk menyelesaikan sebuah proyek jika waktu penyelesaiannya normal dan jika waktu penyelesaian suatu proyek harus dipercepat maka berapa besar biaya dan kegiatan mana yang harus dipercepat agar biaya percepatan total minimum. Dari dua macam kondisi yang diobservasi itu, model ini menurunkan empat macam parameter, yaitu:

- Waktu penyelesaian normal atau waktu normal
- Biaya penyelesaian normal atau biaya normal
- Waktu penyelesaian yang dipercepat atau waktu cepat
- Biaya penyelesaian yang dipercepat atau biaya cepat.

Oleh karena itu percepatan waktu penyelesaian suatu kegiatan sebesar  $W$ , yaitu  $W_n$  ke  $W_c$  akan menyebabkan biaya langsung naik sebesar  $B$ , yaitu  $B_n$  ke  $B_c$ . Jika hubungan antara kedua kondisi waktu dan biaya tersebut linear, maka biaya percepatan persatuan waktu dinyatakan dengan:

$$Bt = \frac{Bc - Bn}{Wn - Wc} = \frac{\Delta B}{\Delta W} \quad (1)$$

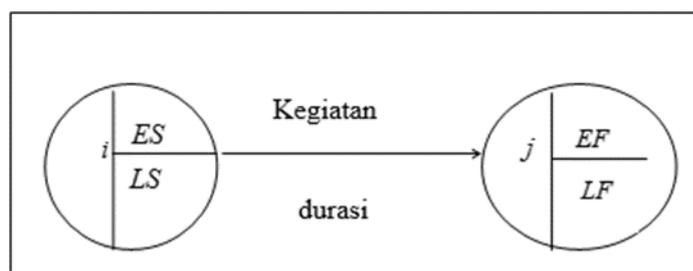
di mana,

- $Bt$  = Biaya tambahan setiap satuan waktu percepatan
- $Bc$  = Biaya dipercepat
- $Bn$  = Biaya normal
- $Wc$  = Waktu dipercepat
- $Wn$  = Waktu normal
- $\Delta B$  = Tambahan biaya persatu satuan waktu
- $\Delta W$  = Tambahan waktu.

Menurut [6] ada beberapa terminologi yang terlibat sehubungan dengan perhitungan maju dan perhitungan mundur metode *Activity On Arrow* (AOA) sebagai berikut:

- Early Start* ( $ES$ ) : waktu paling awal sebuah kegiatan dapat dimulai setelah kegiatan sebelumnya selesai.
- Late Start* ( $LS$ ) : waktu paling akhir sebuah kegiatan dapat diselesaikan tanpa memperlambat penyelesaian jadwal proyek.
- Early Finish* ( $EF$ ) : waktu paling awal sebuah kegiatan dapat diselesaikan jika dimulai pada waktu paling awalnya dan diselesaikan
- Late Finish* ( $LF$ ) : waktu paling akhir sebuah kegiatan dapat dimulai tanpa memperlambat penyelesaian proyek.

Berikut pada Gambar 1 terdapat potongan jaringan kerja AOA.



Gambar 1. Jaringan kerja

Keterangan

ES : Waktu mulai paling cepat (*Early Start*)

LS : Waktu mulai paling lambat (*Late Start*)

EF : Waktu selesai paling cepat (*Early Finish*)

LF : Waktu selesai paling lambat (*Late Finish*)

i : kegiatan yang terdahulu

j : kegiatan yang ditinjau

4). Metode *Program Evaluation and Review Technique* (PERT)

PERT adalah model perencanaan dan pengendalian pada manajemen proyek, PERT menghubungkan kegiatan dan tahap penyelesaian masing-masing kegiatan serta gambaran proyek secara keseluruhan. Estimasi waktu yang digunakan dalam metode PERT, yaitu :

- a. Waktu Optimistik ( $\alpha$ ), waktu kegiatan bila semuanya berjalan baik tanpa hambatan atau penundaan.
- b. Waktu Realistik ( $m$ ) waktu kegiatan yang akan terjadi bila suatu kegiatan dilaksanakan dalam kondisi normal, dengan penundaan tertentu yang dapat diterima.
- c. Waktu Pesimistik ( $b$ ), waktu kegiatan bila terjadi hambatan lebih dari semestinya.

Dari estimasi waktu tersebut, dapat diketahui nilai probabilitas dari kegiatan terutama pada jalur kritis dan sesuai dengan waktu kegiatan yang diharapkan [3].

**HASIL DAN PEMBAHASAN**

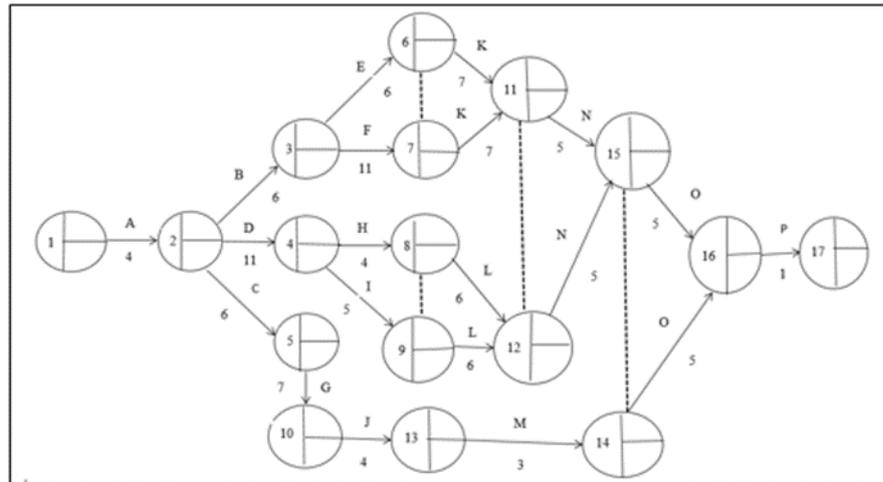
## 1). Penyusunan Jaringan Kerja

Data Rancangan Kegiatan Proyek Pembangunan diperoleh berdasarkan data dan hasil wawancara dengan pihak pelaksana proyek pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur. Adapun rincian kegiatan-kegiatannya seperti pada Tabel 1 berikut.

Tabel 1. Data rancangan proyek

No	Jenis Kegiatan	Simbol Kegiatan	Kegiatan Sebelumnya	Waktu (hari)
1	Pekerjaan persiapan (Bongkar rumah lama)	A	-	4
2	Pekerjaan galian dan urugan	B	A	6
3	Pekerjaan Pondasi	C	A	6
4	Pekerjaan Pasangan	D	A	11
5	Pekerjaan struktur atap	E	B	6
6	Pekerjaan penutup atap	F	B	11
7	Pekerjaan lantai	G	C	7
8	Pekerjaan dinding	H	D	4
9	Pekerjaan instalasi listrik	I	D	5
10	Pekerjaan plafon	J	G	4
11	Pekerjaan kusen, pintu dan jendela	K	E,F	7
12	Pekerjaan perlengkapan pintu dan jendela	L	H,I	6
13	Pekerjaan sanitair	M	J	3
14	Pemasangan instalasi air bersih	N	K,L	5
15	Pekerjaan Pengecatan	O	M,N	5
16	Pekerjaan <i>Finishing</i>	P	O	1

Berdasarkan Tabel 1 di atas terdapat jenis kegiatan dari proses pembuatan rumah kayu yang telah berurut dari awal hingga akhir kegiatan proyek. Kegiatan-kegiatan tersebut disimbol oleh beberapa huruf dan diikuti dengan kegiatan sebelumnya disertai dengan waktu penyelesaian dari setiap kegiatan. Berikut ini ialah jaringan kerja yang diperoleh sebagaimana dapat dilihat pada Gambar 2 berikut.



Gambar 2. Jaringan kerja CPM

Mengacu jaringan kerja pada Gambar 2, langkah selanjutnya adalah dengan melakukan perhitungan maju dan perhitungan mundur untuk mengetahui *Earliest Start (ES)*, *Earliest Finish (EF)*, *Latest Start (LS)* dan *Latest Finish (LF)*. Perhitungan maju didapatkan menggunakan perhitungan ke depan dimulai dari kegiatan awal sedangkan pada perhitungan mundur dimulai dengan kegiatan paling akhir dan dilanjutkan dengan kegiatan sebelumnya menggunakan Persamaan berikut :

$$EET_j = EET_i + \text{durasi}(i, j) \quad (2)$$

$$LET_j = LET_i - \text{durasi}(i, j) \quad (3)$$

dimana,

*EET* = *Earliest Event Time* (durasi tercepat suatu kejadian)

*LET* = *Latest Event Time* (durasi terlambat suatu kejadian)

*i* = kegiatan yang ditinjau

*j* = kegiatan berikutnya

Mengacu jaringan kerja pada Gambar 2, maka langkah selanjutnya adalah dengan melakukan perhitungan maju, perhitungan maju bergerak mulai dari kegiatan awal sampai kegiatan akhir proyek, perhitungan ini dapat dihitung menggunakan Persamaan (2) sehingga didapat hasil berikut:

$$E_f(1,2) = E_s(1,2) + \text{durasi}(A) = 0 + 4 = 4$$

$$E_f(2,3) = E_s(2,3) + \text{durasi}(B) = 4 + 6 = 10$$

$$E_f(2,5) = E_s(2,5) + \text{durasi}(C) = 4 + 6 = 10$$

.....

$$E_f(14,16) = E_s(14,16) + \text{durasi}(O) = 24 + 5 = 29$$

$$E_f(15,16) = E_s(15,16) + \text{durasi}(O) = 33 + 5 = 38$$

$$E_f(16,17) = E_s(16,17) + \text{durasi}(P) = 38 + 1 = 39$$

Langkah selanjutnya adalah dengan melakukan perhitungan mundur, Perhitungan mundur bergerak mulai dari kegiatan akhir sampai kegiatan awal proyek, perhitungan ini dapat dihitung menggunakan Persamaan (3) sehingga didapat hasil:

$$L_s(17,16) = L_f(17,16) - \text{durasi}(P) = 39 - 1 = 38$$

$$L_s(16,15) = L_f(16,15) - \text{durasi}(O) = 38 - 5 = 33$$

$$L_s(16,14) = L_f(16,14) - \text{durasi}(O) = 38 - 5 = 33$$

.....

$$L_s(5,2) = L_f(5,2) - \text{durasi}(C) = 19 - 6 = 13$$

$$L_s(3,2) = L_f(3,2) - \text{durasi}(B) = 10 - 6 = 4$$

$$L_s(2,1) = L_f(2,1) - \text{durasi}(A) = 4 - 4 = 0$$

Setelah mengetahui nilai dari *Earliest Start (ES)*, *Earliest Finish (EF)*, *Latest Start (LS)* dan *Latest Finish (LF)* pada uraian kegiatan, selanjutnya menentukan *float*. *Float* dapat didefinisikan sejumlah waktu yang memungkinkan untuk menunda atau memperlambat kegiatan. Kegiatan kritis terjadi apabila besar selisih antara *EET* dengan *LET* adalah nol, yaitu:

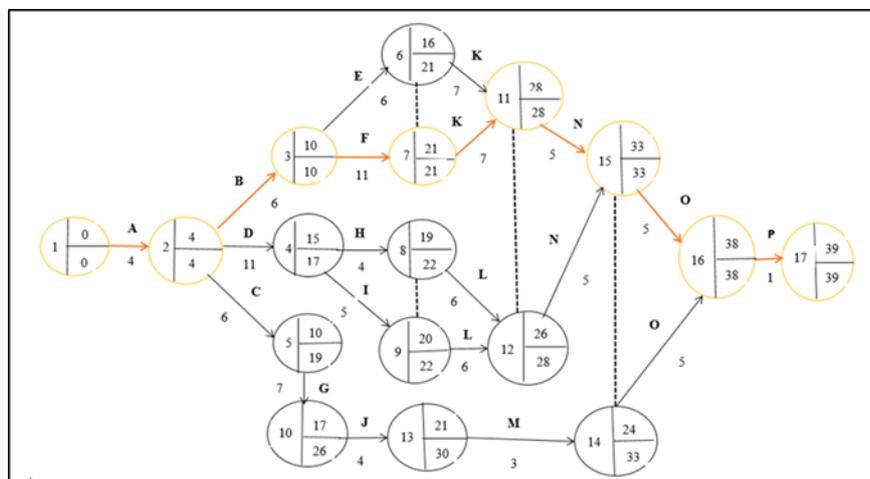
$$EET - LET = 0 \tag{4}$$

Jadi, berdasarkan rumus di atas, diperoleh jalur kritis hasil perhitungan *float* = 0. Adapun notasi TK menyatakan kegiatan tidak berada dalam jalur kritis. Hasil sselengkapnya dapat dilihat pada Tabel 2 berikut.

Tabel 2. Perhitungan *Float Time*

Kegiatan	Kurun Waktu (t)	Awal		Akhir		Total Slack		Jalur Kritis
		(ES)	(EF)	(LS)	(LF)	(LS) - (ES)	(LF) - (EF)	
A	4	0	4	0	4	0	0	Kritis
B	6	4	10	4	10	0	0	Kritis
C	6	4	10	13	19	9	9	TK
D	11	4	15	6	17	2	2	TK
E	6	10	16	15	21	5	5	TK
F	11	10	21	10	21	0	0	Kritis
G	7	10	17	19	26	9	9	TK
H	4	15	19	18	22	3	3	TK
I	5	15	20	17	22	2	2	TK
J	4	17	21	26	30	9	9	TK
K	7	21	28	21	28	0	0	Kritis
L	6	20	26	22	28	2	2	TK
M	3	21	24	30	33	9	9	TK
N	5	28	33	28	33	0	0	Kritis
O	5	33	38	33	38	0	0	Kritis
P	1	38	39	38	39	0	0	Kritis

Hasil dari perhitungan maju dan perhitungan mundur yang telah dihitung *slack* atau *float time*-nya sehingga membentuk sebuah jalur kritis dapat dilihat pada Gambar 3 berikut.



Gambar 3. Jaringan kerja CPM

Berdasarkan gambar 3 di atas, terbentuk jaringan kerja yang sudah diketahui jalur kritisnya yaitu kegiatan A yaitu pekerjaan persiapan (bongkar rumah lama), Kegiatan B yaitu pekerjaan galian dan urugan, Kegiatan F yaitu pekerjaan penutup atap, Kegiatan K yaitu pekerjaan kusen pintu dan jendela, Kegiatan N yaitu pemasangan instalasi air, Kegiatan O yaitu pekerjaan pengecatan dan kegiatan P yaitu pekerjaan *finishing*. Jumlah waktu yang diperlukan dalam proyek pembangunan melalui perhitungan jalur kritis adalah  $(4+6+11+7+5+5+1= 39 \text{ hari})$ .

## 2). Proses *Crashing*

Analisis proses *Crashing* dari beberapa kegiatan yang melewati jalur kritis pada rancangan proyek pembangunan. Rumus untuk menghitung *Crashing* yaitu seperti pada persamaan (1).

Syarat perhitungan untuk menentukan biaya tambahan ialah mengetahui biaya *crash*, biaya normal, waktu normal, waktu *crash* dan jumlah pekerja. Berdasarkan hasil wawancara dengan Tenaga Fasilitator Lapangan (TFL) diperoleh data-data yang diperlukan untuk menghitung biaya tambahan. Sebagai contoh diambil kegiatan A yaitu pekerjaan persiapan (bongkar rumah lama). Data yang diketahui adalah:

1. Waktu normal = 4 hari
2. Waktu *crash* = 2 hari
3. Waktu kerja = 8 jam
4. Jumlah pekerja pada waktu normal tiap kegiatan = 3 orang
5. Jumlah pekerja pada waktu *crash* tiap kegiatan =  $3 + (3 \times 10\%) = 4 \text{ orang}$
6. Jumlah jam pekerja pada waktu normal =  $3 \times 4 \times 8 = 96 \text{ jam}$
7. Jumlah jam pekerja pada waktu *crash* =  $4 \times 2 \times 8 = 64 \text{ jam}$ .
8. Jam kerja lembur yang dibutuhkan =  $96 - 64 = 32 \text{ jam}$  untuk menyelesaikan pekerjaan persiapan (bongkar rumah lama)
9. Biaya gaji perhari = Rp125.000,-
10. Biaya lembur setiap jam = Rp20.000,-
11. Biaya pada waktu *crash* = (Waktu *crash* x jumlah pekerja waktu *crash* x Gaji perhari) + ( Jam kerja lembur total x biaya lembur perjam )  
 $= ( 2 \times 4 \times \text{Rp}125.000,- ) + ( 32 \times \text{Rp}20.000,- )$   
 $= ( \text{Rp}1.000.000,- ) + ( \text{Rp}640.000,- )$   
 $= ( \text{Rp}1.640.000,- )$

Jadi, hasil perhitungan biaya *crash* dari kegiatan A yaitu pekerjaan persiapan (bongkar rumah lama) sebesar Rp1.640.000,-.

Setelah mengetahui biaya *crash* dari kegiatan A selanjutnya mencari biaya tambahan atau *slope* dari kegiatan A menggunakan rumus pada persamaan (1).

$$\text{Biaya tambahan} = \frac{\text{Rp}1.640.000 - \text{Rp}1.500.000}{4 - 2} = 70.000$$

Jadi, biaya tambahan atau biaya *slope* untuk kegiatan A adalah sebesar Rp70.000,- perhari. Hasil perhitungan secara detail dapat dilihat pada Tabel 3 seperti berikut:

Tabel 3. *Crashing* biaya proyek

Kegiatan	Kegiatan Sebelumnya	Waktu (Hari)		Biaya (Rp)		Biaya <i>Slope</i> Per Hari (Rp)
		Normal	Crash	Normal	Crash	
A	-	4	2	1.500.000	1.640.000	70.000
B	A	6	4	2.250.000	2.320.000	35.000
C	A	6	4	2.250.000	2.320.000	35.000
D	A	11	7	4.125.000	4.300.000	43.750
E	B	6	3	2.250.000	2.460.000	70.000
F	B	11	7	4.125.000	4.300.000	43.750
G	C	7	5	2.625.000	2.660.000	17.500
H	D	4	2	1.500.000	1.640.000	70.000
I	D	5	3	1.875.000	1.980.000	52.500
J	G	4	3	1.500.000	1.500.000	0
K	E,F	7	5	2.625.000	2.660.000	17.500
L	H,I	6	4	2.250.000	2.320.000	35.000
M	J	3	2	1.125.000	1.160.000	35.000
N	K,L	5	3	1.875.000	1.980.000	52.500
O	M,N	5	3	1.875.000	1.980.000	52.500
P	O	1	1	375.000	375.000	0

Berdasarkan hasil perhitungan pada Tabel 3 dapat dilakukan perhitungan percepatan proyek pada setiap kegiatan kritis. Lama penyelesaian proyek adalah selama 39 hari dengan biaya langsung total dari seluruh kegiatan awal hingga akhir proyek adalah sebesar Rp34.125.000. Berikut tahap percepatan waktu penyelesaian proyek pada jalur kritis. Sebagai contoh diambil kegiatan K yakni pekerjaan kusen, pintu dan jendela dengan waktu waktu penekanan 2 hari. Dapat kita hitung perhitungan maju dan perhitungan mundur beserta *float time* nya seperti pada persamaan (2), (3) dan (4) dengan waktu waktu penekanan 2 hari sehingga didapat:

$$E_f(1,2) = E_s(1,2) + \text{durasi}(A) = 0 + 4 = 4$$

$$E_f(2,3) = E_s(2,3) + \text{durasi}(B) = 4 + 6 = 10$$

$$E_f(2,5) = E_s(2,5) + \text{durasi}(C) = 4 + 6 = 10$$

.....

.....

$$E_f(14,16) = E_s(14,16) + \text{durasi}(O) = 24 + 5 = 29$$

$$E_f(15,16) = E_s(15,16) + \text{durasi}(O) = 31 + 5 = 36$$

$$E_f(16,17) = E_s(16,17) + \text{durasi}(P) = 36 + 1 = 37$$

Langkah selanjutnya adalah dengan melakukan perhitungan mundur

$$L_s(17,16) = L_f(17,16) - \text{durasi}(P) = 37 - 1 = 36$$

$$L_s(16,15) = L_f(16,15) - \text{durasi}(O) = 36 - 5 = 31$$

$$L_s(16,14) = L_f(16,14) - \text{durasi}(O) = 36 - 5 = 31$$

.....

.....

$$L_s(5,2) = L_f(5,2) - \text{durasi}(C) = 17 - 6 = 11$$

$$L_s(3,2) = L_f(3,2) - \text{durasi}(B) = 10 - 6 = 4$$

$$L_s(2,1) = L_f(2,1) - \text{durasi}(A) = 4 - 4 = 0$$

Setelah mengetahui nilai dari *Earliest Start (ES)*, *Earliest Finish (EF)*, *Latest Start (LS)* dan *Latest Finish (LF)* pada uraian kegiatan, selanjutnya menentukan *float* seperti pada Tabel 4 berikut:

Tabel 4. Perhitungan *float time* percepatan proyek

KEGIATAN	KURUN WAKTU (t)	AWAL		AKHIR		TOTAL SLACK		JALUR KRITIS
		(ES)	(EF)	(LS)	(LF)	(LS) - (ES)	(LF) - (EF)	
A	4	0	4	0	4	0	0	KRITIS
B	6	4	10	4	10	0	0	KRITIS
C	6	4	10	11	17	7	7	TK
E	6	10	16	15	21	5	5	TK
F	11	10	21	10	21	0	0	KRITIS
G	7	10	17	17	24	7	7	TK
H	4	15	19	16	20	1	1	TK
I	5	15	20	15	20	0	0	TK
J	4	17	21	24	28	7	7	TK
K	5	21	26	21	26	0	0	KRITIS
L	6	20	26	20	26	0	0	TK
M	3	21	24	28	31	7	7	TK
N	5	26	31	26	31	0	0	KRITIS
O	5	31	36	31	36	0	0	KRITIS
P	1	36	37	36	37	0	0	KRITIS

Oleh karena telah dilakukan percepatan pada kegiatan K dari 7 hari menjadi 5 hari dan dapat diketahui:

a. Waktu penyelesaian proyek = 37 hari

b. Biaya langsung proyek setelah percepatan = Rp34.125.000,- + ((7-5) x Rp17.500,-)  
= Rp34.160.000,-

c. Jalur kritis tetap yaitu 1→2→3→7→11→11→15→16→17

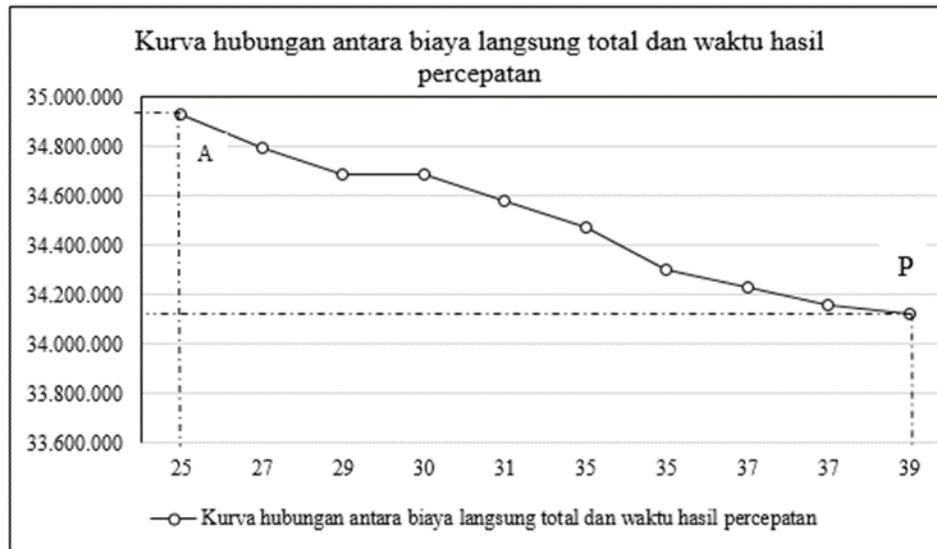
Pada percepatan ini diketahui biaya langsung proyek total tetap naik menjadi Rp34.160.000,- atau naik sebesar Rp35.000,- dari biaya pada waktu normal. Waktu penyelesaian proyek lebih cepat 2 hari, dari 39 hari menjadi 37 hari.

Karena seluruh kegiatan yang berada pada lintasan kritis telah mencapai *crash time*-nya, maka tidak lagi dilakukan pengurangan terhadap waktu penyelesaian proyek pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur, sehingga perhitungan selesai. Adapun data hasil percepatan proyek pada kegiatan kritis dapat dilihat pada Tabel 5 berikut.

Tabel 5. Data hasil percepatan proyek

Kegiatan Kritis	Durasi Normal	Durasi Crash	Waktu Sebelum crash	Waktu Sesudah Crash	Biaya langsung Total (Rupiah)	Slope (Rupiah/hari)
P (16,17)	1	1	39	39	34.125.000	0
K(7,11)	7	5	39	37	34.160.000	17.500
B(2,3)	6	4	37	37	34.230.000	35.000
L(9,12)	6	4	37	35	34.300.000	35.000
F(3,7)	11	7	35	35	34.475.000	43.750
D(2,4)	11	7	35	31	34.580.000	52.500
N(11,15)	5	3	31	30	34.685.000	52.500
J(10,13)	4	3	30	29	34.685.000	0
O(15,16)	5	3	29	27	34.790.000	52.500
A(1,2)	4	2	27	25	34.930.000	70.000

Berdasarkan data pada Tabel 5 di atas, dapat dibentuk kurva seperti pada Gambar 4 berikut.



Gambar 4. Kurva hubungan antara biaya langsung total dan waktu

### 3). Penyusunan Jaringan kerja *Program Evaluation And Review Technique* (PERT)

Data rancangan kerja dicantumkan serta diurutkan proses pengerjaan proyek dari tahap awal pengerjaan hingga tahap penyelesaian proyek. Kegiatan ini diperoleh berdasarkan data dan wawancara dengan pihak pelaksana proyek pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur. Adapun rincian kegiatan-kegiatannya seperti pada Tabel 6.

Penyusunan jaringan kerja menggunakan metode PERT yakni menentukan perkiraan durasi optimistik (a) dan durasi pesimistik (b) dari setiap aktivitas berdasarkan durasi realistik (c) terdapat pada Tabel 6 berikut.

