



**PENERAPAN GRAF KOMPATIBEL PADA
PENENTUAN WAKTU TUNGGU TOTAL OPTIMAL
LAMPU LALU LINTAS DI PERSIMPANGAN JALAN**

skripsi
disajikan sebagai salah satu syarat
untuk memperoleh gelar Sarjana Sains
Program Studi Matematika

PERPUSTAKAAN
UNNES

oleh
Ririn Dwi Hardianti

4150408022

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI SEMARANG
2013**

PERNYATAAN

Saya menyatakan bahwa skripsi ini bebas plagiat dan apabila di kemudian hari terbukti terdapat plagiat dalam skripsi ini, maka saya bersedia menerima sanksi sesuai ketentuan peraturan perundang-undangan.

Semarang, 05 Februari 2013

Ririn Dwi Hardianti

4150408022



PENGESAHAN

Skripsi yang berjudul

Penerapan Graf Kompatibel pada Penentuan Waktu Tunggu Total Optimal

Lampu Lalu Lintas di Persimpangan Jalan

disusun oleh

Ririn Dwi Hardianti

4150408022

telah dipertahankan di hadapan sidang Panitia Ujian Skripsi FMIPA UNNES pada tanggal 05 Februari 2013.

Panitia:

Ketua

Sekretaris

Prof. Dr. Wiyanto, M.Si
196310121988031001

Drs. Arief Agoestanto, M.Si
196807221993031005

Ketua Penguji

Drs. Amin Suyitno, M.Pd
195206041976121001

Anggota Penguji/
Pembimbing Utama

Anggota Penguji/
Pembimbing Pendamping

Dr. Rochmad, M.Si
195711161987011001

Riza Arifudin, S.Pd., M.Cs
198005252005011001

Motto:

“Sesungguhnya Allah tidak akan merubah nasib suatu kaum,
kecuali kaum itu sendiri yang merubah apa-apa yang ada pada diri mereka”

(QS. Ar-Ra'du ayat 11)

Persembahan:

Karya ini saya persembahkan untuk:

1. Bapak dan ibu tercinta (Bapakku Suhadi dan ibuku Sugiharti) motivator dan penyemangat serta penasehat terbesarku
2. Kakakku (Wawan Puji Mugroho) sebagai motivator kedua
3. Seseorang yang saat ini slalu menjadi penyemangat (Rachmat Santoso)
4. Angres, Dany, Putri, Ulya, dan Vinda 4 tahun terakhir ini dan atas kesetiaan dari awal sampai akhir selalu bersama.
5. Teman – teman seperjuangan matematika murni 2008
6. Almamaterku Matematika UNNES

KATA PENGANTAR

Dengan memanjatkan puji dan syukur ke hadirat Allah SWT yang telah melimpahkan rahmat, karunia dan hidayah-Nya, penulis dapat menyelesaikan skripsi dengan judul “Penerapan Graf Kompatibel pada Penentuan Waktu Tunggu Total Optimal Lampu Lalu Lintas di Persimpangan Jalan”.

Skripsi ini dimaksudkan untuk memenuhi salah satu persyaratan guna memperoleh gelar sarjana matematika pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.

Penulis menyadari bahwa penulisan skripsi ini tidak mungkin sukses tanpa adanya bantuan berbagai pihak, baik bantuan moril maupun materiil. Untuk ini penulis dengan rasa rendah hati mengucapkan terimakasih kepada:

1. Prof. Dr. Soedijono Sastroatmojo, M.Si, Rektor Universitas Negeri Semarang (UNNES).
2. Prof. Dr. Wiyanto, M.Si, Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) UNNES.
3. Drs. Arief Agoestanto, M.Si, Ketua Jurusan Matematika FMIPA UNNES.
4. Dr. Rochmad, M.Si, dosen pembimbing I yang telah memberikan bimbingan, motivasi, pengarahan, dan saran-saran, sehingga penulis dapat menyelesaikan skripsi ini.
5. Riza Arifudin, S.Pd, M. Cs, dosen pembimbing II yang dengan sabar memberikan saran, bimbingan, dan meluangkan waktu dalam penyusunan skripsi ini.

6. Bapak dan Ibu tercinta yang telah membimbing, mengasuh, membesarkan, memberikan semangat, dorongan, kasih sayang dan selalu mendoakanku setiap saat demi kebahagiaan dan keberhasilan penulis.
7. Seseorang teristimewa yang telah menemani dalam suka dan duka dalam menyusun skripsi ini serta memberikan semangat.
8. Sahabat-sahabat berenam yang telah berjuang bersama-sama menyelesaikan skripsi.
9. Semua teman-teman angkatan 2008 Fakultas Matematika dan Ilmu Pengetahuan Alam untuk hari-hari yang indah dan menyenangkan selama kuliah.
10. Serta berbagai pihak yang sudah membantu penulis secara langsung maupun tidak langsung.

Akhir kata, penulis berharap semoga skripsi ini dapat bermanfaat dan dapat menambah wawasan keilmuan khususnya Matematika.

Semarang, 05 Februari 2013

Penulis

ABSTRAK

Ririn Dwi Hardianti. 2013 Penerapan Graf Kompatibel pada Penentuan Waktu Tunggu Total Optimal Lampu Lalu Lintas di Persimpangan Jalan. Skripsi, Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang. Pembimbing Utama Dr. Rochmad, M.Si dan Pembimbing Pendamping Riza Arifudin, S.Pd, M.Cs.

Kata Kunci : graf kompatibel, waktu tunggu, lampu lalu lintas, VB 6.0

Arus lalu lintas yang kompatibel yaitu dua buah arus lalu lintas jika keduanya dapat berjalan bersamaan dengan aman atau tidak berpotongan. Graf kompatibel adalah dua buah himpunan di mana titik-titiknya menunjukkan objek-objek yang akan diatur, dan sisi-sisinya menunjukkan pasangan objek yang kompatibel (sesuai). Tujuan penelitian ini adalah (1) mengetahui bentuk graf kompatibilitas dari hasil pemodelan arus lalu lintas di persimpangan jalan, (2) menghitung waktu tunggu total optimal di persimpangan jalan dengan menggunakan graf kompatibilitas, dan (3) mengetahui perhitungan hasil waktu tunggu total optimal berdasarkan graf kompatibilitas dengan pengaturan yang sudah diterapkan.

Sistematika penelitian ini yaitu kajian pustaka dengan langkah-langkah (a) penemuan masalah, (b) perumusan masalah, (c) studi pustaka, (d) kajian pustaka, (b) analisis dan pemecahan masalah, dan (c) tahap pembuatan program program (d) metode pengumpulan data, (e) analisis data, dan (f) penarikan simpulan.

Diperoleh hasil penelitian (1) bentuk graf kompatibel dari hasil pemodelan arus lalu lintas di persimpangan jalan simpang tiga jalan Majapahit-Supriyadi Kota Semarang, (2) waktu tunggu total optimal di simpang tiga jalan Majapahit-Supriyadi Kota Semarang hasil yang didapat di lapangan adalah 277 detik sedangkan dengan menggunakan graf kompatibel menghasilkan 120 detik, (3) bentuk graf kompatibel dari hasil pemodelan arus lalu lintas di persimpangan jalan simpang empat jalan Kaligarang-Kelud Raya-Bendungan Kota Semarang, dan (4) waktu tunggu total optimal di simpang empat jalan Kaligarang-Kelud Raya-Bendungan Kota Semarang hasil yang didapat di lapangan adalah 389 detik sedangkan dengan menggunakan graf kompatibel menghasilkan 120 detik.

Perhitungan hasil waktu tunggu total optimal berdasarkan graf kompatibel dengan pengaturan yang sudah diterapkan yaitu terlihat lebih optimal (minimal). Dalam perhitungan waktu tunggu total optimal ini yang dilihat hanya dari satu variabel yaitu waktu, perlu dilakukan penyempurnaan model graf kompatibel dengan menambah asumsi-asumsi dan variabel-variabel yang digunakan, dengan demikian dapat diperoleh model yang mendekati situasinya.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
PERNYATAAN.....	ii
PENGESAHAN	iii
MOTO DAN PERSEMBAHAN.....	iv
KATA PENGANTAR	v
ABSTRAK	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN.....	xvi
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	7
1.3 Batasan Masalah	7
1.4 Tujuan Penelitian	8
1.5 Batasan Istilah	8
1.6 Manfaat Penelitian	4
1.7 Sistematika Penulisan Skripsi	7
BAB II TINJAUAN PUSTAKA	
2.1 Graf	14
2.1.1 Definisi Graf	14
2.1.2 Jenis-Jenis Graf	16
2.1.3 Terminologi Dasar Graf	23
2.2 Graf Kompatibilitas	25
2.3 Teori Transportasi.....	30
BAB III METODOLOGI PENELITIAN	
3.1 Kajian Pustaka	40
3.2 Persiapan	40

3.3 Pelaksanaan Penelitian.....	40
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	
4.1 Hasil Penelitian	46
4.2 Pembahasan	53
BAB V PENUTUP	
5.1 Simpulan	67
5.2 Saran	70
DAFTAR PUSTAKA	71
LAMPIRAN	73



DAFTAR TABEL

Tabel	Halaman
4.1 Lama siklus waktu lampu lalu lintas di simpang tiga	52
4.1 Lama siklus waktu lampu lalu lintas di simpang empat	52

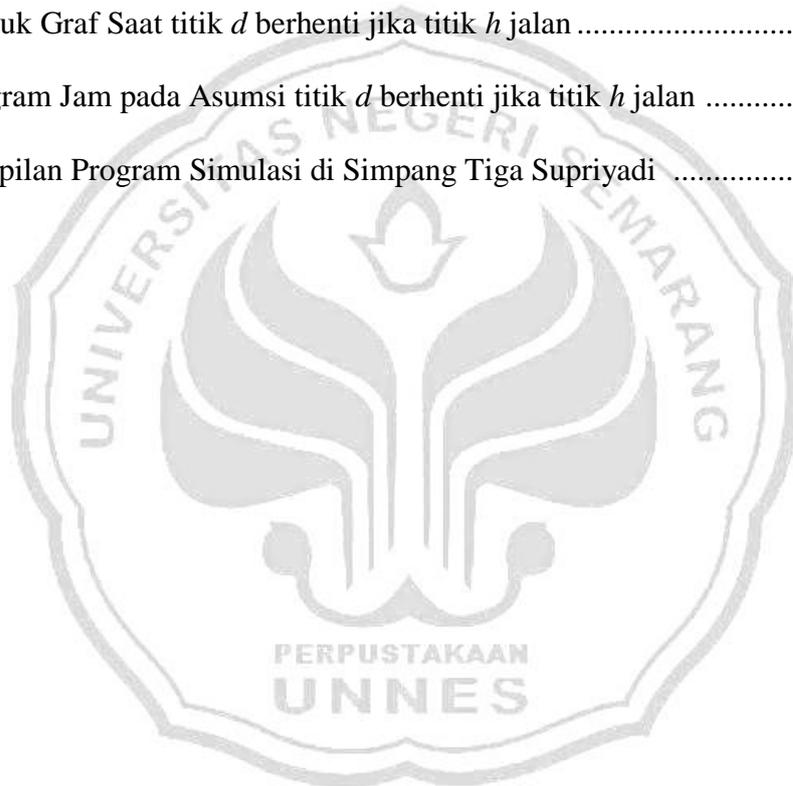


DAFTAR GAMBAR

Gambar	Halaman
1.1 Pertigaan Supriyadi	3
1.2 Perempataan Kaligarang	4
2.1 Contoh Graf G	15
2.2 Graf G	16
2.3 Graf Sederhana	17
2.4 Graf Ganda	18
2.5 Graf Semu	18
2.6 Graf Berhingga.....	19
2.7 Graf Tak Berhingga.....	19
2.8 Graf Tak Berarah	20
2.9 Graf Berarah	21
2.10 Graf Lengkap	21
2.11 Graf Lingkaran.....	22
2.12 Graf Teratur.....	22
2.13 Graf Bipartit	23
2.14 Graf G_1 dan Graf G_2	24
2.15 Graf Kosong	25
2.16 Contoh Persimpangan Jalan	26
2.17 Bentuk Graf Kompatibilitas dari gb. 2.16.....	26
2.18 Diagram Jam	27

2.19 Diagram Jam	28
2.20 Arus Memisah (<i>Diverging</i>)	31
2.21 Arus Menggabung (<i>Merging</i>)	31
2.22 Arus Memotong (<i>Crossing</i>)	32
2.23 Arus Memisah (<i>Weaving</i>)	33
2.24 Skema Persimpangan Jalan Kaligarang-Kelud Raya.....	33
2.25 Skema Persimpangan Jalan Majapahit-Supriyadi.....	34
2.26 Tampilan Utama <i>Visual Basic</i>	36
3.1 Diagram Alur Kerja.....	38
3.2 Bagan Alur Proses Penelitian.....	39
3.3 Diagram Alur Program.....	43
4.1 Diagram Alur (<i>flowchart</i>)	47
4.2 Sistem Lalu Lintas Pada Persimpangan Majapahit	49
4.3 Sistem Lalu Lintas Pada Persimpangan Kaligarang	50
4.4 Foto Lokasi Simpang Tiga Supriyadi	53
4.5 Siklus Waktu Lampu di Simpang Tiga Supriyadi.....	54
4.6 Bentuk Graf Kompatibel pada Simpang Tiga Supriyadi	55
4.7 Graf Lengkap H Waktu Lampu di Simpang Tiga Supriyadi	55
4.8 Diagram Jam	56
4.9 Tampilan Program Simulasi di Simpang Tiga Supriyadi	57
4.10 Foto Lokasi Simpang Empat Kaligarang	58
4.11 Siklus Waktu Lampu saat belok kiri tidak mengikuti lampu.....	58
4.12 Bentuk Graf Kompatibel pada Simpang Empat Kaligarang	59

4.13 Graf Lengkap G Saat Belok Kiri Tidak Mengikuti Lampu	59
4.14 Diagram Jam pada Asumsi Belok Kiri Tidak Mengikuti Lampu	60
4.15 Siklus Waktu Lampu Saat Belok Kiri Mengikuti Lampu	61
4.16 Bentuk Graf Saat Belok Kiri Mengikuti Lampu	62
4.17 Diagram Jam pada Asumsi Belok Kiri Tidak Mengikuti Lampu	63
4.18 Bentuk Graf Saat titik d berhenti jika titik h jalan	63
4.19 Diagram Jam pada Asumsi titik d berhenti jika titik h jalan	64
4.20 Tampilan Program Simulasi di Simpang Tiga Supriyadi	65



DAFTAR LAMPIRAN

Lampiran	Halaman
4.1 Lama siklus waktu lampu lalu lintas di simpang tiga	52
4.1 Lama siklus waktu lampu lalu lintas di simpang empat	52



BAB 1

PENDAHULUAN

1.1 LATAR BELAKANG

Seiring dengan perkembangan zaman dan teknologi, dewasa ini membutuhkan peranan matematika sebagai dasar dari perkembangan ilmu pengetahuan dan teknologi. Dengan demikian, hampir dapat dipastikan bahwa setiap bagian dari ilmu pengetahuan dan teknologi, baik ilmu murni maupun ilmu terapan akan memerlukan peran matematika sebagai ilmu bantunya. Salah satu bagian dari matematika dalam aplikasi kehidupan sehari-hari adalah graf.

Secara umum graf adalah suatu diagram yang memuat informasi tertentu, jika diinterpretasikan secara tepat. Dalam kehidupan sehari-hari graf digunakan untuk menggambarkan berbagai struktur yang ada. Tujuannya adalah sebagai visualisasi objek-objek agar lebih mudah dimengerti. Teori graf merupakan salah satu cabang dari matematika yang bermanfaat di berbagai bidang ilmu. Salah satu contohnya, graf kompatibilitas sering digunakan untuk menentukan waktu tunggu total dan mengatur pergerakan arus lalu lintas.

Teori graf merupakan salah satu topik yang banyak mendapat perhatian, karena model-modelnya sangat berguna untuk aplikasi yang luas, seperti masalah dalam jaringan komunikasi, transportasi, ilmu komputer, dan lain sebagainya. Graf-graf kompatibilitas digunakan secara luas dalam memecahkan masalah yang melibatkan pengaturan data dalam urutan tertentu. Arus lalu lintas tertentu dapat disebut kompatibel jika kedua arus tersebut tidak akan menghasilkan kecelakaan

yang disebabkan oleh kendaraan (Hosseini & Orooji, 2009). Dalam graf ini, titik-titiknya menunjukkan objek-objek yang akan diatur, dan sisi-sisinya menunjukkan objek-objek yang kompatibel.

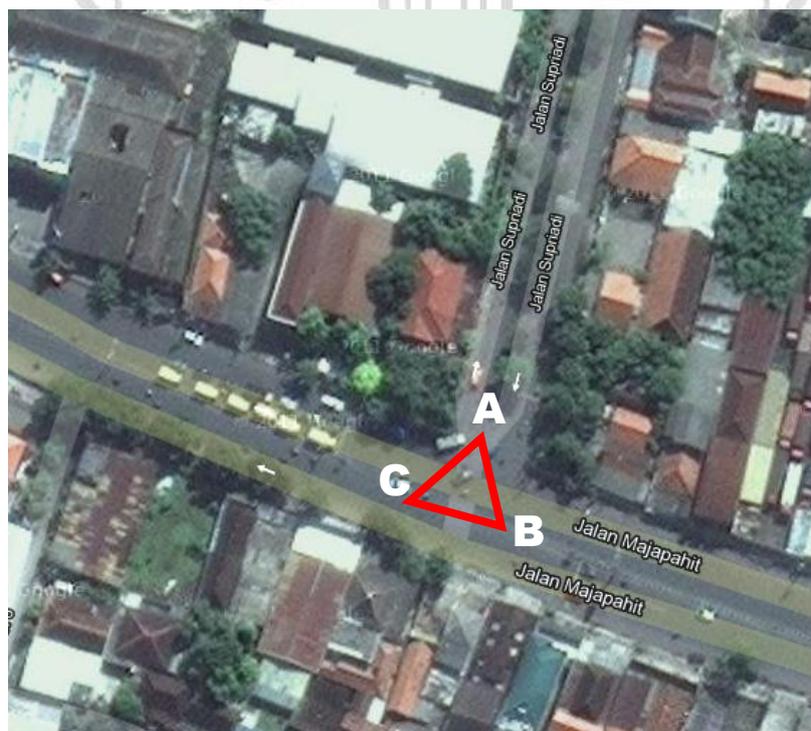
Masalah transportasi secara umum dan lalu lintas pada khususnya adalah merupakan fenomena yang terlihat sehari-hari dalam kehidupan manusia. Semakin tinggi tingkat mobilitas warga suatu kota, akan semakin tinggi juga tingkat perjalanannya. Jika peningkatan perjalanan ini tidak diikuti dengan peningkatan prasarana transportasi yang memadai, maka akan terjadi suatu ketidakseimbangan antara permintaan (*demand*) dan penyediaan (*supply*) yang akhirnya akan menimbulkan suatu ketidak-lancaran dalam mobilitas yaitu berupa kemacetan (Nugroho, 2008). Kemacetan lalu lintas di suatu kota atau tempat sekarang ini bukan merupakan hal yang asing lagi yang dapat terjadi di suatu ruas ataupun persimpangan jalan, kemacetan timbul karena adanya konflik pergerakan yang datang tiap arah kaki simpangnya dan untuk mengurangi konflik ini banyak dilakukan pengendalian untuk mengoptimalkan persimpangan dengan menggunakan lampu lalu lintas.

Lalu lintas adalah suatu keadaan dengan pengaturan menggunakan lampu lalu lintas yang terpasang pada persimpangan dengan tujuan untuk mengatur arus lalu lintas. Pengaturan arus lalu lintas pada persimpangan pada dasarnya dimaksudkan untuk bagaimana pergerakan kendaraan pada masing-masing kelompok pergerakan kendaraan dapat bergerak secara bergantian sehingga tidak saling mengganggu antar arus yang ada. Ada berbagai jenis kendali dengan menggunakan lampu lalu lintas di mana pertimbangan ini sangat bergantung pada

situasi dan kondisi persimpangan seperti volume, geometrik simpang, dan sebagainya.

Arus lalu lintas di kawasan jalan Majapahit terpantau padat merayap terlebih pada waktu pagi hari. Dikarenakan setiap pagi hari dari arah timur dibuat 3 lajur untuk menuju ke arah barat, alhasil lajur sebaliknya hanya 1 lajur. Hal ini disebabkan dengan banyaknya pekerja dan pelajar yang akan berangkat pada jam yang bersamaan. Kemacetan arus lalu lintas terlihat diberapa titik di pertigaan Supriyadi. Beberapa petugas kepolisian yang ada terkadang tak kuasa ikut membantu mengatur semrawutnya arus lalu lintas tersebut, terutama pada waktu sibuk di pagi dan sore hari.

Persimpangan simpang tiga di jalan Majapahit-Supriyadi dapat dilihat pada Gambar 1.1.



Gambar 1.1 Pertigaan Supriyadi

Kepadatan arus lalu lintas di sepanjang Jalan Kaligarang, Semarang, seperti yang terpantau oleh wartawan (Rifki, 5 April 2012) sangat padat dan agak tersendat. Penumpukan kendaraan nampak di perempatan Kaligarang, arah ke Jalan Pamularsih dan Jalan Kelud Raya. Sedangkan arah sebaliknya terpantau ramai lancar. Selain karena jalan yang memang sempit, tersendatnya arus lalu lintas di kawasan ini, memang sering terjadi pada jam-jam sibuk. Kepadatan mulai nampak dari SPBU Kaligarang hingga *traffic light* di perempatan Kaligarang. Akibatnya, banyak para pengendara yang memilih menggunakan jalan kampung yang relatif sepi. Dalam hal ini persimpangan simpang empat di jalan Kaligarang dapat dilihat pada Gambar 1.2.



Gambar 1.2 Perempatan Kaligarang

Persimpangan merupakan tempat yang rawan kecelakaan dan kemacetan. Maka untuk menghindari atau mengurangi kepadatan lalu lintas, salah satu cara yang dipergunakan adalah dipasangnya lampu lalu lintas. Persimpangan adalah

suatu bentuk pertemuan jalan, di mana setiap mulut simpang (akhir jalan/pertemuan dengan jalan lain) memiliki pergerakan lalu lintas, karakteristik, geometrik jalan dan konflik-konflik tertentu yang terjadi pada suatu persimpangan tersebut (Suteja, 2011: 172-173).

Persimpangan jalan adalah merupakan bagian yang terpenting dari jalan raya sebab sebagian besar dari efisiensi, kapasitas lalu lintas, kecepatan biaya operasi, waktu perjalanan, keamanan dan kenyamanan akan tergantung pada perencanaan persimpangan tersebut (Hariyanto, 2004: 2). Setiap persimpangan mencakup pergerakan lalu lintas menerus dan lalu lintas yang saling memotong pada satu atau lebih dari kaki persimpangan dan mencakup juga pergerakan perputaran. Pergerakan lalu lintas ini dikendalikan berbagai cara bergantung pada jenis persimpangannya.

Lalu lintas merupakan perangkat penting dalam mengendalikan persimpangan. Teori lalu lintas adalah fenomena fisik yang bertujuan memahami dan meningkatkan lalu lintas mobil, dan masalah yang terkait dengan itu seperti kemacetan lalu lintas (Baruah & Baruah, 2012). Permasalahannya, penentuan parameter waktu dan pengaturan pergiliran yang kurang sesuai dengan volume dan karakteristik kedatangan kendaraan. Tujuannya adalah sebagai visualisasi objek-objek agar lebih mudah dimengerti model antrian untuk menentukan optimalisasi waktu penyalaan lampu lalu lintas dan meminimalisasi waktu tunggu.

Sistem pengatur lalu lintas adaptif dan sinkron telah banyak digunakan di beberapa negara maju. Dengan adanya sistem pengaturan lalu lintas adaptif,

durasi merah dan hijau disesuaikan dengan kepadatan kedatangan kendaraan. Dengan sistem ini, diharapkan durasi waktu tunggu kendaraan dari semua arah cenderung sama dan tidak akan melebihi satu siklus. Dengan kata lain, hampir tidak ada kendaraan yang mengalami isyarat merah dua kali. Adanya sistem pengatur lalu lintas sinkron digunakan untuk mengurangi waktu tempuh jalan utama. Pengatur lalu lintas saling disinkronkan satu dengan yang lain agar sebagian besar kendaraan di jalan utama tidak terlalu lama menanti isyarat hijau.

Pengatur lalu lintas sinkron digunakan untuk mengurangi durasi waktu tunggu kendaraan di jalan utama. Proses sinkronisasi pada pengatur lalu lintas sinkron cukup rumit. Penentuan waktu dan durasi hijau melibatkan banyak parameter, seperti: waktu hijau arah utama pengatur lalu lintas di sebelahnya, kecepatan dan percepatan kendaraan, serta waktu tempuh kendaraan dari suatu pengatur lalu lintas ke pengatur lalu lintas yang lain. Perhitungan waktu hijau juga harus didukung oleh sensor keberadaan kendaraan di jalan utama yang jumlahnya cukup banyak. Agar tetap sinkron, suatu pengatur lalu lintas sinkron tidak dapat bersifat adaptif.

Adanya peningkatan kepadatan kedatangan kendaraan dari suatu arah dapat meningkatkan durasi waktu tunggu kendaraan dari arah tersebut untuk mendapatkan isyarat lampu hijau. Pada penelitian ini, penulis membuat konsep sistem pengatur lalu lintas sinkron adaptif. Sistem tidak disinkronkan dengan pengatur lalu lintas lain, namun dengan runtun data kepadatan kedatangan kendaraan dari arah utama. Dengan sistem ini, meskipun terdapat perubahan

kepadatan kedatangan kendaraan dari beberapa arah, diharapkan durasi waktu tunggu kendaraan dapat dibuat minimal terutama untuk arah utama.

