



UNIVERSITAS INDONESIA

**PENGEMBANGAN METODE SIMPLEKS DENGAN MENGGUNAKAN
METODE ELIMINASI GAUSSIAN – KRITERIA COSINUS UNTUK
MENDAPATKAN PEMECAHAN DASAR YANG LAYAK
(*BASIC FEASIBLE SOLUTION*) DALAM MENYELESAIKAN
PERMASALAHAN PROGRAM LINIER**

SKRIPSI

**MAOLANA HAKIM KUSMAYANTO
0706274823**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INDUSTRI
DEPOK
JUNI 2011**



UNIVERSITAS INDONESIA

**PENGEMBANGAN METODE SIMPLEKS DENGAN MENGGUNAKAN
METODE ELIMINASI GAUSSIAN – KRITERIA COSINUS UNTUK
MENDAPATKAN PEMECAHAN DASAR YANG LAYAK
(*BASIC FEASIBLE SOLUTION*) DALAM MENYELESAIKAN
PERMASALAHAN PROGRAM LINIER**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

**MAOLANA HAKIM KUSMAYANTO
0706274823**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INDUSTRI
DEPOK
JUNI 2011**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah benar hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun yang dirujuk
telah saya nyatakan dengan benar**

Nama : Maolana Hakim Kusmayanto

NPM : 0706274823

Tanda tangan : 

Tanggal : Juni 2011

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama : Maolana Hakim Kusmayanto
NPM : 07060274823
Program Studi : Teknik Industri
Judul Skripsi : Pengembangan Metode Simpleks dengan Menggunakan Metode Eliminasi Gaussian – Kriteria Cosinus Untuk Mendapatkan Pemecahan Dasar yang Layak (Basic Feasible Solution) dalam Menyelesaikan Permasalahan Program Linier

Telah berhasil dipertahankan di hadapan dewan penguji sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Industri, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Akhmad Hidayatno, ST., MBT.

()

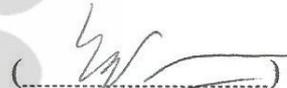
Penguji : Prof. Dr. Ir. Teuku Yuri Zagloel, M.Eng.Sc.

()

Penguji : Ir. M. Dachyar, M.Sc.

()

Penguji : Ir. Yadrifil, M.Sc.

()

Ditetapkan di : Depok

Tanggal : Juni 2010

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Maolana Hakim Kusmayanto
NPM : 0706274823
Program Studi : Teknik Industri
Departemen : Teknik Industri
Fakultas : Teknik
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Non-eksklusif (*Non-exclusive Royalty Free Right*)** atas karya ilmiah saya yang berjudul:

Pengembangan Metode Simpleks dengan Menggunakan Metode Eliminasi Gaussian – Kriteria Cosinus Untuk Mendapatkan Pemecahan Dasar yang Layak (*Basic Feasible Solution*) dalam Menyelesaikan Permasalahan Program Linier

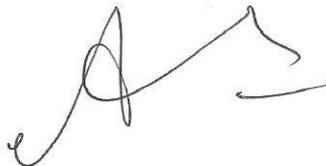
beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-eksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : Juni 2011

Yang menyatakan



(Maolana Hakim Kusmayanto)

KATA PENGANTAR

Pertama-tama, penulis panjatkan puji syukur kepada Allah swt atas segala rahmat dan hidayah-Nya skripsi ini dapat selesai dengan baik. Penulisan skripsi ini dilakukan sebagai persyaratan untuk menuntaskan Program Pendidikan Sarjana Teknik Industri, Fakultas Teknik Universitas Indonesia. Penulisan skripsi ini jauh dari kesempurnaan, dan penulis sadar akan hal itu. Oleh karena itu, penulis berharap adanya masukan yang membangun agar skripsi ini menjadi lebih baik. Penulis sangat berterima kasih atas segala bantuan, saran, komentar, dan bimbingan dari berbagai pihak yang sangat berarti bagi penulis serta dalam proses pembuatan skripsi ini. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak di bawah ini:

1. Bapak Komarudin, ST., M.Eng., sebagai Pembimbing Skripsi yang telah banyak memberikan bantuan, baik berupa ide-saran, masukan, pengarahan, serta bimbingan yang sangat berarti dalam pengerjaan skripsi ini sehingga akhirnya skripsi ini dapat terwujud. Selamat Berjuang di Belgia Pak! Dan maaf telah merepotkan Bapak selama ini.
2. Bapak Akhmad Hidayatno, ST., MBT., yang menjadi Pembimbing Skripsi penulis saat Pak Komar pergi melanjutkan pendidikannya. Saran dan masukan yang Bapak berikan sangat berarti bagi penulisan skripsi ini dan bagi penulis secara khusus.
3. Kedua Orang Tua penulis, Maman Kusmadi dan Eliyati, yang telah memberikan dukungan tak terhingga serta kasih sayang kepada penulis yang menjadi kekuatan terbesar penulis dalam menyelesaikan skripsi ini. Terima kasih atas pelajaran hidup yang telah diberikan hingga saat ini.
4. Makhfudi Kusmayanto dan Diah Laraswati atas dukungannya terhadap penulis. Terima kasih telah mengingatkan dan menjadi panutan penulis dalam banyak hal.
5. Tarida Lusiana, Paulus Bangun, Gersen Samuel, Ariel Wardhana, dan Oscar Sriloka yang telah berjuang bersama-sama dalam menyelesaikan skripsi ini. Terima kasih dukungan, semangat, saran, masukan, serta kebersamaan dalam penulisan skripsi ini. Keberhasilan ini untuk kalian

juga, semoga kita semua sukses, seperti kesuksesan yang telah kita raih bersama-sama selama 5 bulan terakhir.

6. Rendra Satya W. Dan Renaldy Muhamad yang menjadi teman dan pengingat saat penulisan skripsi ini. Terima kasih atas kebersamaan dan pengertian kalian selama 4 tahun ini.
7. Sekar Melati, Gersianto Bagus, Daril Benaya, Christian Tulus, Berry Phann, dan teman-teman lain yang sering mengunjungi dan mengerjakan skripsi di Laboratorium SEM, terima kasih atas canda tawa dan saling mengingatkan. Penulisan skripsi ini menjadi mudah saat bersama kalian semua.
8. Semua mahasiswa TIUI angkatan 2007 atas kebersamaan dan dukungan selama 4 tahun kuliah di TIUI. Saya banyak belajar dari kalian semua atas arti keluarga. Kalian sahabat terbaik selama penulis menjalani pendidikan di TIUI. Penulis sangat beruntung bertemu kalian. Selamat berjuang SAHABAT!
9. Rizky F. Utomo yang telah memberikan saran, masukkan ketika penulis mengalami kegalauan dalam penulisan skripsi ini; mengajarkan penulis dalam menjalani dan menyikapi permasalahan kehidupan.
10. Sawitri yang telah meluangkan waktu, tenaga, dan doa untuk mendukung dan membantu penulis menyelesaikan skripsi ini. Terima kasih telah mendampingi penulis selama ini.
11. Semua pihak yang telah terlibat dan telah membantu penulis dalam menyelesaikan skripsi ini.

Akhir kata, penulis berharap agar skripsi ini bisa memberikan manfaat dan inspirasi bagi semua pihak yang membacanya dan bagi pengembangan ilmu pengetahuan. Penelitian selanjutnya sangat diharapkan untuk menyempurnakan skripsi ini.

Depok, Juni 2011

Penulis

ABSTRAK

Nama : Maolana Hakim Kusmayanto
Program Studi : Teknik Industri
Judul : Pengembangan Metode Simpleks dengan Menggunakan Metode Eliminasi Gaussian – Kriteria Cosinus Untuk Mendapatkan Pemecahan Dasar yang Layak (Basic Feasible Solution) dalam Menyelesaikan Permasalahan Program Linier

Hingga saat ini, pengembangan Metode Simpleks terus dilakukan untuk mendapatkan algoritma efisien yang dapat mengurangi iterasi dan waktu komputasi. Penelitian ini dilakukan sebagai hasil pengembangan dari penelitian terdahulu dan difokuskan pada pembentukan algoritma untuk mendapatkan *Basic Feasible Solution* (BFS) karena penggunaan BFS pada Metode Simpleks terbukti dapat mengurangi iterasi. Algoritma yang dikembangkan menggunakan Kriteria Cosinus serta Eliminasi Gaussian dalam mendapatkan BFS dan diuji pada 13 kasus yang berasal dari jurnal-jurnal Program Linier. Di akhir skripsi, dibahas hasil dari penelitian ini yang menunjukkan bahwa algoritma yang dikembangkan dapat digunakan untuk menyelesaikan masalah Program Linier, dan dapat mengurangi jumlah iterasi yang dilakukan.