Tabel 6. Perkiraan Durasi Optimistik, Pesimistik, dan Realistik

No	Kegiatan	Kegiatan Sebelumnya	Durasi Realistik (m)	Durasi Optimistik (a)	Durasi Pesimistik (b)
1	A	-	4	2	5
2	B	A	6	4	8
3	C	A	6	5	7
4	D	A	11	7	15
5	E	B	6	3	10
6	F	B	11	7	16
7	G	C	7	6	9
8	H	D	4	2	6
9	I	D	5	3	8
10	J	G	4	3	6
11	K	E,F	7	5	9
12	L	H,I	6	5	7
13	M	J	3	2	5
14	N	K,L	5	3	6
15	O	M,N	5	3	6
16	P	O	1	1	2

Setelah membuat estimasi waktu maka dicari nilai waktu yang diharapkan ( $E_t$ ) dengan menggunakan rumus seperti pada persamaan berikut:

$$Et = \frac{a + 4m + b}{6} \quad (5)$$

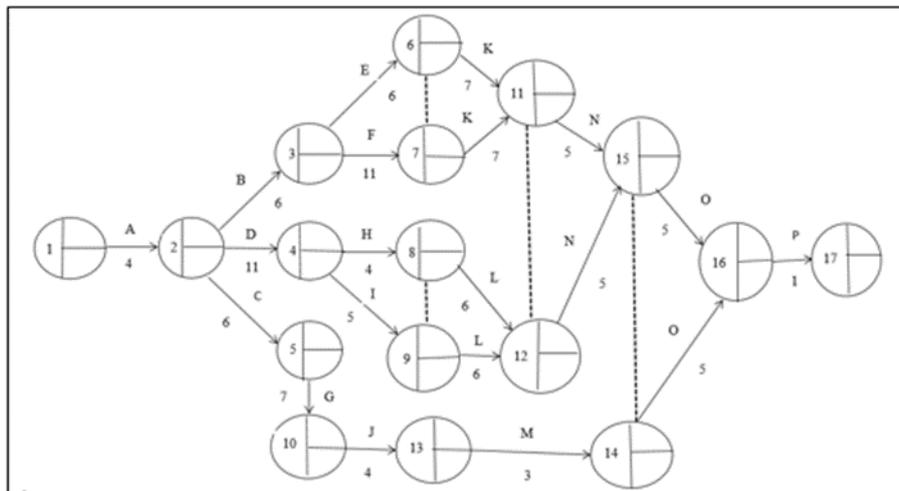
Hasil dari perhitungan untuk mencari nilai yang diharapkan dapat dilihat pada Tabel 7 berikut.

Tabel 7. Waktu yang diharapkan

No	Simbol Kegiatan	Kegiatan Sebelumnya	Waktu yang diharapkan
1	A	-	4
2	B	A	6
3	C	A	6
4	D	A	11
5	E	B	7
6	F	B	12
7	G	C	8
8	H	D	4
9	I	D	6
10	J	G	5
11	K	E,F	7
12	L	H,I	6
13	M	J	4
14	N	K,L	5
15	O	M,N	5
16	P	O	1

Hasil perhitungan waktu yang diharapkan dari tiga estimasi waktu tersebut menghasilkan estimasi waktu baru yang digunakan dalam perhitungan maju dan perhitungan mundur untuk mencari jalur kritis dari metode PERT.

Jaringan kerja dapat diperoleh setelah melaksanakan langkah pertama dalam pendekatan PERT yakni mencari waktu optimis dan waktu pesimis berdasarkan waktu normal. Berikut ini ialah jaringan kerja yang diperoleh, dapat dilihat pada Gambar 5.

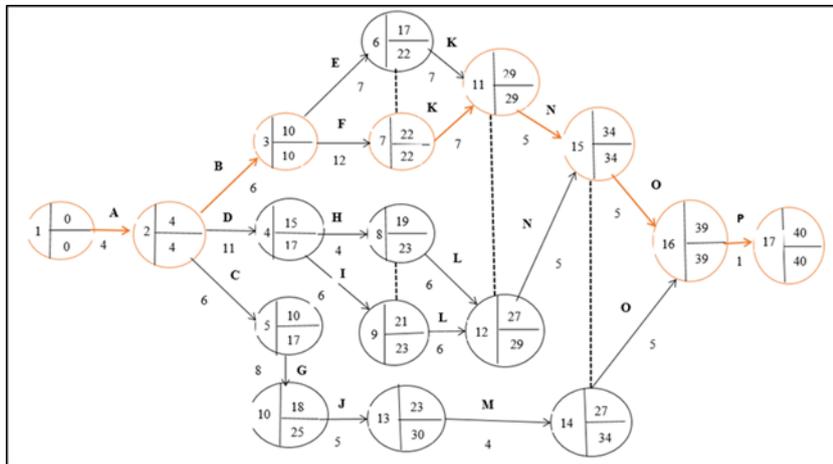


Gambar 5. Jaringan kerja PERT

Mengacu jaringan kerja pada Gambar 5, langkah selanjutnya adalah dengan melakukan perhitungan maju dan perhitungan mundur untuk mengetahui *Earliest Start (ES)*, *Earliest Finish (EF)*, *Latest Start (LS)* dan *Latest Finish (LF)*. Perhitungan maju didapatkan menggunakan perhitungan ke depan dimulai dari kegiatan awal sedangkan pada perhitungan mundur dimulai dengan kegiatan paling akhir dan

dilanjutkan dengan kegiatan sebelumnya menggunakan rumus seperti pada persamaan (2) dan (3).

Setelah mengetahui nilai dari *Earliest Start (ES)*, *Earliest Finish (EF)*, *Latest Start (LS)* dan *Latest Finish (LF)* pada uraian kegiatan, selanjutnya menentukan *float*. Jadi, berdasarkan rumus pada Persamaan (4) di atas, diperoleh jalur kritis hasil perhitungan *float = 0* sebagaimana dapat dilihat pada Gambar 6 berikut.



Gambar 6. Jalur kritis PERT

Berdasarkan Gambar 6, terbentuk jaringan kerja yang sudah diketahui jalur kritisnya yaitu kegiatan A yaitu pekerjaan persiapan (bongkar rumah lama), Kegiatan B yaitu pekerjaan galian dan urugan, Kegiatan F yaitu pekerjaan penutup atap, Kegiatan K yaitu pekerjaan kusen pintu dan jendela, Kegiatan N yaitu pemasangan instalasi air, Kegiatan O yaitu pekerjaan pengecatan dan Kegiatan P yaitu pekerjaan *finishing*. Jumlah waktu yang diperlukan dalam proyek pembangunan melalui perhitungan jalur kritis adalah  $(4+6+12+7+5+5+1=40)$  hari).

#### 4). Deviasi standar

Setelah diketahui jaringan kritis dilakukan perhitungan standar deviasi untuk mengetahui persentase keberhasilan proyek. Perhitungan standar deviasi dari setiap kegiatan pada jalur kritis diperlukan untuk menghitung varian dari kegiatan kritis serta untuk mencari nilai probabilitas. Perhitungan deviasi standar dari metode PERT dapat diperoleh dari persamaan berikut:

$$\sigma = \frac{1}{6}(b - a) \quad (6)$$

$$\sigma_{1,2} = \frac{1}{6}(5 - 2) = 0,5$$

$$\sigma_{2,3} = \frac{1}{6}(8 - 4) = 0,6$$

$$\sigma_{3,7} = \frac{1}{6}(16 - 7) = 1,5$$

$$\sigma_{7,11} = \frac{1}{6}(9 - 5) = 0,6$$

$$\sigma_{11,15} = \frac{1}{6}(6 - 3) = 0,5$$

$$\sigma_{15,16} = \frac{1}{6}(6 - 3) = 0,5$$

$$\sigma_{16,17} = \frac{1}{6}(2 - 1) = 0,1$$

Berdasarkan hasil perhitungan deviasi standar dari kegiatan-kegiatan pada jalur kritis, diperoleh jumlah deviasi standarnya ialah:

$$\begin{aligned}\sum \sigma_{ij} &= \sigma_{1,2} + \sigma_{2,3} + \sigma_{3,7} + \sigma_{7,11} + \sigma_{11,15} + \sigma_{15,16} + \sigma_{16,17} \\ &= 0,5 + 0,6 + 1,5 + 0,6 + 0,5 + 0,5 + 0,1 \\ &= 4,3\end{aligned}$$

Jadi, jumlah deviasi standar dari setiap kegiatan pada jalur kritis ialah sebesar 4,3. Nilai tersebut bisa digunakan untuk mencari nilai variansi.

##### 5). Variansi

Kuadrat dari standar deviasi adalah variansi. Perhitungan variansi dari setiap kegiatan pada jalur kritis di peroleh dari Persamaan berikut :

$$v = \left(\frac{b-a}{6}\right)^2 \quad (7)$$

$$\begin{aligned}v_{1,2} &= \left(\frac{5-2}{6}\right)^2 = 0,25 \\ v_{23} &= \left(\frac{8-4}{6}\right)^2 = 0,36 \\ v_{3,7} &= \left(\frac{16-7}{6}\right)^2 = 2,25 \\ v_{7,11} &= \left(\frac{9-5}{6}\right)^2 = 0,36 \\ v_{11,15} &= \left(\frac{6-3}{6}\right)^2 = 0,25 \\ v_{15,16} &= \left(\frac{6-3}{6}\right)^2 = 0,25 \\ v_{16,17} &= \left(\frac{2-1}{6}\right)^2 = 0,01\end{aligned}$$

Berdasarkan hasil perhitungan variansi dari kegiatan-kegiatan pada jalur kritis diperoleh jumlah variansi ialah:

$$\begin{aligned}\sum v_{ij} &= v_{1,2} + v_{2,3} + v_{3,7} + v_{7,11} + v_{11,15} + v_{15,16} + v_{16,17} \\ &= 0,25 + 0,36 + 2,25 + 0,36 + 0,25 + 0,25 + 0,01 \\ &= 3,73\end{aligned}$$

Jadi, jumlah variansi dari setiap kegiatan pada jalur kritis ialah sebesar 3,73.

##### 6). Nilai Probabilitas

Dalam mencari durasi optimal pembangunan rumah tinggal di kecamatan Rantau Pulung Kutai Timur, maka kita perlu mencari nilai probabilitas mencapai target jadwal. Setelah menentukan variansi dan deviasi standar dari perhitungan PERT selanjutnya mencari perhitungan nilai probabilitas yang dapat di peroleh dari Persamaan berikut :

$$Z = \frac{X-\mu}{\sigma} \quad (8)$$

dengan jumlah waktu penyelesaian proyek yang sudah di harapkan dari hasil wawancara dengan pihak pelaksana proyek ( $X$ ), jumlah waktu yang diharapkan ( $\mu$ ) dengan perhitungan tiga estimasi waktu dan perhitungan waktu deviasi standar ( $\sigma$ ) jalur kritis dengan perhitungan berikut.

Diketahui bahwa  $X = 45$  hari dan  $\mu = 40$  hari. Didapatkan pula  $\sigma = 4,3$  sehingga

$$Z = \frac{45 - 40}{4,3} = 1,16$$

Hasil dari perhitungan mencari nilai probabilitas dari seluruh kegiatan kritis proyek ialah  $Z = 1,16$ . Pada tabel distribusi normal,  $Z = 1,16$  dapat dikonversikan sehingga didapat 0,8770 yaitu 87,70 %. Jadi, besarnya probabilitas keseluruhan proyek kegiatan kritis dapat diselesaikan dalam waktu 45 hari adalah sebesar 87,70%. Kemungkinan ini besar sekali sehingga penetapan waktu penyelesaian rumah tinggal di kecamatan Rantau Pulung Kutai Timur adalah 5 hari lebih lama dari waktu penyelesaian jalur kritis.

## **PENUTUP**

### 1). Kesimpulan

Hasil dari pengoptimalan biaya dan waktu pelaksanaan pembangunan rumah tinggal di kecamatan Rantau Pulung Kutai Timur menggunakan *Critical Path Method* (CPM) dan *Program Evaluation and Review Technique* (PERT) dapat disimpulkan Susunan jaringan kerja pada proyek pembangunan rumah tinggal di Kecamatan Rantau Pulung Kutai Timur menggunakan metode CPM terdiri dari 16 kegiatan dari awal proyek hingga akhir proyek dengan jalur kritis yaitu  $A \rightarrow B \rightarrow F \rightarrow K \rightarrow N \rightarrow O \rightarrow P$ . Hasil perhitungan diperoleh durasi minimum penyelesaian proyek rumah tinggal di kecamatan Rantau Pulung Kutai Timur setelah dilakukan optimalisasi dengan metode PERT adalah selama 40 hari dan optimalisasi dengan metode CPM adalah selama 39 hari percepatan 14 hari dengan biaya upah pekerja sebesar Rp34.930.000,- atau bertambah sebesar Rp805.000,- dari biaya upah pekerja pada waktu normal.

### 2). Saran

Bagi pembaca yang ingin meneliti tentang permasalahan ini dapat mengembangkan dengan membahas sumber daya yang lain seperti biaya material, biaya tidak langsung (transportasi dan lain lain).

## **DAFTAR PUSTAKA**

- [1] Trihudiyatmanto, M. (2018). *Riset Operasional (Operations Research) & Penyelesaian menggunakan Software WinQSB*. Pekalongan: PT.Nasya Expanding Management.
- [2] Syaifuddin, Dedy Takdir. (2011). *Riset Operasi Aplikasi Quantitative Analysis For Management*. Malang: CV Citra Malang.
- [3] Meflinda, A. dan Mahyarni. (2011). *Operations Research (Riset Operasi)*. Pekanbaru: UNRI PRESS.
- [4] Wijaya, Andi. (2012). *Pengantar Riset Operasi*. Jakarta: Mitra Wacana Media.
- [5] Rani, Hafnidar. A. (2016). *Manajemen Proyek Konstruksi*. Yogyakarta: CV Budi Utama.
- [6] Siswanto, Agus B & M.Afif Salim. (2019). *Manajemen Proyek*. Jawa Tengah: CV Pilar Nusantara.
- [7] Herjanto, Eddy. (2007). *Manajemen Operasi Edisi ketiga*. Jakarta: PT.Gramedia Widiasarana Indonesia.
- [8] Siswanto. (2006). *Operations Research Jilid 2*. Bogor: Penerbit Erlangga.

## Penerapan Algoritma Dijkstra dan Algoritma Greedy Pada Optimasi Jalur Evakuasi Banjir

Muliya Wiladi<sup>1</sup>, Wasono<sup>2\*</sup>, Asmaidi<sup>3</sup>

<sup>1,2,3</sup> Universitas Mulawarman

Dikirim: Oktober 2022;

Diterima: November 2022; Dipublikasi: Maret 2023

Alamat Email Korespondensi: [wasono@fmipa.unmul.ac.id](mailto:wasono@fmipa.unmul.ac.id)

### Abstrak

Penyelamatan dan evakuasi merupakan hal yang harus dilakukan dengan segera setelah terjadinya sebuah bencana agar dapat mengurangi jumlah korban dan mengurangi dampak buruk yang ditimbulkan. Salah satu bencana yang sering terjadi di Kota Samarinda yaitu bencana banjir. Pada penelitian ini mengkaji bagaimana pengoptimasian jalur evakuasi warga yang terdampak banjir di Kelurahan Sempaja Timur, Kota Samarinda menggunakan algoritma Dijkstra dan algoritma Greedy. Algoritma Dijkstra merupakan algoritma yang dapat digunakan dalam menentukan jalur terpendek. Algoritma Greedy adalah algoritma yang membuat pilihan terbaik (nilai tertinggi) pada setiap langkahnya. Berdasarkan hasil penelitian, didapatkan 11 lintasan optimal yang dapat digunakan pada proses evakuasi di Jl. Terong dan Jl. Terong Pipit, Kelurahan Sempaja Timur, Kota Samarinda. Sebelas lintasan optimal yang didapatkan merupakan urutan evakuasi banjir yang dapat digunakan agar seluruh warga dapat dievakuasi dan proses evakuasi berjalan optimal.

### Kata Kunci:

*Algoritma Dijkstra, Algoritma Greedy, Masalah Knapsack, Masalah Lintasan Terpendek, Optimasi*

## PENDAHULUAN

Optimasi merupakan suatu proses yang memiliki tujuan mencari nilai minimum atau maksimum suatu fungsi dengan memperhatikan kendala-kendala yang ada. Masalah optimasi adalah masalah besaran tertentu atau disebut juga fungsi objektif (*objective*) yang dimaksimalkan atau diminimumkan dengan bergantung pada sejumlah berhingga variabel masukan (*input variables*). Variabel-variabel ini dapat saling bergantung melalui satu atau lebih kendala (*constrains*) atau juga tidak saling bergantung [7]. Optimasi dalam hal waktu dan beberapa penghematan di bidang lain dapat dilakukan dengan menentukan jalur terpendak. Pekerjaan dapat lebih efektif, cepat dan juga menghemat biaya dengan melewati jalur terpendek. Jalur terpendek merupakan jumlah nilai dari keseluruhan bentuk lintasan yang bernilai minimal [1].

Salah satu masalah optimasi kombinatorial yang juga banyak dipelajari adalah masalah knapsack 0-1. Masalah ini mempunyai tujuan untuk memaksimalkan nilai total dari seluruh objek yang dapat dimuat ke dalam knapsack (suatu wadah atau tas) dengan kendalanya yaitu memastikan total bobot objek kurang dari atau sama dengan kapasitas maksimal *knapsack*. Setiap objek tidak dapat dimasukkan secara tidak utuh atau dimasukkan lebih dari satu kali ke dalam *knapsack* [3]. Terdapat banyak cara yang dapat dilakukan untuk menyelesaikan masalah *knapsack*, salah satunya adalah Algoritma Greedy. Pemecahan masalah optimasi dengan Algoritma Greedy tidak selalu

mendapatkan solusi yang optimum. Prinsip utama dari algoritma ini adalah memilih sebanyak mungkin objek yang dapat diperoleh sekarang [6].

Algoritma Dijkstra merupakan algoritma yang dapat digunakan dalam menentukan jalur terpendek dari titik sumber menuju titik tujuan berdasarkan nilai pada setiap sisi. Nilai tersebut dapat berupa waktu, jarak, biaya maupun yang lainnya. Algoritma Dijkstra bekerja dengan cara melewati setiap titik yang terdapat pada graf dengan dimulai dari titik sumber. Algoritma ini memilih titik-titik terdekat dan menghitung total nilai semua sisi yang dilewati dengan dilakukan secara berulang [5].

Masalah *knapsack* dan masalah lintasan terpendek juga dapat dikerjakan secara bersamaan, seperti penelitian yang dilakukan oleh [10]. Masalah tersebut menjadi masalah kompleks yang menggabungkan dua permasalahan yaitu *knapsack* dan lintasan terpendek pada graf berbobot. Penelitian tersebut menggunakan Algoritma Dijkstra untuk mencari lintasan terpendek dan algoritma Aproksimasi Greedy Dantzig untuk memilih objek pada lintasan terpilih. Penelitian yang dilakukan oleh [10] mendapati hasil yang belum optimal pada kasus dengan pembatasan bobot objek.

Banjir merupakan masalah lingkungan yang biasa terjadi di sejumlah kawasan di Samarinda, Kalimantan Timur, salah satunya adalah Kelurahan Sempaja Timur. Pada tanggal 9 Juni 2019, Kelurahan Sempaja Timur merupakan daerah yang paling parah dilanda banjir. Jumlah penduduk yang terkena bencana di Kelurahan Sempaja Timur, Kecamatan Samarinda Utara adalah sebanyak 2.327 jiwa. Penyebab banjir di Kota Samarinda dikarenakan tingginya intensitas hujan. Intensitas hujan yang tinggi menyebabkan meluapnya Sungai Mahakam dan Sungai Karang Mumus sehingga ketinggian banjir semakin naik. Akibat ketinggian banjir yang naik, tim SAR gabungan harus mengevakuasi sejumlah warga untuk mengungsi [2]. Menurut Peraturan Walikota Samarinda Nomor 6 Tahun 2014 [11], Tanggap darurat bencana adalah serangkaian kegiatan yang dilakukan dengan segera pada saat kejadian bencana untuk menangani dampak buruk yang ditimbulkan, yang meliputi kegiatan penyelamatan dan evakuasi korban, harta benda, pemenuhan kebutuhan dasar, perlindungan, pengurusan pengungsi, penyelamatan, serta pemulihan prasarana dan sarana.

Berdasarkan uraian di atas, penelitian ini akan mengkaji tentang optimasi jalur terpendek yang diselesaikan menggunakan penggabungan Algoritma Dijkstra dan Algoritma Greedy dengan studi kasus di Kelurahan Sempaja Timur, Kota Samarinda.

## **TINJAUAN PUSTAKA**

### **1. Optimasi**

Optimasi merupakan suatu proses maksimasi atau minimasi dengan mencari kondisi yang optimum. Keadaan optimum yang memberikan nilai maksimum (maksimasi) biasanya berkaitan dengan masalah keuntungan. Keadaan optimum yang memberikan nilai minimum (minimasi) biasanya berkaitan dengan masalah pengeluaran. Hal-hal penting dalam pembelajaran optimasi yaitu meliputi fungsi objektif, variabel keputusan (*decision variables*), kendala (*constraints*), dan pernyataan matematika yang menyatakan daerah nilai minimum atau maksimum akan dicari [9].

### **2. Graf**

Graf  $G$  dinotasikan sebagai  $G = (V, E)$  dengan  $V$  merupakan himpunan tidak kosong dari titik-titik (*vertices* atau *node*) dan  $E$  merupakan himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang titik.  $V$  dinyatakan dengan himpunan tak kosong, sedangkan  $E$  boleh kosong, oleh karena itu sebuah graf dimungkinkan untuk tidak

mempunyai sisi tetapi harus memiliki minimal satu titik. Graf yang tidak terdapat sisi dan memiliki satu buah titik dinamakan graf trivial [4].

### 3. Masalah Lintasan Terpendek

Salah satu masalah optimasi di dalam graf adalah masalah pencarian lintasan terpendek. Masalah lintasan terpendek ini menggunakan graf berbobot yang setiap sisinya memiliki satuan nilai atau bobot. Nilai atau bobot tersebut dapat menyatakan jarak antar kota, waktu pengiriman, ongkos pembangunan, dan sebagainya. Asumsi yang digunakan pada masalah lintasan terpendek adalah semua bobot bernilai positif [4].

### 4. Masalah *Knapsack*

Masalah *knapsack* 0-1 adalah salah satu masalah optimasi kombinatorial yang banyak dipelajari. Masalah ini mempunyai tujuan untuk memaksimalkan nilai total dari seluruh objek yang dapat dimuat ke dalam *knapsack* dengan kendalanya yaitu memastikan total bobot objek kurang dari atau sama dengan kapasitas maksimal *knapsack*. Setiap objek tidak dapat dimasukkan secara tidak utuh atau dimasukkan lebih dari satu kali ke dalam *knapsack*. Masalah *knapsack* 0-1 dapat dirumuskan sebagai berikut [3]:

Fungsi tujuan:

$$\sum_{i=1}^m q_i y_i, \quad (1)$$

Keterangan:

$m$  : Jumlah titik

$i$  : Indeks objek

$q_i$  : Nilai objek ke- $i$

$y_i$  : Menunjukkan objek dipilih atau tidak dengan  $y_i$  bernilai 0 atau 1

Fungsi kendala:

$$\sum_{i=1}^m w_i y_i \leq W, \quad (2)$$

Keterangan:

$m$  : Jumlah titik

$i$  : Indeks objek

$w_i$  : Bobot objek ke- $i$

$y_i$  : Menunjukkan objek dipilih atau tidak dengan  $y_i$  bernilai 0 atau 1

$W$  : Kapasitas *knapsack*

### 5. Algoritma

Algoritma merupakan langkah-langkah berurutan untuk melakukan komputasi atau menyelesaikan sebuah masalah. Algoritma merupakan istilah yang diambil dari nama seorang ahli matematika yang hidup sekitar abad ke-9. Awalnya, kata *algorism* merupakan aturan saat melakukan aritmatika menggunakan notasi desimal, namun pada abad ke-18 istilah *algorism* berubah menjadi kata *algorithm* atau dalam bahasa Indonesia disebut algoritma. Seiring meningkatnya minat dalam mesin komputasi, konsep algoritma diberikan arti yang lebih umum, tidak hanya prosedur dalam melakukan aritmatika tetapi juga memuat semua prosedur pasti untuk menyelesaikan masalah [8].

#### 5.1 Algoritma Dijkstra

Algoritma lintasan terpendek yang terkenal salah satunya adalah Algoritma Dijkstra (sesuai dengan nama penemunya, Edsger W. Dijkstra). Algoritma Dijkstra diterapkan untuk mencari lintasan terpendek pada graf berarah. Namun, algoritma ini juga benar untuk graf tak-berarah [4].

Secara singkat Algoritma Dijkstra dijelaskan sebagai berikut [5]:

1. Inisialisasi titik ( $v$ ).
2. Inisialisasi jarak antar titik ( $e$ ).
3. Menentukan titik awal dan titik tujuan.
4. Memberi label permanen 0 (nol) pada titik awal dan label sementara  $\infty$  (tak hingga) pada titik lainnya.
5. Memberi label sementara, yaitu  $\min \{label\ lama\ v_i, (label\ lama\ v_i + e_i)\}$  untuk setiap titik  $v_t$  yang belum mendapat label permanen.
6. Mencari nilai minimum dari semua titik yang masih berlabel sementara.
7. Merubah titik minimum yang berlabel sementara menjadi titik dengan label permanen, jika lebih dari satu titik maka dipilih sembarang titik.
8. Mengulangi langkah 5 sampai 7 hingga semua titik mendapat label permanen.
9. Menyimpan hasil perhitungan.
10. Menampilkan hasil pencarian.

## 5.2 Algoritma Greedy

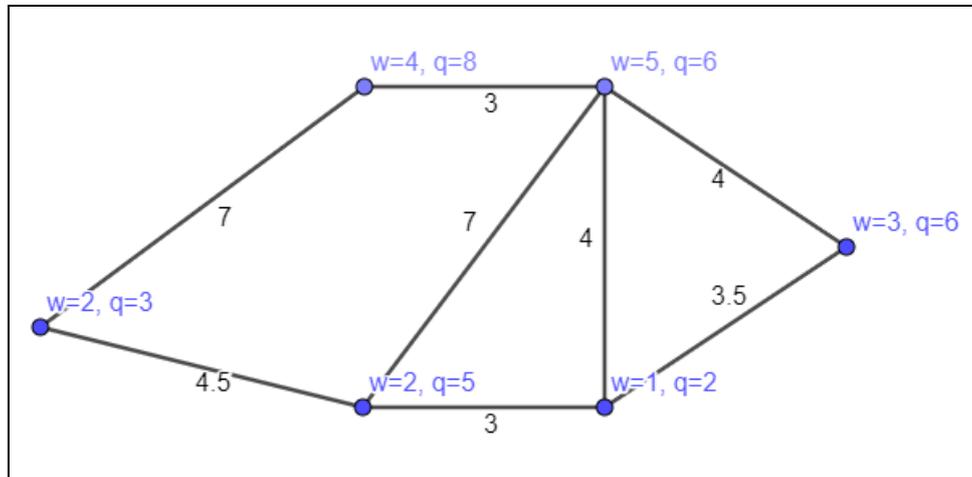
Solusi optimasi biasanya merupakan pendekatan yang paling sederhana. Salah satu pendekatan ini adalah pendekatan yang memilih pilihan terbaik pada setiap langkah, daripada mempertimbangkan semua urutan langkah yang mungkin mengarah pada solusi optimal. Algoritma yang membuat pilihan terbaik (nilai tertinggi) pada setiap langkahnya disebut Algoritma Greedy. Solusi yang didapat dari Algoritma Greedy perlu ditentukan apakah solusi tersebut merupakan solusi optimal [8].