Berdasarkan alasan di atas, penulis akan mengangkat judul **“Penerapan Graf Kompatibel pada Penentuan Waktu Tunggu Total Optimal Lampu Lalu Lintas di Persimpangan Jalan.”**

1.2 RUMUSAN MASALAH

Dari uraian latar belakang di atas dapat dirumuskan masalah yang ingin dipecahkan sebagai berikut:

- a. Bagaimana bentuk graf kompatibel dari hasil pemodelan arus lalu lintas di persimpangan jalan?
- b. Bagaimana hasil perhitungan waktu tunggu total optimal dengan menggunakan graf kompatibel tersebut?
- c. Bagaimana perhitungan hasil waktu tunggu total optimal berdasarkan graf kompatibel dengan pengaturan yang sudah diterapkan?

1.3 BATASAN MASALAH

Agar penulisan skripsi ini tidak meluas, maka pembahasan hanya difokuskan pada persimpangan simpang tiga (pertigaan jalan Majapahit-Supriyadi dan perempatan jalan Kaligarang-Kelud Raya-Bendungan). Yang akan dilakukan peneliti yaitu menghitung waktu tunggu total optimal dengan menggunakan graf kompatibel.

1.4 TUJUAN PENELITIAN

Sesuai dengan rumusan masalah, penulisan skripsi itu bertujuan untuk

- a. Mengetahui bagaimana bentuk graf kompatibel dari hasil pemodelan arus lalu lintas di persimpangan jalan.
- b. Mengetahui bagaimana menghitung waktu tunggu total optimal dengan menggunakan graf kompatibel.
- c. Mengetahui perhitungan hasil waktu tunggu total optimal berdasarkan graf kompatibel dengan pengaturan yang sudah diterapkan.

1.5 BATASAN ISTILAH

a. Graf Kompatibilitas/Kompatibel

Graf-graf komabilitas digunakan secara luas dalam memecahkan masalah yang melibatkan pengaturan data dalam urutan tertentu (Wilson & Watkin, 1976: 63). Dalam graf ini, titik-titiknya menunjukkan objek-objek yang akan diatur, dan sisi-sisinya menunjukkan pasangan objek yang kompatibel (sesuai). Walaupun Pengertian graf komabilitas ini pertama kali muncul dalam kontes genetika, Pengertian ini sekarang telah dipakai dalam bidang-bidang lain seperti arkheologi, psikologi, dan penentuan usia karya tulis klasik yang ditentukan. Dalam penelitian ini, akan dicari bentuk graf kompatibilitas dari persimpangan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan.

b. Lalu Lintas

Lalu lintas merupakan perangkat penting dalam mengendalikan persimpangan. Permasalahannya, penentuan parameter waktu dan pengaturan

pergiliran yang kurang sesuai dengan volume dan karakteristik kedatangan kendaraan (Anggara, 2005). Dalam penelitian ini, arus lalu lintas yang dapat diamati bertujuan agar dapat mengetahui bagaimana keadaan lalu lintas di persimpangan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan sebelum dan sesudah dilakukannya penelitian.

c. Persimpangan Jalan

Persimpangan jalan adalah bagian yang sulit dihindarkan dalam jaringan jalan, karena persimpangan jalan merupakan tempat bertemu dan berganti arah arus lalu lintas dari dua jalan atau lebih. Persimpangan merupakan tempat sumber konflik lalu lintas yang rawan terhadap kecelakaan karena terjadi konflik antara kendaraan dengan kendaraan lainnya ataupun antara kendaraan dengan pejalan kaki. Dalam penelitian ini, persimpangan jalan yang akan diamati yaitu di persimpangan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan.

d. Waktu Tunggu Total Optimal

Waktu merupakan seluruh rangkaian saat ketika proses, perbuatan, atau keadaan berada atau berlangsung di suatu tempat. Waktu tunggu total optimal berarti proses seluruh rangkaian ketika proses berlangsung secara menyeluruh untuk mendapatkan pelayanan yang maksimal. Dalam penelitian ini, akan dihitung waktu tunggu total optimal dan waktu tunggu per jalur dengan menggunakan graf kompatibilitas di persimpangan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan.

e. Pengaturan/Pengendalian Lalu Lintas

Sistem pengaturan/pengendalian lalu lintas pada persimpangan mempunyai cakupan luas antara lain masalah perhitungan besarnya kapasitas persimpangan yang ada, volume lalu lintas, pola pergerakan, jenis kendaraan yang lewat, faktor manusia, perhitungan-perhitungan waktu siklus pada simpang dengan lampu lalu lintas dan hal-hal lainnya yang berkaitan dengan hal tersebut (Hariyanto, 2004). Pemilihan jenis pengendalian pada persimpangan tanpa lalu lintas merupakan hal yang sangat penting. Dalam penelitian ini, pengaturan lalu lintas di persimpangan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan sangat dicermati bagaimana pengendalian lalu lintas di persimpangan tersebut.

1.6 MANFAAT PENELITIAN

a. Bagi penulis

Membantu penulis untuk mengetahui bagaimana menghitung waktu tunggu total optimal lampu lalu lintas di persimpangan jalan dengan penerapan graf kompatibel.

b. Bagi universitas

Dari hasil penelitian ini dapat menjadi referensi yang berkaitan dengan teori graf dalam menyelesaikan masalah menghitung waktu tunggu total optimal.

c. Bagi mahasiswa

Penerapan graf kompatibilitas sangat berguna untuk menghitung jumlah waktu tunggu optimal pada arus lalu lintas di persimpangan jalan. Mahasiswa dapat mengetahui berapakah hasil perhitungan dengan menggunakan graf

kompatibilitas. Penelitian ini juga dapat dipakai sebagai bahan acuan bagi mahasiswa yang ingin melanjutkan penelitian perhitungan waktu tunggu total optimal dengan metode yang berbeda.

d. Bagi instansi

Dari hasil penelitian ini diharapkan dapat menjadi salah satu bahan masukan bagi Dinas Perhubungan untuk menghitung jumlah waktu tunggu total optimal pada persimpangan jalan.

1.7 SISTEMATIKA SKRIPSI

Dalam penulisan skripsi ini secara garis besar dibagi menjadi tiga bagian pokok, yaitu bagian awal, bagian isi, dan bagian akhir.

1.7.1 Bagian awal skripsi memuat:

- a. Halaman sampul
- b. Halaman judul
- c. Abstrak
- d. Lembar pengesahan
- e. Motto dan persembahan
- f. Kata pengantar
- g. Daftar isi

1.7.2 Bagian isi

- a. Bab I : Pendahuluan

Mengemukakan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika skripsi.

b. Bab II : Landasan Teori

Berisi uraian teoritis atau teori-teori yang mendasari pemecahan tentang masalah-masalah yang berhubungan dengan judul skripsi.

c. Bab III : Metode penelitian

Berisi tentang metode-metode yang digunakan dalam penelitian yang meliputi menemukan masalah, merumuskan masalah, studi pustaka, pemecahan masalah, serta penarikan kesimpulan.

d. Bab IV : Hasil penelitian dan pembahasan

Berisi semua hasil penelitian dan pembahasan bentuk graf kompatibilitas dari hasil pemodelan arus lalu lintas di persimpangan jalan simpang tiga, simpang empat, maupun simpang lima. Analisis data dari persimpangan jalan, data jumlah kendaraan yang melalui persimpangan, siklus waktu lampu lalu lintas pada persimpangan jalan, menghitung waktu tunggu total optimal dengan menggunakan graf kompatibilitas, dan menghitung hasil waktu tunggu total optimal berdasarkan graf kompatibilitas dan pengaturan yang sudah diterapkan.

e. Bab V : Penutup

Bab ini berisi tentang simpulan dan saran-saran yang diberikan penulis berdasarkan simpulan yang diambil.

1.7.3 Bagian akhir

Bagian akhir skripsi berisi tentang daftar pustaka dan lampiran-lampiran yang mendukung skripsi.



BAB 2

TINJAUAN PUSTAKA

2.1 Graf

Teori graf merupakan cabang dari Matematika sebenarnya sudah ada sejak lebih dari dua ratus tahun yang silam. Jurnal pertama tentang teori graf muncul pada tahun 1736, oleh matematikawan terkenal dari Swiss bernama Euler. Dari segi matematika, pada awalnya teori graf “kurang” signifikan, karena kebanyakan dipakai untuk memecahkan teka-teki (*puzzle*), namun akhirnya mengalami perkembangan yang sangat pesat yaitu terjadi pada beberapa puluh tahun terakhir ini. Salah satu alasan perkembangan teori graf yang begitu pesat adalah aplikasinya yang sangat luas dalam kehidupan sehari-hari maupun dalam berbagai bidang ilmu seperti: Ilmu Komputer, Teknik, Sains, bahkan Ilmu Sosial (Ketut, 2007: 1).

2.1.1 Definisi Graf

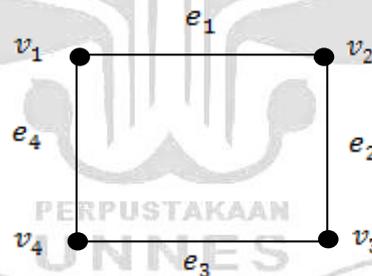
Sebuah graf G berisikan dua himpunan yaitu himpunan berhingga tak kosong $V(G)$ dari objek-objek yang disebut **titik** dan himpunan berhingga (mungkin kosong) $E(G)$ yang elemen-elemennya disebut **sisi** sedemikian hingga setiap elemen e dalam $E(G)$ merupakan pasangan tak berurutan dari titik-titik di $V(G)$ disebut **himpunan titik G** (Ketut, 2007: 1-2). Menurut Munir (2005: 356), graf G didefinisikan sebagai pasangan himpunan (V,E) , ditulis dengan notasi $G = (V,E)$, yang dalam hal ini V adalah himpunan tidak kosong dari titik-titik (*vertices* atau *node*) dan E adalah himpunan sisi (*edges* atau *arcs*) yang menghubungkan

sepasang titik, E boleh kosong. Jadi, sebuah graf dimungkinkan tidak mempunyai sisi, tetapi titiknya harus ada minimal satu. Graf yang hanya mempunyai satu buah titik tanpa sisi dinamakan **graf trivial**.

Misalkan G adalah sebuah graf. Sebuah jalan W disebut tertutup jika titik awal dan titik akhir dari W sama. Jejak tertutup disebut sirkuit. Sirkuit dengan titik awal dan titik akhir internalnya berlainan disebut siklus (*cycle*). Siklus dengan n titik dirotasikan dengan C_n (Sutarno, 2003: 65).

Contoh:

Sebuah graf $G = (V, E)$ dengan $V = \{v_1, v_2, v_3, v_4, v_5\}$ dan $E = \{e_1, e_2, e_3, e_4\}$ dimana $e_1 = v_1, v_2, e_2 = v_2, v_3, e_3 = v_3, v_4, e_4 = v_4, v_1$ dapat dipresentasikan dalam bentuk Gambar 2.1.



Gambar 2.1 Contoh Graf G

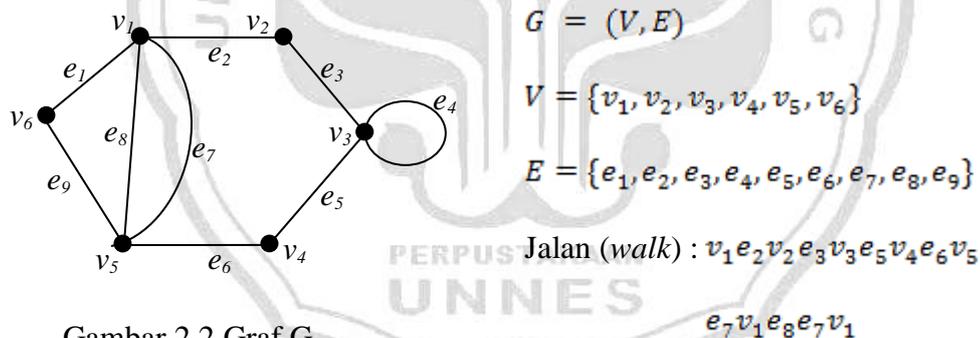
Sebuah sisi yang hanya menghubungkan sebuah titik dengan dirinya sendiri disebut gelung (*loop*). Jika terdapat lebih dari satu sisi yang menghubungkan dua titik u dan v pada suatu graf, maka sisi-sisi tersebut disebut sisi ganda (Ketut, 2007: 3).

Misalkan G adalah sebuah graf. Sebuah jalan (*walk*) di G adalah sebuah barisan berhingga tak kosong $W = (v_0, e_1, v_1, e_2, v_2, \dots, e_k, v_k)$ yang suku-sukunya bergantian titik dan sisi, sedemikian hingga v_{i-1} dan v_i adalah titik-titik

akhir sisi e_i , untuk $1 \leq i \leq k$. Dapat dinyatakan W adalah sebuah jalan dari titik v_0 ke titik v_k , atau jalan (v_0, v_k) . Titik v_0 dan titik v_k berturut-turut disebut titik awal dan titik akhir W . Jika semua sisi $e_1, e_2, e_3, \dots, e_k$ dalam jalan W yang berbeda, maka W disebut jejak (*trail*). Jika semua titik $v_0, v_1, v_2, \dots, v_k$ dalam jalan W juga berbeda, maka W disebut lintasan (*path*) (Ketut, 2007: 6).

Misalkan G adalah sebuah graf. Sebuah jalan W disebut tertutup jika titik awal dan titik akhir dari W sama. Jejak tertutup disebut sirkuit. Sirkuit dengan titik awal dan titik akhir internalnya berlainan disebut siklus (*cycle*). Siklus dengan n titik dinotasikan dengan C_n (Sutarno, 2003: 65).

Contoh:



Gambar 2.2 Graf G

Jejak (*trail*) : v_5, v_6, v_1, v_5, v_4

Lintasan (*path*) : v_1, v_2, v_3, v_4

Jejak tertutup (Sirkuit) :

$v_1, e_8, v_5, e_9, v_6, e_1, v_1$

Siklus (*cycle*) : v_1, v_5, v_6, v_1

Pada Gambar 2.2, Sisi e_4 adalah *loop*. Sisi e_7 dan e_8 merupakan sisi ganda karena kedua sisi tersebut terkait dengan titik v_1 dan v_5 .

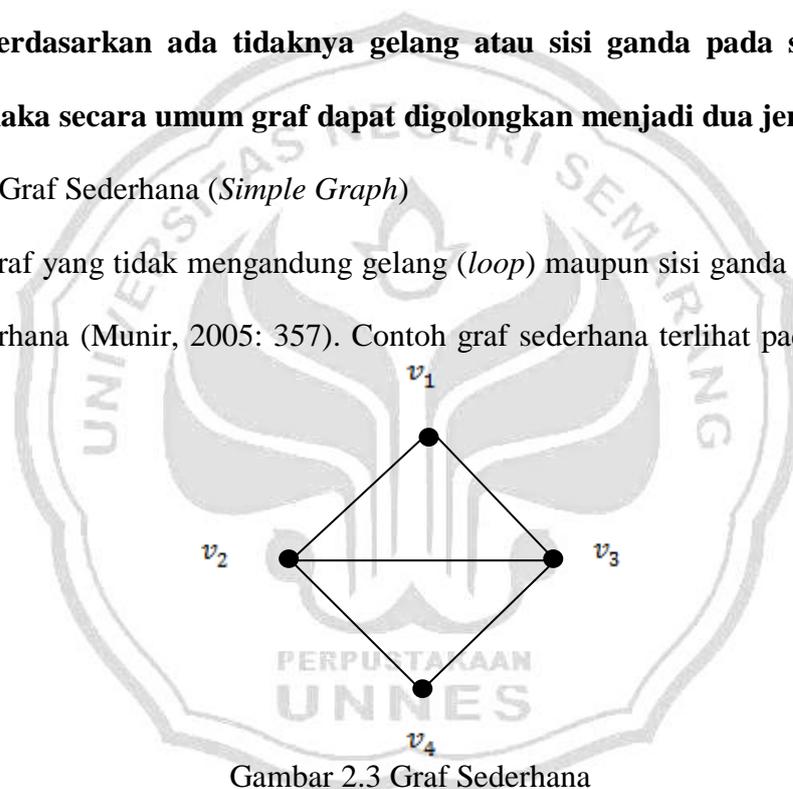
2.1.2 Jenis-Jenis Graf

Graf dapat dikelompokkan menjadi beberapa kategori (jenis) bergantung pada sudut pandang pengelompokannya. Pengelompokan graf dapat dipandang berdasarkan ada tidaknya sisi ganda, berdasarkan jumlah titik, atau berdasarkan orientasi arah pada sisi (Munir, 2005: 357).

2.1.2.1 Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf maka secara umum graf dapat digolongkan menjadi dua jenis, yaitu:

2.1.2.1.1 Graf Sederhana (*Simple Graph*)

Graf yang tidak mengandung gelang (*loop*) maupun sisi ganda dinamakan graf sederhana (Munir, 2005: 357). Contoh graf sederhana terlihat pada Gambar 2.3.



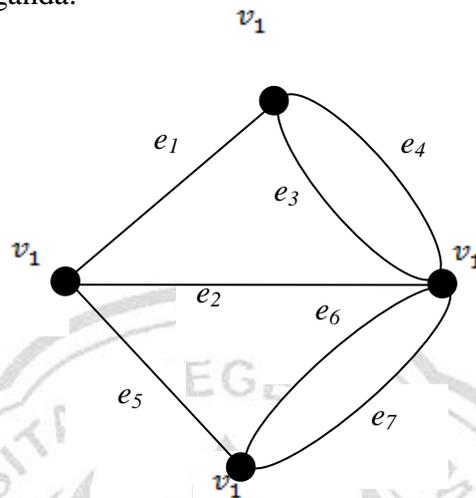
Gambar 2.3 Graf Sederhana

2.1.2.1.2 Graf Tak Sederhana (*Unsimple Graph*)

Menurut Munir (2005, 357) graf yang mengandung sisi ganda atau *loop* dinamakan graf tak sederhana (*unsimple graph*). Ada dua macam graf tak sederhana, yaitu graf ganda (*multigraph*) dan graf semu (*pseudograph*).

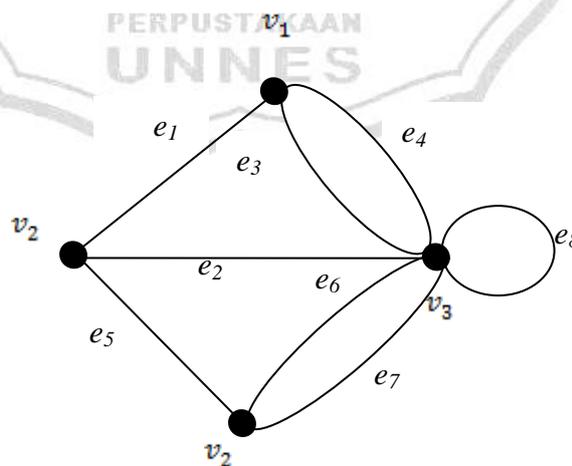
- (1) Graf Ganda (*Multigraph*) adalah graf yang mengandung sisi ganda tetapi tidak memiliki *loop* (gelang). Sisi ganda yang menghubungkan

sepanjang titik dapat lebih dari dua buah. Pada Gambar 2.4 adalah contoh graf ganda.



Gambar 2.4 Graf Ganda

- (2) Graf Semu (*Pseudograph*) adalah graf yang mengandung loop (termasuk bila memiliki sisi ganda sekalipun). Graf semu lebih umum daripada graf ganda, karena sisi pada graf semu dapat terhubung ke dirinya sendiri. Gambar 2.5 adalah contoh graf semu.

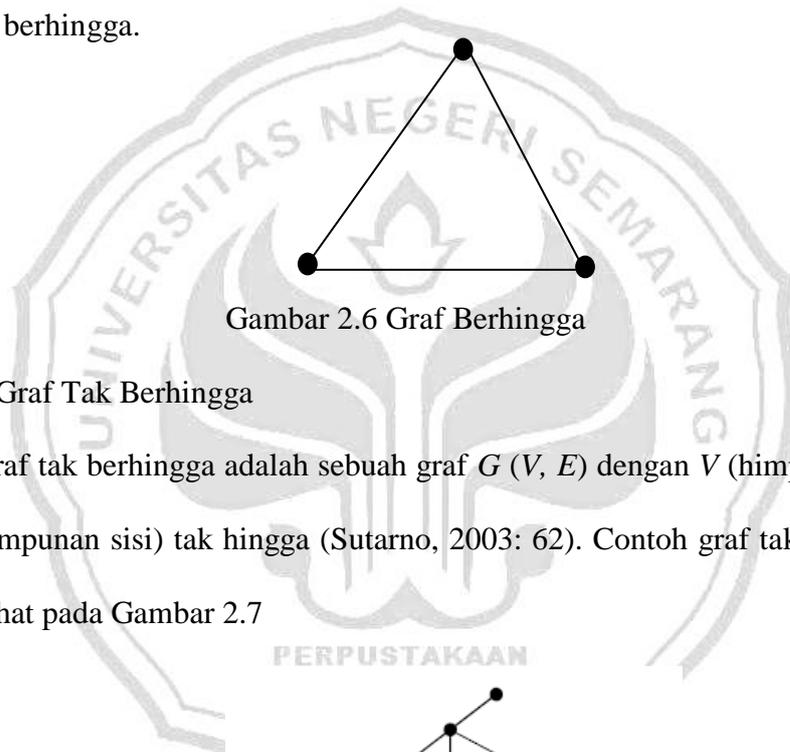


Gambar 2.5 Graf Semu

2.1.2.2 Berdasarkan jumlah titik pada suatu graf, maka secara umum graf dapat digolongkan menjadi dua jenis, yaitu:

2.1.2.2.1 Graf Berhingga

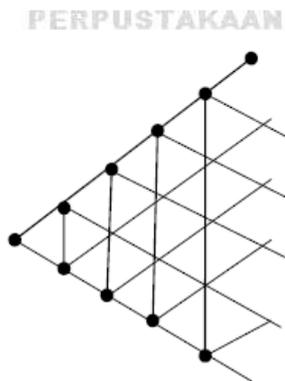
Graf berhingga adalah sebuah graf $G(V, E)$ dengan V (himpunan titik) dan E (himpunan sisi) hingga (Sutarno, 2003: 62). Pada Gambar 2.6 adalah contoh graf yang berhingga.



Gambar 2.6 Graf Berhingga

2.1.2.2.2 Graf Tak Berhingga

Graf tak berhingga adalah sebuah graf $G(V, E)$ dengan V (himpunan titik) dan E (himpunan sisi) tak hingga (Sutarno, 2003: 62). Contoh graf tak berhingga dapat dilihat pada Gambar 2.7

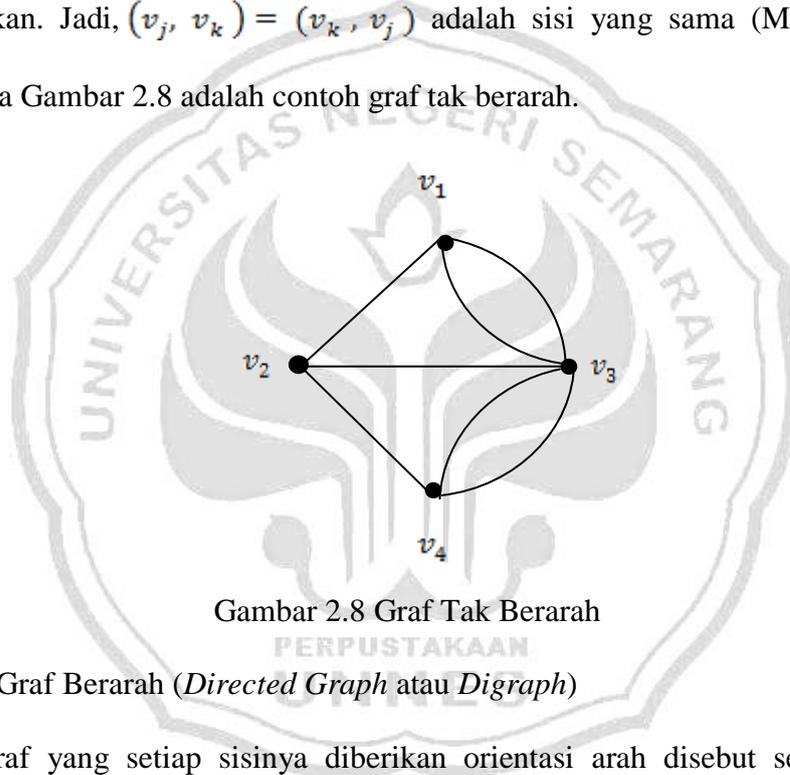


Gambar 2.7 Graf Tak Berhingga

2.1.2.3 Berdasarkan orientasi arah pada sisi, maka secara umum graf dibedakan atas 2 jenis, yaitu:

2.1.2.3.1 Graf Tak Berarah (*Undirected Graph*)

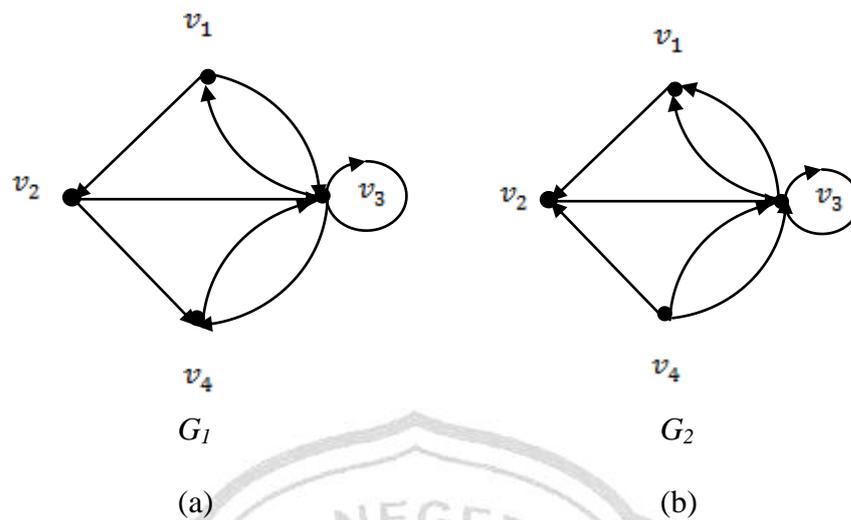
Graf yang sisinya tidak mempunyai orientasi arah disebut graf tak berarah. Pada graf tak berarah, urutan pasangan titik yang dihubungkan oleh sisi tidak diperhatikan. Jadi, $(v_j, v_k) = (v_k, v_j)$ adalah sisi yang sama (Munir, 2005: 358). Pada Gambar 2.8 adalah contoh graf tak berarah.