Kata kunci:

Optimasi, Program Linier, Metode Simplex, Algoritma Simpleks, Algoritma Simpleks Cosinus, Kriteria Cosinus, *Basic Feasible Solution*.

ABSTRACT

Name : Maolana Hakim Kusmayanto
Study program : Industrial Engineering
Title : Developing Simplex Method by Using Gaussian Elimination
– Cosine Criterion to Get Basic Feasible Solution in Solving
Linear Programming

Up until now, researches in developing Simplex Method are done continually to get the most efficient algorithm to reduce iteration and computation time in solving Linear Programming. This research is held as a development of previous researches and focused in forming algorithm to get Basic Feasible Solution (BFS) because using BFS in Simplex Method has been proved in reducing iteration. This algorithm uses Cosine Criterion and Gaussian Elimination to get BFS and is tested by solving 13 problems which are gained from Journal about Linear Programming. At the end, this paper discusses the result from this research which shows that this algorithm can be used to solve Linear Programming and reduce iteration. In solving Linear Programming.

Key words:

Optimization, Linear Programming, Simplex Method , Simplex Algorithm, Cosine Simplex Algorithm, Cosine Criterion, Basic Feasible Solution

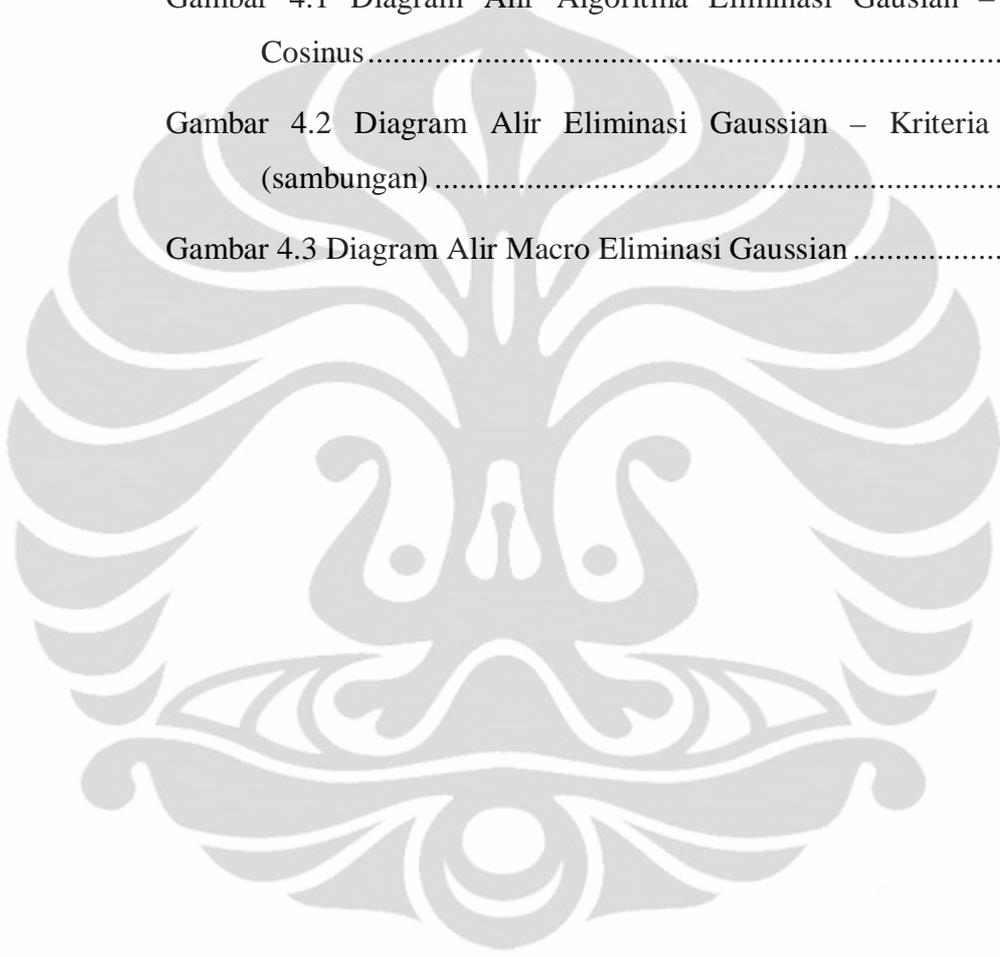
DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS	
AKHIR UNTUK KEPENTINGAN AKADEMIS.....	v
KATA PENGANTAR.....	vi
ABSTRAK.....	viii
ABSTRACT	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN.....	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Diagram Keterkaitan Masalah	2
1.3 Rumusan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Batasan Penelitian	4
1.6 Metodologi Penelitian	5
1.7 Diagram Alir Metodologi Penelitian.....	7
1.8 Sistematika Penulisan.....	7
BAB II DASAR TEORI.....	9
2.1 Program Linear.....	9
2.2 Model Simplex.....	11
2.2.1 Gagasan Keseluruhan tentang Metode Simpleks	11
2.2.2 Batasan	13
2.2.3 Pemecahan Dasar.....	13
2.2.4 Proses Metode Simpleks Primal	14
2.3 Kriteria Cosinus	15
BAB III PENGUMPULAN DATA.....	16
3.1 Data Kasus	16