Terdapat tiga jenis Algoritma Greedy yang dapat digunakan pada penyelesaian masalah *knapsack*, yaitu [6]:

1. *Greedy By Weight*  
Pada setiap langkah, objek yang dipilih adalah objek yang mempunyai berat teringan. Langkah diambil dengan tujuan memaksimalkan keuntungan dengan memasukkan sebanyak mungkin objek ke dalam knapsack. Pertama yang dilakukan adalah mengurutkan secara menaik objek-objek berdasarkan beratnya. Kemudian dipilih satu demi satu objek yang dapat ditampung oleh knapsack hingga knapsack penuh atau tidak dapat lagi memasukkan objek lainnya ke dalam knapsack.
2. *Greedy By Profit*  
Pada setiap langkah, objek yang dipilih adalah objek dengan keuntungan terbesar. Langkah diambil dengan tujuan memaksimalkan keuntungan dengan memasukkan objek yang paling menguntungkan. Pertama yang dilakukan adalah mengurutkan secara menaik objek-objek berdasarkan profit-nya. Kemudian dipilih satu demi satu objek yang dapat ditampung oleh knapsack hingga knapsack penuh atau tidak dapat lagi memasukkan objek lainnya ke dalam knapsack.
3. *Greedy By Density*  
Pada setiap langkah, objek yang dipilih adalah objek yang mempunyai nilai perbandingan terbesar antara profit dan berat. Langkah diambil dengan tujuan memaksimalkan keuntungan dengan memasukkan objek yang mempunyai density terbesar. Pertama yang dilakukan adalah mengurutkan secara menaik objek-objek berdasarkan density-nya. Kemudian dipilih satu demi satu objek yang dapat ditampung oleh knapsack hingga knapsack penuh atau tidak dapat lagi memasukkan objek lainnya ke dalam knapsack.

## 6. Knapsack dan Lintasan Terpendek

Graf berbobot dinotasikan sebagai  $G^* = (V, E)$  dengan  $V$  merupakan himpunan tidak kosong dari titik-titik (*vertices* atau *node*) dan  $E$  merupakan himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang titik. Setiap titik pada graf tersebut diberikan bobot ( $w_i$ ) dan nilai ( $q_i$ ).  $w_i$  dan  $q_i$  pada titik melambangkan bobot atau berat dan nilai atau profit dari sebuah objek. Pada graf tersebut juga diberikan sebuah batas maksimum bobot *Knapsack* yang dilambangkan dengan  $m$  [10].



Gambar 1. Graf berbobot  $G^*$

Graf  $G^*$  yang sudah dijelaskan sebelumnya dicari nilai maksimum objek-objek pada graf tersebut dari lintasan terpendek antara dua titik yang ditentukan (titik awal dan titik tujuan) dengan sebuah batas bobot *Knapsack*. Langkah pertama yaitu menentukan lintasan terpendek antara dua titik yang telah ditentukan. Langkah kedua yaitu mempertimbangkan bobot dan nilai objek pada titik pada hasil sub-graf dari lintasan terpendek terpilih [10]. Penelitian yang dilakukan oleh Voloch [10] menentukan lintasan terpendek antara dua titik dengan menggunakan Algoritma Dijkstra dan memilih objek atau titik dari lintasan terpendek terpilih menggunakan algoritma Aproksimasi Greedy Dantzig.

## HASIL DAN PEMBAHASAN

### 1. Pembuatan Graf Berbobot

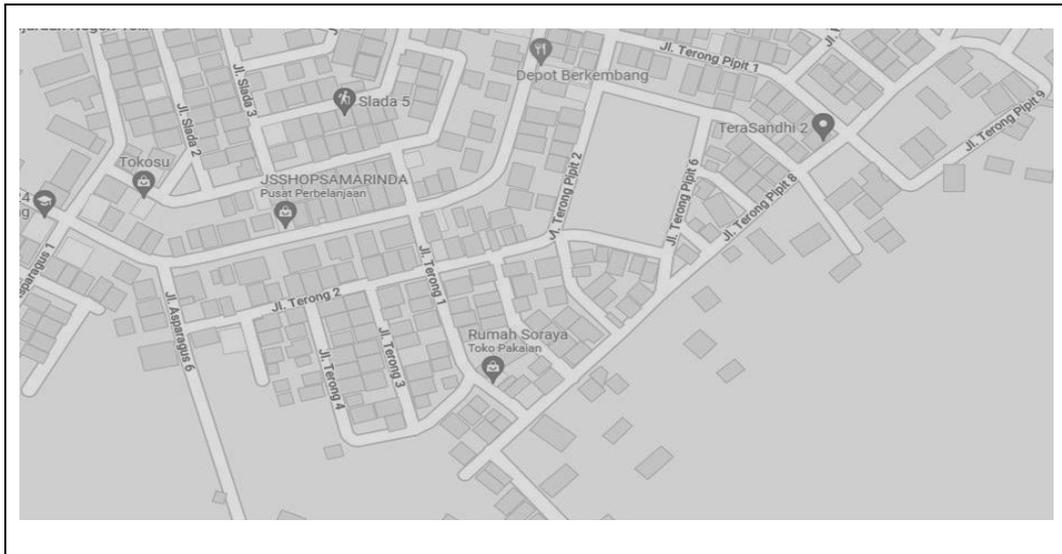
Graf berbobot yang dibuat menggunakan data rute dan data penduduk terdampak banjir pada tanggal 12 Juni 2019 di Kelurahan Sempaja Timur. Berikut tabel data nama jalan dan jumlah penduduk terdampak banjir di Kelurahan Sempaja Timur yang digunakan pada penelitian ini [2]:

Tabel 1. Data Nama Jalan dan Jumlah Penduduk

Nama Jalan	Jumlah Penduduk
Jl. Terong 1 s/d 6	435
Jl. Terong Pipit 1 s/d 12	434

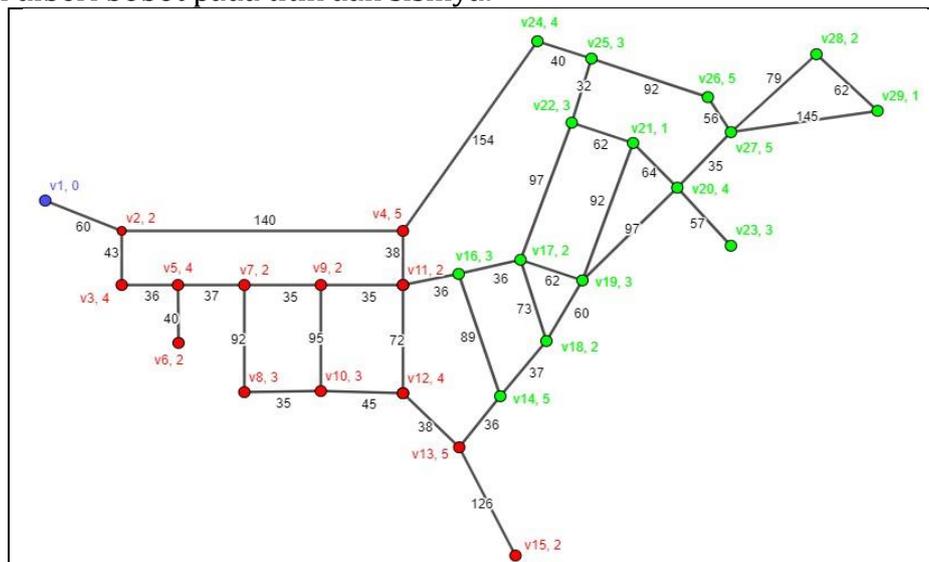
Sumber: Data Kementerian Pekerjaan Umum dan Perumahan Rakyat tahun 2019

Data nama jalan pada Tabel 1 dibuat graf dengan mengikuti rute dari *Google Maps*. Titik pada graf yang dibuat menunjukkan persimpangan jalan dan lokasi titik penjemputan warga. Berikut gambaran lokasi jalan pada Tabel 1 dari *Google Maps*:



Gambar 2. Rute Jl. Terong 1 s/d 6 dan Jl. Terong Pipit 1 s/d 12 pada *Google Maps*

Graf yang telah dibuat, diberi bobot pada setiap sisi dan titiknya. Bobot pada sisi menunjukkan jarak jalan antar persimpangan dalam satuan meter dan nilai pada titik menunjukkan jumlah warga. Sebagai contoh, label  $v_{24}$ , 4 pada titik memiliki arti yaitu titik  $v_{24}$  terdapat jumlah warga sebanyak 4 orang dan label 154 pada sisi memiliki arti yaitu terdapat jarak yang menghubungkan 2 titik dengan besaran 154 meter. Jumlah warga yang digunakan adalah 10% dari 435 warga terdampak pada Jl. Terong yaitu 43 warga dan 10% dari 434 warga terdampak pada Jl. Terong Pipit yaitu 43 warga. Titik berwarna biru adalah titik awal yang merupakan persimpangan pertama sebelum memasuki Jl. Terong. Titik berwarna merah merupakan titik yang berada di Jl. Terong. Titik berwarna hijau merupakan titik yang berada di Jl. Terong Pipit. Empat puluh tiga warga pada Jl. Terong dan Jl. Terong Pipit dibagi secara random sebanyak jumlah titik pada jalan tersebut yaitu 13 titik pada Jl. Terong dan 15 titik pada Jl. Terong Pipit. Hasil random yang didapatkan digunakan sebagai bobot titik pada Jl. Terong dan Jl. Terong Pipit. Berikut graf yang telah diberi bobot pada titik dan sisinya:



Gambar 3. Graf berbobot G

## 2. Pembuatan Matriks Ketetangaan

Graf berbobot pada Gambar 4.2 direpresentasikan dalam bentuk matriks ketetangaan berbobot dengan notasi  $A = a_{ij}; i=1,2,3,\dots,29; j=1,2,3,\dots,29$ . Berikut tabel titik terhubung dan bobot sisi yang menjadi nilai-nilai elemen pada matriks A:

Tabel 2. Titik Terhubung dan Bobot Sisi

No	Titik Terhubung	Bobot Sisi
1.	$v_1 v_2$	60
2.	$v_2 v_3$	43
3.	$v_2 v_4$	140
4.	$v_3 v_5$	36
5.	$v_4 v_{24}$	154
6.	$v_4 v_{11}$	38
7.	$v_5 v_6$	40
8.	$v_5 v_7$	37
9.	$v_7 v_8$	92
10.	$v_7 v_9$	35
11.	$v_8 v_{10}$	35
12.	$v_9 v_{10}$	95
13.	$v_9 v_{11}$	35
14.	$v_{10} v_{12}$	45
15.	$v_{11} v_{12}$	72
16.	$v_{11} v_{16}$	36
17.	$v_{12} v_{13}$	38
18.	$v_{13} v_{14}$	36
19.	$v_{13} v_{15}$	126
20.	$v_{14} v_{16}$	89
21.	$v_{14} v_{18}$	37
22.	$v_{16} v_{17}$	36
23.	$v_{17} v_{18}$	73
24.	$v_{17} v_{22}$	97
25.	$v_{17} v_{19}$	62
26.	$v_{18} v_{19}$	60
28.	$v_{19} v_{21}$	92
29.	$v_{20} v_{21}$	64
30.	$v_{20} v_{23}$	57
31.	$v_{20} v_{27}$	35
32.	$v_{21} v_{22}$	62
33.	$v_{22} v_{25}$	32
34.	$v_{24} v_{25}$	40
35.	$v_{25} v_{26}$	92
36.	$v_{26} v_{27}$	56
37.	$v_{27} v_{28}$	79
38.	$v_{28} v_{29}$	62
39.	$v_{29} v_{27}$	145

Pada matriks A,  $a_{ij}$  bernilai 1 jika  $v_i$  terhubung atau bertetangga dengan  $v_j$ , sedangkan  $a_{ij}$  bernilai 0 saat  $v_i$  tidak terhubung dengan  $v_j$  atau ketika  $v_i$  sama dengan  $v_j$ . Berdasarkan Tabel 2, matriks A direpresentasikan sebagai berikut.



- terdapat nilai yang sama, diambil  $d(v_i)$  dengan nilai  $s(w) + w_i$  terbesar.
- j. Merubah nilai  $d(v_i)$  terpilih menjadi  $d(V)$  dan memberi label 1 pada  $v_i$  terpilih.
  - k. Merubah nilai  $w_i$  dari setiap titik yang dilewati pada lintasan terpilih dengan nilai 0 dan mengembalikan nilai  $w_i$  yang tidak dilewati pada lintasan terpilih.
  - l. Mengulangi langkah h sampai k hingga tidak ada lagi  $d(v_i)$  yang memenuhi langkah i.
  - m. Memilih nilai  $d(v_i)$  maksimum dari semua titik dan merubah nilai  $w_i$  dari setiap titik yang dilewati pada lintasan terpilih secara permanen dengan nilai 0 (nol).
  - n. Menyimpan lintasan terpilih.
  - o. Mengembalikan nilai  $w_i$  yang tidak terpilih.
  - p. Mengulangi langkah e sampai o hingga semua  $w_i$  bernilai 0 (nol).

Berikut perhitungan penentuan lintasan optimal menggunakan Algoritma Dijkstra dan Algoritma Greedy:

1. Perhitungan 1

Pada penentuan lintasan optimal pertama yang selanjutnya disebut perhitungan 1, bobot di setiap titik graf G belum ada yang bernilai 0 selain titik awal. Titik yang menjadi titik awal pada perhitungan ini adalah titik  $v_1$ , sehingga titik  $v_1$  diberi label 1. Nilai densitas lintasan menuju titik  $v_i$  ( $d(v_i)$ ) untuk semua titik diberi nilai  $-\infty$ , nilai densitas lintasan terpilih ( $d(V)$ ) =  $-\infty$ , total bobot lintasan terpilih ( $s(w)$ ) = 0, dan total jarak lintasan terpilih ( $s(e)$ ) = 0.

Iterasi 1:

Pada iterasi 1, titik  $v_1$  hanya bertetangga dengan  $v_2$ .

$$\begin{aligned} d(v_2) \text{ baru} &= \text{maks} \left\{ d(v_2) \text{ lama}; \frac{(s(w)+w_2)}{(s(e)+e_{2,1})} \right\} \\ &= \text{maks} \left\{ -\infty; \frac{(0+2)}{(0+60)} \right\} \\ &= \text{maks} \{ -\infty; 0,3333 \} \\ &= 0,3333 \end{aligned}$$

Berdasarkan Iterasi 1 diperoleh bahwa nilai maksimum  $d(v_i)$  dari semua titik berlabel 0 pada iterasi 1 adalah  $d(v_2) = 0,3333$ , maka nilai  $d(V)$  menjadi 0,3333, dimana  $s(w) = 2$  dan  $s(e) = 60$  dan terdapat nilai  $d(v_i)$  dengan label 0 yang bernilai selain  $-\infty$ , maka dilanjut dengan iterasi 2. Lintasan terpilih untuk iterasi 2 yaitu  $(v_1, v_2)$ . Nilai  $w_1$  dan  $w_2$  dirubah menjadi 0 dan  $v_2$  mendapat label 1.

Iterasi 2:

Pada iterasi 2, titik yang bertetangga dengan  $v_2$  adalah  $v_1$ ,  $v_3$  dan  $v_4$ .

$$\begin{aligned} d(v_1) \text{ baru} &= \text{maks} \left\{ d(v_1) \text{ lama}; \frac{(s(w)+w_1)}{(s(e)+e_{1,2})} \right\} \\ &= \text{maks} \left\{ -\infty; \frac{(2+0)}{(60+60)} \right\} \\ &= \text{maks} \{ -\infty; 0,01667 \} \\ &= 0,01667 \end{aligned}$$

$$\begin{aligned} d(v_3) \text{ baru} &= \text{maks} \left\{ d(v_3) \text{ lama}; \frac{(s(w)+w_3)}{(s(e)+e_{3,2})} \right\} \\ &= \text{maks} \left\{ -\infty; \frac{(2+4)}{(60+43)} \right\} \\ &= \text{maks} \{ -\infty; 0,05825 \} \\ &= 0,05825 \end{aligned}$$

$$d(v_4) \text{ baru} = \text{maks} \left\{ d(v_4) \text{ lama}; \frac{(s(w)+w_4)}{(s(e)+e_{4,2})} \right\}$$

$$\begin{aligned}
 &= \text{maks} \left\{ -\infty; \frac{(2+5)}{(60+140)} \right\} \\
 &= \text{maks} \{ -\infty; 0,035 \} \\
 &= 0,035
 \end{aligned}$$

Berdasarkan iterasi 2 diperoleh bahwa nilai maksimum  $d(v_i)$  dari semua titik berlabel 0 pada iterasi 2 adalah  $d(v_3) = 0,0582$ , maka nilai  $d(V)$  menjadi 0,0582, dimana  $s(w) = 6$  dan  $s(e) = 103$  dan terdapat nilai  $d(v_i)$  dengan label 0 yang bernilai selain  $-\infty$ , maka dilanjut dengan iterasi 3. Lintasan terpilih untuk iterasi 3 yaitu  $(v_1, v_2, v_3)$ . Nilai  $w_1, w_2$  dan  $w_3$  dirubah menjadi 0 dan  $v_3$  mendapat label 1.

Iterasi 3:

Pada iterasi 3, titik yang bertetangga dengan  $v_3$  adalah  $v_2$  dan  $v_5$

$$\begin{aligned}
 d(v_2) \text{ baru} &= \text{maks} \left\{ d(v_2) \text{ lama}; \frac{(s(w)+w_2)}{(s(e)+e_{2,3})} \right\} \\
 &= \text{maks} \left\{ 0,03333; \frac{(6+0)}{(103+43)} \right\} \\
 &= \text{maks} \{ 0,03333; 0,0411 \} \\
 &= 0,0411
 \end{aligned}$$

$$\begin{aligned}
 d(v_5) \text{ baru} &= \text{maks} \left\{ d(v_5) \text{ lama}; \frac{(s(w)+w_5)}{(s(e)+e_{5,3})} \right\} \\
 &= \text{maks} \left\{ -\infty; \frac{(6+4)}{(103+36)} \right\} \\
 &= \text{maks} \{ -\infty; 0,07194 \} \\
 &= 0,07194
 \end{aligned}$$

Berdasarkan iterasi 3 diperoleh bahwa nilai maksimum  $d(v_i)$  dari semua titik berlabel 0 pada iterasi 3 adalah  $d(v_5) = 0,07194$ , maka nilai  $d(V)$  menjadi 0,07194, dimana  $s(w) = 10$  dan  $s(e) = 139$  dan terdapat nilai  $d(v_i)$  dengan label 0 yang bernilai selain  $-\infty$ , maka dilanjut dengan iterasi 4. Lintasan terpilih untuk iterasi 3 yaitu  $(v_1, v_2, v_3, v_5)$ . Nilai  $w_1, w_2, w_3$  dan  $w_5$  dirubah menjadi 0 dan  $v_5$  mendapat label 1.

Iterasi 4:

Pada iterasi 4, titik yang bertetangga dengan  $v_5$  adalah  $v_3, v_6$  dan  $v_7$ . Karena  $(s(w) + w_3) = 10 + 4 \geq W = 10$ ,  $(s(w) + w_6) = 10 + 2 \geq W = 10$  dan  $(s(w) + w_7) = 10 + 2 \geq W = 10$ , maka nilai  $d(v_3), d(v_6)$  dan  $d(v_7)$  tidak berubah.

Berdasarkan iterasi 4 diperoleh bahwa nilai maksimum  $d(v_i)$  dari semua titik berlabel 0 pada iterasi 4 adalah  $d(v_4) = 0,035$ , maka nilai  $d(V)$  menjadi 0,035, dimana  $s(w) = 7$  dan  $s(e) = 200$  dan terdapat nilai  $d(v_i)$  dengan label 0 yang bernilai selain  $-\infty$ , maka dilanjut dengan iterasi 5. Lintasan terpilih untuk iterasi 5 yaitu  $(v_1, v_2, v_4)$ . Nilai  $w_1, w_2$  dan  $w_4$  dirubah menjadi 0 dan  $v_4$  mendapat label 1.

Iterasi 5:

Pada iterasi 5, titik yang bertetangga dengan  $v_4$  adalah  $v_2, v_{11}$  dan  $v_{24}$ . Karena  $(s(w) + w_{24}) = 7 + 4 \geq W = 10$ , maka nilai  $d(v_{24})$  tidak berubah.

$$\begin{aligned}
 d(v_2) \text{ baru} &= \text{maks} \left\{ d(v_2) \text{ lama}; \frac{(s(w)+w_2)}{(s(e)+e_{2,4})} \right\} \\
 &= \text{maks} \left\{ 0,0411; \frac{(7+2)}{(200+140)} \right\} \\
 &= \text{maks} \{ 0,0411; 0,02647 \} \\
 &= 0,0411
 \end{aligned}$$

$$\begin{aligned}
 d(v_{11}) \text{ baru} &= \text{maks} \left\{ d(v_{11}) \text{ lama}; \frac{(s(w)+w_{11})}{(s(e)+e_{11,4})} \right\} \\
 &= \text{maks} \left\{ -\infty; \frac{(7+2)}{(200+38)} \right\} \\
 &= \text{maks} \{ -\infty; 0,03781 \} \\
 &= 0,03781
 \end{aligned}$$

Berdasarkan Tabel iterasi 5 diperoleh bahwa nilai maksimum  $d(v_i)$  dari semua titik berlabel 0 pada iterasi 5 adalah  $d(v_{11}) = 0,03781$ , maka nilai  $d(V)$  menjadi 0,03781, dimana  $s(w) = 9$  dan  $s(e) = 238$  dan terdapat nilai  $d(v_i)$  dengan label 0 yang bernilai selain  $-\infty$ , maka dilanjut dengan iterasi 6. Lintasan terpilih untuk iterasi 6 yaitu  $(v_1, v_2, v_4, v_{11})$ . Nilai  $w_1, w_2, w_4$  dan  $w_{11}$  dirubah menjadi 0 dan  $v_{11}$  mendapat label 1.

Iterasi 6:

Pada iterasi 6, titik yang bertetangga dengan  $v_{11}$  adalah  $v_4, v_9, v_{12}$  dan  $v_{16}$ . Karena  $(s(w) + w_4) = 9 + 5 \geq W = 10$ ,  $(s(w) + w_9) = 9 + 2 \geq W = 10$ ,  $(s(w) + w_{12}) = 9 + 4 \geq W = 10$  dan  $(s(w) + w_{16}) = 9 + 3 \geq W = 10$ , maka nilai  $d(v_4), d(v_9), d(v_{12})$  dan  $d(v_{16})$  tidak berubah.

Pada perhitungan 1, iterasi terakhir yaitu iterasi ke-6. Ketika iterasi ke-6 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_8) = 0,07194$ , dimana  $s(w) = 10$  dan  $s(e) = 139$ . Lintasan optimal terpilih pada perhitungan pertama yaitu  $(v_1, v_2, v_3, v_5)$  dengan jarak tempuh lintasan adalah 139 meter dan total warga yang dievakuasi yaitu 10 orang. Nilai  $w_1, w_2, w_3, w_5$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_1, v_2, v_3, v_5$  telah dievakuasi.

## 2. Perhitungan 2

Pada penentuan lintasan optimal pertama yang selanjutnya disebut perhitungan 1, iterasi terakhir yaitu iterasi ke-13. Ketika iterasi ke-13 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{16}) = 0,0365$ , dimana  $s(w) = 10$  dan  $s(e) = 274$ . Lintasan optimal terpilih pada perhitungan kedua yaitu  $(v_1, v_2, v_4, v_{11}, v_{16})$  dengan jarak tempuh 274 meter dan total warga yang dievakuasi yaitu 10. Nilai  $w_4, w_{11}, w_{16}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_4, v_{11}, v_{16}$  telah dievakuasi.

## 3. Perhitungan 3

Pada penentuan lintasan optimal ketiga yang selanjutnya disebut perhitungan 3, iterasi terakhir yaitu iterasi ke-21. Ketika iterasi ke-21 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_8) = 0,02933$ , dimana  $s(w) = 10$  dan  $s(e) = 341$ . Lintasan optimal terpilih pada perhitungan ketiga yaitu  $(v_1, v_2, v_3, v_5, v_7, v_9, v_{10}, v_8)$  dengan jarak tempuh 341 meter dan total warga yang dievakuasi yaitu 10. Nilai  $w_7, w_9, w_{10}, w_8$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_7, v_9, v_{10}, v_8$  telah dievakuasi.

## 4. Perhitungan 4

Pada penentuan lintasan optimal keempat yang selanjutnya disebut perhitungan 4, iterasi terakhir yaitu iterasi ke-22. Ketika iterasi ke-22 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{22}) = 0,02347$ , dimana  $s(w) = 10$  dan  $s(e) = 426$ .

## 5. Perhitungan 5

Pada penentuan lintasan optimal kelima yang selanjutnya disebut perhitungan 5, iterasi terakhir yaitu iterasi ke-22. Ketika iterasi ke-22 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{13}) = 0,02332$ , dimana  $s(w) = 9$  dan  $s(e) = 386$ . Lintasan

optimal terpilih pada perhitungan kelima yaitu  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13})$  dengan jarak tempuh 386 meter dan total warga yang dievakuasi yaitu 9. Nilai  $w_{12}, w_{13}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{12}, v_{13}$  telah dievakuasi.

#### 6. Perhitungan 6

Pada penentuan lintasan optimal keenam yang selanjutnya disebut perhitungan 6, iterasi terakhir yaitu iterasi ke-23. Ketika iterasi ke-23 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{19}) = 0,01927$ , dimana  $s(w) = 10$  dan  $s(e) = 519$ . Lintasan optimal terpilih pada perhitungan keenam yaitu  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{14}, v_{18}, v_{19})$  dengan jarak tempuh 519 meter dan total warga yang dievakuasi yaitu 10 nilai  $w_{14}, w_{18}, w_{19}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{14}, v_{18}, v_{19}$  telah dievakuasi.

#### 7. Perhitungan 7

Pada penentuan lintasan optimal ketujuh yang selanjutnya disebut perhitungan 7, iterasi terakhir yaitu iterasi ke-25. Ketika iterasi ke-25 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{23}) = 0,01711$ , dimana  $s(w) = 9$  dan  $s(e) = 526$ . Lintasan optimal terpilih pada perhitungan ketujuh yaitu  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{20}, v_{23})$  dengan jarak tempuh 526 meter dan total warga yang dievakuasi yaitu 9. Nilai  $w_{17}, w_{20}, w_{23}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{17}, v_{20}, v_{23}$  telah dievakuasi.

#### 8. Perhitungan 8

Pada penentuan lintasan optimal kedelapan yang selanjutnya disebut perhitungan 8, iterasi terakhir yaitu iterasi ke-29. Ketika iterasi ke-29 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{29}) = 0,01278$ , dimana  $s(w) = 9$  dan  $s(e) = 704$ . Lintasan optimal terpilih pada perhitungan kedelapan yaitu  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{21}, v_{20}, v_{27}, v_{28}, v_{29})$  dengan jarak tempuh 704 meter dan total warga yang dievakuasi yaitu 9. Nilai  $w_{21}, w_{27}, w_{28}, w_{29}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{21}, v_{27}, v_{28}, v_{29}$  telah dievakuasi.