Gambar 2.8 Graf Tak Berarah

2.1.2.3.2 Graf Berarah (*Directed Graph* atau *Digraph*)

Graf yang setiap sisinya diberikan orientasi arah disebut sebagai graf berarah. Sisi berarah disebut juga busur. Pada graf berarah, (v_j, v_k) dan (v_k, v_j) menyatakan dua buah busur yang berbeda, dengan kata lain $(v_j, v_k) \neq (v_k, v_j)$. Untuk busur (v_j, v_k) , titik v_j dinamakan titik asal (*initial vertex*), dan titik v_k dinamakan titik terminal (*terminal vertex*). Pada graf berarah, *loop* diperbolehkan, tetapi sisi ganda tidak diperbolehkan sedangkan graf ganda berarah *loop* dan sisi ganda diperbolehkan (Munir, 2005: 358). Pada Gambar 2.9(a) merupakan graf berarah sedangkan Gambar 2.9(b) merupakan graf ganda berarah.



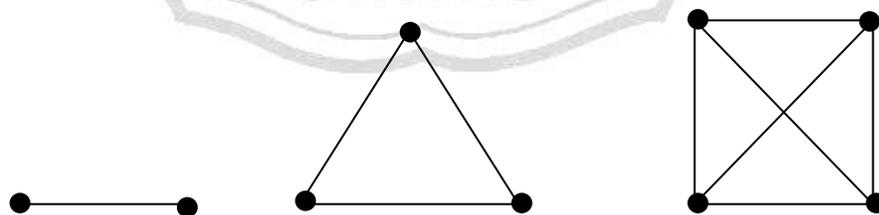
Gambar 2.9 (a) Graf Berarah dan (b) Graf Ganda Berarah

2.1.2.4 Berdasarkan beberapa graf sederhana khusus, maka secara umum graf dibedakan atas 2 jenis, yaitu:

2.1.2.4.1 Graf Lengkap (*Complete Graph*)

Graf lengkap ialah graf sederhana yang setiap titiknya mempunyai sisi ke semua titik lainnya. Graf lengkap dengan n buah titik dilambangkan K_n . setiap titik pada K_n berderajat $n - 1$ (Munir, 2005: 377).

Contoh Graf Lengkap ditunjukkan pada Gambar 2.10.



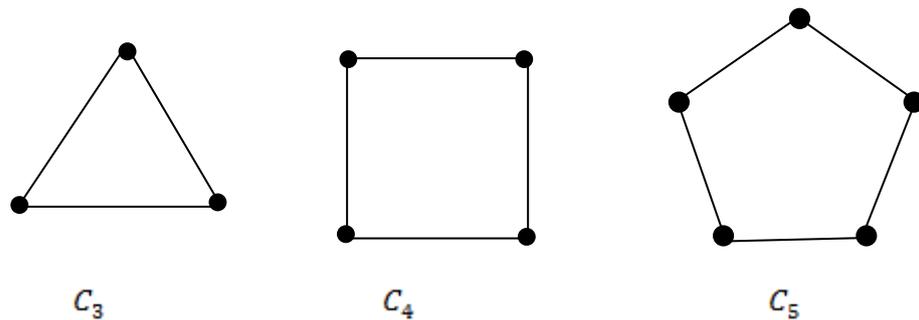
Gambar 2.10 Graf Lengkap

2.1.2.4.2 Graf Lingkaran

Graf lingkaran adalah graf sederhana yang setiap titiknya berderajat dua.

Graf lingkaran dengan n titik dilambangkan dengan C_n (Munir, 2005: 377).

Contoh Graf Lingkaran ditunjukkan pada Gambar 2.11.

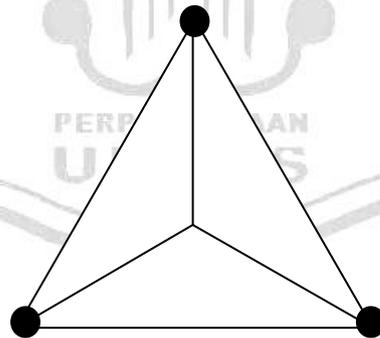


Gambar 2.11 Graf Lingkaran

2.1.2.4.3 Graf Teratur/Graf Reguler

Graf yang setiap titiknya mempunyai derajat yang sama disebut graf teratur. Apabila derajat setiap titik adalah r , maka graf tersebut disebut sebagai graf teratur derajat r . Jumlah sisi pada graf teratur derajat r dengan n buah titik adalah $\frac{nr}{2}$ (Munir, 2005: 378). Contoh Graf Teratur ditunjukkan pada Gambar

2.12.



Gambar 2.12 Graf Teratur dengan 4 titik dan derajat 3

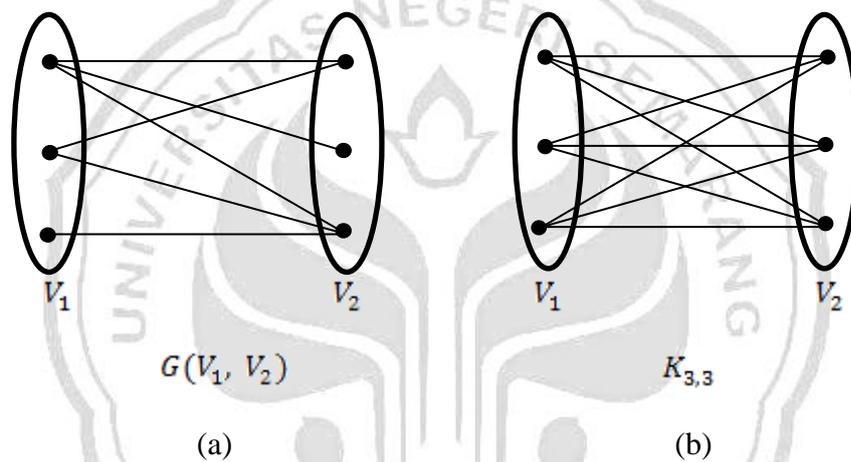
2.1.2.4.4 Graf Bipartit (*Bipartite Graph*)

Graf bipartite adalah graf G himpunan titiknya dapat dikelompokkan menjadi dua himpunan bagian V_1 dan V_2 , sedemikian sehingga setiap sisi di dalam G menghubungkan sebuah titik di V_1 ke sebuah titik di V_2 dan dinyatakan sebagai

$G(V_1, V_2)$. Dengan kata lain, setiap pasang titik di V_1 (demikian pula dengan titik-titik di V_2) tidak bertetangga (Munir, 2005: 379).

Apabila setiap titik di V_1 bertetangga dengan semua titik di V_2 , maka $G(V_1, V_2)$ disebut graf bipartit lengkap, dilambangkan dengan $K_{m,n}$. Jumlah sisi pada graf bipartit lengkap adalah mn (Munir, 2005: 380).

Contoh graf bipartit ditunjukkan pada Gambar 2.13



Gambar 2.13 Graf Bipartit

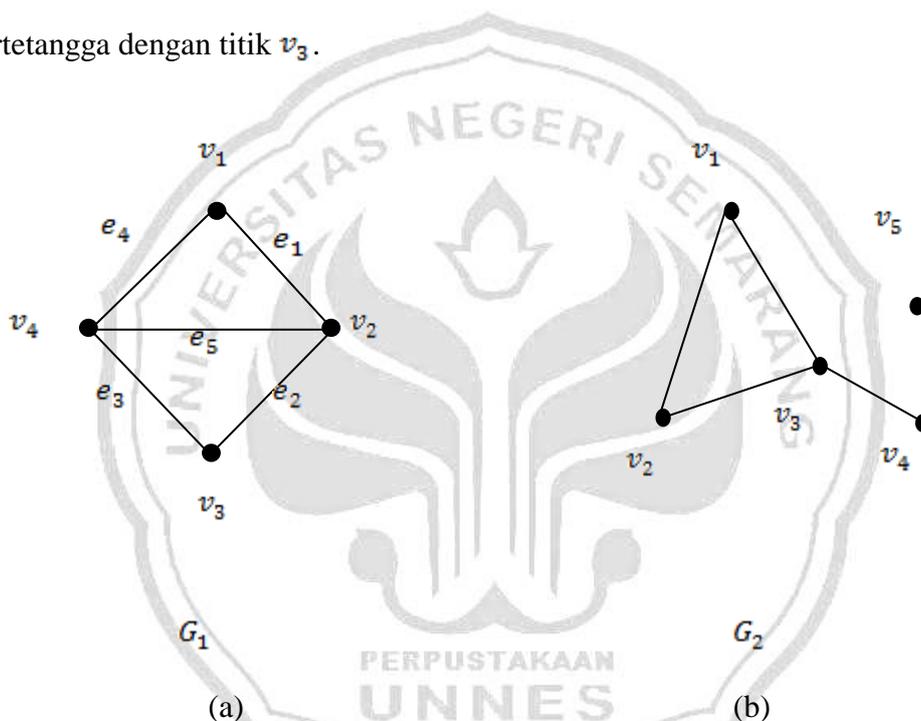
Pada Gambar 2.13(a) $G(V_1, V_2)$ adalah graf bipartit, sedangkan Gambar 2.13(b) $K_{3,3}$ adalah graf bipartit lengkap.

2.1.3 Terminologi Dasar Graf

Dalam pembahasan mengenai graf biasanya sering menggunakan terminology (istilah) yang berkaitan dengan graf. Berikut ini terminology (istilah) yang berkaitan dengan graf yang akan digunakan dalam skripsi ini, yang dirujuk dari Munir (2005: 364-376).

2.1.3.1 Bertetangga (*Adjacent*)

Dua buah titik pada graf tak berarah G dikatakan bertetangga bila keduanya terhubung langsung dengan sebuah sisi. Dengan kata lain v_j bertetangga dengan v_k jika (v_j, v_k) adalah sebuah sisi pada graf G (Munir, 2005: 365). Pada Gambar 2.14(a), titik v_1 bertetangga dengan titik v_2 dan v_4 , titik v_1 tidak bertetangga dengan titik v_3 .



Gambar 2.14 Graf G_1 dan Graf G_2

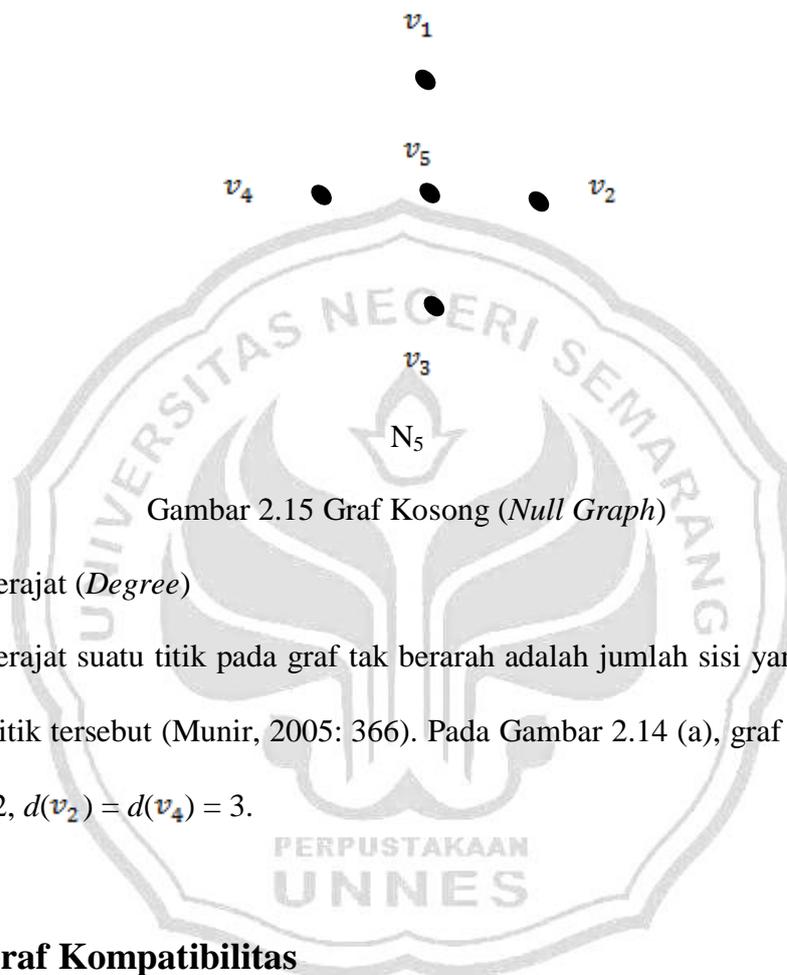
2.1.3.2 Bersisian (*Incident*)

Untuk sembarang sisi $e = (v_j, v_k)$, sisi e dikatakan bersisian dengan titik v_j dan titik v_k (Munir, 2005: 365). Pada Gambar 2.14(a), sisi e_1 bersisian dengan titik v_1 dan titik v_2 , tetapi sisi e_1 tidak bersisian dengan titik v_4 .

2.1.3.3 Graf Kosong (*Null Graph*)

Graf yang himpunan sisinya merupakan himpunan kosong disebut Graf Kosong (*Null Graph*) dan ditulis sebagai N_n , n adalah jumlah titik (Munir, 2005:

366). Menurut (Wilson & Watkin, 1976: 36) Graf kosong (graf nol) adalah graf yang tidak memiliki sisi. Graf kosong dapat ditunjukkan pada Gambar 2.15



Gambar 2.15 Graf Kosong (*Null Graph*)

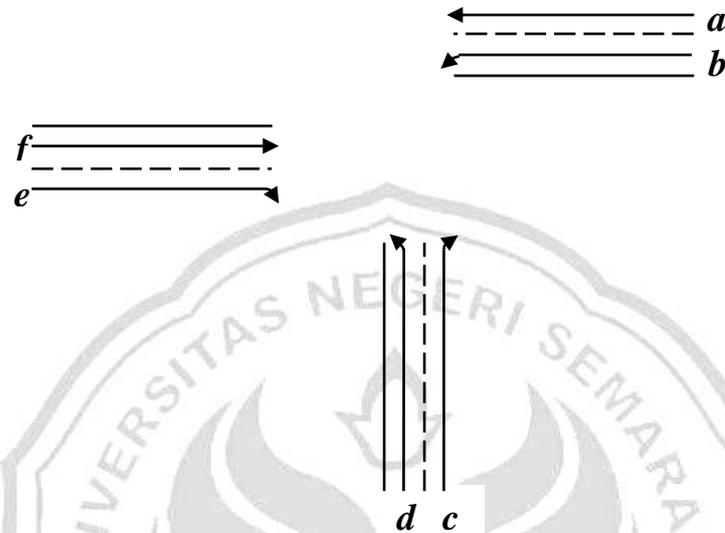
2.1.3.4 Derajat (*Degree*)

Derajat suatu titik pada graf tak berarah adalah jumlah sisi yang bersisian dengan titik tersebut (Munir, 2005: 366). Pada Gambar 2.14 (a), graf G_1 : $d(v_1) = d(v_3) = 2$, $d(v_2) = d(v_4) = 3$.

2.2 Graf Kompatibilitas

Graf-graf kompatibilitas digunakan secara luas dalam memecahkan masalah yang melibatkan pengaturan data dalam urutan tertentu (Wilson & Watkin, 1976: 61-64). Dalam graf ini, titik-titiknya menunjukkan objek-objek yang akan diatur, dan sisi-sisinya menunjukkan pasangan objek yang kompatibel (sesuai). Walaupun pengertian graf kompatibilitas ini pertama kali muncul dalam konteks genetika, pengertian ini sekarang telah dipakai dalam bidang-bidang lain seperti arkheologi, psikologi, dan penentuan usia karya tulis klasik yang

ditemukan. Aplikasi graf kompatibilitas yang pertama akan dibicarakan adalah memfase (mengatur) lampu lalu lintas. Perhatikan persimpangan jalan pada Gambar 2.16



Gambar 2.16 Contoh Persimpangan Jalan

Keterangan:

----- = arus lalu lintas

—————> = menghubungkan pasangan titik yang kompatibel

Beberapa arus lalu lintas pada persimpangan jalan ini adalah kompatibel.

Yaitu arus itu dapat bergerak pada waktu bersamaan tanpa saling membahayakan.

Contoh: arus *a* adalah kompatibel dengan arus *b*, *c*, *e*, dan *f*, tetapi tidak dengan *d*.

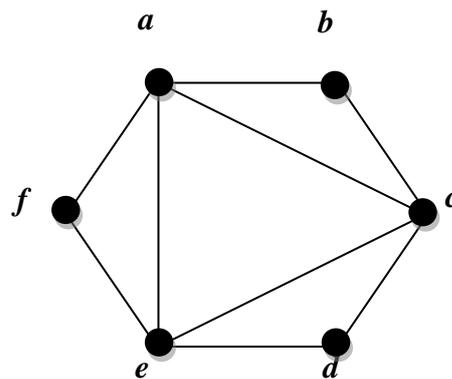
Sedang arus *f* adalah kompatibel dengan arus *a*, dan *e*, tetapi tidak dengan *b*, *c*,

dan *d*. Kompatibilitas macam ini dapat ditunjukkan dengan graf kompatibilitas,

yang titiknya mewakili arus lalu lintas dan sisinya menghubungkan pasangan titik

yang arusnya kompatibel. Graf kompatibilitas dari arus lalu lintas persimpangan

jalan di atas dapat dilihat pada Gambar 2.17

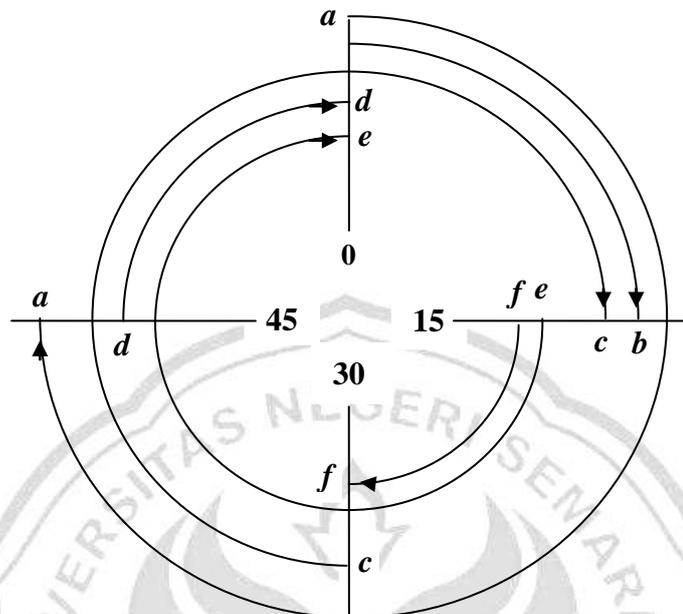


Gambar 2.17 Bentuk graf kompatibilitas dari Gambar 2.16

Untuk titik a menunjukkan arus lalu lintas di a , titik b menunjukkan arus lalu lintas di b , titik c menunjukkan arus lalu lintas di c , dan begitu pula dengan titik d , e , f . Misal sekarang seseorang ahli lalu lintas ingin mengontrol lalu lintas pada persimpangan jalan ini dengan menggunakan lampu lalu lintas. Bagaimana lampunya dapat difase sedemikian hingga arus lalu lintas yang inkompatibel (tidak kompatibel) tidak bergerak pada saat yang bersamaan?

Jika lampu lalu lintas itu beroperasi selama 60 detik putaran, maka salah satu penyelesaian adalah membiarkan setiap arus berjalan selama 10 menit. Penyelesaian ini dapat digambarkan sebagai *diagram jam*. Pengaturan khusus ini kurang memuaskan, karena setiap arus lalu lintas terhenti untuk waktu yang sangat lama menunggu gilirannya bergerak. Ingin dicari penyelesaian yang memperlihatkan kenyataan bahwa arus lalu lintas yang kompatibel dapat berjalan serentak, karena dapat mengurangi 'waktu tunggu total'.

Salah satu pengaturan yang mungkin adalah dengan diagram jam berikut yang memperbolehkan tiga arus lalu lintas yang kompatibel berjalan hampir sepanjang waktu. Salah satu cara penyelesaian dengan menggunakan diagram jam pada Gambar 2.18



Gambar 2.18 Diagram jam

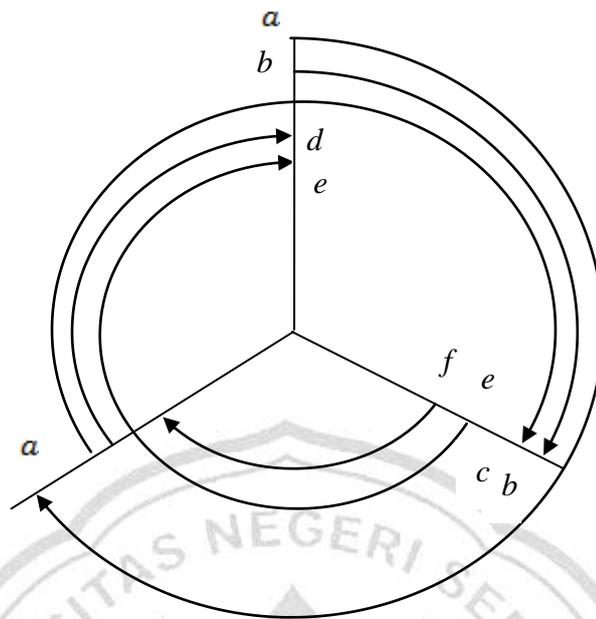
0-15 detik: arus *a*, *b*, dan *c* berjalan

15-39 detik: arus *a*, *e*, dan *f* berjalan

30-45 detik: arus *a*, *c*, dan *e* berjalan

45-50 detik: arus *c*, *d*, dan *e* berjalan

Berarti bahwa dalam setiap periode 60 detik, arus *a*, *c*, dan *e* dapat berjalan selama 45 detik, sedang arus *b*, *d*, dan *f* dapat berjalan selama 15 menit. Sehingga ‘waktu tunggu’ totalnya $(3 \times 15) + (3 \times 45) = 180$ detik, suatu pengurangan sebesar 40% dari waktu tunggu semula yang besarnya (6×50) detik. Penyelesaian yang lain diberikan dalam diagram jam ketiga berikut dan menghasilkan waktu tunggu total yang sama, yaitu 180 detik. Dalam penyelesaian ini masih terdapat tiga arus lalu lintas kompatibel yang berjalan hampir sepanjang waktu.



Gambar 2.19 Diagram jam

0-20 detik: arus a , b , dan c berjalan

20-40 detik: arus a , e , dan f berjalan

40-60 detik: arus c , d , dan e berjalan

Dalam setiap periode 60 detik, arus a , c , dan e dapat berjalan selama 40 detik, sedang arus b , d , dan f dapat berjalan selama 20 detik. Yang mana diantara kedua penyelesaian ini yang terpilih biasanya bergantung pada faktor lainnya seperti banyaknya kendaraan di setiap arus, atau kebutuhan memberikan waktu minimum (katakan) 20 detik kepada setiap arus. Yang diinginkan disini adalah mendapatkan beberapa penyelesaian efisien yang memenuhi semua persyaratan lain itu.

Penyelesaian ini dapat diperoleh dengan melihat graf kompatibilitasnya. Karena tujuannya untuk mendapatkan banyak maksimum arus lalu lintas yang bergerak pada waktu bersamaan, diinginkan untuk mendapatkan subgraf dari graf kompatibilitas yang mencerminkan persyaratan ini. Subgraf lengkap (komplit) ini

berkorespondensi dengan arus yang saling kompatibel. Contoh subgraf lengkap ini adalah segitiga yang terbentuk oleh titik-titik abc , atau aef , atau ace , atau cde . Perhatikan bahwa segitiga itu tepatnya adalah arus lalu lintas pada penyelesaian di atas. Ide ini diaplikasi secara lebih umum sehingga diperoleh pedoman umum berikut ini:

- a) Waktu siklus
- b) Dapatkan subgraf komplit terbesar yang memuat setiap titik graf kompatibilitas itu
- c) Bagilah waktu yang tersedia dengan banyaknya subgraf komplit pada langkah b), dan alokasikan subgraf komplit untuk setiap periode waktu.

Pada contoh di atas, langkah b) menghasilkan subgraf lengkap abc , aef , dan cde , yang bersama-sama memuat keenam titik itu dan memberikan penyelesaian ketiga di atas. Penyelesaian kedua timbul jika subgraf komplit ace juga diikutkan (Wilson & Watkin, 1976: 61-64).