BAB IV PENGOLAHAN DATA DAN ANALISA	22
4.1 Algoritma Eliminasi Gaussian – Kriteria Cosinus.....	22
4.1.1 Macro Eliminasi Gaussian	27
4.2 Pengolahan Data dan Analisa Kasus.....	30
4.2.1 Pengolahan Data dan Analisa Kasus 1.....	31
4.2.2 Pengolahan Data dan Analisa Kasus 2.....	34
4.2.3 Pengolahan Data dan Analisa Kasus 3.....	37
4.2.4 Pengolahan Data dan Analisa Kasus 4.....	39
4.2.5 Pengolahan Data dan Analisa Kasus 5.....	40
4.2.6 Pengolahan Data dan Analisa Kasus 6.....	42
4.2.7 Pengolahan Data dan Analisa Kasus 7.....	44
4.2.8 Pengolahan Data dan Analisa Kasus 8.....	47
4.2.9 Pengolahan Data dan Analisa Kasus 9.....	48
4.2.10 Pengolahan Data dan Analisa Kasus 10.....	51
4.2.11 Pengolahan Data dan Analisa Kasus 11.....	54
4.2.12 Pengolahan Data dan Analisa Kasus 12.....	57
4.2.13 Pengolahan Data dan Analisa Kasus 13.....	60
BAB V KESIMPULAN DAN SARAN.....	62
5.1 Kesimpulan.....	62
5.2 Saran.....	63
DAFTAR REFERENSI	64
LAMPIRAN 1	65

DAFTAR GAMBAR

Gambar 1.1 Diagram Keterkaitan Masalah Penelitian.....	3
Gambar 1.2 Diagram Alir Metodologi Penelitian.....	7
Gambar 4.1 Diagram Alir Algoritma Eliminasi Gaussian – Kriteria Cosinus.....	26
Gambar 4.2 Diagram Alir Eliminasi Gaussian – Kriteria Cosinus (sambungan).....	27
Gambar 4.3 Diagram Alir Macro Eliminasi Gaussian.....	30