#### 9. Perhitungan 9

Pada penentuan lintasan optimal kesembilan yang selanjutnya disebut perhitungan 9, iterasi terakhir yaitu iterasi ke-29. Ketika iterasi ke-29 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_6) = 0,01117$ , dimana  $s(w) = 2$  dan  $s(e) = 179$ . Lintasan optimal terpilih pada perhitungan kesembilan yaitu  $(v_1, v_2, v_3, v_5, v_6)$  dengan jarak tempuh 179 meter dan total warga yang dievakuasi yaitu 2. Nilai  $w_6$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_6$  telah dievakuasi.

#### 10. Perhitungan 10

Pada penentuan lintasan optimal kesepuluh yang selanjutnya disebut perhitungan 10, iterasi terakhir yaitu iterasi ke-29. Ketika iterasi ke-29 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai

lintasan paling optimal yaitu  $d(v_{26}) = 0,00942$ , dimana  $s(w) = 5$  dan  $s(e) = 531$ . Lintasan optimal terpilih pada perhitungan kesepuluh yaitu  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{22}, v_{25}, v_{26})$  dengan jarak tempuh 531 meter dan total warga yang dievakuasi yaitu 5. Nilai  $w_{26}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{26}$  telah dievakuasi.

### 11. Perhitungan 11

Pada penentuan lintasan optimal kesebelas yang selanjutnya disebut perhitungan 11, iterasi terakhir yaitu iterasi ke-29. Ketika iterasi ke-29 semua nilai  $d(v_i)$  dari titik berlabel 0 bernilai  $-\infty$ , sehingga dipilih nilai maksimum  $d(v_i)$  dari semua titik sebagai lintasan paling optimal yaitu  $d(v_{15}) = 0,00391$ , dimana  $s(w) = 2$  dan  $s(e) = 512$ . Lintasan optimal terpilih pada perhitungan kesebelas yaitu  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{15})$  dengan jarak tempuh 512 meter dan total warga yang dievakuasi yaitu 2. Nilai  $w_{15}$  dirubah menjadi 0 (nol) secara permanen yang menandakan bahwa warga pada titik  $v_{15}$  telah dievakuasi.

Perhitungan lintasan optimal menggunakan Algoritma Dijkstra dan Algoritma Greedy pada graf  $G$  yang telah dilakukan mendapatkan hasil yaitu 11 lintasan optimal. Lintasan-lintasan optimal yang diperoleh dari perhitungan 1 hingga perhitungan 11 menunjukkan urutan jalur evakuasi yang dapat digunakan saat evakuasi banjir di Jl. Terong dan Jl. Terong Pipit, Kelurahan Sempaja Timur, Kota Samarinda. Berikut ringkasan lintasan optimal yang diperoleh dari perhitungan 1 hingga perhitungan 11 yang dapat dilihat pada Tabel 3.

Tabel 3. Hasil Perhitungan Lintasan Optimal pada Graf  $G$

No.	Lintasan Optimal	Total Bobot	Jarak Lintasan
1.	$v_1, v_2, v_3, v_5$	10	139
2.	$v_1, v_2, v_4, v_{11}, v_{16}$	10	274
3.	$v_1, v_2, v_3, v_5, v_7, v_9, v_{10}, v_8$	10	341
4.	$v_1, v_2, v_4, v_{24}, v_{25}, v_{22}$	10	426
5.	$v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}$	9	386
6.	$v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{14}, v_{18}, v_{19}$	10	519
7.	$v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{20}, v_{23}$	9	526
8.	$v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{21}, v_{20}, v_{27}, v_{28}, v_{29}$	9	704
9.	$v_1, v_2, v_3, v_5, v_6$	2	179
10.	$v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{22}, v_{25}, v_{26}$	5	531
11.	$v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{15}$	2	512

### KESIMPULAN

Berdasarkan hasil dan pembahasan dari penelitian ini dapat disimpulkan bahwa penggabungan Algoritma Dijkstra dan Algoritma Greedy dapat digunakan pada kasus penentuan jalur evakuasi banjir di Jl. Terong dan Jl. Terong Pipit, Kelurahan Sempaja Timur, Kota Samarinda. Pada penelitian ini didapatkan 11 lintasan optimal dengan total bobot tidak lebih dari batas maksimum yaitu 10. Lintasan-lintasan optimal tersebut yaitu  $(v_1, v_2, v_3, v_5)$ ,  $(v_1, v_2, v_4, v_{11}, v_{16})$ ,  $(v_1, v_2, v_3, v_5, v_7, v_9, v_{10}, v_8)$ ,  $(v_1, v_2, v_4, v_{24}, v_{25}, v_{22})$ ,  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13})$ ,  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{14}, v_{18}, v_{19})$ ,  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{20}, v_{23})$ ,  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{21}, v_{20}, v_{27}, v_{28}, v_{29})$ ,  $(v_1, v_2, v_3, v_5, v_6)$ ,  $(v_1, v_2, v_4, v_{11}, v_{16}, v_{17}, v_{19}, v_{22}, v_{25}, v_{26})$ , dan  $(v_1, v_2, v_3, v_5, v_7, v_8, v_{10}, v_{12}, v_{13}, v_{15})$ . Lintasan-lintasan optimal tersebut merupakan urutan evakuasi banjir yang dapat

digunakan di Jl. Terong dan Jl. Terong Pipit, Kelurahan Sempaja Timur, Kota Samarinda agar seluruh warga dapat dievakuasi dan proses evakuasi berjalan optimal.

#### **DAFTAR PUSTAKA**

- [1] Harahap, M. K., dan Khairina N. (2017). Pencarian Jalur Terpendek dengan Algoritma Dijkstra. *Jurnal & Penelitian Teknik Informatika*, 2(2), 18-23.
- [2] Kementerian Pekerjaan Umum dan Perumahan Rakyat (2019, Juni). Bencana Banjir Kota Samarinda Provinsi Kalimantan Timur. [Laporan Kondisi Terkini]. Samarinda. KPUPR.
- [3] Lin, B., Liu, S., Lin, R., Wu, J., Wang, J., dan Liu, C. (2017). Modeling the 0-1 Knapsack Problem in Cargo Flow Adjustment. *Symmetry*, 9(7), 118.
- [4] Munir, R. (2010). *Matematika Diskrit*. Bandung : Informatika.
- [5] Puspika, B. N., Rachmat C., A., dan Kurniawan, E. (2012). Implementasi Algoritma Dijkstra Dalam Penentuan Jalur Terpendek Di Yogyakarta Menggunakan GPS Dan Qt Geolocation. *Informatika*, 8(2), 141-149.
- [6] Rachmawati, D. dan Candra, A. (2013). Implementasi Algoritma Greedy untuk Menyelesaikan Masalah Knapsack Problem. *Jurnal Saintikom*, 12(3), 185-192.
- [7] Rahmawati. (2018). Penerapan Fuzzy Linear Programming Pada Optimasi Pembangunan Rumah Susun (Rusun) Di Kawasan Pondok Cina Provinsi Jawa Barat. *Jurnal Sains Matematika dan Statistika*, 4(1), 79-89.
- [8] Rosen, K. H. (2019). *Discrete Mathematics and Its Applications*, Eighth Edition. New York : McGraw-Hill Education.
- [9] Sualang, C. S., Kaunang MT., Ir. S. T. G., dan Lumenta ST. MT., A. S. M. (2015). Pembuatan Prototype Aplikasi Pentarifan Rental Mobil Dengan Metode Polinomial Lagrange. *E-Journal Teknik Elektro dan Komputer*. 54-59.
- [10] Voloch, N. (2017). A Complex Problem of Knapsack and Shortest Paths on Weighted Graphs. *International Journal of Advanced Computational Engineering and Networking*, 5(1), 31-34.
- [11] Walikota Samarinda (2014, Februari). Peraturan Walikota (PERWALI) tentang Penyelenggaraan Penanggulangan, Pegaturan Pendanaan Serta Penetapan Besaran Santunan/Bantuan Korban Bencana. [Peraturan Walikota]. Samarinda. Pemerintah Daerah Kota Samarinda.

## Aplikasi Pemrograman Integer pada Masalah Penjadwalan Mata Pelajaran di SMPN 4 Tanah Grogot

Ahmad Topik<sup>1</sup>, Syaripuddin<sup>2,\*</sup>, Qonita Qurrota A'yun<sup>3</sup>

<sup>1,2,3</sup> *Laboratorium Matematika Komputasi Program Studi Matematika Jurusan Matematika FMIPA Universitas Mulawarman*

Dikirim: November 2022; Diterima: Maret 2023;

Dipublikasi: Maret 2023

Alamat Email Korespondensi: syarifrahman2014@gmail.com

### Abstrak

Penjadwalan mata pelajaran sekolah adalah penentuan mata pelajaran, hari sesi dimulainya dan ruangan yang akan digunakan di sekolah untuk kegiatan belajar-mengajar. Masalah penjadwalan mata pelajaran sering ditemukan di sekolah-sekolah termasuk SMPN 4 Tanah Grogot. Di antara metode untuk menyelesaikan masalah penjadwalan adalah dengan pemrograman *integer*. Tujuan dari penelitian ini adalah untuk menentukan model penjadwalan optimal mata pelajaran SMPN 4 Tanah Grogot menggunakan model pemrograman *integer*. Pada penelitian ini model penjadwalan mata pelajaran SMPN 4 Tanah Grogot dibatasi hanya untuk data kelas VIII saja. Data diperoleh dari bidang kurikulum SMPN 4 Tanah Grogot yang meliputi tiga kelas VIII, sepuluh mata pelajaran, enam hari aktif sekolah dengan tiga sesi pembelajaran per hari dan 22 guru. Dari 22 guru tersebut, terdapat dua guru Agama, dua guru PPKN, tiga guru Bahasa Indonesia, tiga guru Matematika, lima guru IPA, tiga guru IPS, tiga guru Bahasa Inggris, dua guru Seni Budaya, dua guru Olahraga dan dua guru Prakarya. Terdapat lima guru yang mengajar lebih dari satu mata pelajaran. Model masalah penjadwalan yang dihasilkan dalam penelitian ini berupa model pemrograman nonlinier *integer*. Solusi optimal diperoleh dengan menyelesaikan model menggunakan *software* LINGO 11.0 yaitu penjadwalan mata pelajaran dengan 15 guru nomor urut tertentu yang mengajar mata pelajaran sesuai bidang yang dikuasai. Terdapat juga tujuh guru yang tidak mengajar di kelas VIII.

### Kata Kunci:

*Pemrograman integer nonlinier, Penjadwalan, Mata pelajaran, Sekolah*

## PENDAHULUAN

Riset operasi adalah pendekatan dalam pengambilan keputusan yang ditandai dengan penggunaan pengetahuan ilmiah melalui usaha kelompok antar disiplin yang bertujuan menentukan penggunaan terbaik sumber daya yang terbatas [1].

Pemrograman linier merupakan salah satu bagian dari riset operasi yang digunakan untuk memecahkan persoalan optimasi (maksimum atau minimum) dengan menggunakan persamaan dan pertidaksamaan linier dalam rangka untuk mencari pemecahan yang optimal dengan memperhatikan pembatasan-pembatasan yang ada [1]. Pemrograman linier dengan tambahan kendala integer yang biasa disebut dengan pemrograman integer. Jika terdapat kendala berupa persamaan/pertidaksamaan nonlinier maka pemrograman tersebut dinamakan pemrograman nonlinier integer.

Di antara metode yang sering digunakan untuk menyelesaikan pemrograman integer yaitu metode *Branch and Bound* sebagaimana yang telah diaplikasikan dari beberapa referensi [2]-[4]. Dalam masalah ini metode *Branch and Bound* menghasilkan

penyelesaian optimal yang menghasilkan variabel-variabel keputusan bilangan bulat dengan membatasi penyelesaian optimum yang akan menghasilkan bilangan pecahan dengan cara membuat cabang atas dan bawah bagi masing-masing variabel keputusan yang bernilai pecahan agar bernilai bulat sehingga setiap pembatasan akan menghasilkan cabang baru [5]. Salah satu aplikasi pemrograman integer adalah permasalahan penjadwalan sekolah sebagaimana yang telah dilakukan [6].

Penelitian ini menghasilkan model penjadwalan mata pelajaran SMPN 4 Tanah Grogot yang dibatasi hanya untuk data kelas VIII saja. Data pada penelitian ini diperoleh dari bidang kurikulum SMPN 4 Tanah Grogot yang meliputi tiga kelas VIII, sepuluh mata pelajaran, enam hari aktif sekolah dengan tiga sesi pembelajaran per hari dan 22 guru.

### **METODE PENELITIAN**

Penelitian ini termasuk ke dalam penelitian kuantitatif dengan populasi yang digunakan adalah data bidang kurikulum SMPN 4 Tanah Grogot. Adapun sampel yang digunakan adalah data bidang kurikulum SMPN 4 Tanah Grogot yang meliputi data mata pelajaran dan guru pengajar untuk kelas VIII Semester Genap Tahun Ajaran 2021/2022. Adapun beberapa langkah-langkah yang harus dilakukan untuk menganalisis data adalah sebagai berikut:

1. Pendeskripsian masalah dengan menentukan parameter utama yang digunakan dalam penjadwalan mata pelajaran, tujuan dan batasan-batasan yang digunakan dalam pemodelan penjadwalan mata pelajaran.
2. Merumuskan variabel keputusan untuk masalah penjadwalan mata pelajaran dengan variabel yang ditentukan seperti hari, periode waktu, kelas, guru, dan mata pelajaran.
3. Perumusan model dilakukan berdasarkan tujuan pemodelan penjadwalan yang akan dijadikan fungsi objektif dan batasan-batasan dalam masalah penjadwalan dirumuskan menjadi fungsi kendala. Fungsi kendala yang digunakan sebanyak 8 kendala.
4. Penyusunan *syntax* pada *software* LINGO 11.0 dilakukan setelah memformulasikan fungsi tujuan dan fungsi kendala. Dengan menentukan judul atau *title, sets*, data yang digunakan dalam penelitian ini data yang diinput pada program merupakan jumlah periode waktu tiap kelas, bobot (koefisien) dan jumlah periode guru mengajar untuk setiap kelas, dan kendala yang digunakan.
5. Setelah model diformulasikan menjadi *syntax* program kemudian dilakukan *running* dan menghasilkan *output*. *Output* tersebut kemudian disusun menjadi jadwal mata pelajaran untuk setiap kelas dan jadwal mengajar untuk setiap guru.

### **HASIL DAN PEMBAHASAN**

SMPN 4 Tanah Grogot memiliki sebelas kelas untuk kelas VII terdapat empat kelas, untuk kelas VIII terdapat tiga kelas, dan untuk kelas IX terdapat empat kelas. Dari data sekolah tersebut sampel yang akan diambil adalah kelas VIII yang terbagi menjadi tiga kelas, yaitu kelas VIII A dengan indeks  $k = 1$ , kelas VIII B dengan indeks  $k = 2$ , dan kelas VIII C dengan indeks  $k = 3$ . Adapun data selengkapnya mengenai mata pelajaran, banyak sesi dan bobot dari tiap kelas dapat dilihat pada Tabel 1.

Tabel 1. Mata Pelajaran, Banyak Sesi, dan Bobot

Indeks (p)	Mata Pelajaran	Sing- katan	Jumlah sesi per minggu			Bobot		
			Kelas VIII A	Kelas VIII B	Kelas VIII C	Sesi 1	Sesi 2	Sesi 3
1	Matematika	MAT	3	3	3	1	2	3
2	Ilmu Pengetahuan Alam	IPA	3	3	3	1	2	3
3	Bahasa Indonesia	IND	3	2	3	1	1	1
4	Bahasa Inggris	ING	2	3	3	1	1	1
5	Ilmu Pengetahuan Sosial	IPS	2	2	2	1	1	1
6	PPKn	PKN	1	1	1	1	1	1
7	Pendidikan Agama & Budi Pekerti	AGA	1	1	1	1	1	1
8	Penjasorkes	PJOK	1	1	0	1	1	3
9	Seni Budaya	SBK	1	1	1	1	1	1
10	Prakarya	PKR	1	1	1	2	1	3

Pada Tabel 1 dapat dilihat setiap kelas memiliki sepuluh mata pelajaran dan jumlah sesi untuk tiap minggu dan pembagian bobot di setiap sesi. Hal ini berdasarkan data yang diperoleh di lapangan.

Tabel 2. Mata Pelajaran yang dikuasai Guru

Indeks guru (g)	Mata pelajaran yang dikuasai
1	Matematika
2	Matematika
3	Matematika, Prakarya
4	Bahasa Inggris
5	Bahasa Inggris
6	Bahasa Inggris
7	Ilmu Pengetahuan Sosial
8	Ilmu Pengetahuan Sosial
9	Bahasa Indonesia
10	Bahasa Indonesia
11	Bahasa Indonesia
12	Penjasorkes
13	Pendidikan Agama & Budi Pekerti, Penjasorkes
14	Pendidikan Agama & Budi Pekerti
15	Ilmu Pengetahuan Alam
16	Ilmu Pengetahuan Alam
17	PKN
18	PKN
19	Ilmu Pengetahuan Sosial, Ilmu Pengetahuan Alam
20	Seni Budaya
21	Seni Budaya, Ilmu Pengetahuan Alam
22	Prakarya, Ilmu Pengetahuan Alam

SMPN 4 Tanah Grogot memiliki 22 guru dengan bidang keahlian yang berbeda-beda. Hal ini ditunjukkan pada Tabel 2 yang dimana ada beberapa guru yang menguasai lebih dari satu mata pelajaran yang sudah sesuai dengan keahliannya masing-masing.

SMPN 4 Tanah Grogot memiliki beberapa batasan khusus yaitu:

1. Setiap pelajaran di setiap kelas maksimal hanya satu sesi per hari,
2. Pelajaran olahraga tidak diselenggarakan pada hari Senin dan Kamis,
3. Setiap guru maksimal hanya mengajar dua sesi per hari,
4. Memperhatikan ketidaksediaan guru di sesi atau hari tertentu, maka model matematika untuk masalah penjadwalan ini adalah sebagai berikut:  
Fungsi objektif dari kasus ini ialah:

$$\min \sum_{p=1}^{10} \sum_{k=1}^3 \sum_{h=1}^6 \sum_{s=1}^3 c_{ps} y_{pkhs}$$

Kendala-kendalanya ialah

1. Setiap mata pelajaran harus memenuhi jumlah sesi yang ditetapkan,

$$\sum_{h=1}^6 \sum_{s=1}^3 y_{pkhs} = b_{pk}, \quad \forall p = 1, 2, \dots, 10; k = 1, 2, 3.$$

2. Dalam satu sesi hanya ada satu guru dan satu mata pelajaran untuk satu kelas,

$$\sum_{g=1}^{22} \sum_{p=1}^{10} x_{gpk} y_{pkhs} = 1, \quad \forall k = 1, 2, 3; h = 1, 2, \dots, 6; s = 1, 2, 3.$$

3. Setiap guru maksimal hanya mengajar di satu kelas di tiap sesi,

$$\sum_{p=1}^{10} \sum_{k=1}^3 x_{gpk} y_{pkhs} \leq 1, \quad \forall g = 1, 2, \dots, 22; h = 1, 2, \dots, 6; s = 1, 2, 3.$$

4. Setiap mata pelajaran dan kelas hanya terdapat satu guru yang mengajar,

$$\sum_{g=1}^{22} x_{gpk} = 1, \quad \forall p = 1, 2, \dots, 10; k = 1, 2, 3.$$

5. Guru hanya mengajar sesuai dengan bidang keahliannya,

$$x_{gpk} \leq a_{gp} \quad \forall g = 1, 2, \dots, 22; p = 1, 2, \dots, 10; k = 1, 2, 3.$$

6. Setiap pelajaran maksimal hanya satu sesi per hari di setiap kelas,

$$\sum_{s=1}^3 y_{pkhs} \leq 1, \quad \forall p = 1, 2, \dots, 10; k = 1, 2, 3; h = 1, 2, \dots, 6$$

7. Prakarya tidak dilaksanakan di hari Senin dan Kamis,

$$\sum_{s=1}^3 y_{10k1s} = 0, \quad \forall k = 1, 2, 3.$$

8. Setiap guru maksimal mengajar dua sesi per hari

$$\sum_{p=1}^{10} \sum_{k=1}^3 \sum_{s=1}^3 x_{gpk} y_{pkhs} \leq 2, \quad \forall g = 1, 2, \dots, 22; h = 1, 2, \dots, 6.$$

## 9. Kendala kesediaan guru

Guru tujuh tidak bisa mengajar di Sesi tiga

$$x_{7pk}y_{pkh3} = 0, \quad \forall p = 1,2, \dots, 10; k = 1,2,3; h = 1,2, \dots, 6.$$

Guru delapan tidak bisa mengajar di Sesi satu

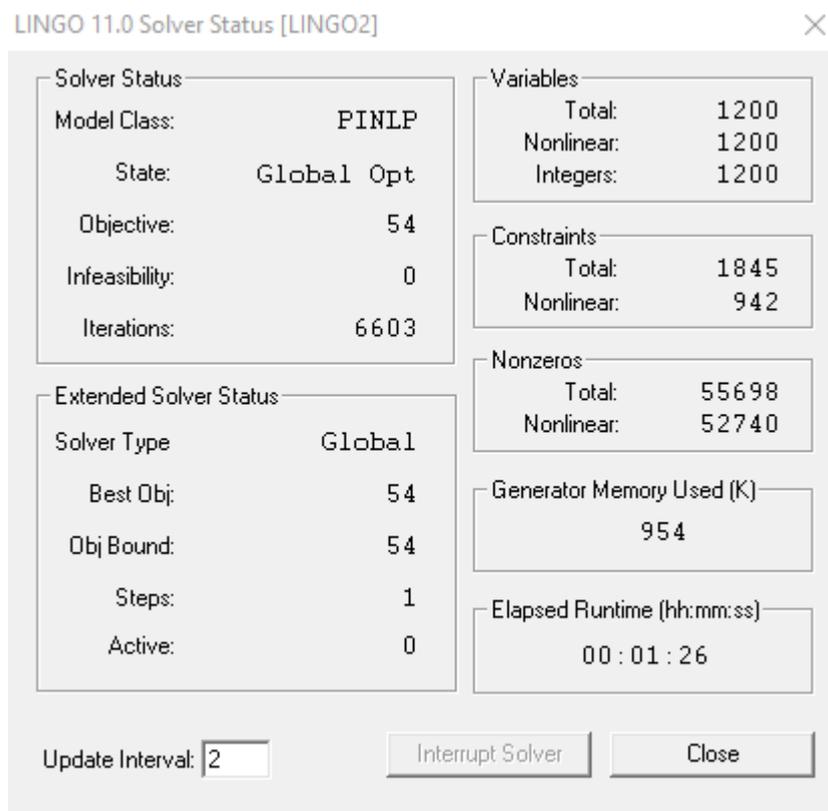
$$x_{8pk}y_{pkh1} = 0, \quad \forall p = 1,2, \dots, 10; k = 1,2,3; h = 1,2, \dots, 6.$$

## 10. Kendala biner

$$x_{gpk} \in \{0,1\}, \quad \forall g = 1,2, \dots, 22; p = 1,2, \dots, 10; k = 1,2,3.$$

$$y_{pkhs} \in \{0,1\}, \quad \forall p = 1,2, \dots, 10; k = 1,2,3; h = 1,2, \dots, 6; s = 1,2,3.$$

Selanjutnya dibahas hasil solusi dari model Matematika dengan bantuan *software* LINGO 11.0 sehingga didapatkan nilai optimal 54 sebagaimana pada Gambar 1



Gambar 1. Hasil Solusi Model Matematika dengan *Software* LINGO 11.0

Berdasarkan hasil tersebut diperoleh jadwal yang ditunjukkan di Tabel 3.

Tabel 3. Jadwal Pelajaran dan Guru

Hari	Sesi	Kelas 8A		Kelas 8B		Kelas 8C	
		Mapel	Guru	Mapel	Guru	Mapel	Guru
Senin	sesi 1	IPA	19	MAT	3	MAT	2
	sesi 2	PJOK	12	IPS	8	IPS	7
	sesi 3	SBK	20	IND	9	PKN	17
Selasa	sesi 1	MAT	1	IPA	15	MAT	2
	sesi 2	IND	9	PJOK	12	PKR	22
	sesi 3	ING	6	SBK	21	IND	9
Rabu	sesi 1	IPA	19	IPA	15	IPA	22
	sesi 2	ING	6	PKR	22	IPS	7
	sesi 3	PKN	17	IND	9	AGA	13
Kamis	sesi 1	IPA	19	IPA	15	IPA	22
	sesi 2	IPS	7	ING	6	IND	9
	sesi 3	IND	9	PKN	17	ING	6
Jumat	sesi 1	MAT	1	MAT	3	MAT	2
	sesi 2	IPS	7	ING	6	SBK	21
	sesi 3	IND	9	IPS	8	ING	6
Sabtu	sesi 1	MAT	1	MAT	3	IPA	22
	sesi 2	PKR	22	AGA	13	ING	6
	sesi 3	AGA	13	ING	6	IND	9

Pada Tabel 4 diberikan jumlah sesi per mata pelajaran di setiap minggunya. Dari tabel tersebut dapat dilihat pembagian setiap mata pelajaran di setiap kelasnya sudah terpenuhi, tidak ada bentrok mata pelajaran dan bentrok guru di setiap sesi dihari yang sama. Hal ini dapat disimpulkan bahwa pembagian jadwal mata pelajaran dan guru untuk sekolah tersebut sudah optimal berdasarkan hasil LINGO 11.0.

Tabel 4. Jumlah Sesi per Mata Pelajaran

Mata pelajaran	Bobot di sesi			Terjadwal di sesi			Total sesi
	1	2	3	1	2	3	
Matematika	1	2	3	9	0	0	9
Ilmu Pengetahuan Alam	1	2	3	9	0	0	9
Bahasa Indonesia	1	1	1	0	2	6	8
Bahasa Inggris	1	1	1	0	4	4	8
Ilmu Pengetahuan Sosial	1	1	1	0	5	1	6
PPKn	1	1	1	0	0	3	3
Pendidikan Agama & Budi Pekerti	1	1	1	0	1	2	3
Penjasorkes	1	1	3	0	2	0	2
Seni Budaya	1	1	1	0	1	2	3
Prakarya	2	1	3	0	3	0	3

Pada Tabel 5 diberikan total mengajar setiap guru setiap harinya. Dari tabel tersebut dapat dilihat beberapa kendala terpenuhi. Nilai bobot dari masing-masing mata pelajaran tersebut sudah berdasarkan Tabel 4. Adapun hasil yang didapatkan sangat baik karena tidak ada satu sesi pun yang dijadwalkan pada sesi dengan bobot dua maupun tiga, dikarenakan semakin rendah angka bobotnya maka semakin pantas mata pelajaran itu diajarkan pada sesi tersebut.