2.3 Teori Transportasi

Persimpangan jalan adalah suatu daerah umum di mana dua atau lebih ruas jalan (*link*) saling bertemu atau berpotongan yang mencakup fasilitas jalur jalan (*roadway*) dan tepi jalan (*road side*), di mana lalu lintas dapat bergerak di dalamnya (Harianto, 2004: 2). Setiap persimpangan mencakup pergerakan lalu lintas menerus. Lalu lintas yang saling memotong pada satu atau lebih dari kaki persimpangan dan mencakup juga pergerakan perputaran.

Pergerakan lalu lintas ini dikendalikan berbagai cara, bergantung pada jenis persimpangannya. Dilihat dari bentuknya ada beberapa macam jenis persimpangan sebidang, sebagai berikut:

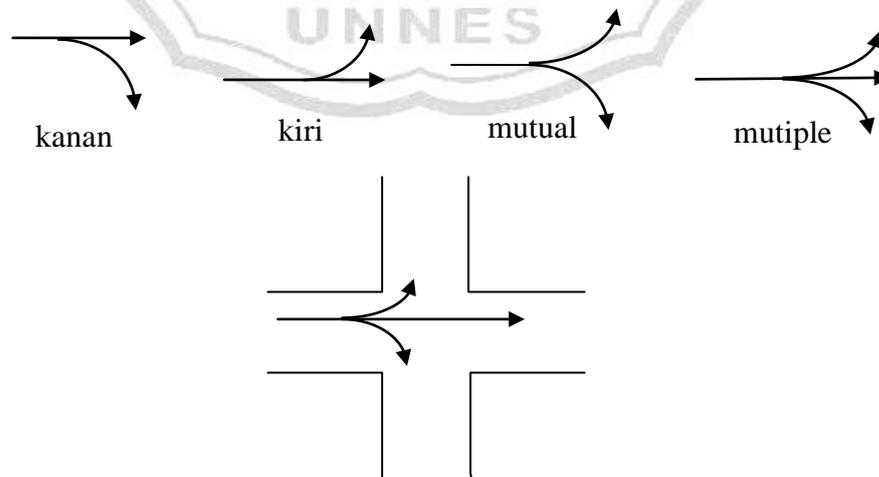
- (1) Pertemuan atau persimpangan sebidang bercabang tiga.
- (2) Pertemuan atau persimpangan sebidang bercabang empat.
- (3) Pertemuan atau persimpangan sebidang bercabang banyak.
- (4) Bundaran (*Rotary Intersection*).

2.3.1 Bentuk Alih Gerak (*manuver*)

Dari sifat dan tujuan gerakan di daerah persimpangan, dikenal beberapa bentuk alih gerak (*manuver*) antara lain, *diverging* (memisah), *merging* (menggabung), *crossing* (memotong), dan *weaving* (menyilang).

2.3.1.1 *Diverging* (memisah)

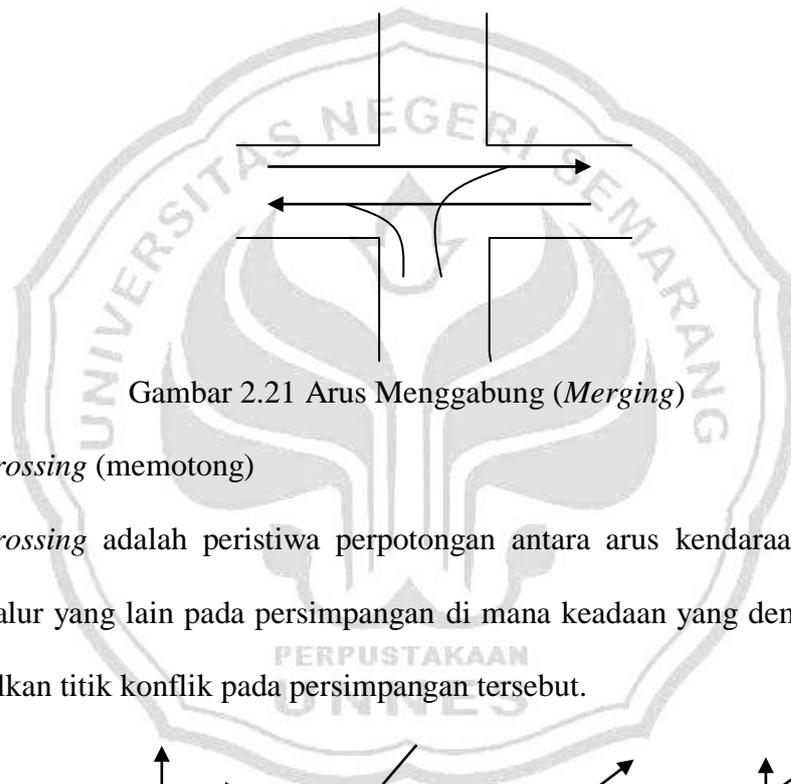
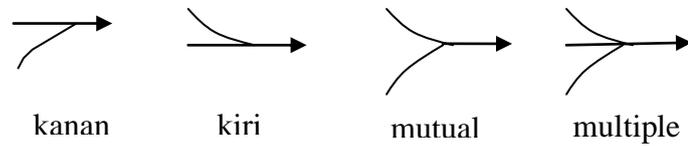
Diverging adalah peristiwa memisahkannya kendaraan dari suatu arus yang sama ke jalur yang lain:



Gambar 2.20 Arus Memisah (*Diverging*)

2.3.1.2 *Merging* (menggabung)

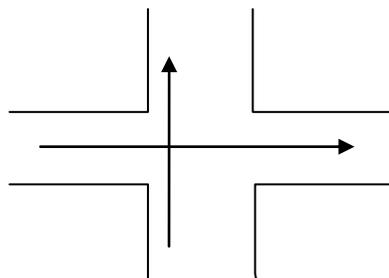
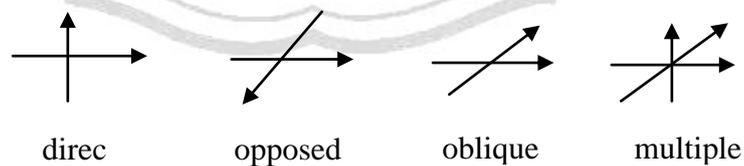
Merging adalah peristiwa menggabungkannya kendaraan dari suatu jalur ke jalur yang lain:



Gambar 2.21 Arus Menggabung (*Merging*)

2.3.1.3 *Crossing* (memotong)

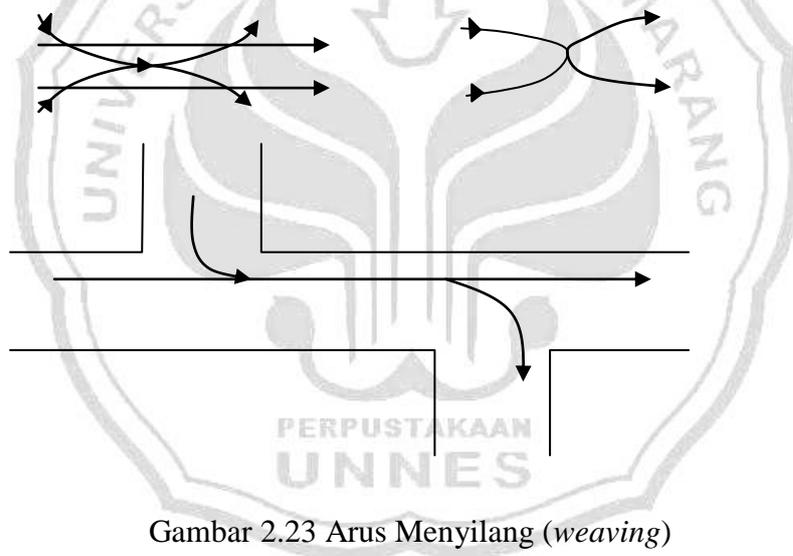
Crossing adalah peristiwa perpotongan antara arus kendaraan dari satu jalur ke jalur yang lain pada persimpangan di mana keadaan yang demikian akan menimbulkan titik konflik pada persimpangan tersebut.



Gambar 2.22 Arus Memotong (*Crossing*)

2.3.1.4 *Weaving* (menyilang)

Weaving adalah pertemuan dua arus lalu lintas atau lebih yang berjalan menurut arah yang sama sepanjang suatu lintasan di jalan raya tanpa bantuan rambu lalu lintas. Gerakan ini sering terjadi pada suatu kendaraan yang berpindah dari suatu jalur ke jalur lain misalnya pada saat kendaraan masuk ke suatu jalan raya dari jalan masuk, kemudian bergerak ke jalur lainnya untuk mengambil jalan keluar dari jalan raya tersebut keadaan ini juga akan menimbulkan titik konflik pada persimpangan tersebut.

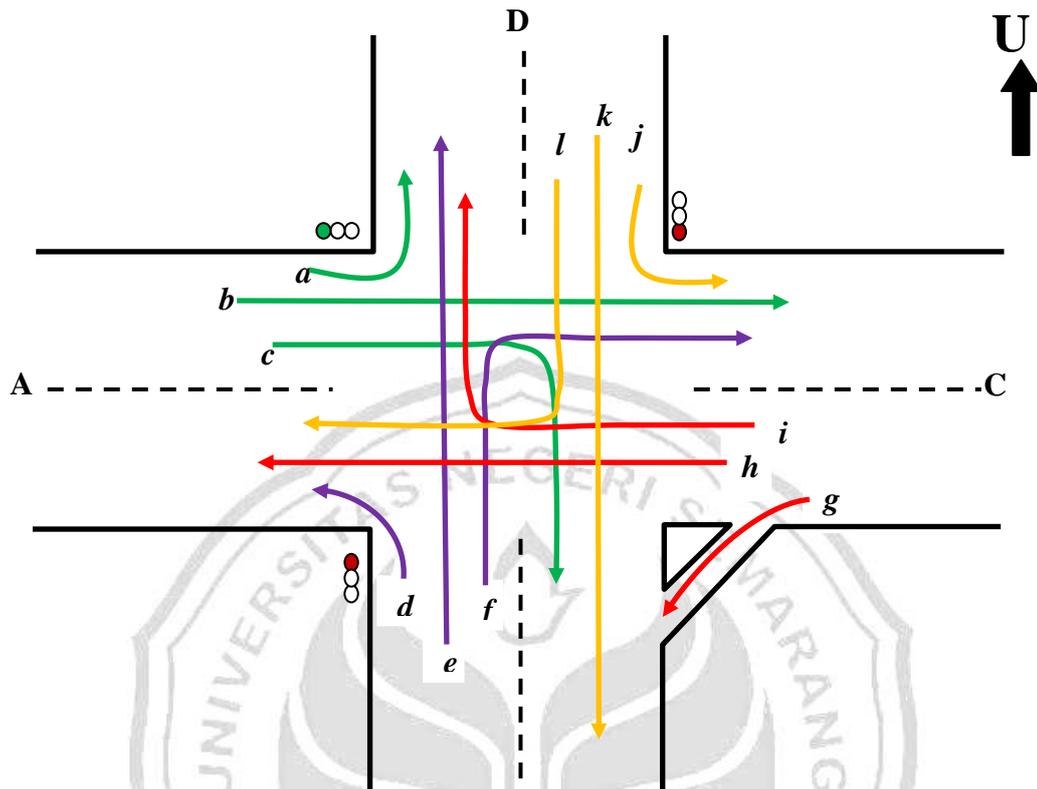


Gambar 2.23 Arus Menyilang (*weaving*)

2.3.2 Skema Persimpangan Simping Empat dan Simping Tiga

2.3.2.2 Persimpangan simping empat jalan Kaligarang-Kelud Raya

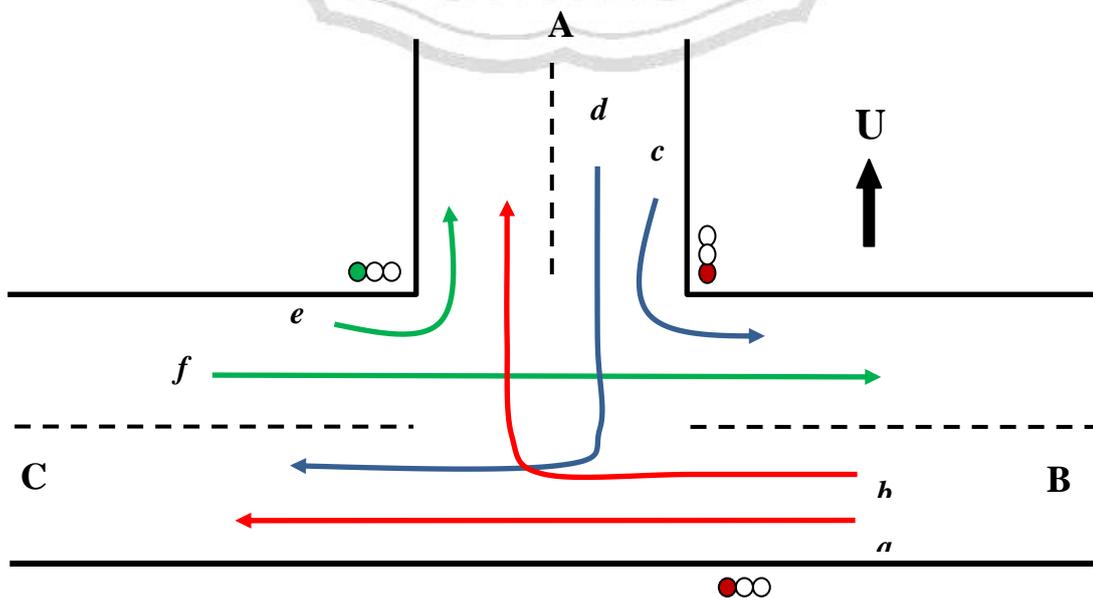
Untuk dapat melihat skema persimpangan simping empat jalan Kaligarang-Kelud Raya-Bendungan, bisa dilihat pada Gambar 2.24.



Gambar 2.24 Skema Persimpangan Jalan Kaligarang–Kelud Raya-Bendungan

2.3.2.2 Persimpangan simpang tiga jalan Majapahit-Supriyadi

Untuk dapat melihat skema persimpangan simpang tiga jalan Majapahit-Supriyadi, dapat dilihat pada Gambar 2.25.



Gambar 2.25 Skema Persimpangan Jalan Majapahit-Supriyadi

2.4 Visual Basic 6.0

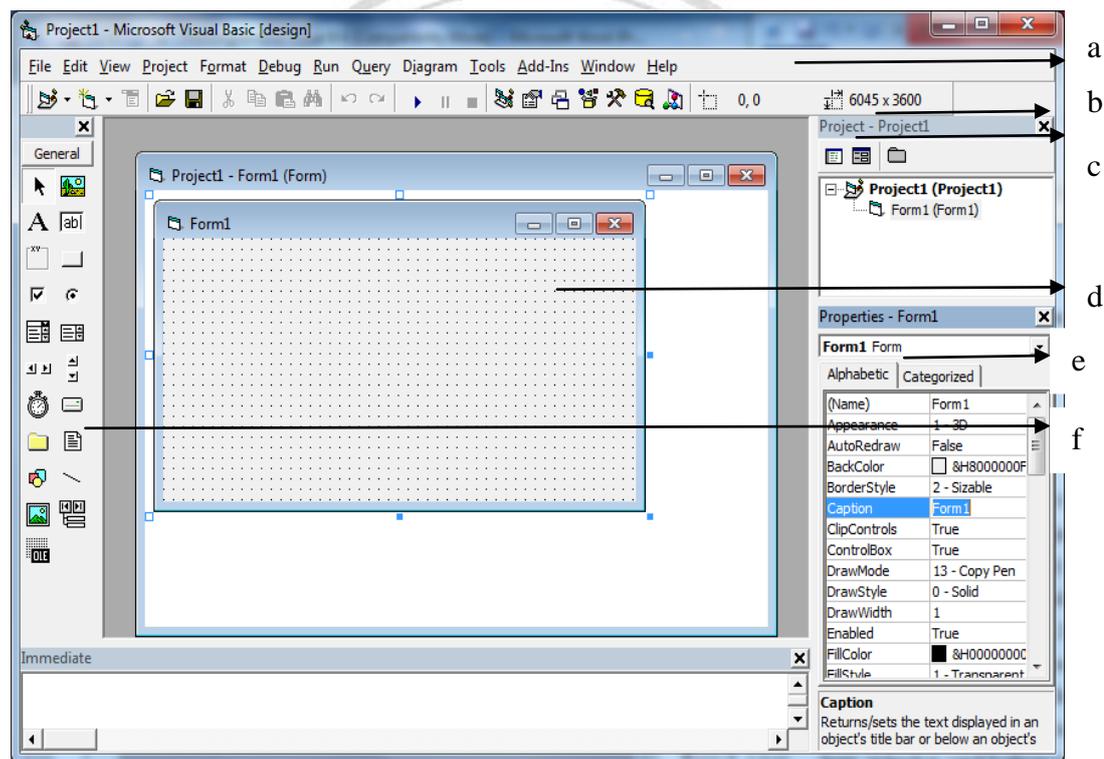
Microsoft *Visual Basic 6.0* merupakan bahasa pemrograman yang cukup populer dan mudah untuk dipelajari serta dapat membuat program dengan aplikasi GUI (*Graphical User Interface*) atau program yang memungkinkan pemakai komputer berkomunikasi dengan komputer tersebut dengan menggunakan modus grafik atau gambar (Madcoms, 2001: 3).

Visual Basic adalah salah satu *development tool* untuk membangun aplikasi dalam lingkungan windows. Dalam pengembangan aplikasi, *visual basic* menggunakan pendekatan visual untuk merancang *user interface* dalam bentuk form, sedangkan untuk kodingnya menggunakan pendekatan bahasa basic yang cenderung mudah dipelajari. Basic yang cenderung mudah dipelajari. *Visual Basic* telah menjadi tools yang terkenal bagi para pemula maupun para developer dalam pengembangan aplikasi skala kecil sampai ke skala besar. Dalam lingkungan *Window's User-interface* sangat memegang peranan penting, karena dalam pemakaian aplikasi yang kita buat, pemakai senantiasa berinteraksi dengan *user interface* tanpa menyadari bahwa dibelakangnya berjalan instruksi-instruksi program yang mendukung tampilan dan proses yang dilakukan.

Pada pemrograman *Visual Basic*, pengembangan aplikasi dimulai dengan pembentukan *user interface*, kemudian mengatur properti dari objek-objek yang digunakan dalam *user interface*, dan baru dilakukan penulisan kode program untuk menangani kejadian-kejadian (*event*). Tahap pengembangan aplikasi demikian dikenal dengan istilah pengembangan aplikasi dengan pendekatan *Bottom Up*.

2.4.1. Komponen Utama *Microsoft Visual Basic*

Mengenal komponen-komponen *Visual Basic* merupakan hal yang sangat penting. Komponen-komponen ini akan membantu kita dalam pembuatan program (Yuniar, 2006:7). Pertama kali menjalankan *Visual Basic* akan tampil beberapa komponen, yaitu Baris Menu, *Toolbar*, *Form*, Jendela Projek, Jendela *Properties*, dan Jendela *Form Layout*, seperti terlihat pada Gambar 2.26



Gambar 2.26. Tampilan Utama *Visual Basic*

a. Baris Menu

Menyimpan seluruh perintah yang terdapat pada *Visual Basic*.

b. *Toolbar*

Merupakan kumpulan ikon-ikon objek untuk membuat tampilan program atau *form*.

c. *Toolbar*

Merupakan kumpulan ikon-ikon perintah yang sering dipakai pada *Visual Basic*.

d. *Form*

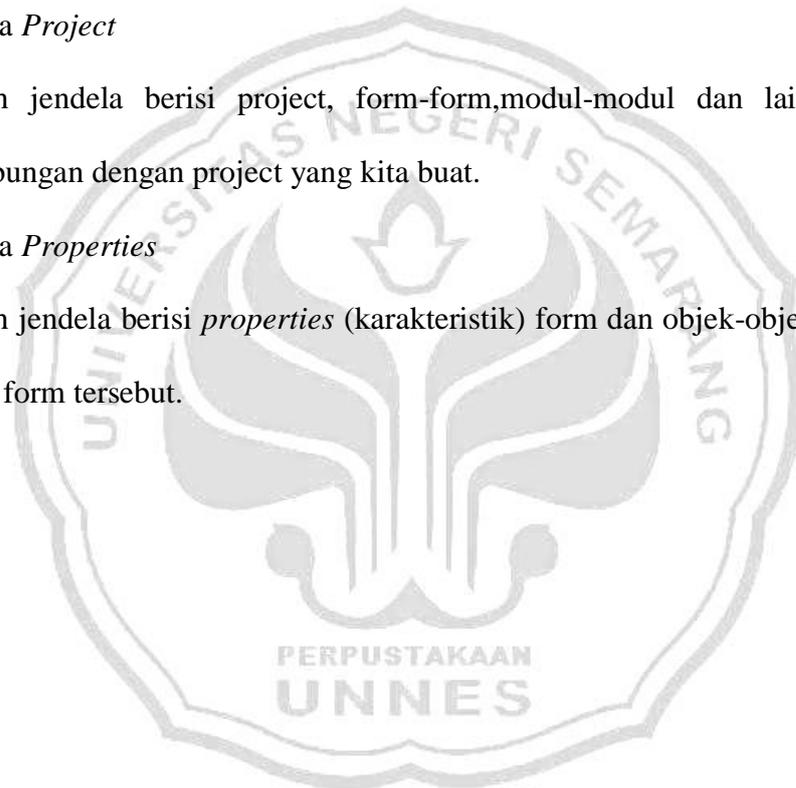
Tempat untuk meletakkan objek-objek sebagai tampilan program.

e. *Jendela Project*

Adalah jendela berisi project, form-form, modul-modul dan lainnya yang berhubungan dengan project yang kita buat.

f. *Jendela Properties*

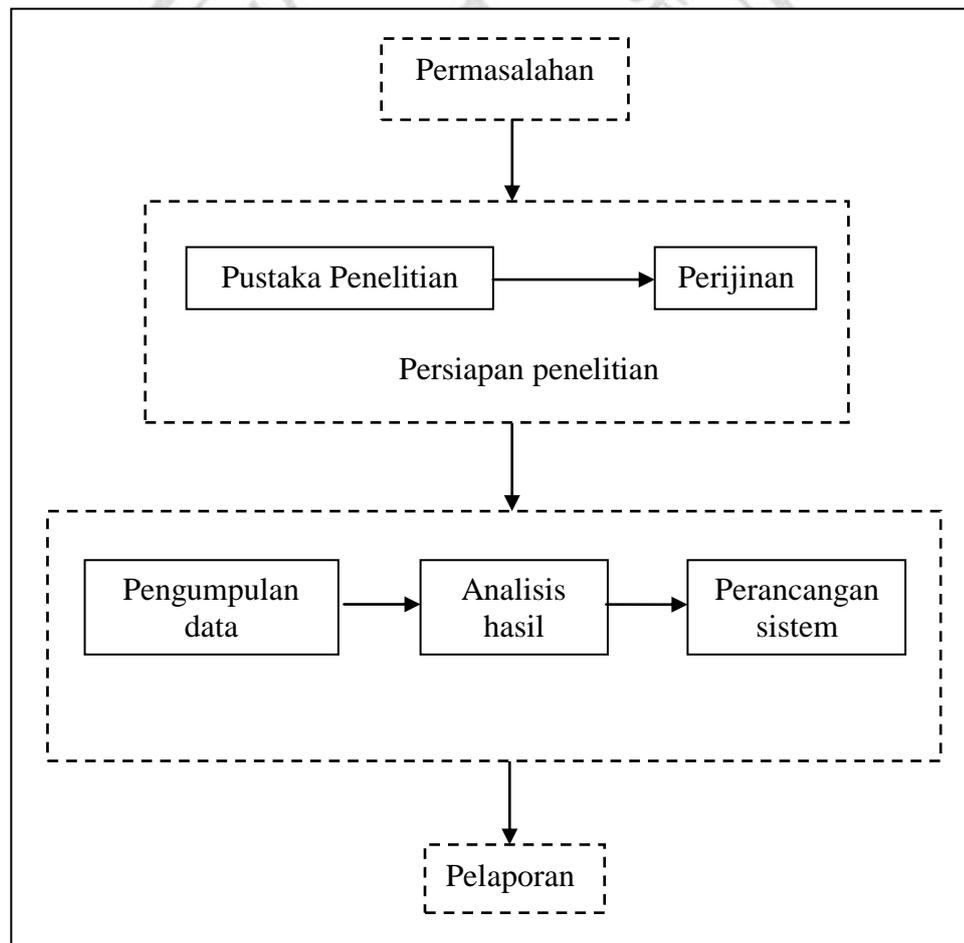
Adalah jendela berisi *properties* (karakteristik) form dan objek-objek yang ada dalam form tersebut.