Tabel 5. Total Mengajar (Dalam Sesi) Setiap Guru

Guru ( <i>g</i> )	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
1	0	1	0	0	1	1
2	1	1	0	0	1	0
3	1	0	0	0	1	1
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	1	1	2	2	2
7	1	0	1	1	1	0
8	1	0	0	0	1	0
9	1	2	1	2	1	1
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	1	1	0	0	0	0
13	0	0	1	0	0	2
14	0	0	0	0	0	0
15	0	1	1	1	0	0
16	0	0	0	0	0	0
17	1	0	1	1	0	0
18	0	0	0	0	0	0
19	1	0	1	1	0	0
20	1	0	0	0	0	0
21	0	1	0	0	1	0
22	0	1	2	1	0	2

Pada hasil penjadwalan mata pelajaran SMPN 4 Tanah Grogot dengan bantuan *software* LINGO 11.0, terdapat 15 guru yang mengajar mata pelajaran sesuai bidang yang dikuasai, yaitu guru dengan urutan 1, 2, 3, 6, 7, 8, 9, 12, 13, 15, 17, 18, 19, 20, 21 dan 22. Selain itu diperoleh pula bahwa terdapat tujuh guru yang tidak mengajar sama sekali di kelas VIII, yaitu guru dengan no urutan 4, 5, 10, 11, 14, 16 dan 18. Hal ini dikarenakan hasil dari *software* LINGO 11.0 yang mendahulukan satu guru untuk mengajar di sesi yang berbeda dengan pertimbangan sedikitnya jumlah kelas dan mata pelajaran yang terbatas.

Jadwal mata pelajaran SMPN 4 Tanah Grogot untuk masalah penjadwalan ini adalah sebagai berikut:

- Matematika dilaksanakan pada hari Senin, Selasa, Jumat, dan Sabtu di sesi satu oleh guru satu, guru dua, maupun guru tiga.
- Ilmu Pengetahuan Alam dilaksanakan pada hari Senin, Selasa, Rabu, Kamis, dan Sabtu di sesi satu oleh guru 15, guru 19 maupun guru 22.
- Bahasa Indonesia dilaksanakan pada hari Senin, Selasa, Rabu, Kamis, Jumat, dan Sabtu di sesi dua dan sesi tiga oleh guru sembilan.
- Bahasa Inggris dilaksanakan pada hari selasa, Rabu, Kamis, Jumat dan Sabtu di sesi dua dan sesi tiga oleh guru enam.
- Ilmu Pengetahuan Sosial dilaksanakan pada hari Senin, Rabu, Kamis, dan Jumat di sesi dua dan sesi tiga oleh guru tujuh maupun guru delapan.
- PPKn dilaksanakan pada hari Senin, Rabu, dan Kamis di sesi tiga oleh guru 17.

- g. Pendidikan Agama & Budi Pekerti dilaksanakan pada hari Rabu dan Sabtu di sesi dua dan sesi tiga oleh guru 13.
- h. Penjasorkes dilaksanakan pada hari Senin dan Selasa di sesi dua oleh guru 12.
- i. Seni Budaya dilaksanakan pada hari Senin, Selasa, dan Jumat di sesi dua dan sesi tiga oleh guru 20 maupun guru 21.
- j. Prakarya dilaksanakan pada hari Selasa, Rabu, dan Sabtu di sesi dua oleh guru 22.

## **PENUTUP**

Berdasarkan hasil dan pembahasan dapat ditarik kesimpulan bahwa penjadwalan mata pelajaran SMPN 4 Tanah Grogot dapat dimodelkan menjadi model pemrograman *integer* nonlinier melalui fungsi objektif yang meminimumkan total bobot mata pelajaran dan memenuhi sepuluh kendala yang ada pada sekolah tersebut dan jadwal mata pelajaran yang dihasilkan dengan bantuan *software* LINGO 11.0 yaitu, terdapat 15 guru yang mengajar mata pelajaran sesuai bidang yang dikuasai dan terdapat tujuh guru yang tidak mengajar di kelas VIII dikarenakan hasil dari *software* LINGO 11.0 yang mendahulukan satu guru untuk mengajar di sesi yang berbeda dengan pertimbangan sedikitnya jumlah kelas dan mata pelajaran yang terbatas.

## **DAFTAR PUSTAKA**

- [1] Taha, A.H. (2014). *Riset Operasi*. Jilid 2. Tangerang: Binarupa Aksara Publisher.
- [2] Supatimah, S.S., Farida, F., & Andriani, S. (2019). Optimasi Keuntungan dengan Metode *Branch and Bound*. *AKSIOMA: Jurnal Matematika dan Pendidikan Matematika*, 10 (1), 13-23.
- [3] Suryati, P., & Kurniawati, D. (2016). Aplikasi Penjadwalan Kaos dengan Metode *Branch and Bound*. *JIKO (Jurnal Informatika dan Komputer)*, 1 (2).
- [4] Lesmana, N.I. (2016). Penjadwalan Produksi untuk Meminimalkan Waktu Produksi dengan Menggunakan Metode *Branch and Bound*. *Jurnal Teknik Industri*, 17 (1), 42-50.
- [5] Winston, W.L. (2004). *Operations Research Applications and Algorithms*. New York: Duxbury.
- [6] Mile, A.R., Katili, M.R., & Nurwan, N. (2022). Penjadwalan Mata Pelajaran Menggunakan Integer Nonlinear Programming. *Research in the Mathematical and Natural Sciences*, 1 (1), 12-18.

## Implementasi Algoritma *K-Means Clustering* dengan Jarak *Euclidean* dalam Mengelompokkan Daerah Penyebaran *COVID-19* di Kabupaten Bogor

Muhammad Iqbal<sup>1</sup>, Syaripuddin<sup>2,\*</sup>, Moh. Nurul Huda<sup>3</sup>

<sup>1,2,3</sup> *Laboratorium Matematika Komputasi Jurusan Matematika FMIPA Universitas Mulawarman*

Dikirim: November 2022; Diterima: Maret 2023; Dipublikasi: Maret 2023

Alamat Email Korespondensi: [syarifrahman2014@gmail.com](mailto:syarifrahman2014@gmail.com)

### Abstrak

*K-Means clustering* merupakan algoritma pengelompokan iteratif yang melakukan partisi himpunan data ke dalam sejumlah  $K$  kluster yang sudah ditentukan di awal, sedemikian sehingga setiap data masuk dalam pusat kluster (*centroid*) terdekat. Pada penelitian dilakukan pengelompokan data menggunakan metode *K-Means clustering* dengan jarak *Euclidean* pada data daerah penyebaran *COVID-19* di Kabupaten Bogor. Penentuan jumlah kluster optimal yang ditentukan sejak awal menggunakan metode *Elbow* dan *Silhouette Index*. Penelitian bertujuan untuk mengetahui hasil dari penerapan algoritma *K-means clustering* untuk mengelompokkan daerah penyebaran *COVID-19* di Kabupaten Bogor. Adapun variabel yang digunakan yaitu kepadatan penduduk ( $X_1$ ), jumlah kasus terinfeksi *COVID-19* ( $X_2$ ) dan jumlah kasus meninggal *COVID-19* ( $X_3$ ). Berdasarkan hasil penelitian diperoleh dua kluster tingkat risiko penyebaran *COVID-19* yaitu kluster C1 dan C2. Pada kluster C1 memiliki tingkat risiko penyebaran sedang terdiri atas 36 Kecamatan, sedangkan pada kluster C2 dengan tingkat risiko penyebaran tinggi terdiri atas 4 Kecamatan. Nilai validitas yang didapat sebesar 0.71676 yang berarti struktur dari hasil kluster tersebut kuat.

### Kata Kunci:

*COVID-19, Jarak Euclidean, K-Means Clustering, Metode Elbow, Silhouette Index*

### PENDAHULUAN

Pada awal tahun 2020 dunia dilanda pandemi virus Corona atau *COVID-19*. *COVID-19* (*Coronavirus Disease 2019*) adalah penyakit menular yang disebabkan oleh virus *SARS-CoV-2* (*Severe Acute Respiratory Syndrome Coronavirus-2*). Orang yang terinfeksi virus ini akan mengalami penyakit pernapasan ringan hingga sedang dan sembuh tanpa memerlukan perawatan khusus. Namun, beberapa kasus akan mengalami sakit yang berat dan memerlukan perhatian medis [1].

Kasus virus *pneumonia* yang tidak diketahui, sekarang dikenal *COVID-19* pertama kali dilaporkan di Kota Wuhan, Provinsi Hubei, China pada Desember 2019 [2]. Sementara di Indonesia kasus *COVID-19* pertama kali dilaporkan pada 2 Maret 2020, ketika dua orang terkonfirmasi tertular dari seorang warga negara Jepang. Setelah terkonfirmasi adanya kasus terinfeksi *COVID-19* atau virus Corona, pemerintah Indonesia melakukan berbagai cara atau upaya untuk menghindari terjadinya penularan *COVID-19* yang berkepanjangan di Indonesia. Pada tanggal 9 April 2020, pandemi sudah menyebar ke 34

provinsi, salah satunya provinsi Jawa Barat, virus ini telah menyebar ke 27 Kabupaten/Kota di Jawa Barat salah satunya adalah Kabupaten Bogor [3].

Penyebaran kasus *COVID-19* yang cukup cepat dan merata di seluruh provinsi di Indonesia berdampak negatif pada seluruh bidang. Luasnya wilayah Indonesia memungkinkan diperlukannya *clustering* (pengelompokan) berdasarkan wilayah di Indonesia khususnya Kabupaten Bogor, sebab Kabupaten Bogor merupakan Kabupaten/Kota dengan populasi penduduk terbesar di Indonesia [4]. *Clustering* ini akan menghasilkan titik-titik pusat penyebaran dan risiko penyebaran kasus *COVID-19* di Kabupaten Bogor.

Algoritma *K-Means clustering* merupakan salah satu metode *clustering* non-hierarki yang berusaha mengelompokkan data ke dalam satu kelompok yang sama [5]. Ukuran kesamaan yang digunakan adalah ukuran jarak antar objek. Kedua objek yang memiliki jarak paling dekat akan bergabung menjadi satu kelompok. Kedekatan jarak yang dimiliki menunjukkan bahwa kedua objek tersebut memiliki tingkat kesamaan karakteristik. Pada penelitian ini untuk menentukan jumlah kluster yang optimal dalam mengelompokkan daerah penyebaran *COVID-19* menggunakan metode *Elbow* dan *Silhouette Index*.

Penelitian terkait penerapan algoritma *K-Means clustering* telah dilakukan oleh beberapa peneliti sebelumnya, antara lain pada penelitian Sari dan Sukestiyarno [6]. Penelitian tersebut bertujuan untuk mengetahui tingkat persebaran kasus *COVID-19* kategori tinggi, sedang dan rendah pada masing-masing provinsi di Indonesia dengan menggunakan metode *K-Means clustering*. Berdasarkan pada beberapa penelitian sebelumnya untuk mengetahui tingkat penyebaran kasus *COVID-19* dengan tingkat risiko tinggi, sedang dan rendah di Kabupaten Bogor, ada beberapa aspek yang dapat diukur seperti kepadatan penduduk, jumlah kasus terinfeksi *COVID-19* dan jumlah kasus kematian *COVID-19*.

Berdasarkan uraian di atas, penelitian ini mengkaji tentang implementasi algoritma *K-Means clustering* dengan jarak *Euclidean* dalam mengelompokkan daerah penyebaran *COVID-19* di Kabupaten Bogor. Tujuan penelitian ini adalah mengetahui cara menentukan jumlah kluster yang paling optimal dengan menggunakan metode *Elbow* dan *Silhouette Index* dan mengetahui karakteristik kluster yang terbentuk berdasarkan tingkat risiko penyebaran *COVID-19* di Kabupaten Bogor.

## METODE PENELITIAN

Penelitian ini termasuk ke dalam penelitian kuantitatif dengan populasi yang digunakan adalah seluruh data kepadatan penduduk, data jumlah kasus terinfeksi *COVID-19* dan data jumlah kasus kematian *COVID-19* Kabupaten Bogor. Adapun sampel pada penelitian ini adalah data kepadatan penduduk (km<sup>2</sup>), data jumlah kasus terinfeksi *COVID-19* (jiwa) dan data jumlah kasus kematian *COVID-19* (jiwa) pada rentang 1 sampai 31 Juli 2021. Adapun tahapan analisis data yang dilakukan pada penelitian ini adalah sebagai berikut:

1. Pengumpulan Data
2. Normalisasi Data

Menormalisasikan data dapat mempercepat tahap pembelajaran dan dapat mengatasi masalah pada perbedaan rentang nilai yang jauh pada setiap variabel [7]. Normalisasi data pada penelitian ini menggunakan metode *Min-max normalization*. Adapun rumus persamaannya sebagai berikut:

$$x' = \left( \frac{x - X_{min}}{X_{max} - X_{min}} \right), \quad (1)$$

keterangan:

- $x$  = nilai data  $x$ ,
- $x'$  = nilai baru data  $x$ ,
- $X_{min}$  = nilai minimum pada variabel  $X$ ,
- $X_{max}$  = nilai maksimum pada variabel  $X$ .

### 3. Menentukan Nilai $K$

Yaitu menentukan jumlah kluster terbaik atau paling optimal dalam mengelompokkan data menggunakan metode *Elbow* dan *Silhouette Index*. Adapapun rumus persamaan metode *Elbow* sebagai berikut:

$$SSE = \sum_{k=1}^K \sum_{x_i} |x_i - c_k|^2, \quad (2)$$

keterangan:

- $K$  = banyak kluster,
- $x_i$  = nilai variabel ke- $i$  dari data  $x$ ,
- $c_k$  = *centroid* kluster ke- $k$ .

### 4. Metode *K-Means Clustering*

Mengelompokkan set data dengan menggunakan algoritma *K-Means Clustering*. Adapapun algoritma *K-Means clustering* adalah sebagai berikut:

- i. Inisialisasi: menentukan nilai  $K$  sebagai jumlah kluster yang diinginkan dan metrik ketidakmiripan (jarak). Pada penelitian ini digunakan metrik ketidakmiripan jarak *Euclidean*. Adapapun rumus persamaannya sebagai berikut:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}, \quad (3)$$

keterangan:

- $d(x, y)$  = jarak data  $x$  ke data  $y$ ,
- $x_i$  = nilai variabel ke- $i$  dari data  $x$ ,
- $y_i$  = nilai variabel ke- $i$  dari data  $y$ .

- ii. Memilih  $K$  data dari set data  $X$  sebagai *centroid* atau pusat kluster.
- iii. Mengalokasikan semua data ke *centroid* terdekat dengan metrik jarak yang sudah ditetapkan (memperbarui kluster pada setiap data).
- iv. Menghitung kembali *centroid*  $C$  berdasarkan data yang mengikuti kluster masing-masing.
- v. Mengulangi langkah ke iii dan iv hingga kondisi konvergen tercapai, yaitu
  - a. Perubahan fungsi objektif sudah di bawah ambang batas yang diinginkan; atau
  - b. Tidak ada data yang berpindah kluster; atau
  - c. Perubahan posisi *centroid* sudah di bawah ambang batas yang ditetapkan.

### 5. Menghitung Nilai Validitas

Melakukan perhitungan hasil *clustering* dengan *Silhouette Index* untuk mengukur validitas hasil kluster. Adapapun untuk menghitung nilai *SI* dari sebuah data ke- $i$ , ada dua komponen yaitu  $a_i$  dan  $b_i$ .  $a_i$  adalah rata-rata jarak data ke- $i$  terhadap semua data lainnya dalam satu kluster, sedangkan  $b_i$  adalah rata-rata jarak data ke- $i$  terhadap semua data dari kluster lain yang tidak dalam satu kluster dengan data ke- $i$ , kemudian diambil yang terkecil. Adapapun rumus persamaannya sebagai berikut:

$$a_i^j = \frac{1}{m_j - 1} \sum_{\substack{r=1 \\ r \neq i}}^{m_j} d(x_i^j, x_r^j), \quad i = 1, 2, \dots, m_j. \quad (4)$$

$d(x_i^j, x_r^j)$  adalah jarak data ke- $i$  dengan data ke- $r$  dalam satu kluster  $j$ , sedangkan  $m_j$  adalah jumlah data dalam kluster ke- $j$ . Sementara rumus  $b_i^j$ :

$$b_i^j = \min_{\substack{n=1, \dots, k \\ n \neq j}} \left\{ \frac{1}{m_n} \sum_{\substack{r=1 \\ r \neq i}}^{m_n} d(x_i^j, x_r^n) \right\}, \quad i = 1, 2, \dots, m_n, \quad (5)$$

untuk mendapatkan *Silhouette Index* ( $SI$ ) data ke- $i$  menggunakan persamaan berikut:

$$SI_i^j = \frac{b_i^j - a_i^j}{\max\{a_i^j, b_i^j\}}. \quad (6)$$

Sementara nilai  $SI$  global didapatkan dengan menghitung rata-rata  $SI$  dari semua kluster seperti pada persamaan berikut:

$$SI = \frac{1}{k} \sum_{j=1}^k SI_j. \quad (7)$$

Interpretasi nilai dari *Silhouette Index* menurut Kauffman dan Rousseeuw [8] ditunjukkan pada Tabel 1.

Tabel 1. Interpretasi Nilai *Silhouette Index*

<i>Silhouette Index</i>	Interpretasi
0.71 – 1.00	Struktur yang dihasilkan kuat
0.51 – 0.70	Struktur yang dihasilkan baik
0.26 – 0.50	Struktur yang dihasilkan lemah
$\leq 0.25$	Tidak terstruktur

## HASIL DAN PEMBAHASAN

Data yang digunakan pada penelitian ini berjumlah 40 data, masing-masing data merepresentasikan Kecamatan di Kabupaten Bogor. Adapun Variabel  $X_1$  adalah data kepadatan penduduk yang diambil dari data Badan Pusat Statistik Kabupaten Bogor [9]. Sementara  $X_2$  adalah data jumlah kasus terinfeksi *COVID-19* dan  $X_3$  adalah jumlah kasus kematian *COVID-19* yang diambil dari laman Geoportal Kabupaten Bogor [10]. Data tersebut dapat dilihat pada Tabel 2.

Tabel 2. Data *COVID-19* Kabupaten Bogor

No.	Kecamatan	$X_1$	$X_2$	$X_3$
1.	Babakan Madang	2,619.84	312	3
2.	Bojonggede	10,145.32	1,671	52
3.	Caringin	2,777.83	236	0
⋮	⋮	⋮	⋮	⋮
38.	Tanjungsari	667.54	109	1
39.	Tenjo	880.51	114	1
40.	Tenjolaya	1,539.12	51	3

## Normalisasi Data

Selanjutnya dilakukan normalisasi data pada Tabel 2 dengan rentang 0 sampai 1 menggunakan Persamaan (1)

$$\begin{aligned}
 x'_{1,1} &= \left( \frac{x_{1,1} - X_{min}}{X_{max} - X_{min}} \right) \\
 &= \left( \frac{2,619.84 - 303.33}{10,145.32 - 303.33} \right) = 0.23537 \\
 &\vdots \\
 x'_{40,1} &= \left( \frac{x_{40,1} - X_{min}}{X_{max} - X_{min}} \right) \\
 &= \left( \frac{1,539.12 - 303.33}{10,145.32 - 303.33} \right) = 0.12556
 \end{aligned}$$

Perhitungan yang sama dilakukan seterusnya pada setiap data sampai dengan data ke-40 dan dilanjutkan untuk semua variabel. Berikut adalah hasil normalisasi data yang diperoleh, dapat dilihat pada Tabel 3.

Tabel 3. Data Normalisasi

No.	$X_1$	$X_2$	$X_3$
1.	0.23537	0.09859	0.03947
2.	1.00000	0.57711	0.68421
3.	0.25142	0.07183	0.00000
⋮	⋮	⋮	⋮
38.	0.03701	0.02711	0.01316
39.	0.05864	0.02887	0.01316
40.	0.12556	0.00669	0.03947

## Menentukan Nilai $K$ yang Optimal

Pada penelitian ini untuk menentukan nilai  $K$  atau jumlah kluster yang optimal menggunakan dua metode yakni Metode *Elbow* dan *Silhouette Index*, kedua metode ini digunakan untuk memperkuat dan memastikan jumlah kluster yang paling optimal pada data di penelitian.

### 1. Metode *Elbow*

Grafik hasil perhitungan nilai *SSE* (*Sum of Squared Error*) menggunakan *syntax* Python dapat dilihat pada Gambar 1.

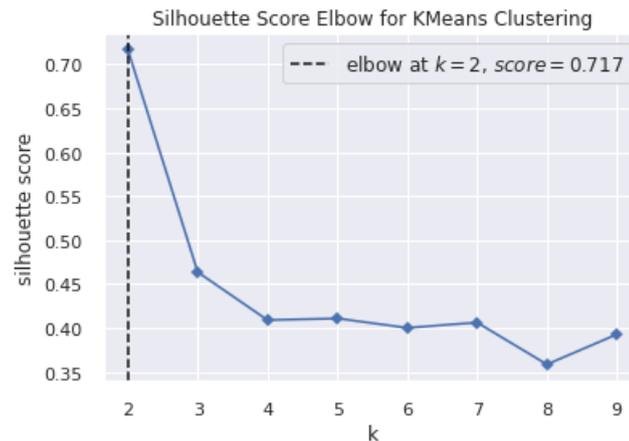


Gambar 1. Grafik Hasil Perhitungan *SSE* Metode *Elbow*

Berdasarkan metode *Elbow* didapatkan jumlah kluster terbaik yang dilakukan pada penelitian ini adalah sebanyak dua kluster. pada saat  $K = 2$  nilai *SSE* sebesar 2.02272, setelah melewati  $K = 2$  nilai *SSE* cenderung stabil, serta terdapat sedikit sudut siku pada saat  $K = 2$ . Maka pada penelitian digunakan  $K = 2$  atau membagi data ke dalam dua kluster.

## 2. *Silhouette Index*

Sementara itu, hasil perhitungan nilai *Silhouette Index* dengan *syntax* pemrograman Python dapat dilihat pada Gambar 2.



Gambar 2. Grafik Rata-rata Nilai *Silhouette Index*

Berdasarkan Gambar 2, maka didapatkan jumlah kluster paling optimal menggunakan *Silhouette Index* sebanyak dua kluster. Karena nilai rata-rata *Silhouette Index* tertinggi adalah ketika  $k = 2$ .

## Metode *K-Means Clustering*

Pada tahap ini dilakukan proses utama yaitu penerapan algoritma *K-Means clustering*. Berdasarkan metode *Elbow* dan *Silhouette Index* jumlah kluster pada penelitian ini menggunakan dua kluster, kluster pertama direpresentasikan dengan C1 dan kluster kedua direpresentasikan dengan C2. Selanjutnya dipilih dua data sebagai *centroid* atau pusat kluster yang terdapat pada Tabel 4.

Tabel 4. Data *Centroid* Awal

C1				C2	
0.18895	0.44930	0.32895	1.00000	0.57711	0.68421

Selanjutnya adalah menghitung jarak antara data dengan kedua *centroid* menggunakan jarak *Euclidean* pada Persamaan (3).

$$\begin{aligned}
 d_{1,c1} &= \sqrt{\sum_{i=1}^3 (x_i - C_1)^2}, \\
 &= \sqrt{(0.23537 - 0.18895)^2 + (0.09859 - 0.44930)^2 + (0.03947 - 0.32895)^2}, \\
 &= \sqrt{0.20895}, = 0.45710, \\
 d_{1,c2} &= \sqrt{\sum_{i=1}^3 (x_i - C_2)^2},
 \end{aligned}$$

$$= \sqrt{(0.23537 - 0.25142)^2 + (0.09859 - 0.07183)^2 + (0.03947 - 0.00000)^2},$$

$$= \sqrt{1.22933} = 1.10875.$$

Hasil perhitungan data ke-1 didapatkan jarak data ke *centroid* terdekat adalah ke C1 yakni sebesar 0.45710, maka data pertama masuk kedalam kelompok atau kluster C1. Selanjutnya dilakukan perhitungan yang sama pada seluruh data ke setiap *centroid* C1 dan C2, hasil perhitungan iterasi ke-1 dapat dilihat pada Tabel 5.

Tabel 5. Perhitungan Iterasi ke-1

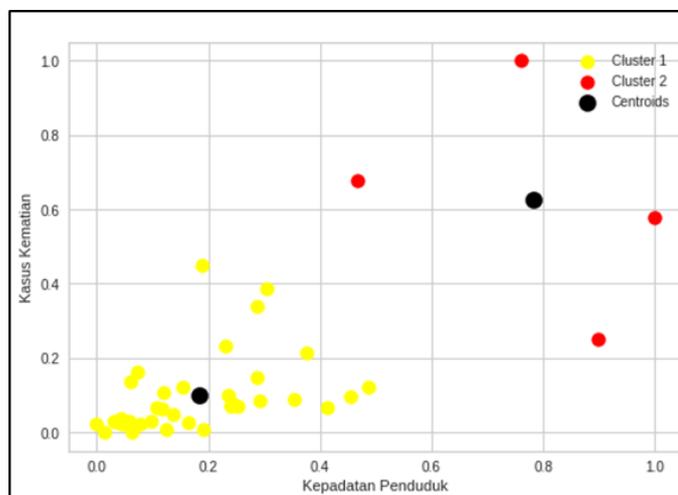
No.	Jarak Data ke <i>Centroid</i>		Cluster
	C1	C2	
1.	0.45710	1.10875	C1
2.	0.89462	0.00000	C2
3.	0.50457	1.13306	C1
⋮	⋮	⋮	⋮
38.	0.54868	1.29621	C1
39.	0.54172	1.27946	C1
40.	0.53265	1.22707	C1

Selanjutnya dihitung rata-rata data tiap kluster untuk mendapatkan nilai *centroid* baru. Lalu kembali hitung jarak tiap data dengan *centroid* baru sampai keadaan konvergen. Pada penelitian ini didapatkan hasil konvergen pada iterasi ke-3. Pada kluster C1 terdapat 36 Kecamatan sementara pada kluster C2 terdiri atas 4 Kecamatan, hasil tersebut dapat dilihat pada Tabel 6.

Tabel 6. Hasil *K-Means Clsuetring*

Cluster	
C1	<ul style="list-style-type: none"> <li>Babakan Madang, Caringin, Cariu, Ciampea, Ciawi, Cibungbulang, Cigombong, Cigudeg, Cijeruk, Cileungsi, Cisarua, Ciseeng, Citeureup, Dramaga, Gunungsindur, Jasinga, Jonggol, Kemang, Klapanunggal, Leuwiliang, Leuwisadeng, Megamendung, Nanggung, Pamijahan, Parung, Parungpanjang, Rancabungur, Rumpin, Sukajaya, Sukaraja, Sukamakmur, Tajurhalang, Tamansari, Tanjungsari, Tenjo, Tenjolaya</li> </ul>
C2	<ul style="list-style-type: none"> <li>Bojonggede, Cibinong, Ciomas, Gunung Putri</li> </ul>

Hasil algoritma *K-Means clustering* pada penelitian dapat dilihat dalam bentuk visualisasi grafik *scatter*, visualisasi sebaran *dataset* dapat dilihat pada Gambar 3.

Gambar 3. Grafik Plot Data Hasil *K-Means Clustering*

Data kluster C1 direpresentasikan dengan warna kuning, sementara data kluster C2 direpresentasikan dengan warna merah dan warna hitam adalah *centroid*.