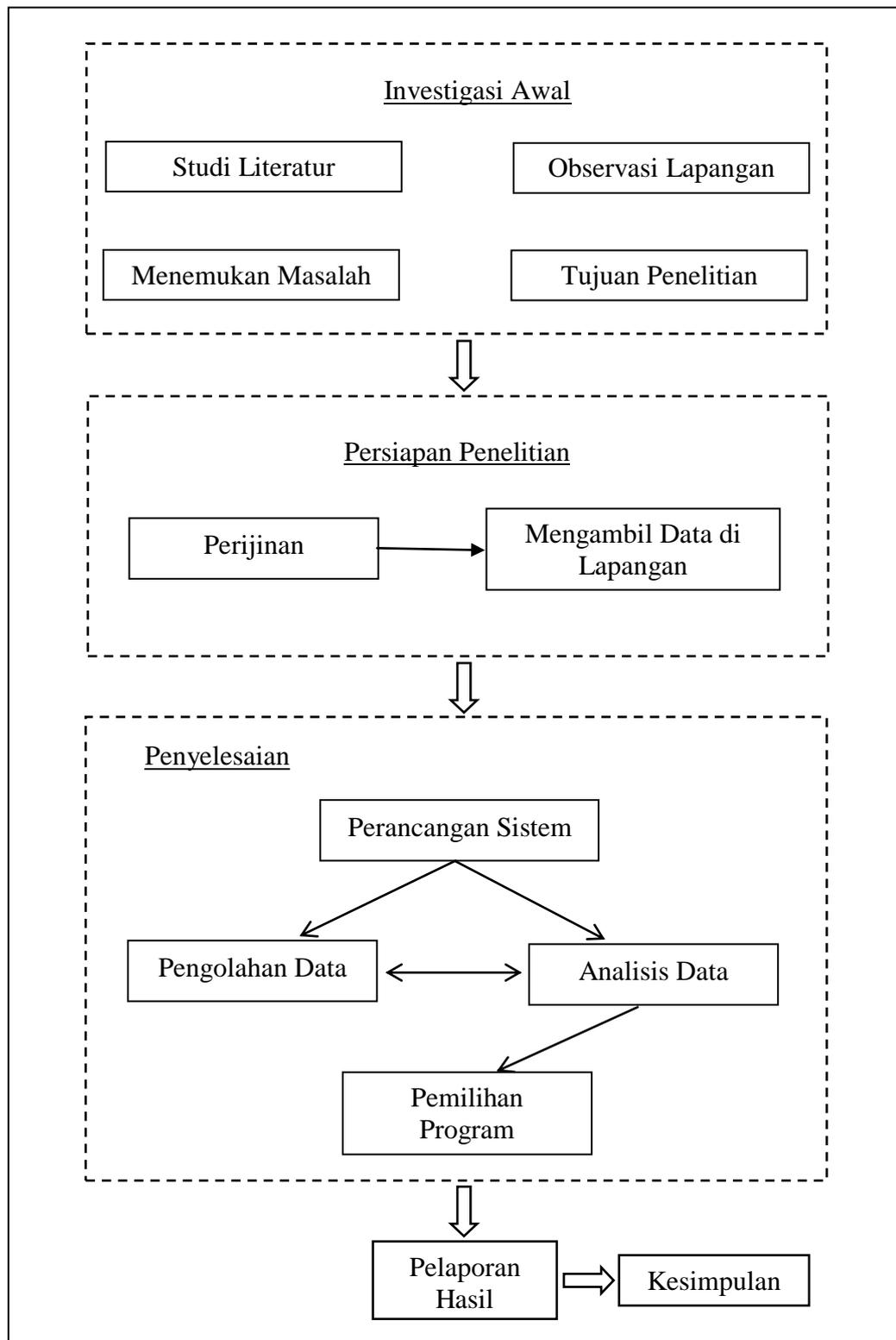
BAB 3

METODE PENELITIAN

Untuk melakukan penelitian harus memperhatikan prosedur dan langkah-langkah yang akan dilakukan untuk memulai penelitian sehingga dapat terarah dan terlaksana dengan baik dalam hal pelaporan penelitian. Langkah-langkah dasar yaitu persiapan penelitian, pelaksanaan penelitian dan pelaporan. Adapun alur yang menggambarkan kerja pada penelitian ini terdapat pada Gambar 3.1.



Gambar 3.1 Diagram Alur Kerja



Gambar 3.2 Bagan Alur Proses Penelitian

3.1 Kajian Pustaka

Terdapat materi yang menarik terkait dengan bidang matematika diskrit yang pernah disinggung dalam perkuliahan tapi tidak diangkat dalam bentuk tulisan yaitu graf. Melakukan telaah pustaka dari berbagai referensi yang ada dan melakukan konfirmasi dan konsultasi dengan dosen yang membidangi masalah tersebut membuahakan gagasan untuk menuliskannya dalam bentuk skripsi.

3.2 Persiapan

3.2.1 Pengumpulan Pustaka Penelitian

Dalam pengumpulan pustaka penelitian ini diambil berbagai sumber seperti buku-buku, artikel, jurnal dan skripsi yang berkaitan dengan konsep dasar teori graf dan *Visual Basic 6.0* selain itu juga berupa artikel maupun buku-buku yang berhubungan dengan objek pebelitian yaitu tentang waktu tunggu di persimpangan simpang empat jalan Kaligarang-Kelud Raya dan simpang tiga jalan Majapahit-Supriyadi, Semarang.

3.2.2 Menentukan Objek Penelitian

Objek penelitian dalam penelitian ini adalah waktu tunggu optimal pada persimpangan simpang empat jalan Kaligarang-Kelud Raya dan simpang tiga jalan Majapahit-Supriyadi. Objek penelitian yang dilakukan adalah menghitung waktu tunggu pada lampu lalu lintas.

3.3 Pelaksanaan Penelitian

3.3.2 Pengumpulan data

Dalam penelitian ini data yang diambil adalah data primer, yaitu data yang diperoleh dengan pengamatan langsung dan pencatatan secara langsung dari

tempat penelitian, yaitu waktu tunggu total pada lampu lalu lintas di persimpangan simpang empat jalan Kaligarang-Kelud Raya dan simpang tiga jalan Majapahit-Supriyadi.

3.3.2 Tempat dan Waktu Penelitian

Penelitian dilakukan di persimpangan simpang empat jalan Kaligarang. Bagian yang diamati adalah berapa waktu tunggu total optimal pada lampu lalu lintas. Waktu penelitian dilakukan pada pagi, siang, dan sore.

3.3.4 Teknik Pengumpulan Data

Metode yang digunakan dalam penelitian ini adalah metode observasi yang merupakan metode pengumpulan data dengan cara pengamatan atau peninjauan langsung terhadap objek penelitian. Pengambilan data dilakukan melalui penelitian secara langsung. Data yang akan diambil sedemikian rupa sehingga tiap objek penelitian dari populasi mempunyai kesempatan yang sama untuk dipilih. Objek populasi dari penelitian ini adalah jumlah kendaraan serta jumlah jalur yang ada pada persimpangan simpang empat jalan Kaligarang.

Hasil dari pengambilan data ini nantinya akan disajikan dalam bentuk graf kompatibel serta simulasi dengan menggunakan bantuan program *Visual Basic* 6.0. Data yang diambil dianggap telah mewakili semua hari di hari-hari biasa.

3.3.5 Analisis Hasil

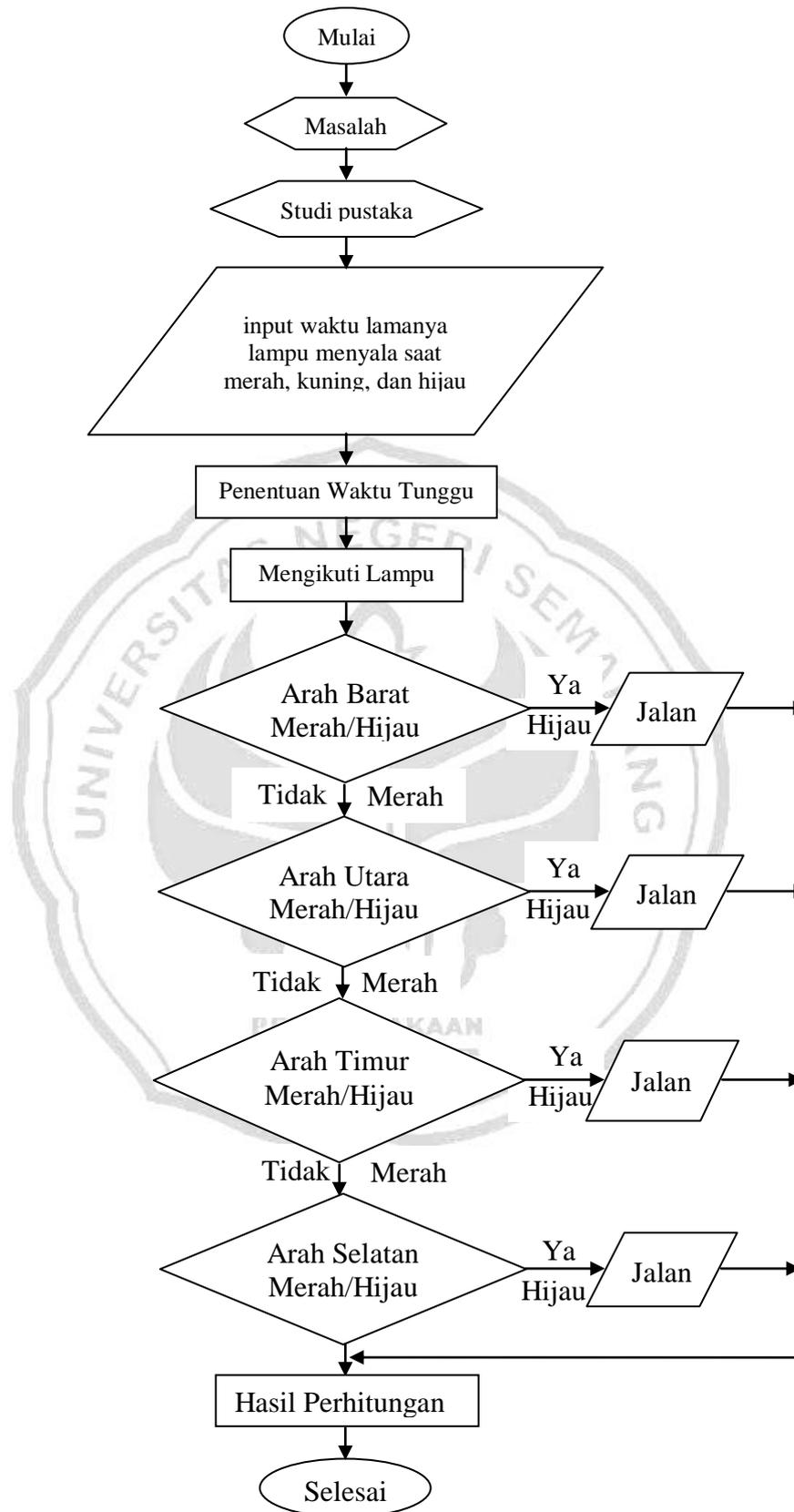
Proses analisis hasil ini akan dilakukan pada bab pembahasan pada penelitian ini, sedangkan proses yang terakhir yaitu penelitian dapat dibuktikan dengan terselesaikannya skripsi ini.

3.3.6 Prosedur Pengolahan Data

Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Pengolahan data adalah manipulasi data kedalam bentuk yang lebih berarti berupa informasi. Sedangkan informasi adalah hasil dari kegiatan-kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dan suatu kegiatan atau peristiwa.

Data pada penelitian ini diperoleh dengan mengamati jumlah waktu pada persimpangan simpang empat Kaligarang secara langsung (data primer). Data yang dikumpulkan meliputi jumlah waktu selama lampu lalu lintas menyala merah, kuning, hijau, dan situasi keadaan pada saat itu. Pengamatan dilakukan tanpa mengganggu arus lalu lintas yang sedang berlangsung, yaitu mengambil tempat di tepi jalan raya.

Pengolahan data merupakan bagian yang amat penting dalam metode ilmiah karena dengan pengolahan data-data tersebut dapat diberi arti dan makna yang berguna dalam memecahkan masalah penelitian. Pengolahan data secara sederhana diartikan sebagai proses mengartikan data-data lapangan sesuai dengan tujuan, rancangan, dan sifat penelitian. Prosedur pengolahan data dapat digambarkan dengan flowchart seperti yang terdapat pada Gambar 3.3.



Gambar 3.3 Diagram Alur Program

Adapun penjelasan dari flowchart di atas adalah sebagai berikut:

1. Adanya permasalahan yang diangkat dalam skripsi ini sehingga dilakukan penelitian dan melakukan analisis lebih lanjut.
2. Untuk memulai penelitian, terlebih dahulu dengan melakukan studi pustaka mengenai topik yang akan diangkat pada penelitian. Dalam hal ini, harus memahami permasalahan dan metode yang digunakan, cara pengumpulan data, dan teknik pengolahan data.
3. Menentukan tempat penelitian dan melakukan penelitian secara langsung. Dalam penelitian ini, harus didapat data waktu nyala lampu lalu lintas dalam satuan waktu yang ditentukan oleh peneliti.
4. Data yang sudah didapat yaitu berapa lama masing-masing waktu nyala lampu merah, kuning, hijau dalam setiap jalur. Jika waktu yang didapat tidak sesuai dengan situasi tempat penelitian, maka dapat diperbaiki dengan melihat situasi yang memungkinkan.
5. Membuat subgraf lengkap terbesar dari bentuk persimpangan yang telah diamati.
6. Dalam simulasi program nanti akan dibuat langkah bagaimana hasil tampilan akhirnya. Ketika mengikuti lampu merah dari masing-masing arah, maka kendaraan yang melintas akan berhenti sesuai nyala lampu dan menunggu nyala lampu hijau. Ketika mengikuti warna lampu hijau dari masing-masing arah, maka kendaraan yang melintas akan langsung jalan mengikuti warna lampu.

7. Setelah membuat bentuk subgraf lengkap terbesar, maka akan diperoleh hasil waktu tunggu optimal dan waktu tunggu per jalur pada setiap masing-masing jalur, yaitu dengan memilih waktu tunggu yang paling efisien agar bisa diteapkan pada tempat penelitian.
8. Menentukan hasil dan pembahasan yang dapat diperoleh dari ukuran waktu tunggu dan mengambil kesimpulan.



BAB 4

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Hasil Penelitian

Di pembahasan ini mengkaji tentang penerapan graf kompatibel untuk pengaturan lalu lintas di persimpangan jalan, dengan lokasi penelitian pada persimpangan simpang empat jalan Kaligarang-Kelud Raya-Bendungan dan simpang tiga jalan Majapahit-Supriyadi. Penelitian ini memerlukan data tentang bentuk persimpangan jalan dari kedua lokasi serta menentukan arus yang terjadi pada persimpangan tersebut.

4.1.1 Penerapan Graf Kompatibel pada Penentuan Waktu Tunggu Total Optimal di Persimpangan Jalan

Berdasarkan pengamatan yang dilakukan di lapangan, selanjutnya dibuat gambar persimpangan tersebut kemudian diubah kedalam bentuk graf kompatibel. Sebelum mengubah ke dalam bentuk graf kompatibel, diperoleh definisi arus lalu lintas yang kompatibel, yaitu: dua buah arus lalu lintas dikatakan kompatibel jika keduanya dapat berjalan bersamaan dengan aman atau tidak berpotongan.

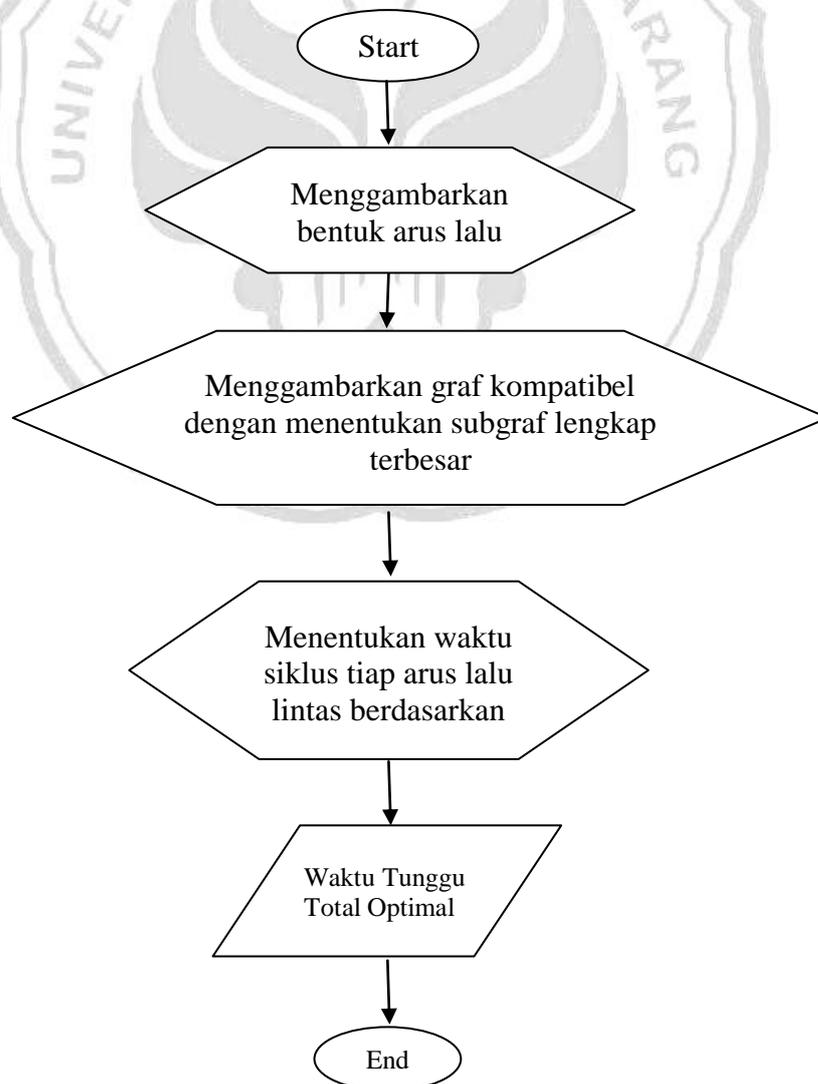
4.1.2 Algoritma

Algoritma pemodelan lalu lintas dengan graf kompatibel adalah sebagai berikut:

1. Menggambarkan graf kompatibel, dimana titik-titiknya menunjukkan arus lalu lintas yang akan diatur, dan sisi-sisinya menunjukkan pasangan objek yang kompatibel. Dua buah titik dihubungkan dengan sisi jika dua arus lalu lintas kompatibel.

2. Menentukan subgraf lengkap terbesar. Graf lengkap ialah graf sederhana yang setiap titiknya mempunyai sisi ke semua titik lainnya.
3. Menentukan waktu siklus tiap arus lalu lintas berdasarkan banyaknya subgraf lengkap. Membagi 1 periode waktu dari jumlah banyaknya subgraf lengkap terbesar, kemudian mengalokasikan siklus waktu tiap jalur.
4. Menentukan waktu tunggu total. Di mana waktu tunggu total dihitung dengan menggunakan hasil alokasi periode waktu tiap jalur.

4.1.3 Diagram Alur (*Flowchart*)



Gambar 4.1 Diagram Alur (flowchart)

4.1.4 Tahap Pengambilan Data

Langkah pertama adalah menentukan lokasi penelitian. Penelitian akan dilaksanakan di persimpangan jalan Kaligarang-Bendungan-Kelud Raya dan jalan Majapahit-Supriyadi. Pada penelitian ini, dibuat penerapan graf kompatibel untuk menentukan waktu tunggu total optimal lampu lalu lintas dan program simulasi untuk menggambarkan keadaan yang mirip dengan objek penelitian, yaitu persimpangan simpang empat jalan Kaligarang-Kelud Raya-Bendungan dan simpang tiga jalan Majapahit-Supriyadi Semarang. Selanjutnya gambar persimpangan tersebut diubah ke bentuk graf kompatibel atau dibuat sub graf lengkap terbesar, kemudian dilakukan proses untuk mencari beberapa arah yang dapat berjalan secara bersamaan dengan aman dan konsisten berdasarkan waktu tunggu tiap jalur.

Asumsi-asumsi yang digunakan pada penelitian ini adalah sebagai berikut:

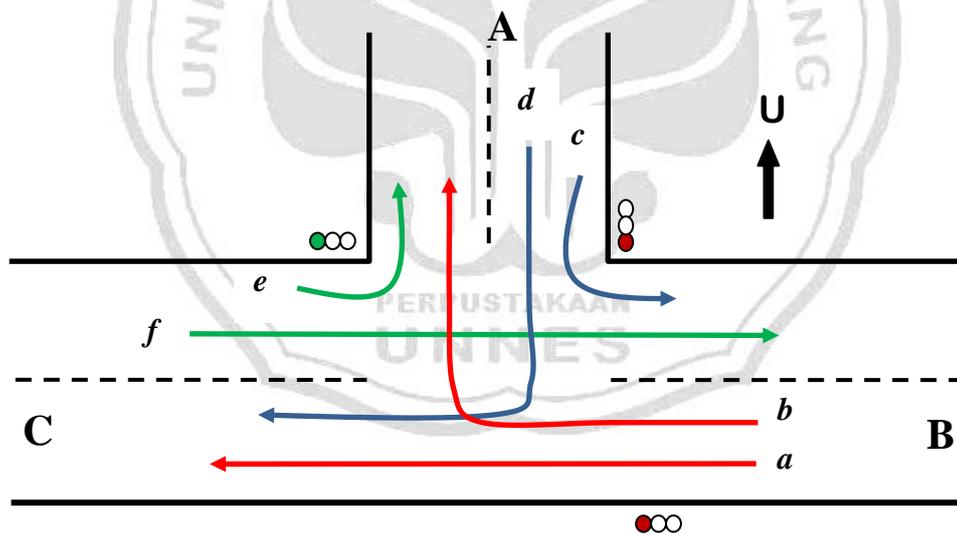
1. Waktu pengambilan data akan dibagi pada tiga periode waktu, yaitu:
 - a. Pagi hari, dibatasi pada pukul 06.30-07.30 WIB, dengan asumsi banyaknya pekerja dan pelajar yang berangkat pada jam tersebut.
 - b. Siang hari, dibatasi pada pukul 12.30-13.30 WIB, dengan asumsi banyaknya pelajar yang pulang dan aktivitas lain pada jam tersebut.
 - c. Sore hari, dibatasi pada pukul 16.30-17.30 WIB, dengan asumsi banyaknya pekerja yang pulang.
2. Data yang diamati pada tiap ruas jalan dari dua arah hanya kendaraan bermotor dan roda empat, sedangkan pejalan kaki dan penyeberang jalan diabaikan.

3. Arus lalu lintas yang diamati yaitu yang berbelok kiri mengikuti lampu dan belok kiri tidak mengikuti lampu.

4.1.5 Gambar Sistem Arus Lalu Lintas

Jika akan menggambar arus lalu lintas perlu melakukan observasi awal untuk menentukan banyaknya lintasan yang diperbolehkan melintas pada persimpangan tersebut dan untuk menentukan lintasan mana saja yang diperbolehkan melintas. Berdasarkan pengamatan yang dilakukan di lapangan, sistem lalu lintas yang diterapkan pada persimpangan jalan dapat dilihat pada Gambar 4.2 dan Gambar 4.3.

4.1.5.1 Simpang tiga Jl. Majapahit-Supriyadi



Gambar 4.2 Sistem Lalu Lintas pada Persimpangan Majapahit

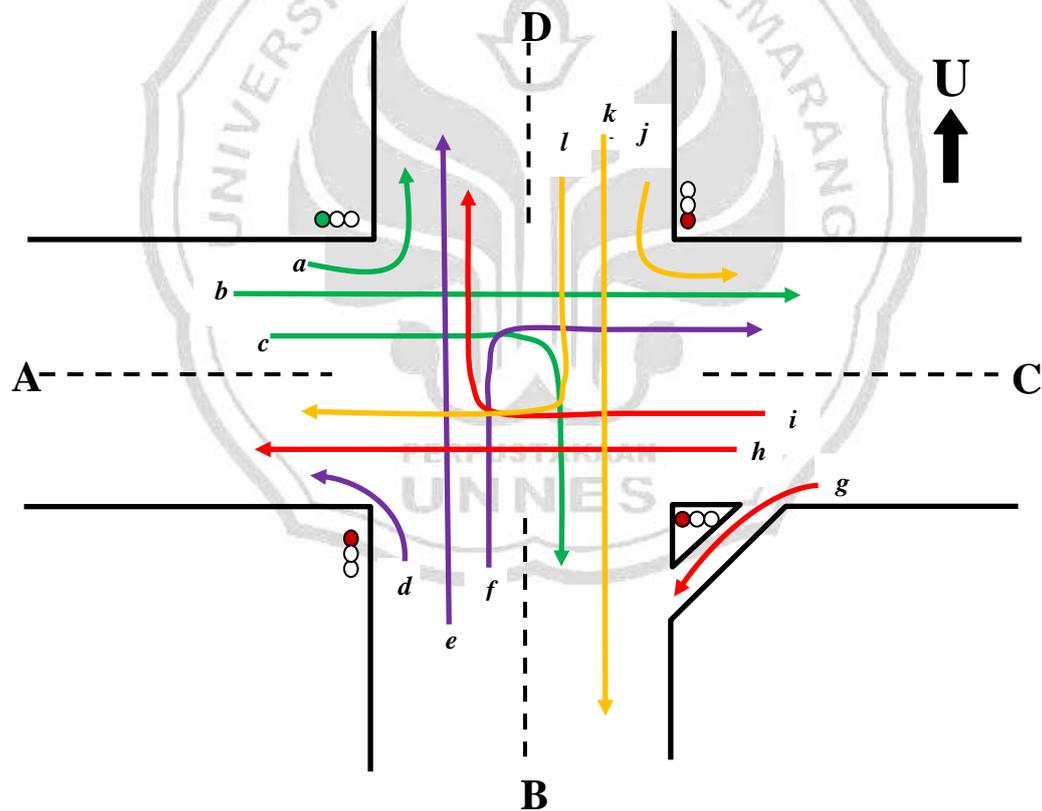
Keterangan gambar:

- A. Jl. Supriyadi
- B. Jl. Majapahit dari arah timur
- C. Jl. Majapahit dari arah barat

Pada Gambar 4.1 terdapat beberapa lintasan, yaitu:

1. Lintasan AB: dari arah Supriyadi ke Pedurungan
2. Lintasan AC: dari arah Supriyadi ke Gayam Sari
3. Lintasan BA: dari arah Pedurungan ke Supriyadi
4. Lintasan BC: dari arah Pedurungan ke Gayam Sari
5. Lintasan CA: dari arah Gayam Sari ke Supriyadi
6. Lintasan CB: dari arah Gayam Sari ke Pedurungan

4.1.5.2 Simpang empat Jl. Kaligarang-Kelud Raya-Bendungan



Gambar 4.3 Sistem Lalu Lintas pada Persimpangan Kaligarang

Keterangan gambar:

A = Jalan Kelud Raya

B = Jalan Kaligarang arah Karyadi

C = Jalan Bendungan

D = Jalan Kaligarang arah Sampokong

Untuk *a, b, c, d, e, f, g, h, i, j, k*, dan *l* menunjukkan arus lalu lintas dari masing-masing jalur.