Daerah pada kluster C1 memiliki rata-rata kepadatan penduduk 2,108.49 orang/km<sup>2</sup>, rata-rata kasus terinfeksi *COVID-19* sebanyak 315.44 orang dan rata-rata kasus kematian 6.03 orang. Sementara daerah pada kluster C2 memiliki rata-rata kepadatan penduduk 7,994.66 orang/km<sup>2</sup>, rata-rata kasus terinfeksi *COVID-19* sebanyak 1,809 orang dan rata-rata kasus kematian 48.05 orang. Berdasarkan nilai rata-rata tersebut daerah pada kluster C1 memiliki kepadatan penduduk, kasus terinfeksi *COVID-19* dan kasus kematian lebih rendah dibandingkan dengan daerah pada kluster C2. Maka dapat diartikan bahwa daerah kluster C1 termasuk dalam Kecamatan yang memiliki tingkat risiko penyebaran *COVID-19* yang sedang, sementara daerah pada kluster C2 memiliki tingkat risiko penyebaran *COVID-19* yang tinggi.

### Menghitung Nilai Validitas dengan *Silhouette Index*

Pada tahap ini dilakukan validasi hasil kluster yang telah dilakukan menggunakan *Silhouette Index*. Menghitung nilai rata-rata *SI* secara manual dengan jumlah kluster  $k = 2$ . Pertama adalah mencari nilai  $a_i$  untuk data yang berada pada kluster C2 dengan Persamaan (4)

$$\begin{aligned} a_1^{C2} &= \frac{1}{4-1} \sum_{r=1}^4 d(x_1^1, x_r^1), \\ &= \frac{1}{4-1} (d(x_1^1, x_1^1) + d(x_1^1, x_2^1) + d(x_1^1, x_3^1) + d(x_1^1, x_4^1)), \\ &= \frac{1}{3} (0.00000 + 0.57926 + 0.57443 + 0.54389) = 0.56586. \end{aligned}$$

Hal yang sama dilakukan pada data lainnya di kluster C2. Perhitungan yang sama juga dilakukan pada data di kluster C1.

Selanjutnya yaitu menghitung nilai  $b_i$  untuk data yang berada pada kluster C2 menggunakan Persamaan (5)

$$\begin{aligned} b_1^{C2} &= \min_{n=1} \left\{ \frac{1}{36} \sum_{r=1}^{36} d(x_1^{C2}, x_r^{C1}) \right\}, \\ &= \min \left\{ \frac{1}{36} (1.10875 + 1.13306 + \dots + 1.22707) \right\}, \\ &= \min\{1.13016\} = 1.13016. \end{aligned}$$

Hal yang sama dilakukan pada data lainnya di kluster C2. Perhitungan yang sama juga dilakukan pada data di kluster C1.

Selanjutnya yaitu menghitung nilai  $SI_i$  (*Silhouette Index*) untuk data yang berada pada kluster C2 dengan menggunakan Persamaan (6)

$$\begin{aligned} SI_1^{C2} &= \frac{b_1^{C2} - a_1^{C2}}{\max\{a_1^{C2}, b_1^{C2}\}}, \\ &= \frac{1.13016 - 0.56586}{\max\{1.13016, 0.56586\}}, \\ &= \frac{1.13016 - 0.56586}{1.13016} = 0.49931. \end{aligned}$$

Hal yang sama dilakukan pada data lainnya di kluster C2. Perhitungan yang sama juga dilakukan pada data di kluster C1.

Langkah terakhir yaitu menghitung rata-rata nilai *SI (Silhouette Index)* dengan menggunakan Persamaan (7)

$$\begin{aligned} SI &= SI = \frac{1}{40} \sum_{j=1}^{40} SI_j, \\ &= \frac{1}{40} (0.82522 + 0.49931 + \dots + 0.83617), \\ &= \frac{1}{40} (28.67048) = 0.71676. \end{aligned}$$

Pada penelitian didapatkan hasil nilai rata-rata *SI (Silhouette Index)* sebesar 0.71676, maka berdasarkan Tabel 2.1 dapat diartikan bahwa implementasi algoritma *K-Means clustering* pada penelitian ini memiliki struktur kluster yang kuat.

## PENUTUP

Berdasarkan hasil penerapan algoritma *K-Means clustering* pada penelitian ini didapatkan hasil 36 Kecamatan masuk ke dalam kluster C1 dan 4 Kecamatan masuk ke dalam kluster C2, Karakteristik kluster C1 memiliki tingkat risiko penyebaran sedang dengan rata-rata kepadatan penduduk 2,108.49 orang/km<sup>2</sup>, rata-rata kasus terinfeksi *COVID-19* sebanyak 315.44 orang dan rata-rata kasus kematian 6.03 orang. Sementara daerah pada kluster C2 memiliki tingkat risiko penyebaran tinggi dengan rata-rata kepadatan penduduk 7,994.66 orang/km<sup>2</sup>, rata-rata kasus terinfeksi *COVID-19* sebanyak 1,809 orang dan rata-rata kasus kematian 48.05 orang. Nilai validitas *clustering* dengan *Silhouette Index* pada penelitian ini didapatkan hasil sebesar 0.71676 berdasarkan hasil tersebut, maka struktur yang dihasilkan algoritma *K-means clustering* pada penelitian ini memiliki struktur yang kuat.

## DAFTAR PUSTAKA

- [1] World Health Organization (WHO). (2020). *Coronavirus disease 2019 (COVID-19): Situation Report*.
- [2] Yang, Y., Peng F., Wang, R., Guan, K., Jiang, T., Xu, G., Sun, J. & Chang, C. (2020). The Deadly Coronaviruses: The 2003 SARS Pandemic and The 2020 Novel Coronavirus Epidemic in China. *J Autoimmun* 2020:102434.
- [3] Ramadanti, E. & Muslih, M. (2021). Analisis Persebaran Kasus COVID-19 di Jawa Barat Menggunakan Metode K-Means Clustering. *Jurnal SISMATIK Nusa Putra*, 319-326.
- [4] Kementerian Dalam Negeri RI (2022). Rilis Data Kemendagri: Jumlah Penduduk Jabar Terbanyak, Kaltara Tersedikit. Diakses pada 30 Juni 2022, dari <https://www.dukcapil.kemendagri.go.id/berita/baca/1033/rilis-data-kemendagri-jumlah-penduduk-jabar-terbanyak-kaltara-tersedikit>.
- [5] Prasetyo, Eko. (2013). *Data Mining: Mengolah Data Menjadi Informasi Menggunakan Matlab*. Yogyakarta: Penerbit ANDI.
- [6] Sari, D. N. P. & Sukestiyarno, Y. L. (2021). Analisis Cluster Dengan Metode K-Means Pada Persebaran Kasus COVID-19 Berdasarkan Provinsi di Indonesia. *PRISMA; Prosiding Seminar Nasional Matematika 2021. Vol 4, 602-610*.

- [7] Li, W. & Liu, Z. (2011). A Method of SVM with Normalization in Intrusion Detection. *Procedia Environ. Sci*, vol. 11, pp. 256–262, 2011.
- [8] Kauffman, L. & Rousseeuw, P. (1990). *Finding Groups in Data*. Kanada: John Willey & Sons, Inc.
- [9] Badan Pusat Statistik Kabupaten Bogor (BPS). (2021). *Kabupaten Bogor Dalam Angka*. Bogor: Badan Pusat Statistik Kabupaten Bogor.
- [10] Geoportal Kabupaten Bogor. (2021). Info COVID-19 Kabupaten Bogor. Diakses pada 13 Oktober 2021, dari <https://geoportal.bogorkab.go.id/covid19/>.

## Penerapan Aljabar Max-Plus pada Pengaturan Durasi Waktu Lalu Lintas di Simpang Empat Air Putih Samarinda

Leniy Eka Watiy<sup>1</sup>, Syaripuddin Syaripuddin<sup>2,\*</sup>, Qonita Qurrota A'yun<sup>3</sup>

<sup>1,2,3</sup> *Laboratorium Matematika Komputasi, Program Studi Matematika, Jurusan Matematika, FMIPA Universitas Mulawarman*

Dikirim: Maret 2023;

Diterima: Maret 2023;

Dipublikasi: Maret 2023

Alamat Email Korespondensi: [syarifrahman2014@gmail.com](mailto:syarifrahman2014@gmail.com)

### Abstrak

Aljabar max-plus dapat digunakan untuk memodelkan dan menganalisis masalah jaringan secara aljabar. Tujuan penelitian ini adalah untuk mengetahui penerapan aljabar max-plus pada pengaturan durasi waktu lampu lalu lintas di simpang empat Air Putih Samarinda. Data yang digunakan pada penelitian ini merupakan data primer berupa durasi waktu lampu lalu lintas. Kemudian, disusun graf yang menggambarkan kondisi persimpangan dan merepresentasikan arah dari pergerakan masing-masing jalur. Selanjutnya disusun aturan sinkronisasi yang sesuai dengan graf sehingga diperoleh model aljabar max-plus. Dari model aljabar max-plus, dapat diperoleh pengaturan waktu nyala lampu lalu lintas dan keperiodikannya berdasarkan hasil perhitungan vektor eigen dan nilai eigen model. Durasi waktu lampu lalu lintas yang diperoleh dari hasil perhitungan ini akan dianalisis dengan membandingkan hasil perhitungan dengan data awal.

### Kata Kunci:

*Aljabar max-plus, Algoritma power, Lampu lalu lintas*

### PENDAHULUAN

Kemacetan lalu lintas telah terjadi di mana-mana terutama di kota-kota besar [1]. Meningkatnya jumlah kendaraan menyebabkan bertambahnya volume kendaraan yang mengakibatkan kemacetan di jam-jam tertentu seperti pagi hari dan sore hari [5]. Salah satu titik rawan terjadinya kemacetan adalah persimpangan. Banyak konflik lalu lintas yang mungkin terjadi di persimpangan. Untuk mengurangi konflik tersebut maka perlu dilakukan pengaturan lalu lintas guna mengoptimalkan fungsi simpang [10]

Salah satu usaha untuk mengurangi terjadinya konflik di persimpangan jalan adalah dengan dipasang lampu lalu lintas (*traffic light*) [4]. Akan tetapi masih banyak ditemui lampu lalu lintas dengan durasi waktu lampu merah yang lama dan durasi waktu lampu hijau yang singkat. Hal ini menimbulkan antrean yang menumpuk sehingga terjadi kemacetan. Oleh karena itu dibutuhkan sebuah pengoptimalisasian dan pengintegrasian pengaturan lampu lalu lintas, khususnya di persimpangan yang memiliki kepadatan kendaraan yang tinggi.

Aljabar max-plus dapat digunakan untuk memodelkan dan menganalisis masalah jaringan secara aljabar, seperti masalah penjadwalan dan sistem antrean. Aljabar max-plus merupakan semifield komutatif atas himpunan  $\mathbb{R} \cup \{-\infty\}$  dengan  $\mathbb{R}$  himpunan semua bilangan riil yang dilengkapi dengan operasi maksimum, dinotasikan dengan  $\oplus$  dan operasi penjumlahan yang dinotasikan dengan  $\otimes$  [8].

Terdapat beberapa penelitian yang sudah dilaksanakan terkait masalah lalu lintas menggunakan aljabar max-plus. Misalnya [6] meneliti tentang penggunaan aljabar max-plus dalam pengaturan waktu nyala lampu lalu lintas, kemudian [3] meneliti tentang analisis durasi nyala lampu lalu lintas pada persimpangan berdekatan dengan penerapan aljabar max-plus dan [10] meneliti tentang penerapan aljabar max-plus pada pengaturan sistem antrean traffic light.

Berdasarkan uraian di atas, penulis ingin mengkaji lebih dalam mengenai sistem persamaan linear aljabar max-plus serta penerapannya pada sistem antrian lampu lalu lintas dan penerapannya pada pengaturan durasi waktu lampu lalu lintas. Studi teori Aljabar max-plus digunakan untuk memodelkan dan menganalisis masalah jaringan secara aljabar, seperti masalah penjadwalan dan sistem antrean. Berdasarkan karakteristik permasalahan tersebut penelitian ini dikaji dengan menggunakan model aljabar max-plus untuk menyimulasikan pengaturan lampu lalu lintas di simpang empat Air Putih Samarinda.

Batasan masalah pada penelitian ini adalah diasumsikan kondisi lalu lintas pada waktu sibuk pagi dan sore yaitu pukul 07.00 – 09.00 WITA dan 15.00-17.00 WITA pada hari kerja dan tidak ada kegiatan besar di sekitar lokasi dan faktor penyebab kemacetan dalam penelitian ini dibatasi pada durasi waktu lampu lalu lintas. Tujuan dari penelitian ini adalah untuk mengetahui penerapan aljabar max-plus pada pengaturan durasi waktu lampu lalu lintas di simpang empat Air Putih Samarinda yang diharapkan dapat bermanfaat untuk meminimalisir kemacetan.

## METODE PENELITIAN

Penelitian ini merupakan penelitian dengan pengembangan ilmu aljabar max-plus. Pada penelitian ini dipilih simpang empat Air Putih Samarinda sebagai objek penelitian. Simpang ini dipilih karena merupakan pertemuan arus kendaraan dari empat ruas jalan utama di Kota Samarinda, yaitu jalan Ir. H. Juanda, jalan P. Antasari, jalan Mayjend. MT. Haryono dan jalan P. Suryanata. Pada jam sibuk pagi dan sore sering terjadi antrean kendaraan di simpang tersebut. Data diperoleh dari observasi langsung ke lapangan. Pada penelitian ini tahapan analisis data yang dilakukan adalah sebagai berikut:

1. Mengkonstruksi graf berarah pada sistem lampu lalu lintas.  
Penelitian dimulai dari penggambaran sistem lampu lalu lintas yang kemudian dikonstruksi ke dalam bentuk graf yang menjelaskan keadaan dari sistem tersebut. Kemudian ditentukan aturan arah sistem yang berlaku dari awal sampai akhir sistem. Dari graf berarah kemudian disusun model aljabar max-plus (aturan sinkronisasi) pada sistem lampu lalu lintas.
2. Mengkonstruksi model berdasarkan aturan aljabar max-plus  $x(k + 1) = A \otimes x(k)$ .  
Aturan aljabar max-plus terutama pada penerapannya di sistem antrean dan penjadwalan memiliki bentuk  $(k + 1) = A \otimes x(k)$ . Sistem tersebut menjadi acuan untuk penyesuaian model persimpangan dan membentuk matriks sesuai dengan persamaan yang telah diperoleh.
3. Menghitung nilai eigen dan vektor eigen atas matriks aljabar max-plus.  
Proses selanjutnya yaitu mencari nilai eigen dan vektor eigen atas matriks yang menggambarkan kondisi sistem lampu lalu lintas dengan menggunakan algoritma

khusus pada aljabar max-plus yang menjadi pembahasan awal pada penelitian ini, yaitu dengan menggunakan algoritma power. Algoritma power dipilih karena memudahkan proses pencarian nilai eigen dan vektor eigen dari suatu matriks dengan langkah-langkah yang sudah ditentukan.

4. Membandingkan hasil durasi lampu lalu lintas lama dengan durasi baru.  
Proses terakhir yaitu analisis hasil dari sistem lampu lalu lintas untuk mengetahui durasi waktu lampu lalu lintas lama dengan yang baru

## HASIL DAN PEMBAHASAN

Sebelum membahas lebih jauh mengenai pemodelan aljabar max-plus akan diberikan terlebih dahulu definisi dan notasi dari aljabar max-plus sebagai berikut.

**Definisi 1.** [9] Untuk  $\mathbb{R}$  himpunan semua bilangan riil, diberikan  $\mathbb{R}_{max} = \mathbb{R} \cup \{\varepsilon\}$  dengan  $\varepsilon := -\infty$  dan  $e := 0$ . Untuk setiap  $x, y \in \mathbb{R}_{max}$ , didefinisikan operasi  $\oplus$  dan  $\otimes$  sebagai berikut

$$\begin{aligned}x \oplus y &:= \max\{x, y\} \\ x \otimes y &:= x + y\end{aligned}$$

Selanjutnya, himpunan  $\mathbb{R}_{max}$  dengan operasi  $\oplus$  dan  $\otimes$  sebagai mana didefinisikan di atas disebut dengan Aljabar Max-Plus, dinotasikan dengan  $(\mathbb{R}_{max}, \oplus, \otimes)$ .

Operasi  $\oplus$  dan  $\otimes$  pada  $\mathbb{R}_{max}$  dapat diperluas untuk operasi-operasi matriks dalam  $\mathbb{R}_{max}^{n \times m}$  seperti dalam definisi berikut.

**Definisi 2.** [7] Diberikan  $\mathbb{R}_{max}^{n \times m} := \{A = (A_{ij}) | A_{ij} \in \mathbb{R}_{max}, i \in \underline{n} \text{ dan } j \in \underline{m}\}$

i) Diketahui  $\alpha \in \mathbb{R}_{max}, A, B \in \mathbb{R}_{max}^{n \times m}$ . Didefinisikan

$$\alpha \otimes A = \alpha \otimes a_{i,j}$$

ii) Diketahui skalar  $\alpha \in \mathbb{R}_{max}, A, B \in \mathbb{R}_{max}^{n \times m}$ . Didefinisikan

$$\begin{aligned}[A \oplus B]_{i,j} &= a_{i,j} \oplus b_{i,j} \\ &= \max\{a_{i,j}, b_{i,j}\}\end{aligned}$$

iii) Diketahui  $A \in \mathbb{R}_{max}^{n \times p}$  dan  $B \in \mathbb{R}_{max}^{p \times m}$ . Didefinisikan

$$\begin{aligned}[A \otimes B]_{i,j} &= \bigoplus_{k=1}^p a_{i,k} \otimes b_{k,j} \\ &= \max_{k \in \underline{p}} \{a_{i,k}, b_{i,j}\}\end{aligned}$$

iv) Diketahui  $k \in \mathbb{N}, k \neq 0, A \in \mathbb{R}_{max}^{n \times n}$ . Didefinisikan

$$\begin{aligned}[A^{\otimes k}]_{i,j} &= \bigoplus_{r_{k-1}=1}^n a_{i,r_{k-1}} \dots \left( \bigoplus_{r_1=1}^n (a_{r_2,r_1} \otimes a_{r_1,j}) \right) \\ &= \max_{1 \leq r_1, \dots, r_{k-1} \leq n} \{a_{i,r_{k-1}} + \dots + a_{r_2,r_1} + a_{r_1,j}\}\end{aligned}$$

Nilai eigen dan vektor eigen yang bersesuaian dari matriks bujursangkar  $A$  berukuran  $n \times n$  sebagaimana dijumpai dalam aljabar linear juga dijumpai dalam aljabar max-plus, yaitu bila diberikan suatu persamaan:

$$A \otimes x = \lambda \otimes x$$

Nilai eigen dari matriks persegi  $A$  diberikan dalam teorema berikut.

**Teorema 3.** [9] Jika keadaan awal  $x(0) \neq \varepsilon$  sistem persamaan memenuhi  $x(p) = c \otimes x(q)$  untuk beberapa bilangan bulat  $p$  dan  $q$  dengan  $p > q \geq 0$  dan beberapa bilangan riil  $c$ , maka

$$\lim_{k \rightarrow \infty} \frac{x(k)}{k} = (\lambda \lambda \dots \lambda)^T$$

dengan  $\lambda = \frac{c}{p-q}$  selanjutnya  $\lambda$  adalah suatu nilai eigen dari matriks  $A$  dengan vektor eigen diberikan oleh

$$\mathbf{v} = \bigoplus_{i=1}^{p-q} \left( \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right)$$

Bukti. Misalkan  $l = p - q$ , didapat

$$\begin{aligned} \lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} &= \lim_{k \rightarrow \infty} \frac{\mathbf{x}(q+il)}{q+il} \\ &= \lim_{k \rightarrow \infty} \frac{c^{\otimes i} \otimes \mathbf{x}(q)}{q+il} \\ &= \left( \lim_{k \rightarrow \infty} \frac{ic}{q+il} \right) \otimes \left( \lim_{k \rightarrow \infty} \frac{\mathbf{x}(q)}{q+il} \right) \\ &= \left( \lim_{k \rightarrow \infty} \frac{c^{\otimes i}}{q+il} \right) \otimes \left( \lim_{k \rightarrow \infty} \frac{\mathbf{x}(q)}{q+il} \right) \\ &= \frac{c}{l} \otimes \mathbf{0} \text{ (sebab } \mathbf{x}(q) \in \mathbb{R}^n \text{)} \end{aligned}$$

Dengan vektor

$$\mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Jika bila  $\lambda = \frac{c}{p-q}$ , maka vektor waktu *cycle* adalah

$$\lim_{k \rightarrow \infty} \frac{\mathbf{x}(k)}{k} = \begin{bmatrix} \lambda \\ \lambda \\ \vdots \\ \lambda \end{bmatrix}$$

Selanjutnya bila

$$\mathbf{v} = \bigoplus_{i=1}^{p-q} \left( \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right),$$

Maka

$$\begin{aligned} A \otimes \mathbf{v} &= A \otimes \left( \bigoplus_{i=1}^{p-q} \left( \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right) \right) \\ &= \bigoplus_{i=1}^{p-q} A \otimes \left( \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i-1) \right) \\ &= \bigoplus_{i=1}^{p-q} \lambda^{\otimes(p-q-i)} \otimes (A \otimes \mathbf{x}(q+i-1)) \\ &= \bigoplus_{i=1}^{p-q} \lambda^{\otimes(p-q-i)} \otimes \mathbf{x}(q+i) \\ &= \bigoplus_{j=2}^{p-q+1} \lambda^{\otimes(p-q-j+1)} \otimes \mathbf{x}(q+j-1) \\ &= \lambda \otimes \left( \bigoplus_{j=2}^{p-q+1} \left( \lambda^{\otimes(p-q-j)} \otimes \mathbf{x}(q+j-1) \right) \right) \end{aligned}$$

$$= \lambda \otimes \left( \bigoplus_{j=1}^{p-q} \left( \lambda^{\otimes(p-q-j)} \otimes x(q+j-1) \right) \right) = \lambda \otimes v$$

Persamaan terakhir diperoleh dari

$$x(p) = \lambda^{\otimes(p-q)} \otimes x(q)$$

Yang berakibat bahwa

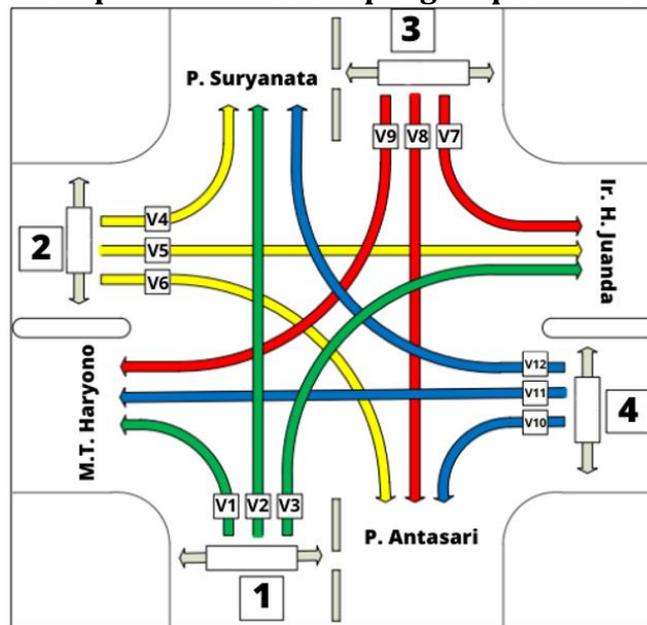
$$\lambda^{\otimes(-1)} \otimes x(p) = \lambda^{\otimes(p-q-1)} \otimes x(q) \quad \blacksquare$$

Berdasarkan Teorema 3 dapat dijabarkan menjadi suatu algoritma yang dapat digunakan untuk menentukan nilai eigen dan vektor eigen suatu matriks persegi yaitu algoritma power. Algoritma ini mempermudah penyelesaian model aljabar max-plus.

**Algoritma Power [8]**

1. Memulai dari sebarang vektor awal  $x(0) \neq \varepsilon$ .
2. Iterasi  $x(k + 1) = A \otimes x(k)$  sampai ada bilangan bulat  $p > q \geq 0$  dan bilangan rill  $c$  sehingga suatu perilaku periodik terjadi, yaitu  $x(p) = c \otimes x(q)$ .
3. Mencari nilai  $\lambda = \frac{c}{p-q}$ .
4. Mencari calon vektor eigen dengan  $v = \bigoplus_{i=1}^{p-q} \left( \lambda^{\otimes(p-q-i)} \otimes x(q+i-1) \right)$ .
5. Tunjukkan bahwa  $A \otimes v = \lambda \otimes v$  Jika terpenuhi maka  $v$  merupakan vektor eigen yang bersesuaian dengan nilai eigen  $\lambda$  selanjutnya algoritma berhenti.
6. Jika tidak, mulai  $x(k + 1) = A \otimes x(k)$  dengan  $x(0) = v$  sampai ada bilangan  $r \geq 0$ , sehingga  $x(r + 1) = \lambda \otimes x(r)$ . Maka  $x(r)$  merupakan vektor eigen yang bersesuaian dengan nilai eigen  $\lambda$ .

**Pembahasan Sistem Lampu Lalu Lintas Simpang Empat Air Putih**

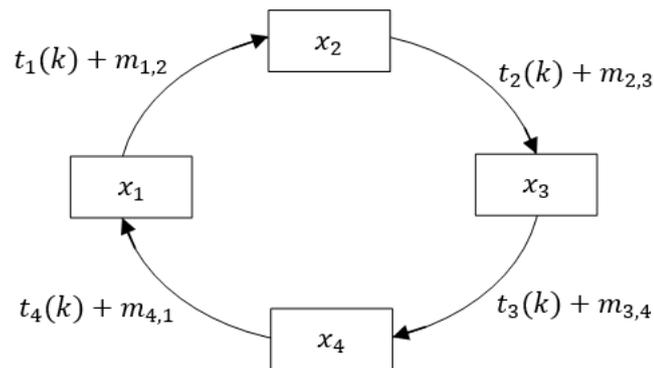


Gambar 1. Sistem Lampu Lalu Lintas Simpang Empat Air Putih Samarinda  
 Gambar 1 merepresentasikan arus-arus pada simpang empat Air Putih Samarinda dan aturan yang diterapkan pada pergerakan arus simpang tersebut. Jenis data yang dikumpulkan antara lain data durasi waktu hijau, durasi waktu kuning, durasi waktu merah dan durasi waktu selang lampu lalu lintas. Pada penelitian ini yang menjadi populasi adalah durasi waktu lampu lalu lintas di simpang empat Air Putih Samarinda.

Tabel 1. Durasi Waktu Lampu Lalu Lintas yang Diterapkan Sekarang

Simpang	Durasi Waktu Hijau (detik)	Durasi Waktu Kuning (detik)	Durasi Waktu Selang (detik)	Durasi Waktu Merah (detik)
1	38	5	6	143
2	33	5	6	148
3	48	5	6	133
4	38	5	6	143
<b>Total 1 Fase</b>			186 detik	

Data primer Tabel 1 tersebut diambil pada 1 Maret 2022, pukul 07.00-08.30 WITA dan pukul 16.00-17.30 WITA. Dari dua kali pengamatan diketahui bahwa tidak terdapat perbedaan durasi waktu lampu lalu lintas antara pagi hari dan sore hari.