Pada Gambar 4.3 terdapat beberapa lintasan, sebagai berikut.

1. Lintasan AC : dari arah Pamularsih ke Karyadi
2. Lintasan AB : dari arah Pamularsih ke Kelud Raya
3. Lintasan AD : dari arah Pamularsih ke Bendungan
4. Lintasan BD : dari arah Kelud ke Bendungan
5. Lintasan BC : dari arah Kelud ke Karyadi
6. Lintasan BA : dari arah Kelud ke Pamularsih
7. Lintasan CA : dari arah Karyadi ke Pamularsih
8. Lintasan CD : dari arah Karyadi ke Bendungan
9. Lintasan CB : dari arah Karyadi ke Kelud
10. Lintasan DB : dari arah Bendungan ke Kelud
11. Lintasan DA : dari arah Bendungan ke Pamularsih
12. Lintasan DC : dari arah Bendungan ke Karyadi

4.1.6 Hasil Pengambilan Data

4.1.6.1 Lokasi simpang tiga Jl. Majapahit-Supriyadi

Pengambilan data ini dilakukan selama 3 hari dengan tiga waktu yang berbeda, yaitu pagi, siang dan sore hari. Berdasarkan hasil yang didapat setelah dilakukannya pengamatan di lapangan, ternyata data yang diperoleh menunjukkan sama dalam pengambilan pada waktu yang berbeda.

Tabel 4.1 Lama siklus waktu awal lampu lalu lintas pada persimpangan
jalan Majapahit-Supriyadi

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Supriyadi (A)	83	3	40
Jalan Majapahit dari arah timur (B)	106	3	17
Jalan Majapahit dari arah barat (C)	88	3	35

Dari hasil pengamatan yang sudah dilakukan pada waktu yang berbeda, diperoleh bahwa ternyata siklus lampu lalu lintas pada persimpangan Supriyadi sama. Hal ini sangat tidak efisien, dikarenakan berdasarkan jumlah/kepadatan kendaraan yang melintas pada waktu pagi, siang, maupun sore berbeda.

4.1.6.2 Lokasi simpang empat Jl. Kaligarang-Kelud Raya-Bendungan

Pengambilan data ini dilakukan selama 3 hari dengan tiga waktu yang berbeda, yaitu pagi, siang dan sore hari. Berdasarkan hasil yang didapat setelah dilakukannya pengamatan di lapangan, ternyata data yang diperoleh menunjukkan sama dalam pengambilan pada waktu yang berbeda.

Tabel 4.2 Lama siklus waktu awal lampu lalu lintas pada persimpangan
jalan Kaligarang-Kelud-Bendungan

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Kaligarang arah Pamularsih (A)	83	3	40
Jalan Kelud Raya (B)	106	3	17
Jalan Kaligarang arah Karyadi (C)	88	3	35
Jalan Bendungan (D)	112	3	11

Dari hasil pengamatan yang sudah dilakukan pada waktu yang berbeda, diperoleh bahwa ternyata siklus lampu lalu lintas pada persimpangan Kaligarang sama. Hal ini sangat tidak efisien, dikarenakan berdasarkan jumlah/kepadatan kendaraan yang melintas pada waktu pagi, siang, maupun sore berbeda.

4.2 Pembahasan

Pemodelan graf kompatibel pada masing-masing persimpangan diawali dengan penggambaran bentuk persimpangan serta arus yang terjadi pada persimpangan kemudian dilakukan mencari subgraf lengkap terbesar.

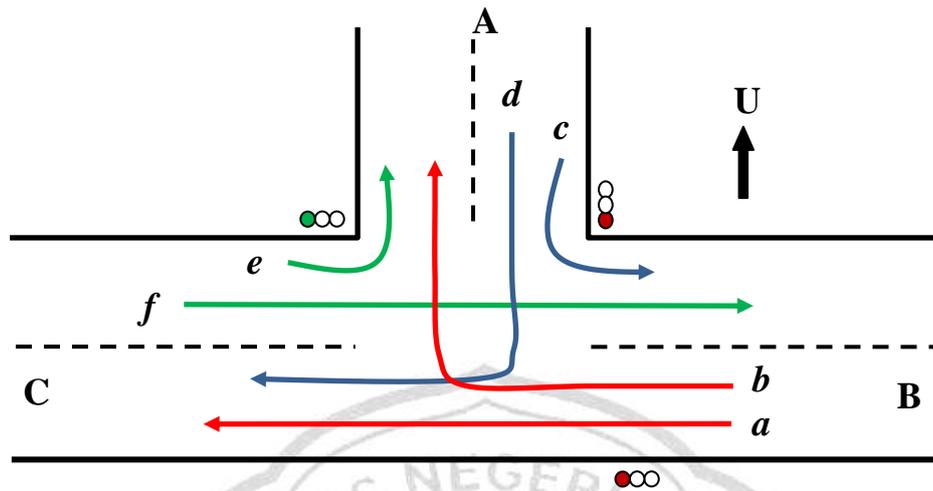
4.2.1 Simpang tiga jalan Majapahit-Supriyadi

Untuk lokasi simpang tiga jalan Majapahit-Supriyadi, dapat dilihat pada Gambar 4.4



Gambar 4.4 Foto lokasi simpang tiga Supriyadi

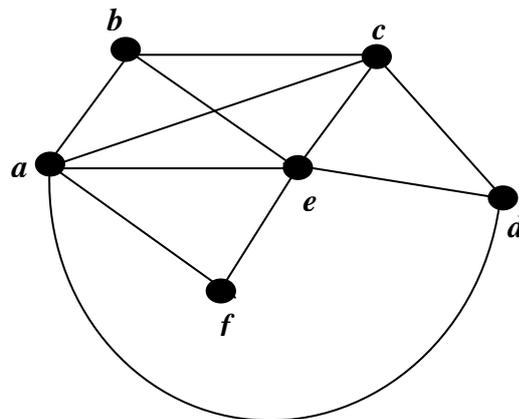
Dalam skema siklus waktu lampu lalu lintas simpang tiga jalan Majapahit-Supriyadi, dapat dilihat pada Gambar 4.5.



Gambar 4.5 Siklus waktu lampu di simpang tiga jalan Majapahit-Supriyadi

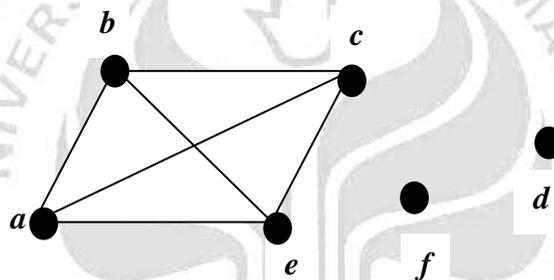
Dari skema siklus waktu lalu lintas di simpang tiga jalan Majapahit-Supriyadi, bisa dijelaskan untuk yang warna merah (jalur *a* dan *b*) adalah arus dari arah pedurungan menuju ke Gayamsari (*a*) dan ke Supriyadi (*b*), warna biru (jalur *c* dan *d*) adalah arus dari arah Supriyadi menuju ke Pedurungan (*c*) dan ke Gayamsari (*d*), dan warna hijau adalah arus dari arah Gayamsari menuju ke Supriyadi (*e*) dan ke pedurungan (*f*), untuk menyelesaikan masalah ini hanya dilakukan dengan asumsi yang sesuai dengan di lapangan.

Arus lalu lintas tertentu dapat disebut kompatibel jika kedua arus tidak akan menghasilkan apapun kecelakaan. Misalnya pada Gambar 4.5 arus *a* dan *b* adalah kompatibel, sedangkan *b* dengan *d*, *e* dan *f* tidak kompatibel. Pentahapan lampu harus sedemikian rupa sehingga ketika lampu hijau akan menyala selama dua arus, kedua arus tersebut harus kompatibel. Dari siklus lampu di persimpangan tersebut akan dibentuk graf kompatibel, seperti yang terlihat pada Gambar 4.6.



Gambar 4.6 Bentuk graf kompatibel pada simpang tiga jalan Majapahit-Supriyadi

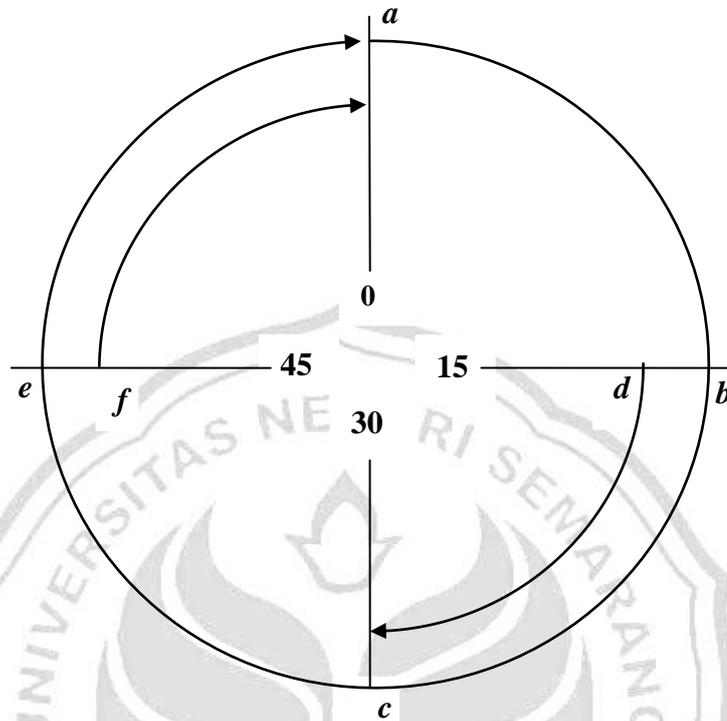
Selanjutnya sistem lalu lintas pada Gambar 4.6 akan dicari subgraf lengkap terbesarnya. Untuk titik a menunjukkan arus lalu lintas di a , titik b menunjukkan arus lalu lintas di b , titik c menunjukkan arus lalu lintas di c , dan begitu pula dengan titik d, e, f .



Gambar 4.7 Graf Lengkap H waktu lampu di simpang tiga jalan Majapahit-Supriyadi

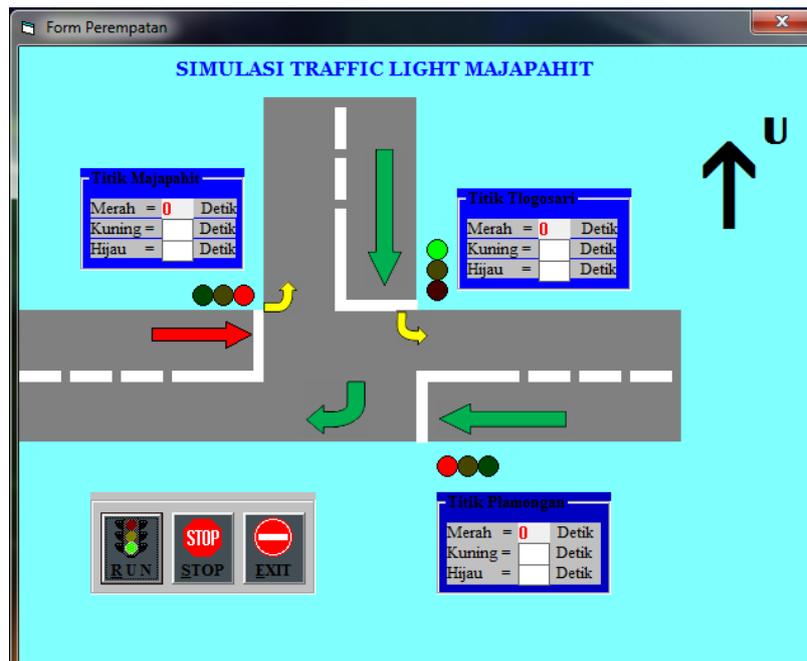
Lintasan-lintasan pada Gambar 4.7 dinyatakan sebagai titik pada graf kompatibel dengan mendapatkan subgraf lengkap, sehingga diperoleh himpunan titik $\{abce, d, \text{ dan } f\}$. Sisi-sisi pada graf lengkap H merupakan sisi graf lengkap yang diperoleh dari himpunan keterkaitan lintasan-lintasan pada Gambar 4.7

Dari Gambar 4.7 diperoleh 3 subgraf lengkap terbesar dengan setiap subgraf terbesar memuat 4 titik dan 2 titik. Dengan mengasumsikan lampu lalu lintas itu beroperasi selama 60 detik tiap putaran, maka salah satu penyelesaian adalah membiarkan setiap arus berjalan selama 60 detik: 3 subgraf lengkap = 20 detik. Untuk lebih jelasnya bisa dilihat pada diagram jam.



Gambar 4.8 Diagram jam pada asumsi sesuai di lapangan

Berarti bahwa dalam setiap periode 60 detik, arus dari tiap jalur ada yang berjalan selama 4×20 detik = 80 detik dan 2×20 detik = 40 detik. Karena terdapat 6 titik pada subgraf lengkap di atas, maka untuk 'waktu tunggu total' nya 6×20 detik = **120** detik. Untuk melihat bagaimana hasil tampilan program simulasi *traffic light* simpang tiga jalan Majapahit-Supriyadi, dapat dilihat pada Gambar 4.9.



Gambar 4.9 Tampilan program simulasi *traffic light* pertigaan jalan Majapahit-Supriyadi

Gambar 4.9 menampilkan program simulasi *traffic light*. Terlihat pada gambar, permodelan pertigaan Majapahit-Supriyadi beserta lampu lalu lintas pada setiap jalurnya. Berdasarkan pengamatan di lapangan gambar tersebut juga menggambarkan situasi ketika dari arah Supriyadi lampu menyala hijau, maka dari arah Simpang Lima berhenti kecuali dari arah Supriyadi untuk belok kiri jalan terus dan dari arah Simpang Lima juga jalan. Untuk dari arah Plamongan, boleh tetap jalan lurus karena dari arah Supriyadi jumlah kendaraan yang jalan tidak terlalu padat maka tidak akan terjadi kemacetan dan tidak terlalu berbahaya. Untuk mengetahui waktu tunggu atau waktu lampu merah dari masing-masing jalur didapatkan dari jumlah akumulasi waktu lampu kuning dan hijau dari 2 jalur.

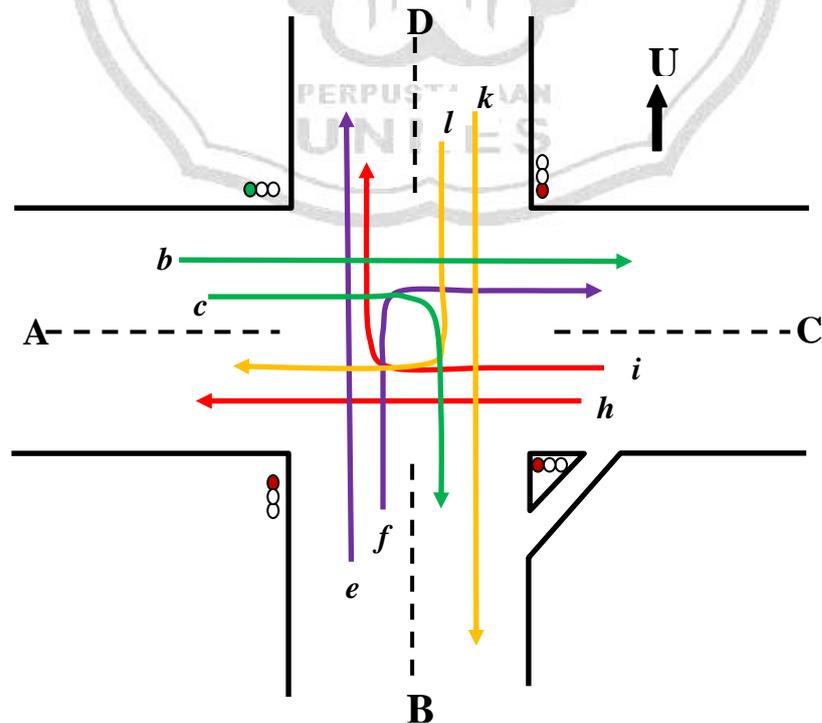
4.2.2 Simpang empat jalan Kaligarang-Kelud Raya-Bendungan



Gambar 4.10 Foto lokasi simpang empat jalan Kaligarang-Kelud Raya-Bendungan

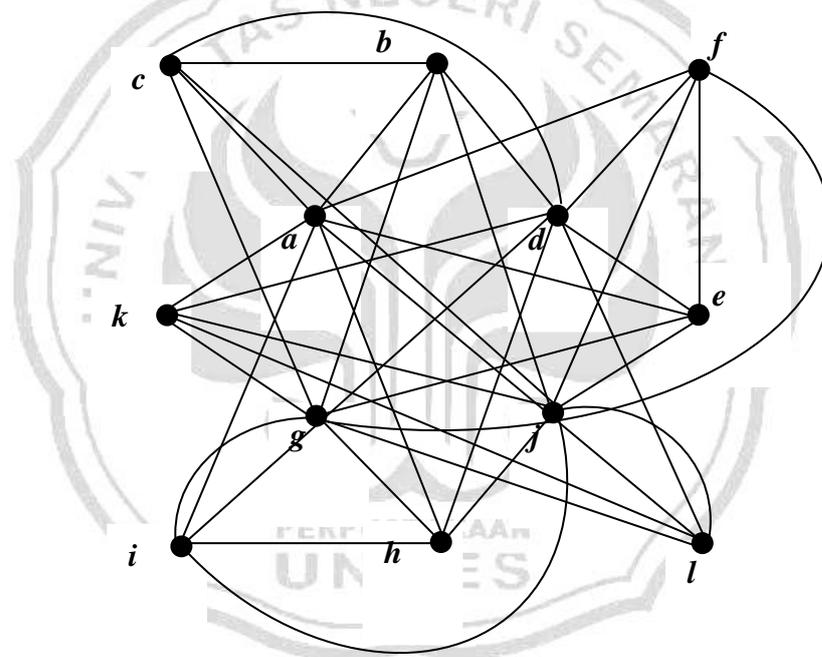
Pada penelitian di simpang empat ini digunakan beberapa asumsi, diantaranya:

4.2.2.1 Asumsi belok kiri **tidak mengikuti** lampu



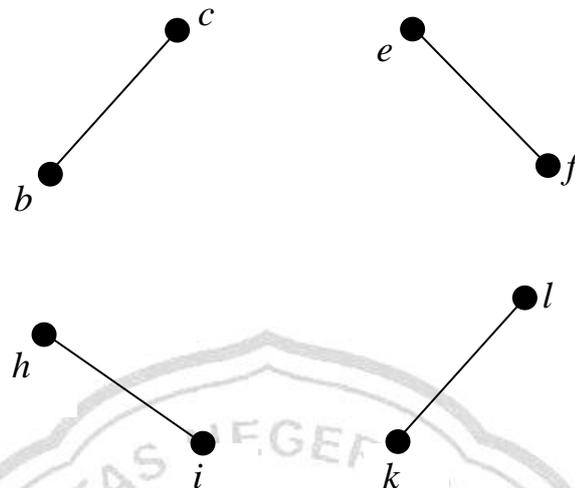
Gambar 4.11 Siklus waktu lampu saat belok kiri **tidak mengikuti** lampu

Arus lalu lintas tertentu dapat disebut kompatibel jika kedua arus tidak akan menghasilkan apapun kecelakaan. Misalnya pada Gambar 4.11, arus b dan c adalah kompatibel, sedangkan b dengan e , f , i , k , dan l tidak kompatibel. Pentahapan lampu harus sedemikian rupa sehingga ketika lampu hijau akan menyala selama dua arus, kedua arus tersebut harus kompatibel. Dari siklus lampu di persimpangan tersebut akan dibentuk graf kompatibel, seperti yang terlihat pada Gambar 4.12



Gambar 4.12 Bentuk graf kompatibel pada simpang empat jalan Kaligarang-Kelud-Bendungan

Selanjutnya sistem lalu lintas pada Gambar 4.12 akan dimodelkan dalam bentuk graf kompatibel dengan mendapatkan subgraf lengkap. Untuk titik a menunjukkan arus lalu lintas di a , titik b menunjukkan arus lalu lintas di b , titik c menunjukkan arus lalu lintas di c , dan begitu pula dengan titik d , e , f , g , h , i , j , k , dan l .

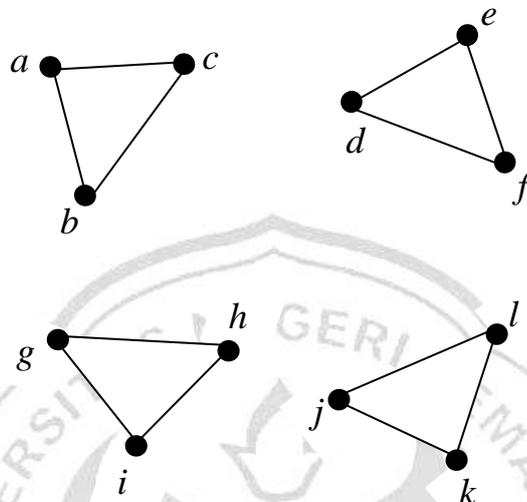


Gambar 4.13 Graf lengkap G saat belok kiri **tidak mengikuti**

Lintasan-lintasan pada Gambar 4.11 dinyatakan sebagai titik pada graf kompatibel dengan mendapatkan subgraf lengkap, sehingga diperoleh himpunan titik $\{bc, ef, hi, kl\}$. Sisi-sisi pada G merupakan sisi graf lengkap yang diperoleh dari himpunan keterkaitan lintasan-lintasan pada Gambar 4.13.

Dari Gambar 4.13 diperoleh 4 subgraf lengkap terbesar dengan setiap subgraf terbesar memuat 2 titik. Untuk itu dengan mengasumsikan lampu lalu lintas itu beroperasi selama 60 detik tiap putaran, maka salah satu penyelesaian adalah membiarkan setiap titik berjalan selama 60 detik : 4 subgraf lengkap = 15 detik tiap periode. Untuk lebih jelasnya bisa dilihat pada diagram jam.

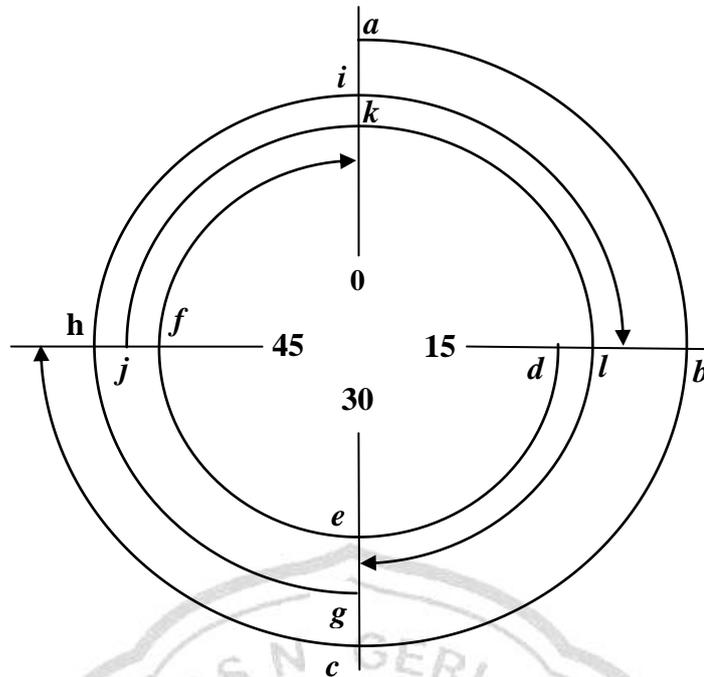
Selanjutnya sistem lalu lintas pada Gambar 4.15 akan dimodelkan dalam bentuk graf kompatibel dengan mendapatkan subgraf lengkap.



Gambar 4.16 Saat belok kiri **mengikuti** lampu

Lintasan-lintasan pada Gambar 4.15 dinyatakan sebagai titik pada graf kompatibel dengan mendapatkan subgraf lengkap, sehingga diperoleh himpunan titik $\{abc, def, ghi, jkl\}$. Sisi-sisi pada G merupakan sisi graf lengkap yang diperoleh dari himpunan keterkaitan lintasan-lintasan pada Gambar 4.16.

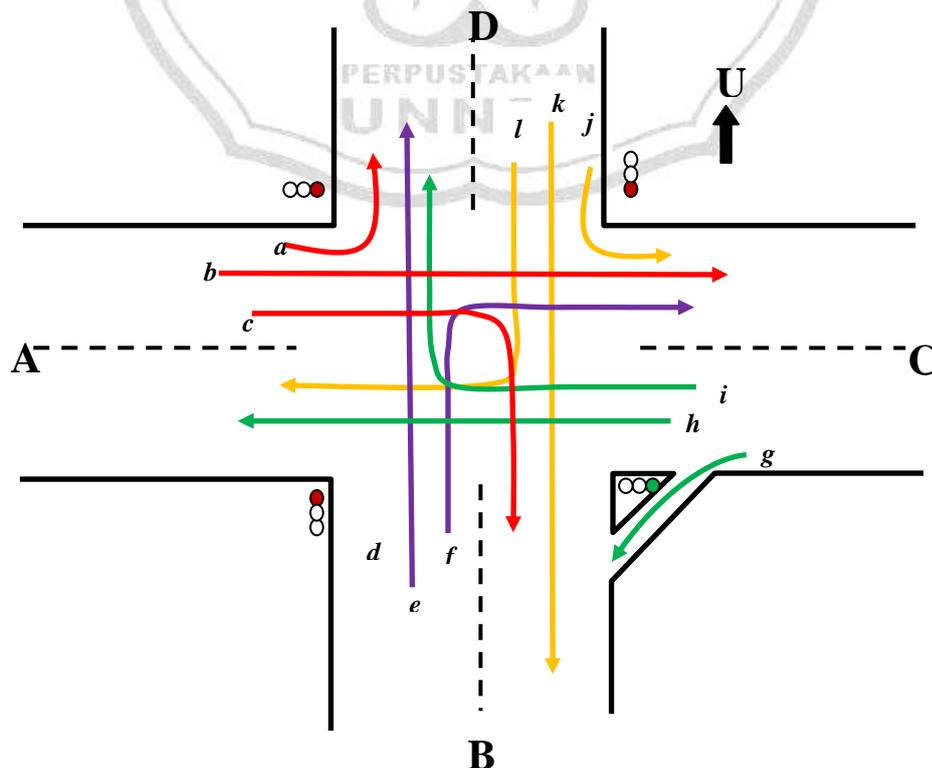
Dari Gambar 4.15 diperoleh 4 subgraf lengkap terbesar dengan setiap subgraf terbesar memuat 3 titik. Untuk itu dengan mengasumsikan lampu lalu lintas itu beroperasi selama 60 detik tiap putaran, maka salah satu penyelesaian adalah membiarkan setiap arus berjalan selama 60 detik : 4 subgraf lengkap = 15 detik. Untuk lebih jelasnya bisa dilihat pada diagram jam.



Gambar 4.17 Diagram jam pada asumsi belok kiri mengikuti lampu

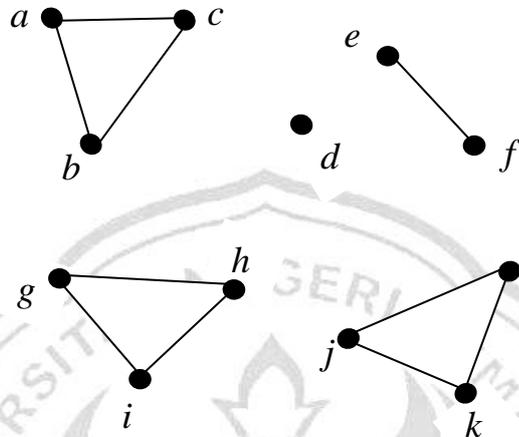
Karena dalam 1 subgraf lengkap terdiri dari 3 titik maka dalam setiap periode 60 detik, arus dari tiap jalur berjalan selama 3×15 detik = 45 detik. Karena terdapat 12 titik pada subgraf lengkap diatas, maka untuk 'waktu tunggu total' nya 12×15 detik = **180** detik.

4.2.2.3 Asumsi titik *d* berhenti jika titik *h* jalan



Gambar 4.18 Siklus waktu lampu saat titik *d* berhenti jika titik *h* jalan

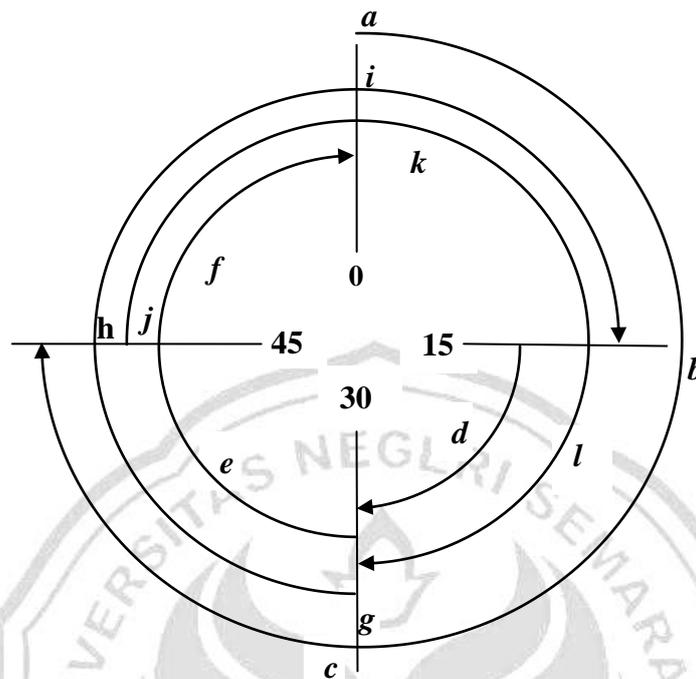
Pada saat titik d berhenti dan titik h jalan, selanjutnya akan langsung dimodelkan dalam bentuk graf kompatibel dengan mendapatkan subgraf lengkap yang dapat dilihat pada Gambar 4.18



Gambar 4.19 Saat titik d berhenti jika titik h jalan

Lintasan-lintasan pada Gambar 4.15 dinyatakan sebagai titik pada graf kompatibel dengan mendapatkan subgraf lengkap, sehingga diperoleh himpunan titik $\{abc, d, ef, ghi, jkl\}$. Sisi-sisi pada G merupakan sisi graf lengkap yang diperoleh dari himpunan keterkaitan lintasan-lintasan pada Gambar 4.18.

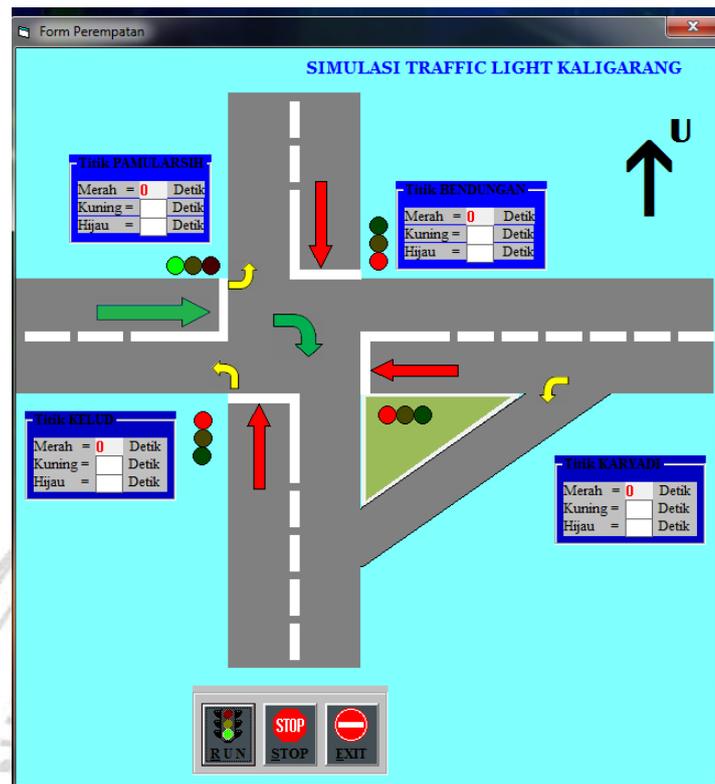
Dari Gambar 4.18 diperoleh 5 subgraf lengkap terbesar dengan setiap subgraf lengkap terbesar memuat 3 titik, 2 titik, dan 1 titik. Untuk itu dengan mengasumsikan lampu lalu lintas itu beroperasi selama 60 detik tiap putaran, maka salah satu penyelesaian adalah membiarkan setiap arus berjalan selama 60 detik: 5 subgraf lengkap = 12 detik.



Gambar 4.20 Diagram jam pada asumsi titik d berhenti jika titik h jalan

Karena dalam 1 subgraf lengkap terdiri dari 3 titik, 2 titik, dan 1 titik, maka dalam setiap periode 60 detik arus dari tiap jalur ada yang berjalan selama 3×12 detik = 36 detik, 2×12 detik = 24 detik, dan 1×12 detik = 12 detik. Karena terdapat 12 titik pada subgraf lengkap di atas, maka untuk 'waktu tunggu total' nya 12×12 detik = **144** detik.

Setelah dilakukan dengan menggunakan tiga asumsi tersebut, didapat hasil yang optimal yaitu pada asumsi belok kiri tidak mengikuti lampu dengan hasil perhitungan waktu tunggu total **120** detik. Untuk melihat bagaimana hasil tampilan program simulasi *traffic light* simpang empat jalan Kaligarang-Kelud Raya-Bendungan, dapat dilihat padan Gambar 4.16.



Gambar 4.21 Tampilan program simulasi traffic light perempatan jalan
Kaligarang-Kelud Raya-Bendungan

Gambar 4.21 menampilkan *form* utama dari program simulasi *traffic light*. Terlihat pada gambar, permodelan perempatan Kaligarang beserta lampu lalu lintas pada setiap jalurnya. Berdasarkan pengamatan di lapangan gambar tersebut juga menggambarkan situasi ketika dari arah Pamularsi lampu menyala hijau, maka dari arah yang lain berhenti kecuali untuk belok kiri jalan terus dari arah Karyadi dan Kelud Raya. Untuk belok kiri dari arah Bendungan berhenti dikarenakan jika belok kiri jalan terus maka berbahaya ketika dari arah Pamularsi jalan dan bisa juga menyebabkan kemacetan. Untuk mengetahui waktu tunggu atau waktu lampu merah dari masing-masing jalur didapatkan dari jumlah akumulasi waktu lampu kuning dan hijau dari 3 jalur lainnya.

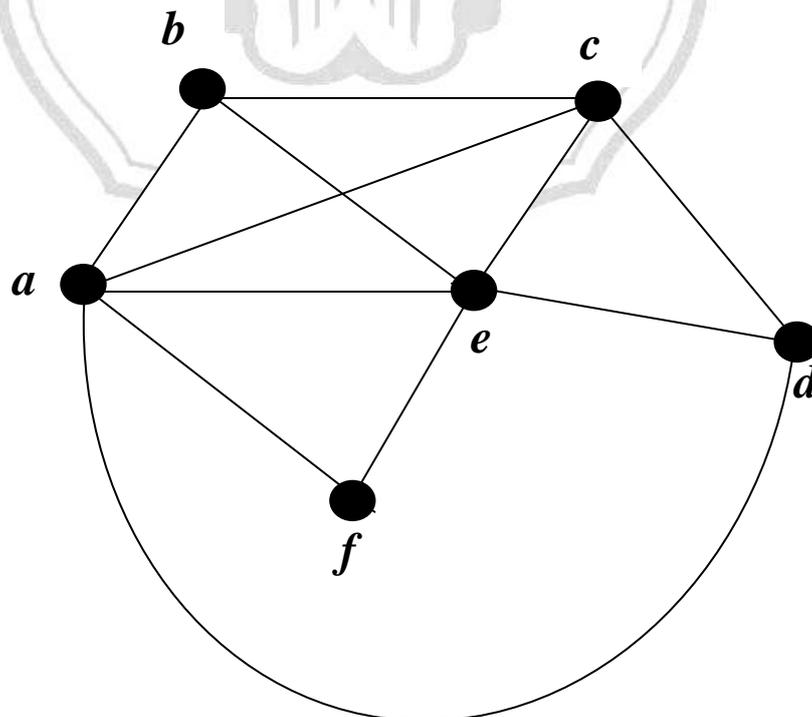
BAB 5

PENUTUP

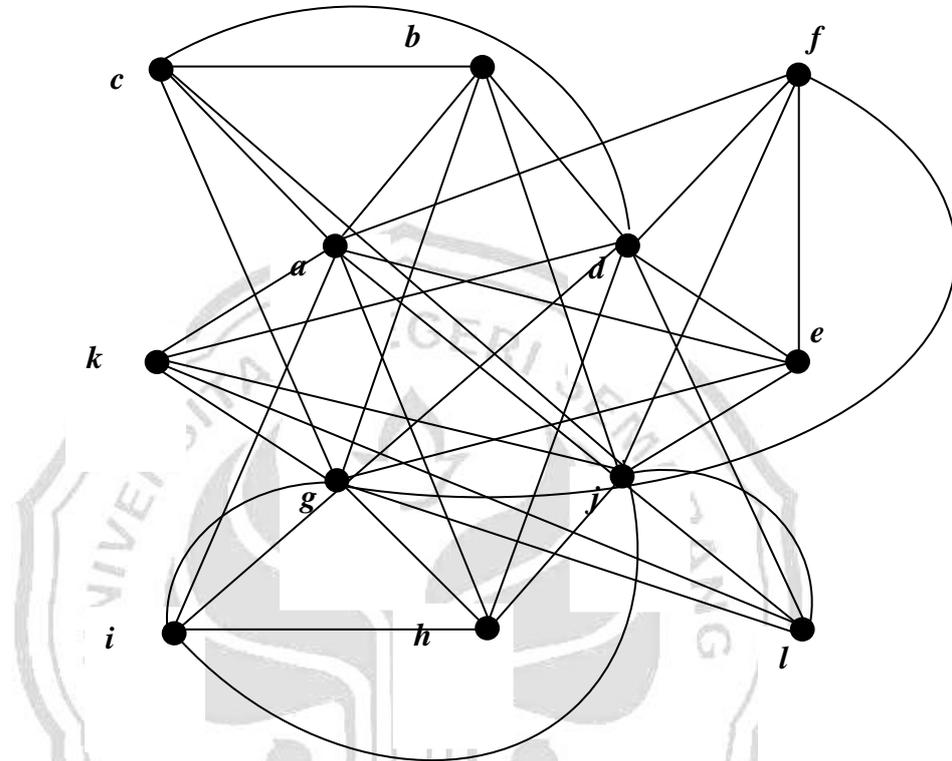
5.1 Simpulan

Berdasarkan pembahasan yang telah diuraikan pada bab-bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut.

1. Berdasarkan hasil penelitian didapat bentuk graf kompatibel dari hasil pemodelan arus lalu lintas di persimpangan jalan simpang tiga jalan Majapahit-Supriyadi dan simpang empat jalan Kaligarang-Kelud Raya-Bendungan sebagai berikut.
 - a. Bentuk graf kompatibel pada simpang tiga jalan Majapahit-Supriyadi.



- b. Bentuk graf kompatibel pada simpang empat jalan Kaligarang-Kelud Raya-Bendungan.



2. Hasil perhitungan waktu tunggu total optimal dengan menggunakan graf kompatibel dari masing-masing persimpangan yaitu:
- Untuk lokasi simpang tiga jalan Majapahit-Supriyadi diperoleh hasil perhitungan dengan asumsi sesuai di lapangan waktu tunggu totalnya adalah 120 detik.
 - Untuk lokasi simpang empat jalan Kaligarang-Kelud Raya-Bendungan diperoleh hasil perhitungan waktu tunggu totalnya dalam 3 asumsi, yaitu 120 detik, 180 detik, dan 144 detik.

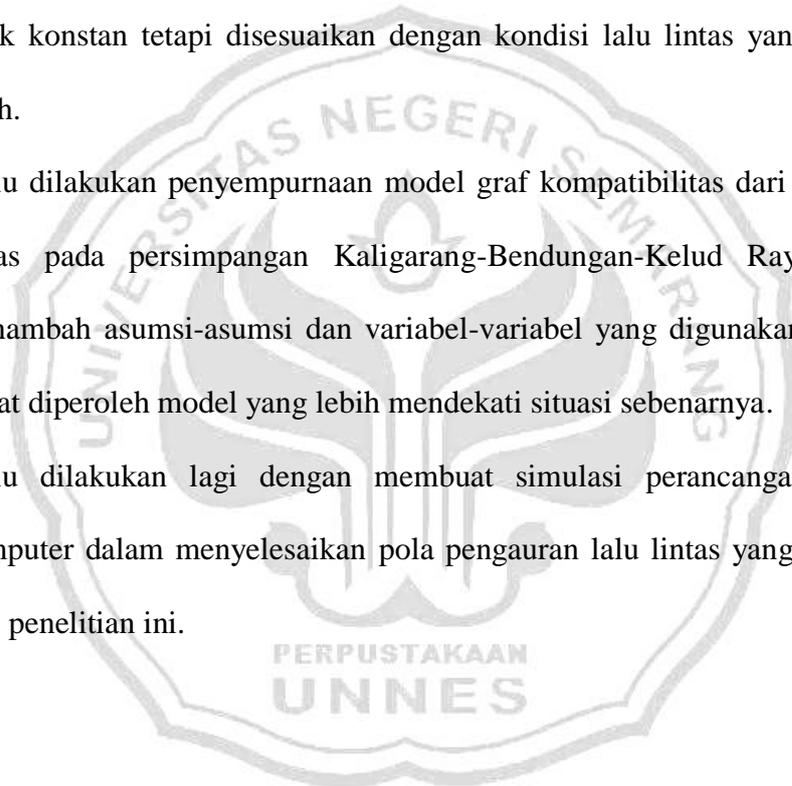
3. Perhitungan hasil waktu tunggu total optimal berdasarkan graf kompatibel dengan pengaturan yang sudah diterapkan dari masing-masing persimpangan yaitu:
 - a. Untuk lokasi simpang tiga jalan Majapahit-Supriyadi diperoleh hasil perhitungan dengan asumsi sesuai di lapangan waktu tunggu total optimal berdasarkan graf kompatibel adalah 120 detik sedangkan pengaturan yang sudah diterapkan 277 detik.
 - b. Untuk lokasi simpang empat jalan Kaligarang-Kelud Raya-Bendungan diperoleh hasil perhitungan waktu tunggu total optimal berdasarkan graf kompatibel dalam 3 asumsi yaitu belok kiri tidak mengikuti lampu, belok kiri mengikuti lampu, dan ketika titik d berhenti saat titik h jalan berbeda jauh dengan yang sudah diterapkan di lapangan. Hasil perhitungan waktu tunggu total optimal yang diperoleh dari lapangan adalah 389 detik sedangkan dengan menggunakan graf kompatibel menghasilkan 120 detik dari hasil salah satu asumsi yang paling minimal.

Hasil perhitungan waktu tunggu total optimal berdasarkan graf kompatibilitas dengan pengaturan yang sudah diterapkan di setiap persimpangan terlihat sangat berbeda, ini disebabkan karena dalam penyelesaian dengan menggunakan graf kompatibilitas variabel yang diamati hanya waktu tidak melibatkan variabel lainnya seperti jumlah kendaraan, pejalan kaki, dan lebar jalan.

5.2 Saran

Pada penelitian selanjutnya, dapat dilakukan hal-hal berikut.

1. Model graf kompatibilitas/kompatibel yang dihasilkan dari penelitian ini yang kami rekomendasikan untuk diterapkan adalah model siklus lampu lalu lintas yang berbeda, karena model ini menghasilkan pola pengaturan yang tidak konstan tetapi disesuaikan dengan kondisi lalu lintas yang berubah-ubah.
2. Perlu dilakukan penyempurnaan model graf kompatibilitas dari sistem lalu lintas pada persimpangan Kaligarang-Bendungan-Kelud Raya, dengan menambah asumsi-asumsi dan variabel-variabel yang digunakan, sehingga dapat diperoleh model yang lebih mendekati situasi sebenarnya.
3. Perlu dilakukan lagi dengan membuat simulasi perancangan program komputer dalam menyelesaikan pola pengaturan lalu lintas yang dihasilkan dari penelitian ini.



DAFTAR PUSTAKA

- Baruah, A.K, & Baruah, Niky. 2012. *Signal Group of Compatible Graph in Traffic Control Problems. Int. J. Advanced Networking and Applications*. Vol:04 Issue:01 Pages: 1473-1480 ISSN: 0975-0290
- Budayasa, I.K. 2007. *Teori graf dan Aplikaisnya*. Surabaya :Unesa University Press.
- Hariyanto, J. 2004. *Sistem Pengendalian Lalu Lintas Pada Pertemuan Jalan Sebidang*. Sumatera Utara: Jurnal Jurusan Teknik Sipil Universitas Sumatera Utara
- Hosseini, S.M, & Orooji, H. 2009. *Phasing of Traffic Light at a Road Junction*. Applied Mathematical Science. Vol.3. No.30:1487-1492.
- Johnsonbaugh, R. 1986. *Discrete Mathematics Revised Edition*. New York: Macmillian Publishing Company.
- Munir, R. 2005. *Matematika Diskrit*. Bandung: Informatika.
- Nugroho, A. D. 2008. *Analisis Penerapan Belok Kiri Langsung Terhadap Tundaan Lalu Lintas Pada Pendekat Persimpangan Bersinyal*. Tesis. Semarang: Program Magister Teknik Sipil Universitas Diponegoro.
- Rifki. 2012. Jam Pulang, Lalin di Jalan Kaligarang. Semarang: *Suara Merdeka.Com*. Online. Tersedia di <http://berita-news-terkini.blogspot.com/2012/04/jam-pulang-lalin-di-jalan-kaligarang.html> [diakses 10 Mei 2012].
- Siang, J. J. 2004. *Matematika Diskrit dan Aplikasinya pada Ilmu Komputer*. Yogyakarta: Andi Offset.
- Sutarno, H. 2003. *Common Text Book Matematika Diskrit*. Jakarta: Universitas Pendidikan Indonesia
- Wayan, S. I. 2011. *Analisis Kebutuhan Penanganan Simpang Empat Gerung Di Lombok Barat*. Mataram: Jurnal Ilmiah Teknik Sipil Teknik Universitas Mataram.
- Wilson, R. J, & Watkins, J. J. 1976. *Graphs An Introductory Approach*. New York: Published simultaneously in Canada.