Gambar 2. Graf Bobot Sistem Lampu Lalu Lintas Simpang Empat Air Putih Samarinda

Graf bobot pada Gambar 2 menunjukkan bahwa sistem simpang empat Air Putih Samarinda terhubung dengan kuat karena terdapat lintasan pada setiap titik dari fase awal ke fase akhir suatu siklus. Selanjutnya akan dikonstruksi model aljabar max plus (aturan sinkronisasi) pada sistem lampu lalu lintas yang dirancang dalam graf bobot.

Dalam mengkonstruksi model pada sistem lampu lalu lintas, didefinisikan:

$x_i(k)$  = Waktu awal hijau menyala pada simpang  $i$  di fase ke- $k$ ,  $i \in 1,2,3,4$ ,  $k \in \mathbb{R}$

$t_i(k)$  = Durasi waktu hijau pada simpang  $i$  di fase ke- $k$ ,  $i \in 1,2,3,4$ ,  $k \in \mathbb{R}$

$m_{i,j}$  = Durasi waktu selang antara akhir lampu hijau simpang  $i$  dan awal lampu hijau simpang  $j$ ,  $i, j \in 1,2,3,4$

Berdasarkan variabel tersebut, dapat dibentuk model matematika sebagai berikut:

$$\begin{aligned}
 x_1(k+1) &= \max(x_4(k) + t_4(k) + m_{4,1}, -\infty) \\
 x_2(k+1) &= \max(x_1(k) + t_1(k) + m_{1,2}, -\infty) \\
 x_3(k+1) &= \max(x_2(k) + t_2(k) + m_{2,3}, -\infty) \\
 x_4(k+1) &= \max(x_3(k) + t_3(k) + m_{3,4}, -\infty)
 \end{aligned} \tag{1}$$

untuk  $k = 0,1,2,3,4$

Model (1) dapat ditulis dengan notasi aljabar max-plus sebagai berikut:

$$\begin{aligned}
 x_1(k+1) &= x_4(k) \otimes t_4(k) \otimes m_{4,1} \oplus -\infty \\
 x_2(k+1) &= x_1(k) \otimes t_1(k) \otimes m_{1,2} \oplus -\infty \\
 x_3(k+1) &= x_2(k) \otimes t_2(k) \otimes m_{2,3} \oplus -\infty \\
 x_4(k+1) &= x_3(k) \otimes t_3(k) \otimes m_{3,4} \oplus -\infty
 \end{aligned} \tag{2}$$

untuk  $k = 0,1,2,3,4$

Model (2) dapat ditulis dalam bentuk persamaan matriks berikut:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & t_4(k) + m_{4,1} \\ t_1(k) + m_{1,2} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & t_2(k) + m_{2,3} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & t_3(k) + m_{3,4} & \varepsilon \end{bmatrix} \otimes x(k) \oplus \varepsilon$$

untuk  $k = 0, 1, 2, 3, 4$  dengan  $x(k) = [x_1(k) \ x_2(k) \ x_3(k) \ x_4(k)]^T$  dan  $\varepsilon = [\varepsilon \ \varepsilon \ \varepsilon \ \varepsilon]^T$  atau dapat ditulis

$$x(k+1) = A \otimes x(k) \oplus B$$

Dari model (2) diperoleh entri dari matriks A adalah

$$A = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & 44 \\ 39 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 54 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 44 & \varepsilon \end{bmatrix}$$

Nilai eigen dan vektor eigen atas matriks A dapat dicari dengan langkah-langkah sebagai berikut.

1. Memulai dari sebarang vektor awal  $x(0) \neq \varepsilon$ .

$$x(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

2. Iterasi  $x(k+1) = A \otimes x(k)$  sampai ada bilangan bulat  $p > q \geq 0$  dan bilangan rill  $c$  sehingga suatu perilaku periodik terjadi, yaitu  $x(p) = c \otimes x(q)$ .

Iterasi pertama

$$A \otimes x(0) = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & 44 \\ 39 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 54 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 44 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 44 \\ 39 \\ 54 \\ 44 \end{bmatrix}$$

Iterasi kedua

$$A \otimes x(1) = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & 44 \\ 39 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 54 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 44 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 44 \\ 39 \\ 54 \\ 44 \end{bmatrix} = \begin{bmatrix} 88 \\ 83 \\ 93 \\ 98 \end{bmatrix}$$

Iterasi tersebut berlanjut sampai menemukan kondisi  $x(p) = c \otimes x(q)$  dalam algoritma power.

$x(0)$	$x(1)$	$x(2)$	$x(3)$	$x(4)$	$x(5)$	$\dots$
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 44 \\ 39 \\ 54 \\ 44 \end{bmatrix}$	$\begin{bmatrix} 88 \\ 83 \\ 93 \\ 98 \end{bmatrix}$	$\begin{bmatrix} 142 \\ 127 \\ 137 \\ 137 \end{bmatrix}$	$\begin{bmatrix} 181 \\ 181 \\ 181 \\ 181 \end{bmatrix}$	$\begin{bmatrix} 225 \\ 220 \\ 235 \\ 225 \end{bmatrix}$	$\dots$

Dari proses iterasi yang dilakukan, maka didapatkan hasil yang memenuhi algoritma power yaitu

$$\begin{bmatrix} 181 \\ 181 \\ 181 \\ 181 \end{bmatrix} = 181 \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow x(4) = 181 \otimes x(0)$$

3. Mencari nilai  $\lambda = \frac{c}{p-q}$ .

$$\lambda = \frac{c}{p-q} = \frac{181}{4-0} = 45,25$$

4. Mencari calon vektor eigen dengan  $v = \bigoplus_{i=1}^{p-q} (\lambda^{\otimes(p-q-i)} \otimes x(q+i-1))$ .

Jadi vektor eigen yang bersesuaian adalah

$$v = \bigoplus_{i=1}^4 (45,25^{\otimes 4-i} \otimes x(i-1)) = \begin{bmatrix} 142 \\ 137,75 \\ 144,5 \\ 143,25 \end{bmatrix}$$

5. Munjukkan bahwa  $A \otimes v = \lambda \otimes v$  Jika terpenuhi maka  $v$  merupakan vektor eigen yang bersesuaian dengan nilai eigen  $\lambda$  selanjutnya algoritma berhenti.

$$A \otimes v = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & 44 \\ 39 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 54 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 44 & \varepsilon \end{bmatrix} \otimes \begin{bmatrix} 142 \\ 137,75 \\ 144,5 \\ 143,25 \end{bmatrix} = \begin{bmatrix} 187,25 \\ 181 \\ 191,75 \\ 188,5 \end{bmatrix}$$

$$\lambda \otimes v = 45,25 \otimes \begin{bmatrix} 142 \\ 137,75 \\ 144,5 \\ 143,25 \end{bmatrix} = \begin{bmatrix} 187,25 \\ 181 \\ 191,75 \\ 188,5 \end{bmatrix}$$

dengan demikian  $A \otimes v = \lambda \otimes v$ . Jadi vektor eigen atas matriks  $A$  yang bersesuaian

dengan  $\lambda = 45,25$  adalah  $\begin{bmatrix} 142 \\ 137,75 \\ 144,5 \\ 143,25 \end{bmatrix}$ .

Vektor eigen yang diperoleh digunakan untuk menentukan waktu awal nyala lampu lalu lintas. Waktu awal nyala lampu lalu lintas ini diperoleh dengan mengurangkan setiap elemen vektor eigen dengan elemen vektor eigen terkecil yaitu 137,75 kemudian hasil pengurangannya dibulatkan ke satuan terdekat. Sedangkan, nilai eigen yang diperoleh digunakan untuk menentukan keperiodikan waktu nyala lampu lalu lintas tersebut. Pembulatan juga dilakukan terhadap nilai eigen yang menggambarkan periode waktu nyala lampu lalu lintas sehingga diperoleh periode 45 detik. Pengaturan waktu nyala lampu lalu lintas untuk setiap fase dituliskan pada Tabel 2 dengan waktu awal nyala lampu lalu lintas aliran kendaraan pada fase ke-1 yaitu 0:00:00. Pada Tabel 2, waktu nyala hijau ditandai dengan warna hijau dan waktu nyala merah ditandai dengan warna merah.

Tabel 2. Waktu Periodik Lampu Lalu Lintas (menit.detik)

Simpang	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	dst
1	00.04	00.49	01.34	02.19	03.04	...
2	00.00	00.45	01.30	02.15	03.00	...
3	00.07	00.52	01.37	02.22	03.07	...
4	00.06	00.51	01.36	02.21	03.06	...

Dari analisis model simpang Simpang empat Air Putih Samarinda diperoleh durasi lampu lalu lintas baru sebagai berikut.

Tabel 3. Durasi Waktu Hijau Baru (detik)

Simpang $i$	Akhir Hijau Menyala Pada Simpang $i+1$	Awal Hijau Menyala Pada Simpang $i$	Durasi Waktu Selang	Durasi Waktu Hijau
1	45	4	6	$45 - 4 - 6 = 35$
2	97	45	6	$97 - 45 - 6 = 46$
3	141	97	6	$141 - 97 - 6 = 38$
4	184	141	6	$184 - 141 - 6 = 37$

**PENUTUP**

Waktu nyala lampu lalu lintas di persimpangan dapat dimodelkan menggunakan aljabar max-plus. Penentuan durasi waktu nyala lampu lalu lintas menggunakan aljabar max-plus memerlukan dua input, yaitu durasi waktu hijau dari masing-masing simpang dan durasi waktu selang antar simpang. Dari model aljabar max-plus dapat ditentukan vektor eigen dan nilai eigen yang akan digunakan untuk menentukan waktu nyala lampu lalu lintas serta keperiodikannya.

Saran diharapkan pada penelitian selanjutnya dapat menggunakan waktu rekayasa dan menambahkan faktor-faktor lain seperti volume kendaraan, lebar badan jalan, panjang antrian kendaraan dan lain-lain yang menyebabkan terjadinya antrian kendaraan di simpang empat Air Putih Samarinda agar hasil perhitungan dapat dibandingkan.

**DAFTAR PUSTAKA**

- [1] Hanna, R., Kreindler, G. dan Olken, B.A. (2017). Citywide effects of high-occupancy vehicle restrictions: Evidence from "three-in-one" in Jakarta. *Journal Science*. 357(6346), 89-93.
- [2] Hurit, R.U. dan Rudhito, A. (2019). Max-plus algebraic modeling of three crossroad traffic queue systems with one underpass. *Journal of Physics: Conference Series* 1307 012013.
- [3] Iswati, R.P.E. (2017). *Analisis Durasi Nyala Lampu Lalu Lintas pada Persimpangan Berdekatan Dengan Penerapan Aljabar Max-Plus (Skripsi)*. Universitas Negeri Yogyakarta.
- [4] Lazuardi, Rizki, Musthofa. (2013). Pengaturan Durasi Waktu Nyala Lampu Lalu Lintas di Persimpangan Gondomanan dengan Menggunakan Aljabar Max-Plus. *Jurnal Pendidikan Matematika dan Sains*, 2(4).
- [5] Mufidah, N. (2018). *Pemodelan Antrean Kendaraan Bermotor Menggunakan Model Antrean M/M/1 di Simpang Tiga Ringroad Utara Yogyakarta Pada Pagi Hari dan Sore Hari (Skripsi)*. Universitas Islam Indonesia.
- [6] Pradanti, P. dan Sari, M.R.A. (2016). Penggunaan aljabar max-plus dalam pengaturan waktu nyala lampu lalu lintas. *Prosiding Seminar Nasional Aljabar, Penerapan dan Pembelajarannya*. Universitas Sanata Dharma.
- [7] Prastiwi, L. dan Listiana, Y. The Application of Max-Plus Algebra to Determine The Optimal Time of Ikat Kupang Woven Production. *International Journal Of Computing Science And Applied Mathematics*, 3(2), 77-80.
- [8] Subiono. (2015). *Aljabar Min-Max Plus dan Terapannya (Modul)*. Surabaya: Jurusan Matematika FMIPA ITS.
- [9] Wibowo, A., Wijayanti, K. dan Veronica, R. (2018). Penerapan Aljabar Max-Plus pada Pengaturan Sistem Antrean Traffic Light. *UNNES Journal of Mathematics*, 7(2), 192-205.

## Penerapan *Minimum Spanning Tree* dalam Menentukan Rute Terpendek Distribusi Naskah Soal USBN di SMA Negeri se-Sleman

Deddy Rahmadi <sup>1,\*</sup>, Hardina Sandariria <sup>2</sup>

<sup>1</sup> Program Studi Matematika, Universitas Islam Negeri Sunan Kalijaga Yogyakarta

<sup>2</sup> Program Studi Matematika, Universitas Mulawarman Samarinda

Dikirim: Januari 2023; Diterima: Maret 2023; Dipublikasi: Maret 2023

Alamat Email Korespondensi: [deddy.rahmadi@uin-suka.ac.id](mailto:deddy.rahmadi@uin-suka.ac.id)

### Abstrak.

Perkembangan teknologi merupakan hal yang tidak dapat dihindari sehingga matematika menjadi salah satu pilar yang sangat penting. Salah satu penerapan matematika dalam perkembangan teknologi adalah menentukan jalur optimal pada suatu graf. Tujuan dari penelitian ini adalah untuk menentukan jalur optimal untuk distribusi naskah guna menunjang pelaksanaan USBN di SMA Negeri se-Sleman. Algoritma Prim adalah sebuah algoritma dalam teori graf yang mencari *minimum spanning tree* untuk sebuah graf berbobot dan terhubung. Algoritma Prim merupakan salah satu algoritma untuk menentukan jalur terpendek dari lokasi salah satu SMA menuju SMA lainnya.

### Kata Kunci:

*prim, graf, tree, jalur terpendek, usbn*

## PENDAHULUAN

Seiring dengan perkembangan teknologi di zaman sekarang, penerapan ilmu pengetahuan di bidang matematika sangat berkembang nyata, khususnya dalam menentukan jalur terpendek pada suatu graf. Teori Graf banyak digunakan dalam kehidupan sehari-hari, khususnya dalam menentukan jalur terpendek [5], [2].

Graf merupakan salah satu cabang ilmu dari matematika diskrit. Permasalahan pada graf yang dapat menentukan rute terpendek adalah *Minimum Spanning Tree* [6] [7]. *Minimum Spanning Tree* (MST) merupakan salah satu cara untuk menentukan rute terpendek, dimana kita harus memilih bobot dari sisi-sisi yang paling minimal sehingga semua titik pada graf dapat terhubung [6]. Pada tahun 1956 dan 1957 Joseph B. Kruskal dan Robert C. Prim masing-masing menggambarkan algoritma yang jauh lebih efisien untuk membangun *minimum spanning tree* [8]. Bahkan untuk graf yang cukup besar, kedua algoritma dapat diimplementasikan sehingga memakan waktu komputasi yang relatif singkat.

Beberapa permasalahan yang dapat diselesaikan dengan MST adalah menentukan rute terpendek, persoalan minimasi jaringan, persoalan aliran maksimum. Permasalahan dalam menentukan rute terpendek telah diterapkan dalam beberapa kasus, yaitu optimalisasi pemasangan pipa air [4], pemasangan jaringan listrik [1], dan distribusi bahan naskah USBN di SD/MI [7].

Ujian Sekolah Berstandar Nasional (USBN) dahulu dikenal sebagai Ujian Sekolah (US). Perbedaan yang mendasar terletak pada standar yang dipergunakan. USBN menggunakan standar nasional, sedangkan US menggunakan standar sekolah [3]. Dikarenakan menggunakan standar nasional, maka lembar soal USBN didistribusikan dari pusat balai pendidikan ke sekolah-sekolah. Oleh sebab itu diperlukan rute terpendek untuk mendistribusikan soal.

Dalam menentukan rute terpendek, dapat menggunakan algoritma Prim dan algoritma Kruskal. Adapun perbedaan dari keduanya yaitu algoritma Prim berorientasi pada titik graf, sedangkan algoritma Kruskal berorientasi pada bobot sisi graf. Pada penelitian ini, peneliti menggunakan metode algoritma Prim karena lebih efektif saat graf memiliki banyak sisi dengan titik yang sedikit (graf lengkap), sedangkan algoritma Kruskal lebih efektif saat graf memiliki banyak titik dengan sisi yang sedikit.

Berdasarkan latar belakang di atas, tujuan dari penelitian ini adalah mengetahui bentuk graf dari jaringan peta antar SMA Negeri se-Sleman dan mengetahui jarak terpendek antar SMA Negeri se-Sleman sehingga dapat meminimalkan biaya.

## **METODE PENELITIAN**

Pada penelitian ini terdapat langkah-langkah penelitian yang dilakukan. Adapun langkah-langkahnya adalah sebagai berikut :

1. Melakukan pengambilan data peta SMA Negeri se-Sleman dan Balai Pendidikan Menengah Kabupaten Sleman dan mencari jarak antara masing-masing lokasi (dalam kilometer).
2. Melakukan pembentukan Graf awal dari data peta tersebut.
3. Melakukan pencarian *Minimum Spanning Tree* dari Graf yang diperoleh dengan menggunakan Algoritma Prim (manual) dan melakukan perhitungan komputasi menggunakan *software* Python.
4. Melakukan perbandingan hasil antara pencarian MST menggunakan Algoritma Prim dengan Graf peta SMA Negeri se-Sleman.
5. Didapatkan perbandingan antara jumlah bobot awal graf sebelum dan sesudah dilakukan pencarian MST, kemudian dilakukan penarikan kesimpulan.

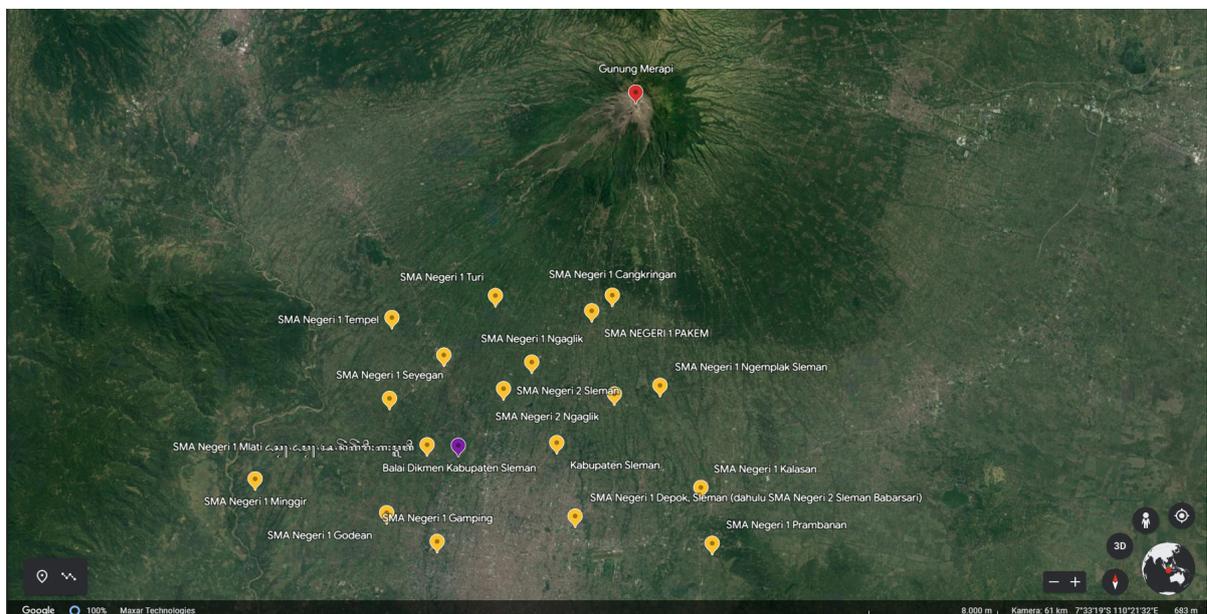
## **HASIL DAN PEMBAHASAN**

Penelitian ini bertujuan untuk menentukan jalur optimal untuk distribusi naskah USBN, dimana permasalahan pada penelitian ini adalah mencari *Minimum Spanning Tree* untuk distribusi naskah USBN agar menempuh jarak seminimal mungkin dengan menggunakan algoritma Prim.

Berikut adalah langkah-langkah dalam melakukan Algoritma Prim, yaitu:

1. Menentukan titik awal kemudian ambil sisi dari titik tersebut pada graf  $G$  yang mempunyai bobot terkecil, masukkan ke dalam subgraf  $T$ .
2. Selanjutnya pilih sisi  $e$  yang mempunyai bobot minimum dan bersisian dengan simpul di  $T$ , tetapi  $e$  tidak membentuk *circuit* di  $T$ .
3. Masukkan ke dalam  $T$ , Ulangi langkah kedua sebanyak  $(n-2)$  kali.

Gambar 1 merupakan persebaran SMA Negeri di Kabupaten Sleman. Terdapat 17 SMA Negeri yang dilambangkan dengan titik berwarna kuning dan Balai Pendidikan Menengah Kab. Sleman dilambangkan dengan titik berwarna ungu.



Gambar 1. Peta Persebaran SMA Negeri di Kabupaten Sleman

Diasumsikan Balai Pendidikan Menengah (Baldikmen) dan SMA Negeri se-Sleman adalah titik dan jalan yang menghubungkannya adalah sisi yang memiliki bobot. Diperoleh hasil pengamatan dari peta pada Gambar 1 seperti yang tertera pada Tabel 1.

Tabel 1. Kode huruf dan tempat distribusi naskah USBN

No	Kode Huruf	Keterangan
1.	A	SMA N 1 Cangkringan
2.	B	SMA N 1 Depok
3.	C	SMA N 1 Gamping
4.	D	SMA N 1 Godean
5.	E	SMA N 1 Kalasan
6.	F	SMA N 2 Minggir
7.	G	SMA N 1 Mlati
8.	H	SMA N 1 Ngaglik
9.	I	SMA N 2 Ngaglik
10.	J	SMA N 1 Ngemplak
11.	K	SMA N 1 Pakem
12.	L	SMA N 1 Prambanan
13.	M	SMA N 1 Seyegan
14.	N	SMA N 1 Sleman
15.	O	SMA N 2 Sleman
16.	P	SMA N 1 Tempel
17.	Q	SMA N 1 Turi
18.	R	Balai Pendidikan Menengah Kab. Sleman

Berdasarkan peta, diperoleh graf awal dengan 18 titik, 153 sisi, dan bobot sebesar 2402. Kemudian dilakukan pencarian minimum spanning tree dengan menggunakan Algoritma Prim dan diperoleh 18 titik, 17 sisi, dan bobot sebesar 97,2 km. Adapun waktu yang dibutuhkan untuk proses MST dengan *software* Python adalah 0.01635515099999907 detik. Sisi yang tetap digunakan setelah diterapkan Algoritma Prim disajikan pada Tabel 2.

Tabel 2. Sisi dan Bobot hasil Algoritma Prim.

No.	Sisi	Bobot (dalam kilometer)
1.	L-A	10
2.	D-C	5,1
3.	G-B	6,3
4.	L-E	4,4
5.	D-F	8,7
6.	R-G	2,3
7.	K-H	6,8
8.	H-I	6,8
9.	I-J	3,3
10.	A-K	2,9
11.	J-L	8,7
12.	G-M	4,8
13.	M-N	5,1
14.	H-O	3,1
15.	N-P	6,2
16.	H-Q	6,5
17.	O-P	6,2

## PENUTUP

Berdasarkan hasil yang diperoleh, dengan menerapkan Algoritma Prim untuk mencari *Minimum Spanning Tree* diperoleh hasil yaitu graf dengan 18 titik, 34 sisi, dan bobot sebesar 97,2 dari graf awal yaitu 18 titik, 153 sisi, dan bobot sebesar 2402. Penghapusan sisi dilakukan karena mempertimbangkan jarak yang optimal untuk distribusi naskah USBN.

**DAFTAR PUSTAKA**

- [1] Damayanti, A.A. & Rochmad, R.A. (2013). Penerapan Algoritma Kruskal Pada Jaringan Listrik Perumahan Kampong Harmoni Di Ungaran Barat. *UNNES Journal of Mathematics*, 2(1), 9-16
- [2] Greenberg, H.J. (1998). Greedy Algorithm for Minimum Spanning Tree <http://glossary.computing.society.informs.org/notes/spanningtree.pdf>, University of Coloardo, Denver.
- [3] Mahmuzah, R., Aklimawati, Meilizza, & Asri, K. (2019). Pengaruh Nilai Ujian Sekolah Berstandar Nasional (USBN) terhadap Prestasi Belajar Siswa MTs Negeri Rukoh Kota Banda Aceh pada Mata Pelajaran Matematika. *Jurnal Serambi PTK*, VI (2), 64-69.
- [4] Ramadhan, A. F. (2017). Aplikasi Algoritma Prim dalam Penentuan Pohon Rentang Minimum untuk Jaringan Pipa PDAM Kota Tangerang. *Jurnal Ilmiah*, 2(1), 30-38.
- [5] Rosen, Kenneth H. (1997.) *Exploring Discrete Mathematics With Maple*. Singapore: McGraw-Hill Book Co. Siang.
- [6] Sari, R.F., Widyasari, R., & Marpaung, F. A. (2023). Optimasi Pemasangan Jalur Pipa Air Bersih Melalui *Minimum Spanning Tree* dengan Algoritma Prim. *G-Tech: Jurnal Teknologi Terapan*, 7(1), 70-74.
- [7] Sudibyoy, N.A., Purwanto, T., & Rahmadi, D. (2020). Minimum Spanning Tree pada Distribusi Bahan Naskah USBN SD/MI di Kabupaten Sragen, *Riemann: Research of Mathematics and Mathematics Education*, 2(2), 64-69.
- [8] Wamiliana, Usman, M., Warsono, Warsito, & Daoud, J. I. (2020). Using Modification of Prim's Algorithm and GNU Octave and to Solve the Multiperiods Installation Problem. *IIUM Engineering Journal*, 21(1), 100-112.

## Hubungan antara Indeks Pembangunan Manusia dengan Garis Kemiskinan di Sumatera Barat, Sumatera Selatan dan Bengkulu

Dea Alvionita Azka<sup>1,\*</sup>, Dia Cahyawati<sup>2</sup>

<sup>1</sup> Institut Teknologi Muhammadiyah Sumatera

<sup>2</sup> Universitas Insan Cita Indonesia

Dikirim: Maret 2023;

Diterima: Maret 2023;

Dipublikasi: Maret 2023

Alamat Email Korespondensi: dealvionitazka@gmail.com

### Abstrak

Pembangunan manusia adalah paradigma pembangunan yang menempatkan manusia sebagai fokus dan sasaran akhir dari seluruh kegiatan pembangunan yang bertujuan untuk penguasaan sumber daya guna kehidupan yang lebih layak, peningkatan derajat kesejahteraan dan Pendidikan. Ukuran untuk mengetahui sejauh mana capaian pembangunan manusia disebut dengan Indeks Pembangunan Manusia (IPM) Pembangunan manusia sendiri bukan merupakan proses yang instant, melainkan sebuah proses yang memerlukan waktu atau berkelanjutan, dan untuk menjaga pembangunan manusia yang berkelanjutan tersebut, diperlukan sumber daya (*resources*) di mana sumber daya tersebut didapatkan dari pertumbuhan ekonomi. Berbicara tentang pertumbuhan ekonomi tentu tidak bisa lepas dari kemiskinan yang terjadi di Indonesia. Terdapat perubahan garis kemiskinan di beberapa daerah dari tahun 2014, 2015 dan 2016, salah satunya adalah Sumatera Barat, Sumatera Selatan dan Bengkulu. Oleh karena ini penelitian ini bermaksud untuk menganalisis, apakah terdapat hubungan antara IPM dan garis kemiskinan suatu daerah, dengan menggunakan regresi linear multilevel dengan 2-level, yaitu kabupaten sebagai level individu (level-1) dan pulau sebagai level grup (level-2). Berdasarkan analisis data diperoleh pola hubungan antara garis kemiskinan dengan indeks pembangunan manusia di Provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu tahun 2015 membentuk garis lurus dan hubungannya positif. Artinya, semakin tinggi indeks pembangun masyarakat, maka garis kemiskinan akan semakin tinggi juga.