LAMPIRAN



Lampiran 1

1. Pengambilan data I (Senin, 15 Oktober 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Supriyadi (A)	83	3	40
Jalan Majapahit dari arah timur (B)	106	3	17
Jalan Majapahit dari arah barat (C)	88	3	35

2. Pengambilan data II (Selasa, 16 Oktober 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Supriyadi (A)	83	3	40
Jalan Majapahit dari arah timur (B)	106	3	17
Jalan Majapahit dari arah barat (C)	88	3	35

3. Pengambilan data III (Rabu, 17 Oktober 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Supriyadi (A)	83	3	40
Jalan Majapahit dari arah timur (B)	106	3	17
Jalan Majapahit dari arah barat (C)	88	3	35

Lampiran 2

1. Pengambilan data I (Kamis, 9 Agustus 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Kaligarang arah Pamularsih (A)	83	3	40
Jalan Kelud Raya (B)	106	3	17
Jalan Kaligarang arah Karyadi (C)	88	3	35
Jalan Bendungan (D)	112	3	11

2. Pengambilan data II (Kamis, 9 Agustus 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Kaligarang arah Pamularsih (A)	83	3	40
Jalan Kelud Raya (B)	106	3	17
Jalan Kaligarang arah Karyadi (C)	88	3	35
Jalan Bendungan (D)	112	3	11

3. Pengambilan data III (Jumat, 10 Agustus 2012)

Pengamatan Awal	Merah (detik)	Kuning (detik)	Hijau (detik)
Jalan Kaligarang arah Pamularsih (A)	83	3	40
Jalan Kelud Raya (B)	106	3	17
Jalan Kaligarang arah Karyadi (C)	88	3	35
Jalan Bendungan (D)	112	3	11

Lampiran 3

1. Foto Lokasi Persimpangan Supriyadi



2. Foto Lokasi Persimpangan Kaligarang



Lampiran 4

Form dan code program Visual Basic 6.0 yaitu sebagai berikut:

- Langkah-langkah membuat program Visual Basic 6.0 yaitu:
 1. Desain *form* pertama untuk tampilan utama seperti gambar dibawah ini:

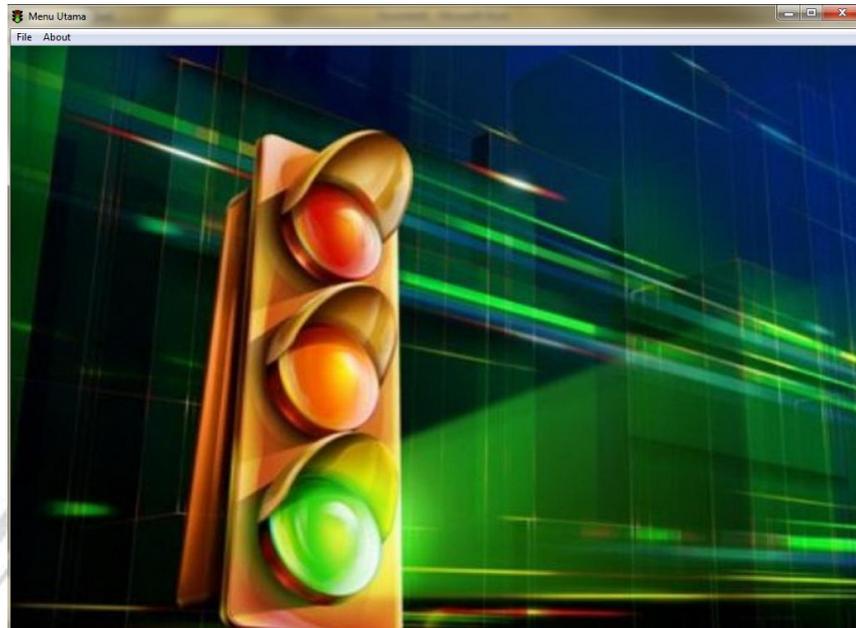


2. Tulis *Code* Visual Basic pada command seperti *code* dibawah ini:

```

Private Sub Form_Unload(Cancel As Integer)
    End
End Sub
Private Sub mAbout_Click()
    frmAbout.Show
End Sub
Private Sub mKeluar_Click()
    MsgBox "TERIMAKASIH TELAH MENCOBA PROGRAM
SIMULASI INI"
    Unload Me
    End
End Sub
Private Sub mP3_Click()
    frmPPP.Show
End Sub
Private Sub mP4_Click()
    frmP4.Show
End Sub
  
```

3. *Form* utama untuk simulasi terlihat pada gambar dibawah ini:



4. Tulis *Code* Visual Basic pada command seperti *code* dibawah ini:

```
Private Sub Form_Unload(Cancel As Integer)
    End
End Sub
```

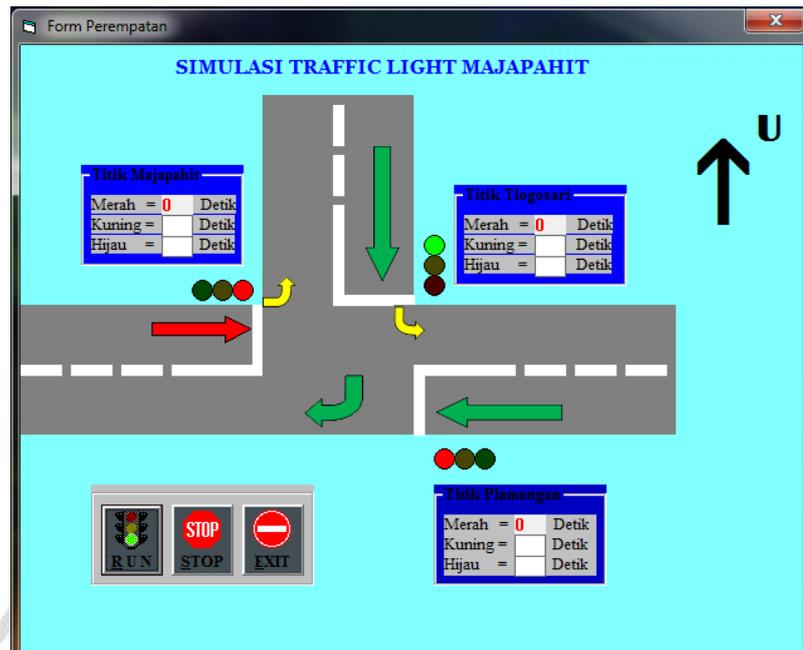
```
Private Sub mAbout_Click()
    frmAbout.Show
End Sub
```

```
Private Sub mKeluar_Click()
    MsgBox "TERIMAKASIH TELAH MENCOBA PROGRAM
SIMULASI INI"
    Unload Me
    End
End Sub
```

```
Private Sub mP3_Click()
    frmPPP.Show
End Sub
```

```
Private Sub mP4_Click()
    frmP4.Show
End Sub
```

5. Tampilan hasil simulasi untuk simpang tiga sebagai berikut:



```

Private Declare Function Inp Lib "inpout32.dll" Alias "Inp32"
(ByVal PortAddress As Integer) As Integer
Private Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As
Integer)
Option Explicit
Private m_index, m_idx, m_ind As Integer
Private m_NumImages, m_NumImg, m_NumberImg As Integer
Dim pantul As Integer
Dim idxLampuHijau As Integer
Private Sub LampuMati()
    Dim ctl As Control
    Out &H378, 256
    Out &H37A, 11
    For Each ctl In Me.Controls
        If TypeOf ctl Is Shape Then
            If ctl.Name = "shpLampuMerah" Then ctl.BackColor =
RGB(70, 0, 0)
            If ctl.Name = "shpLampuKuning" Then ctl.BackColor =
RGB(70, 70, 0)
            If ctl.Name = "shpLampuHijau" Then ctl.BackColor =
RGB(0, 70, 0)
            End If
        End If
    Next
End Sub
Private Sub LampuMerahNyala(Index As Integer)
    Select Case Index
        Case 0
    
```

```

        Out &H378, 1 + Val(Inp(&H378))
    Case 1
        Out &H378, 2 + Val(Inp(&H378))
    Case 2
        Out &H378, 4 + Val(Inp(&H378))
    End Select
    shpLampuMerah(Index).BackColor = vbRed
End Sub
Private Sub LampuMerahMati(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, Val(Inp(&H378)) - 1
    Case 1
        Out &H378, Val(Inp(&H378)) - 2
    Case 2
        Out &H378, Val(Inp(&H378)) - 4
    End Select
    shpLampuMerah(Index).BackColor = RGB(50, 0, 0)
End Sub
Private Sub LampuKuningNyala(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, 16 + Val(Inp(&H378))
    Case 1
        Out &H378, 32 + Val(Inp(&H378))
    Case 2
        Out &H378, 64 + Val(Inp(&H378))
    End Select
    shpLampuKuning(Index).BackColor = vbYellow
End Sub
Private Sub LampuKuningMati(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, Val(Inp(&H378)) - 16
    Case 1
        Out &H378, Val(Inp(&H378)) - 32
    Case 2
        Out &H378, Val(Inp(&H378)) - 64
    End Select
    shpLampuKuning(Index).BackColor = RGB(50, 50, 0)
End Sub
Private Sub LampuHijauNyala(Index As Integer)
    ResetArahAnim
    Select Case Index
    Case 0

```

```

        Out &H37A, 3
        idxLampuHijau = 0
    Case 1
        Out &H37A, 15
        idxLampuHijau = 1
    Case 2
        Out &H37A, 9
        idxLampuHijau = 2
    End Select
    shpLampuHijau(Index).BackColor = vbGreen
    tmrArah.Enabled = True
End Sub
Private Sub LampuHijauMati(Index As Integer)
    tmrArah.Enabled = False
    ResetArahAnim
    Select Case Index
    Case 0
        Out &H37A, 11
    Case 1
        Out &H37A, 11
    Case 2
        Out &H37A, 11
    End Select
    shpLampuHijau(Index).BackColor = RGB(0, 50, 0)
End Sub
Private Sub cmdExit_Click()
    Unload Me
    frmUtama.Show
End Sub
Private Sub cmdRun_Click()
    Dim intNum As Integer
    LampuMati
    tmrLampu.Interval = 1
    tmrLampu.Enabled = True
End Sub
Private Sub cmdStop_Click()
    tmrArah.Enabled = False
    LampuMati
    tmrLampu.Enabled = False
End Sub
Private Sub ResetArahAnim()
    img_mbl(0).Visible = False
    img_mbl(1).Visible = False
    img_mbl(2).Visible = False

```

```

img_mbl_ka(1).Visible = False
img_mbl_ka(2).Visible = False
'With img_mbl(0)
    '.Move 0 - .Width, (picArah(0).ScaleHeight - .Height) / 2
'End With
'With img_mbl(1)
    '.Move (picArah(1).ScaleWidth - .Width) / 2, 0 - .Height
'End With
'With img_mbl(2)
    '.Move picArah(2).ScaleWidth + .Width,
(picArah(2).ScaleHeight - .Height) / 2
'End With
'With img_mbl(3)
    '.Move (picArah(3).ScaleWidth - .Width) / 2,
picArah(3).ScaleHeight + .Height
'End With
End Sub
Private Sub Form_Load()
    ResetArahAnim
    LampuMati
    pantul = 100
    m_NumImages = ImageList4.ListImages.Count
    m_NumImg = ImageList3.ListImages.Count
    m_NumberImg = ImageList1.ListImages.Count
    m_index = 1
    m_idx = 1
    m_ind = 1
End Sub
Private Sub Form_Unload(Cancel As Integer)
    LampuMati
End Sub
Private Sub lblLampuHijau_Click(Index As Integer)
    LampuMati
    LampuHijauNyala Index
End Sub
Private Sub lblLampuHijau_DblClick(Index As Integer)
    LampuHijauMati Index
End Sub
Private Sub lblLampuKuning_Click(Index As Integer)
    LampuMati
    LampuKuningNyala Index
End Sub
Private Sub lblLampuKuning_DblClick(Index As Integer)
    LampuKuningMati Index
End Sub

```

```

Private Sub lblLampuMerah_Click(Index As Integer)
    LampuMati
    LampuMerahNyala Index
End Sub
Private Sub lblLampuMerah_DblClick(Index As Integer)
    LampuMerahMati Index
End Sub
Private Sub tmrAnim_Timer()
    With lblJudul
        .Left = .Left + pantul
        If .Left < 0 Then pantul = 100
        If .Left > Me.ScaleWidth - .Width Then pantul = -100
    End With
End Sub
Private Sub tmrArah_Timer()
    m_idx = (m_idx Mod m_NumImg) + 1
    m_ind = (m_ind Mod m_NumberImg) + 1
    With img_mbl(idxLampuHijau)
        Select Case idxLampuHijau
        Case 0
            '.Left = .Left + 20
            'If .Left > picArah(idxLampuHijau).ScaleWidth Then .Left
            = 0 - .Width
            img_mbl(2).Picture =
            ImageList1.ListImages(m_idx).Picture
            img_mbl(0).Picture =
            ImageList3.ListImages(m_idx).Picture
            img_mbl(0).Visible = True
            img_mbl(1).Visible = False
            img_mbl(2).Visible = True
            img_mati(0).Visible = False
            img_mati(1).Visible = True
            img_mati(2).Visible = False
            img_mbl_ka(1).Visible = False
            img_mbl_ka(2).Visible = False

        Case 1
            '.Top = .Top + 20
            'If .Top > picArah(idxLampuHijau).ScaleHeight Then
            .Top = 0 - .Height
            img_mbl(1).Picture =
            ImageList5.ListImages(m_idx).Picture
            img_mbl(2).Picture =
            ImageList1.ListImages(m_idx).Picture

```

```

        img_mbl_ka(1).Picture =
        ImageList6.ListImages(m_ind).Picture
        img_mbl(0).Visible = False
        img_mbl(1).Visible = True
        img_mbl(2).Visible = True
        img_mati(0).Visible = True
        img_mati(1).Visible = False
        img_mati(2).Visible = False
        img_mbl_ka(1).Visible = True
        img_mbl_ka(2).Visible = False
    Case 2
        '.Left = .Left - 20
        'If .Left < 0 - .Width Then .Left =
        picArah(idxLampuHijau).ScaleWidth
        img_mbl(2).Picture =
        ImageList1.ListImages(m_idx).Picture
        img_mbl_ka(2).Picture =
        ImageList2.ListImages(m_ind).Picture
        img_mbl(0).Visible = False
        img_mbl(1).Visible = False
        img_mbl(2).Visible = True
        img_mati(0).Visible = True
        img_mati(1).Visible = True
        img_mati(2).Visible = False
        img_mbl_ka(1).Visible = False
        img_mbl_ka(2).Visible = True
    End Select
End With
End Sub
Private Sub tmrKedip_Timer()
    m_index = (m_index Mod m_NumImages) + 1
    img_mbl_ki(0).Picture =
    ImageList4.ListImages(m_index).Picture
    img_mbl_ki(1).Picture =
    ImageList7.ListImages(m_index).Picture
End Sub
Private Sub tmrLampu_Timer()
    Static Index As Integer
    Static intLampu As Integer
    Dim intNum As Integer
    Select Case intLampu
    Case 0 'Hijau
        LampuMati
        tmrLampu.Interval = Val(txtHijau(Index).Text) * 1000
        LampuHijauNyala Index

```

```

For intNum = 0 To 2
  If intNum <> Index Then LampuMerahNyala intNum
  If intNum <> Index Then
    merah (intNum)
  Else
    LabelMerah(Index).Caption = 0
  End If
Next
intLampu = 1
Case 1 'Kuning
LampuMati
tmrLampu.Interval = Val(txtKuning(Index).Text) * 1000
LampuKuningNyala Index
For intNum = 0 To 2
  If intNum <> Index Then LampuMerahNyala intNum
Next
intLampu = 0
Index = Index + 1
If Index = 3 Then Index = 0
End Select
End Sub
Private Sub merah(idxEmp As Integer)
  Select Case idxEmp
  Case 0
    LabelMerah(idxEmp).Caption = Val(txtKuning(1).Text) +
Val(txtHijau(1).Text) + _
Val(txtKuning(2).Text) +
Val(txtHijau(2).Text)
  Case 1
    LabelMerah(idxEmp).Caption = Val(txtKuning(0).Text) +
Val(txtHijau(0).Text) + _
Val(txtKuning(2).Text) +
Val(txtHijau(2).Text)
  Case 2
    LabelMerah(idxEmp).Caption = Val(txtKuning(0).Text) +
Val(txtHijau(0).Text) + _
Val(txtKuning(1).Text) +
Val(txtHijau(1).Text)
  End Select
End Sub
Private Sub txtHijau_Change(Index As Integer)
  With txtHijau(Index)
    If IsNumeric(.Text) = False Then SendKeys vbBack: Exit
  End With
End Sub

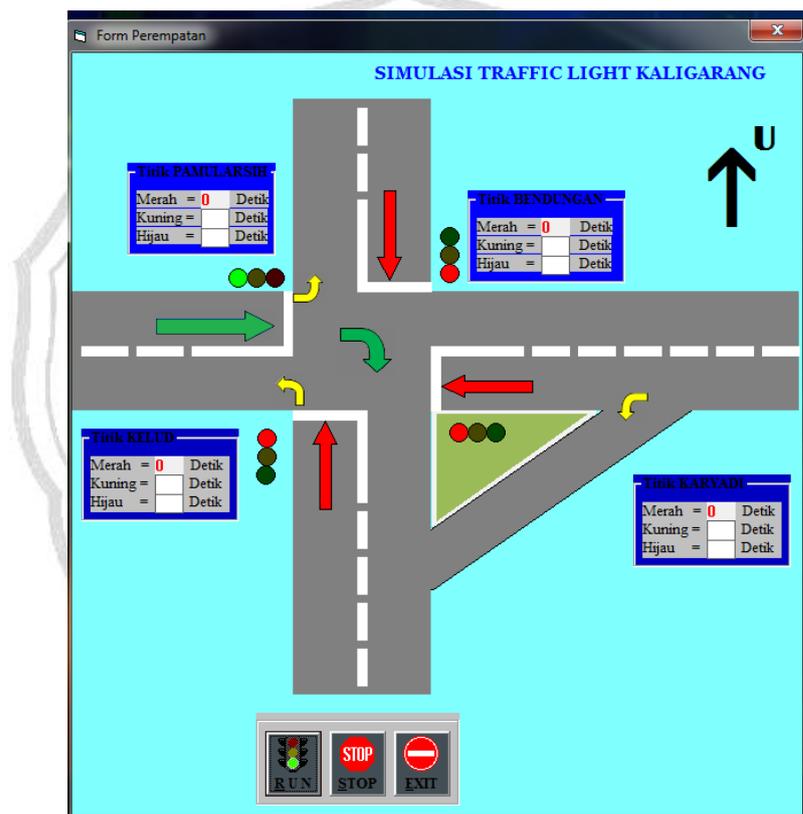
```

```

End Sub
Private Sub txtKuning_Change(Index As Integer)
    With txtKuning(Index)
        If IsNumeric(.Text) = False Then SendKeys vbBack: Exit
    End With
End Sub

```

6. Tampilan hasil simulasi untuk simpang tiga sebagai berikut:



```

Private Declare Function Inp Lib "inpout32.dll" Alias "Inp32"
    (ByVal PortAddress As Integer) As Integer
Private Declare Sub Out Lib "inpout32.dll" _
    Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As
    Integer)
Option Explicit
Private m_index, m_idx, m_ind As Integer
Private m_NumImages, m_NumImg, m_NumberImg As Integer
Dim pantul As Integer
Dim idxLampuHijau As Integer
Private Sub LampuMati()
    Dim ctl As Control
    Out &H378, 256

```

```

Out &H37A, 11
For Each ctl In Me.Controls
    If TypeOf ctl Is Shape Then
        If ctl.Name = "shpLampuMerah" Then ctl.BackColor =
RGB(70, 0, 0)
        If ctl.Name = "shpLampuKuning" Then ctl.BackColor =
RGB(70, 70, 0)
        If ctl.Name = "shpLampuHijau" Then ctl.BackColor =
RGB(0, 70, 0)
    End If
Next
End Sub
Private Sub LampuMerahNyala(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, 1 + Val(Inp(&H378))
    Case 1
        Out &H378, 2 + Val(Inp(&H378))
    Case 2
        Out &H378, 4 + Val(Inp(&H378))
    End Select
    shpLampuMerah(Index).BackColor = vbRed
End Sub
Private Sub LampuMerahMati(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, Val(Inp(&H378)) - 1
    Case 1
        Out &H378, Val(Inp(&H378)) - 2
    Case 2
        Out &H378, Val(Inp(&H378)) - 4
    End Select
    shpLampuMerah(Index).BackColor = RGB(50, 0, 0)
End Sub
Private Sub LampuKuningNyala(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, 16 + Val(Inp(&H378))
    Case 1
        Out &H378, 32 + Val(Inp(&H378))
    Case 2
        Out &H378, 64 + Val(Inp(&H378))
    End Select
    shpLampuKuning(Index).BackColor = vbYellow
End Sub

```

```

Private Sub LampuKuningMati(Index As Integer)
    Select Case Index
    Case 0
        Out &H378, Val(Inp(&H378)) - 16
    Case 1
        Out &H378, Val(Inp(&H378)) - 32
    Case 2
        Out &H378, Val(Inp(&H378)) - 64
    End Select
    shpLampuKuning(Index).BackColor = RGB(50, 50, 0)
End Sub
Private Sub LampuHijauNyala(Index As Integer)
    ResetArahAnim
    Select Case Index
    Case 0
        Out &H37A, 3
        idxLampuHijau = 0
    Case 1
        Out &H37A, 15
        idxLampuHijau = 1
    Case 2
        Out &H37A, 9
        idxLampuHijau = 2
    End Select
    shpLampuHijau(Index).BackColor = vbGreen
    tmrArah.Enabled = True
End Sub
Private Sub LampuHijauMati(Index As Integer)
    tmrArah.Enabled = False
    ResetArahAnim
    Select Case Index
    Case 0
        Out &H37A, 11
    Case 1
        Out &H37A, 11
    Case 2
        Out &H37A, 11
    End Select
    shpLampuHijau(Index).BackColor = RGB(0, 50, 0)
End Sub
Private Sub cmdExit_Click()
    Unload Me
    frmUtama.Show
End Sub
Private Sub cmdRun_Click()

```

```

Dim intNum As Integer
LampuMati
tmrLampu.Interval = 1
tmrLampu.Enabled = True
End Sub
Private Sub cmdStop_Click()
tmrArah.Enabled = False
LampuMati
tmrLampu.Enabled = False
End Sub
Private Sub ResetArahAnim()
img_mbl(0).Visible = False
img_mbl(1).Visible = False
img_mbl(2).Visible = False
img_mbl_ka(1).Visible = False
img_mbl_ka(2).Visible = False
'With img_mbl(0)
.Move 0 - .Width, (picArah(0).ScaleHeight - .Height) / 2
'End With
'With img_mbl(1)
.Move (picArah(1).ScaleWidth - .Width) / 2, 0 - .Height
'End With
'With img_mbl(2)
.Move picArah(2).ScaleWidth + .Width,
(picArah(2).ScaleHeight - .Height) / 2
'End With
'With img_mbl(3)
.Move (picArah(3).ScaleWidth - .Width) / 2,
picArah(3).ScaleHeight + .Height
'End With
End Sub
Private Sub Form_Load()
ResetArahAnim
LampuMati
pantul = 100
m_NumImages = ImageList4.ListImages.Count
m_NumImg = ImageList3.ListImages.Count
m_NumberImg = ImageList1.ListImages.Count
m_index = 1
m_idx = 1
m_ind = 1
End Sub
Private Sub Form_Unload(Cancel As Integer)
LampuMati
End Sub

```

```

Private Sub lblLampuHijau_Click(Index As Integer)
    LampuMati
    LampuHijauNyala Index
End Sub
Private Sub lblLampuHijau_DblClick(Index As Integer)
    LampuHijauMati Index
End Sub
Private Sub lblLampuKuning_Click(Index As Integer)
    LampuMati
    LampuKuningNyala Index
End Sub
Private Sub lblLampuKuning_DblClick(Index As Integer)
    LampuKuningMati Index
End Sub
Private Sub lblLampuMerah_Click(Index As Integer)
    LampuMati
    LampuMerahNyala Index
End Sub
Private Sub lblLampuMerah_DblClick(Index As Integer)
    LampuMerahMati Index
End Sub
Private Sub tmrAnim_Timer()
    With lblJudul
        .Left = .Left + pantul
        If .Left < 0 Then pantul = 100
        If .Left > Me.ScaleWidth - .Width Then pantul = -100
    End With
End Sub
Private Sub tmrArah_Timer()
    m_idx = (m_idx Mod m_NumImg) + 1
    m_ind = (m_ind Mod m_NumberImg) + 1
    With img_mbl(idxEmpuk)
        Select Case idxLampuHijau
        Case 0
            .Left = .Left + 20
            'If .Left > picArah(idxEmpuk).ScaleWidth Then .Left
            = 0 - .Width
            img_mbl(2).Picture =
            ImageList1.ListImages(m_idx).Picture
            img_mbl(0).Picture =
            ImageList3.ListImages(m_idx).Picture
            img_mbl(0).Visible = True
            img_mbl(1).Visible = False
            img_mbl(2).Visible = True

```

```

img_mati(0).Visible = False
img_mati(1).Visible = True
img_mati(2).Visible = False
img_mbl_ka(1).Visible = False
img_mbl_ka(2).Visible = False
Case 1
'.Top = .Top + 20
'.If .Top > picArah(idxLampuHijau).ScaleHeight Then
.Top = 0 - .Height
img_mbl(1).Picture =
ImageList5.ListImages(m_idx).Picture
img_mbl(2).Picture =
ImageList1.ListImages(m_idx).Picture
img_mbl_ka(1).Picture =
ImageList6.ListImages(m_ind).Picture
img_mbl(0).Visible = False
img_mbl(1).Visible = True
img_mbl(2).Visible = True
img_mati(0).Visible = True
img_mati(1).Visible = False
img_mati(2).Visible = False
img_mbl_ka(1).Visible = True
img_mbl_ka(2).Visible = False
Case 2
'.Left = .Left - 20
'.If .Left < 0 - .Width Then .Left =
picArah(idxLampuHijau).ScaleWidth
img_mbl(2).Picture =
ImageList1.ListImages(m_idx).Picture
img_mbl_ka(2).Picture =
ImageList2.ListImages(m_ind).Picture
img_mbl(0).Visible = False
img_mbl(1).Visible = False
img_mbl(2).Visible = True
img_mati(0).Visible = True
img_mati(1).Visible = True
img_mati(2).Visible = False
img_mbl_ka(1).Visible = False
img_mbl_ka(2).Visible = True
End Select
End With
End Sub
Private Sub tmrKedip_Timer()
m_index = (m_index Mod m_NumImages) + 1

```

```

img_mbl_ki(0).Picture =
ImageList4.ListImages(m_index).Picture
img_mbl_ki(1).Picture =
ImageList7.ListImages(m_index).Picture
End Sub
Private Sub tmrLampu_Timer()
    Static Index As Integer
    Static intLampu As Integer
    Dim intNum As Integer
    Select Case intLampu
    Case 0 'Hijau
        LampuMati
        tmrLampu.Interval = Val(txtHijau(Index).Text) * 1000
        LampuHijauNyala Index
        For intNum = 0 To 2
            If intNum <> Index Then LampuMerahNyala intNum
            If intNum <> Index Then
                merah (intNum)
            Else
                LabelMerah(Index).Caption = 0
            End If
        Next
        intLampu = 1
    Case 1 'Kuning
        LampuMati
        tmrLampu.Interval = Val(txtKuning(Index).Text) * 1000
        LampuKuningNyala Index
        For intNum = 0 To 2
            If intNum <> Index Then LampuMerahNyala intNum
        Next
        intLampu = 0
        Index = Index + 1
        If Index = 3 Then Index = 0
    End Select
End Sub
Private Sub merah(idXLamp As Integer)
    Select Case idXLamp
    Case 0
        LabelMerah(idXLamp).Caption = Val(txtKuning(1).Text) +
Val(txtHijau(1).Text) + _
Val(txtKuning(2).Text) +
Val(txtHijau(2).Text)
    Case 1
        LabelMerah(idXLamp).Caption = Val(txtKuning(0).Text) +
Val(txtHijau(0).Text) + _

```

```
                Val(txtKuning(2).Text) +  
Val(txtHijau(2).Text)  
    Case 2  
        LabelMerah(idxLamp).Caption = Val(txtKuning(0).Text) +  
Val(txtHijau(0).Text) + _  
                Val(txtKuning(1).Text) +  
Val(txtHijau(1).Text)  
    End Select  
End Sub  
Private Sub txtHijau_Change(Index As Integer)  
    With txtHijau(Index)  
        If IsNumeric(.Text) = False Then SendKeys vbBack: Exit  
Sub  
    End With  
End Sub  
Private Sub txtKuning_Change(Index As Integer)  
    With txtKuning(Index)  
        If IsNumeric(.Text) = False Then SendKeys vbBack: Exit  
Sub  
    End With  
End Sub
```