### Kata Kunci:

*Indeks Pembangunan Manusia, Garis Kemiskinan, Regresi Linear Multilevel*

### PENDAHULUAN

Pembangunan manusia adalah paradigma pembangunan yang menempatkan manusia sebagai fokus dan sasaran akhir dari seluruh kegiatan pembangunan. Dimana tujuannya adalah untuk penguasaan atas sumber daya guna memperoleh pendapatan untuk mencapai hidup layak, peningkatan derajat kesehatan agar membangun manusia yang sehat dan berumur panjang dan meningkatkan pendidikan terutama dalam kemampuan baca tulis dan keterampilan untuk dapat berpartisipasi dalam masyarakat dan kegiatan ekonomi [1].

Ukuran untuk mengetahui sejauh mana capaian pembangunan manusia dikenal dengan dengan Indeks Pembangunan Manusia (IPM). IPM mengukur pencapaian pembangunan manusia di suatu daerah dalam 3 dimensi dasar pembangunan manusia, yaitu *longevity*/umur panjang dan sehat, *knowledge*/pengetahuan, serta *decent living standard*/standar hidup layak [2]. Dengan berpijak pada anggapan bahwa 3 dimensi

tersebut merupakan dimensi dasar yang harus dipenuhi oleh manusia. Nilai IPM diperoleh dari agregat rata-rata ukur (geometrik) yaitu indeks kesehatan, indeks pendidikan, dan indeks pengeluaran. IPM sebagai ukuran kualitas manusia dapat digunakan untuk menganalisis status pembangunan dari tahun ke tahun dengan melihat pergeseran status, dari rendah menjadi sedang, sedang menjadi tinggi, dan tinggi menjadi sangat tinggi.

Pembangunan manusia sendiri bukan merupakan proses yang instant, melainkan sebuah proses yang memerlukan waktu atau berkelanjutan, dan untuk menjaga pembangunan manusia yang berkelanjutan tersebut, diperlukan sumber daya (resources) di mana sumber daya tersebut didapatkan dari pertumbuhan ekonomi [3].

Berbicara tentang pertumbuhan ekonomi tentu tidak bisa lepas dari kemiskinan yang terjadi di Indonesia. Badan Pusat Statistik (BPS) mengumumkan angka penduduk miskin per September 2016. Berdasarkan catatan BPS, angka penduduk miskin di Indonesia menurun sebesar 250 ribu jiwa menjadi 27,76 juta penduduk. Banyaknya penduduk miskin ini dikategorikan berdasarkan banyaknya penduduk yang berada di bawah garis kemiskinan. [4]

Garis kemiskinan adalah representasi dari jumlah rupiah minimum yang dibutuhkan untuk memenuhi kebutuhan pokok makanan dan kebutuhan pokok non makanan. Garis kemiskinan selama Maret 2016-September 2016 juga meningkat rata-rata sekitar 2,15 persen. Untuk kategori pedesaan, garis kemiskinan berada pada level Rp 350.420. Adapun kategori perkotaan berada pada level Rp 372.114. Penyebab naiknya garis kemiskinan mengikuti barang dan jasa yang dikonsumsi. Selain itu, kenaikan garis kemiskinan versi BPS menunjukkan daya beli masyarakat yang semakin tinggi. Kenaikan biaya hidup, dan kesejahteraan dan pendapatan masyarakat yang bertambah juga mempengaruhi kenaikan garis kemiskinan [5]. Terdapat perubahan garis kemiskinan di beberapa daerah dari tahun 2014, 2015 dan 2016, salah satunya adalah Sumatera Barat, Sumatera Selatan dan Bengkulu. Oleh karena ini penelitian ini bermaksud untuk menganalisis, apakah terdapat hubungan antara IPM dan garis kemiskinan suatu daerah, dengan menggunakan regresi linear multilevel dengan 2-level, yaitu kabupaten sebagai level individu (level-1) dan pulau sebagai level grup (level-2).

## **METODE PENELITIAN**

Jenis penelitian ini adalah penelitian kuantitatif dengan menggunakan data sekunder yang diperoleh dari Badan Pusat Statistik tahun 2015. Variabel tak bebas yang digunakan dalam tulisan ini adalah garis kemiskinan di provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu tahun 2015 yang di notasikan dengan GK. Sedangkan variabel bebasnya adalah indeks pembangunan masyarakat di tahun 2015. Penelitian ini melibatkan 48 kabupaten kota, di antaranya 19 kabupaten yang ada di Provinsi Sumatera Barat, 18 kabupaten/kota di Sumatera Selatan dan 11 kabupaten/kota di Provinsi Bengkulu.

Teknik analisis data menggunakan regresi linear multilevel, yaitu variabel-variabel yang diamati dibagi menjadi dua level. Level pertama untuk kabupaten/kota, dan level kedua untuk provinsi, sehingga diperoleh:

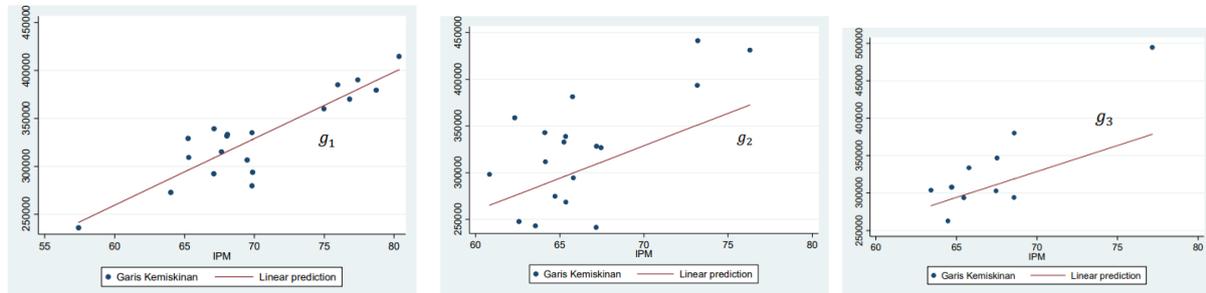
Variabel dependen ( $y_{ij}$ ) : Garis kemiskinan penduduk tahun 2015

Variabel independent ( $x_{ij}$ ) : Indeks Pembangunan Masyarakat di Provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu tahun 2015.

Selanjutnya akan dibentuk formulasi model multilevel untuk data tersebut.

## HASIL DAN PEMBAHASAN

Sebelum membuat pemodelan data, hal pertama yang harus dilakukan adalah melihat scatter plot yang terbentuk, untuk melihat sebaran data hubungan antar variabel di masing-masing provinsi. Berikut adalah hasil dari scatter plot dari kabupaten/kota masing-masing provinsi sebagaimana dinyatakan pada Gambar 1.



Gambar 1. Sebaran Data Garis Kemiskinan di Kabupaten/Kota di Sumatera Barat ( $g_1$ ), Sumatera Selatan ( $g_2$ ), dan Bengkulu ( $g_3$ )

Berdasarkan Gambar 1, 2 dan 3 di atas, jelas bahwa  $g_1, g_2, g_3$  membentuk garis lurus. maka dapat digunakan regresi linear untuk menganalisis data. Dengan kata lain, pemodelan yang digunakan adalah pemodelan multilevel regresi linear.

Selanjutnya, dilakukan penggabungan nilai estimasi *intercept* dan *slope* pada dataset untuk memisahkan data *inter* dan *slope* sesuai dengan provinsinya. Akibatnya diperoleh bahwa *intercept* dan *slope* setiap titik berbeda, sehingga dapat disimpulkan bahwa model multilevel yang sesuai untuk memodelkan kasus garis kemiskinan penduduk adalah model koefisien random.

Tahap selanjutnya akan dibentuk pemodelan multilevel secara umum dari permasalahan ini. Model multilevel ini nanti yang akan merepresentasikan model-model dari setiap provinsi. Inti dari pemodelan umum ini adalah bagaimana membentuk intercept dan slope fix dari pemodelan dengan menggunakan metode Likelihood Estimator, dimana intercept dan slope random bergantung pada masing-masing provinsi. Berdasarkan hasil analisis Maximum Likelihood Estimated (MLE) untuk model koefisien random pada pemodelan tersebut, diperoleh nilai fiks intercept dari model yaitu  $\beta_1$  adalah -333565,3 dengan standar error 115397,6. Sedangkan untuk  $\beta_2$  yaitu fix slope-nya adalah 9804,436 dengan standar errornya sebesar 17779,303. Akibatnya diperoleh model multilevel umum sebagai berikut

$$\begin{aligned}
 y_{ij} &= (\beta_1 + \zeta_{1j}) + (\beta_2 + \zeta_{2j})x_{ij} \\
 &= (-333565,3 + \zeta_{1j}) + (9804,436 + \zeta_{2j})x_{ij}
 \end{aligned} \tag{1}$$

Dengan

$y_{ij}$  : Garis kemiskinan di kabupaten/kota  $i$  yang terletak di provinsi  $j$

$x_{ij}$  : IPM di kabupaten/kota  $i$  yang terletak di provinsi  $j$

$\beta_1$  : intercept fiks model = -333565,3

$\zeta_{1j}$  : intercept random model

$\beta_2$  : slope fiks model = 9804,436

$\zeta_{2j}$  : slope random model

```

. xtmixed GK ipm || provinsi:ipm, cov(unstructured)mle

Performing EM optimization:

Performing gradient-based optimization:

Iteration 0:  log likelihood = -571.03478
Iteration 1:  log likelihood = -570.39443
Iteration 2:  log likelihood = -570.17922
Iteration 3:  log likelihood = -569.93785
Iteration 4:  log likelihood = -569.92041
Iteration 5:  log likelihood = -569.92037
Iteration 6:  log likelihood = -569.92036

Computing standard errors:

Mixed-effects ML regression              Number of obs   =       48
Group variable: provinsi                 Number of groups =        3

                                         Obs per group:
                                         min =          11
                                         avg =          16.0
                                         max =          19

                                         Wald chi2(1)    =       30.36
Log likelihood = -569.92036              Prob > chi2     =       0.0000

```

GK	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
ipm	9804.436	1779.303	5.51	0.000	6317.065	13291.81
_cons	-333565.3	115397.6	-2.89	0.004	-559740.4	-107390.2

Random-effects Parameters	Estimate	Std. Err.	[95% Conf. Interval]	
provinsi: Unstructured				
sd(ipm)	2388.017	1570.604	657.9611	8667.117
sd(_cons)	150632.2	103940.1	38955.34	582463.3
corr(ipm,_cons)	-1	1.80e-06	-1	1
sd(Residual)	33132.4	3484.364	26961.02	40716.4

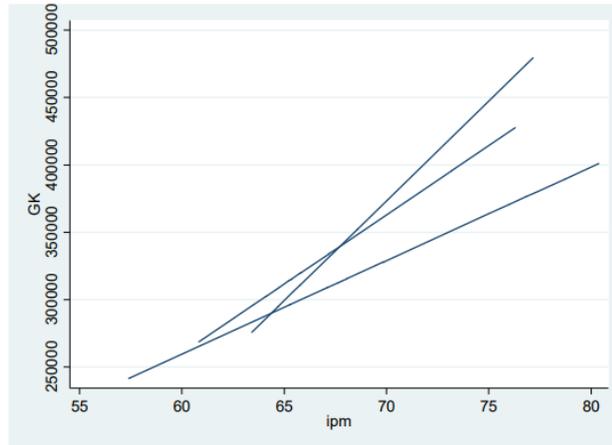
LR test vs. linear model: chi2(3) = 4.98      Prob > chi2 = 0.1735

Gambar 2. Model Multilevel secara Umum Menggunakan MLE

Nilai kovariansi antara intercept random dengan slope random bernilai negatif berarti pada persamaan garis regresi untuk setiap provinsi ( $j= 1,2,3$ ) jika nilai slope random ( $\zeta_{2j}$ ) semakin besar, mengakibatkan nilai intercept random ( $\zeta_{1j}$ ) semakin kecil dan sebaliknya jika nilai slope random ( $\zeta_{2j}$ ) semakin kecil, maka nilai intercept random ( $\zeta_{1j}$ ) semakin besar. Berdasarkan persamaan (1), nilai intercept ( $\zeta_{1j}$ ) dan nilai slope random ( $\zeta_{2j}$ ) bergantung pada masing-masing provinsi ( $j$ ), yaitu:

- i. Model (i), yaitu Model untuk Provinsi Sumatera Barat, dengan nilai  $j=1$  diperoleh  $y_{i1} = -157507,6 + 6949,265x_{i1}$
- ii. Model (ii), yaitu Model untuk Provinsi Sumatera Selatan, dengan nilai  $j=2$  diperoleh  $y_{i2} = -356287,6 + 10273x_{i2}$
- iii. Model (iii), yaitu Model untuk Provinsi Bengkulu, dengan nilai  $j=3$  diperoleh  $y_{i3} = -664419,2 + 14825,18x_{i3}$

Berdasarkan tiga model yang telah terbentuk di atas, dapat dibuat garis regresi prediksi untuk masing-masing provinsi, yaitu dengan menghitung nilai yang memenuhi model (i) untuk provinsi Sumatera Barat, model (ii) untuk provinsi Sumatera Selatan dan model (iii) untuk provinsi Bengkulu sebagaimana pada Gambar 3.



Gambar 3. Garis Regresi Prediksi untuk Provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu

Berdasarkan interpretasi di atas, apabila ketiga garis regresi untuk masing-masing provinsi digambarkan dalam satu grafik dengan sumbu Y menunjukkan garis kemiskinan dan sumbu X menunjukkan IPM, maka bentuk ketiga garisnya akan cenderung menuju ke arah yang sama untuk nilai garis kemiskinan yang membesar. Berdasarkan Gambar 3 di atas, setiap garis memiliki slope yang positif. Selain itu, karena pada model multilevel umum untuk ketiga provinsi (persamaan 1), nilai slope fiksnya ( $b_2$ ) adalah 9804,436 (positif), maka ketiga garis regresi tersebut naik untuk IPM yang membesar.

## PENUTUP

Berdasarkan hasil analisis data dan pembahasan yang telah dilakukan, maka dapat disimpulkan bahwa:

1. Pola hubungan antara garis kemiskinan dengan indeks pembangunan manusia di Provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu tahun 2015 membentuk garis lurus dan hubungannya positif. Artinya semakin tinggi indeks pembangunan masyarakat, maka garis kemiskinan akan semakin tinggi juga.
2. Berdasarkan perhitungan statistik, indeks pembangunan manusia mempengaruhi kenaikan garis kemiskinan di Provinsi Sumatera Barat, Sumatera Selatan dan Bengkulu tahun 2015, dan pengaruhnya sangat tinggi untuk masing-masing provinsi. Untuk Provinsi Sumatera Barat derajat hubungannya sebesar 0,8895, untuk Provinsi Sumatera Selatan sebesar 0,693 dan untuk untuk Provinsi Bengkulu sendiri 0,891.

## **DAFTAR PUSTAKA**

- [1] B. P. Statistik. (2004). *Laporan Pembangunan Manusia Indonesia*. Jakarta: BPS-Bappenas.
- [2] R. S. K. Siregar. (2015). *Perbandingan Regresi Nonparametrik dengan Pendekatan*. Bandung: Universitas Padjajaran.
- [3] G. d. S. F. Ranis. (2001). Growth And Human Development: Comparative Latin American Experience. *Economic Growth Center*. 826.
- [4] B. P. Statistik. (2017). *Badan Pusat Statistik*. 3 Desember 2017. [Online]. Available: <https://sirusa.bps.go.id/index.php?r=indikator/view&id=50>.
- [5] B. P. Statistik. (2017). *Badan Pusat Statistik*. 3 Desember 2017. [Online]. Available: <https://www.bps.go.id/subjek/view/id/23>.
- [6] J. Guilford. (1956). *Fundamental Statistics in Psychology and Education*. New York: McGraw.
- [7] B. d. U. Badan Pusat Statistik. (2004). *Laporan Pembangunan Manusia Indonesia 2004*. Jakarta: BPS-Bappenas-UNDP.
- [8] B. P. Statistik. (2017). *Badan pusat Statistik*. 3 Desember 2017. [Online]. Available: <https://www.bps.go.id/subjek/view/id/26>.

## The Implementation of Minimum Spanning Tree in Finding Algebraically the Shortest Path of National-Exam-Sheet Distribution in All Senior High Schools over Bantul Regency

Wakhid Fitri Albar <sup>1,\*</sup>, Deddy Rahmadi<sup>2</sup>, Katya Royhana Dewi<sup>2</sup>

<sup>1</sup> *Mathematics Department, Universitas Negeri Semarang, Indonesia*

<sup>2</sup> *Mathematics Department, Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Indonesia*

Received: March 2023; Accepted: March 2023; Published: March 2023

Correspondence Email: [wakhid.albar@mail.unnes.ac.id](mailto:wakhid.albar@mail.unnes.ac.id)

### Abstract.

Mathematics is a very important pillar in this rapidly technological developments. One application of mathematics in technological developments is determining the optimal path on a graph. The purpose of this research is to determine the optimal path for delivering national exam sheets (USBN) in SMA (public high schools) of all over Bantul regency. Kruskal's algorithm is an algorithm in graph theory that seeks a minimum spanning tree for a weighted and connected graph. Kruskal's algorithm is one of the algorithms for determining the shortest path from the location of one SMA to another SMA.

### Keywords:

*Kruskal, graph, tree, shortest path, USBN, algebra*

## INTRODUCTION

Along with today's technological developments, the application of science in the field of mathematics grows significantly, especially in determining the shortest path in a graph. Graph theory is widely used in everyday life including in determining the shortest path [3] [6].

Graph is a branch of discrete mathematics. The problem with a graph which can determine the shortest route is the Minimum Spanning Tree (MST) [4]. In MST, we have the minimum edges so that all vertices in the graph can be connected [1]. In 1956 and 1957 Joseph B. Kruskal and Robert C. Prim respectively described a much more efficient algorithm for building an MST [9]. Even for fairly large graphs, both algorithms can be implemented such that it takes relatively short computation time.

One of the problems which can be solved with MST is finding the shortest path in delivering packages. It can optimize the cost, energy, and time for the industry and its couriers. Wirabudi et. al., amplified that MST can truly increase the effectiveness in package delivery [10].

In the past, national exams (UN) and Final School Exam (US) were done to determine the graduation of students. Since the UN was demolished, then the US was

standardized to be National Level of School Exam which is known as USBN [5].

USBN exams are centralized at the capital of each province. Therefore, different province may have different questions of the exams. Since it is centered at the province, then the USBN sheets are produced at the capital of the province [7]. The sheets are then distributed to each region (Kabupaten/Kota) and are followed by distributing them to each school.

In each region, the sheets are collected at the Dinas Pendidikan or Baldikmen. The Dinas has to send the sheets to each school. Due to the various distances and landscape of the schools, the Dinas may find it difficult to deliver the sheets effectively. In this case, MST can be a powerful tool in optimizing the deliveries [8].

Based on the background, the purpose of this study is to determine the shape of the graph of the distribution of National Exam Sheets in all senior high schools over Bantul regency and to find out the shortest distance between the schools so as to minimize costs.

## **RESEARCH METHODS**

The steps which were conducted in this research were:

1. Retrieve map data for all SMAN in Bantul Regency.
2. Create the initial graph of the map data.
3. Search the Minimum Spanning Tree of the Graph obtained using the Kruskal's Algorithm (manually) and Python software.
4. Compare the results between the MST search using the Kruskal's Algorithm and the map graphs of SMA Negeri throughout Bantul.
5. Compare between the initial weights of the graph before and after the MST search is performed, then conclusions are drawn.

## **RESULT AND DISCUSSION**

This study aims to determine the optimal path for the distribution of USBN sheets, where the problem in this research is to find a Minimum Spanning Tree for the distribution of USBN sheets to cover the minimum distance possible using the Kruskal's algorithm.

It is assumed that the Middle Education Center (Baldikmen) and Bantul Public High Schools (SMAN) are the points and the roads which connect them are the sides which have weight. The results of our observations from the map are shown in the following table.

Table 1. Letter Code and School-targeted for USBN-sheet delivery.

No.	Letter Code	Place
1.	A	Balai Pendidikan Menengah Kab. Bantul
2.	B	SMA N 1 Bambanglipuro
3.	C	SMA N 1 Banguntapan
4.	D	SMA N 2 Banguntapan
5.	E	SMA N 1 Bantul
6.	F	SMA N 2 Bantul
7.	G	SMA N 3 Bantul
8.	H	SMA N 1 Dlingo
9.	I	SMA N 1 Imogiri
10.	J	SMA N 1 Jetis
11.	K	SMA N 1 Kasihan
12.	L	SMA N 1 Kretek
13.	M	SMA N 1 Pajangan
14.	N	SMA N 1 Piyungan
15.	O	SMA N 1 Pleret
16.	P	SMA N 1 Pundong
17.	Q	SMA N 1 Sanden
18.	R	SMA N 1 Sedayu
19.	S	SMA N 1 Sewon
20.	T	SMA N 1 Srandakan

The picture of the schools on the real map can be explored at the following image.

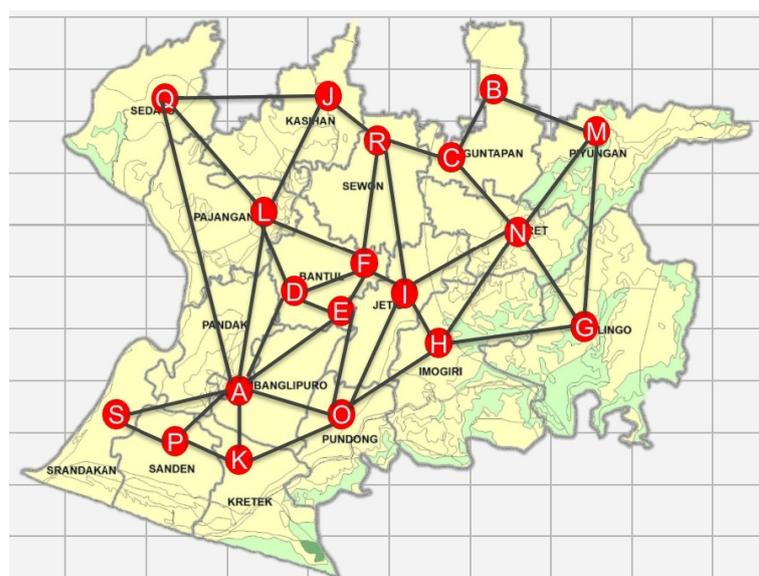


Figure 1. The Map of SMAN over Bantul Regency

The total weight in which it is the total distance of all schools is 224,5 km. In forming the shortest path, we used the Kruskal's Algorithm. The Kruskal's Algorithm as cited from Dutta in 2019 [2] can be seen as follows:

1. Arrange the edges in ascending order of weight.
2. Select the smallest one among them. If the selected edge does not form a cycle with the partial spanning tree, then insert it into the spanning tree else do not insert it.
3. Repeat step 2 to include edges one less than the total number of vertices.”

After applying the Kruskal's algorithm, we get the final graph with 20 vertices, 19 edges, and 73,72 km of total weight.

## **CONCLUSION**

Kruskal's Algorithm is one of the most effective ways in finding the shortest route in distributing USBN sheets. In the case of all Public Senior High Schools (SMAN) in Bantul Regency, before applying the Kruskal's Algorithm, the total distance which should be covered are 224,5 km. But, after applying the Kruskal's Algorithm, we get that starting from the Baldikmen, we will have all 19 SMANs covered with the total travel distance only 73,72 km. We recommend that a further research related with this USBN-sheet distribution will be conducted with various methods including the Modified Kruskal's Algorithm.

## **ACKNOWLEDGEMENT**

Thank you for the team from UIN Sunan Kalijaga who supported this research.

## **REFERENCES**

- [1] Bazgan, C., Toubaline, S., & Vanderpooten, D. (2013). Critical edges/nodes for the minimum spanning tree problem: complexity and approximation. *Journal of Combinatorial Optimization*, 26, 178-189.
- [2] Dutta, J., Barma, P. S., Kar, S., & De, T. (2019). A modified Kruskal's algorithm to improve genetic search for open vehicle routing problem. *International Journal of Business Analytics (IJBAN)*, 6(1), 55-76.
- [3] Greenberg, H.J. (1998). *Greedy Algorithm for Minimum Spanning Tree* <http://glossary.computing.society.informs.org/notes/spanningtree.pdf>, University of Coloardo, Denver.
- [4] Lestari, A. D., Pertiwi, D. A. A., & Muslim, M. A. (2023). Increasing package delivery efficiency through the application of the prim algorithm to find the shortest route on the expedition route. *Journal of Student Research Exploration*, 1(1), 7-14.

- [5] Mahmuzah, R., Aklimawati, Meilizza, & Asri, K. (2019). Pengaruh Nilai Ujian Sekolah Berstandar Nasional (USBN) terhadap Prestasi Belajar Siswa MTs Negeri Rukoh Kota Banda Aceh pada Mata Pelajaran Matematika. *Jurnal Serambi PTK*, VI(2), 64-69.
- [6] Rosen, Kenneth H. (1997). *Exploring Discrete Mathematics With Maple*. Singapore: McGraw-Hill Book Co. Siang.
- [7] Rosidin, U., Herpratiwi, Suana, W., & Firdaus, R. (2019). Evaluation of national examination (UN) and nationalbased school examination (USBN) in Indonesia. *European Journal of Educational Research*, 8(3), 827-837.
- [8] Sudibyoy, N.A., Purwanto, T., & Rahmadi, D. (2020). Minimum Spanning Tree pada Distribusi Bahan Naskah USBN SD/MI di Kabupaten Sragen, *Riemann: Research of Mathematics and Mathematics Education 2* (2), 64-69.
- [9] Wamiliana, Usman, M., Warsono, Warsito, & Daoud, J. I. (2020). Using Modification of Prim's Algorithm and GNU Octave and to Solve the Multiperiods Installation Problem. *IIUM Engineering Journal*, 21 (1), 100-112.
- [10] Wirabudi, A. A., Hafiza, L., & Fachrurrozi, N. R. (2022, October). Design Autonomous Drone Control For Delivery Package using Prim Algorithm and Waypoint Method. In *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)* (pp. 1183-1188). IEEE.

[ ]

**Basis**  
Jurnal ilmiah matematika

ISSN 2962-6013



PRORAM STUDI S-1 MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA & ILMU PENGETAHUAN ALAM  
UNIVERSITAS MULAWARMAN

<http://jurnal.fmipa.unmul.ac.id/index.php/Basis>