DAFTAR TABEL

Tabel 2.1 Contoh Penyelesaian Kasus Menggunakan Metode Simpleks 15	
Tabel 4.1 Tabel Permasalahan Kasus 1	31
Tabel 4.2 Tabel Perhitungan Nilai Cosinus Kasus 1.....	31
Tabel 4.3 Tabel Simpleks Kondisi Awal Kasus 1	32
Tabel 4.4 Tabel Simpleks Kasus 1 Ketika BFS Tercapai	32
Tabel 4.5 Tabel Simpleks Iterasi ke-1 Kasus 1	33
Tabel 4.6 Tabel Simpleks Hasil Iterasi ke-1.....	33
Tabel 4.7 Tabel Permasalahan Kasus 2.....	34
Tabel 4.8 Tabel Perhitungan Nilai Cosinus Kasus 2.....	35
Tabel 4.9 Tabel Simpleks Kasus 2 Ketika Kondisi BFS Tercapai	36
Tabel 4.10 Tabel Permasalahan Kasus 3	37
Tabel 4.11 Tabel Perhitungan Nilai Cosinus Kasus 3.....	38
Tabel 4.12 Tabel Simpleks Kasus 3 Ketika BFS Tercapai.....	38
Tabel 4.13 Tabel Simpleks Proses Iterasi Kasus 3	38
Tabel 4.14 Tabel Permasalahan Kasus 4.....	39
Tabel 4.15 Tabel Perhitungan Nilai Cosinus Kasus 4.....	39
Tabel 4.16 Tabel Simpleks Kasus 4 Ketika BFS Tercapai.....	40
Tabel 4.17 Tabel Permasalahan Kasus 5	40
Tabel 4.18 Tabel Perhitungan Nilai Cosinus Kasus 5.....	41
Tabel 4.19 Tabel Simpleks Kasus 5 Ketika BFS Tercapai.....	41
Tabel 20 Tabel Simpleks Proses Iterasi Kasus 5	42
Tabel 4.21 Tabel Permasalahan Kasus 6	42
Tabel 4.22 Tabel Perhitungan Nilai Cosinus Kasus 6.....	43
Tabel 4.23 Tabel Simpleks Kasus 6 Ketika BFS Tercapai.....	43
Tabel 24 Tabel Simpleks proses Iterasi Kasus 6	43
Tabel 4.25 Tabel Permasalahan Kasus 7.....	44
Tabel 4.26 Tabel Perhitungan Nilai Cosinus Kasus 7.....	45
Tabel 4.27 Tabel Simpleks Kasus 7 Ketika BFS Tercapai.....	45
Tabel 4.28 Tabel Simpleks proses Iterasi Kasus 7.....	46
Tabel 4.29 Tabel Permasalahan Kasus 8.....	47
Tabel 4.30 Tabel Perhitungan Nilai Cosinus Kasus 8.....	47

Tabel 4.31 Tabel Simpleks Kasus 8 Ketika BFS Tercapai.....	48
Tabel 4.32 Tabel Permasalahan Kasus 9.....	48
Tabel 33 Tabel Perhitungan Nilai Cosinus Kasus 9.....	49
Tabel 4.34 Tabel Simpleks Kasus 9 Ketika BFS Tercapai.....	49
Tabel 4.35 Tabel Simpleks Proses Iterasi Kasus 9	50
Tabel 4.36 Tabel Permasalahan Kasus 10	51
Tabel 4.37 Tabel Perhitungan Nilai Cosinus Kasus 10.....	51
Tabel 4.38 Tabel Simpleks Kasus 10 Ketika BFS Tercapai.....	52
Tabel 4.39 Tabel Simpleks Proses Iterasi Kasus 10	52
Tabel 4.40 Tabel Permasalahan Kasus 11	54
Tabel 4.41 Tabel Perhitungan Nilai Cosinus Kasus 11.....	54
Tabel 4.42 Tabel Simpleks Kasus 11 Ketika BFS Tercapai.....	55
Tabel 4.43 Tabel Simpleks Proses Iterasi Kasus 11	56
Tabel 4.44 Tabel Permasalahan Kasus 12	57
Tabel 4.45 Tabel Perhitungan Nilai Cosinus Kasus 12.....	57
Tabel 4.46 Tabel Simpleks Kasus 12 Ketikas BFS Tercapai.....	58
Tabel 4.47 Tabel Simpleks Proses Iterasi Kasus 12	59
Tabel 4.48 Tabel Permasalahan Kasus 13	60
Tabel 4.49 Tabel Perhitungan Nilai Cosinus Kasus 13.....	60
Tabel 4.50 Tabel Simpleks Kasus 13 Ketika BFS Tercapai.....	61
Tabel 4.51 Tabel Simplex Proses Iterasi Kasus 13.....	61

DAFTAR LAMPIRAN

Lampiran 1: Tabel Simplex Proses Iterasi Kasus 4



BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Revolusi Industri yang terjadi pada akhir abad ke-18 di Eropa telah banyak mengubah keadaan industri yang ada pada zaman itu. Revolusi ini dipicu oleh perkembangan ilmu pengetahuan yang melahirkan penemuan-penemuan dan riset-riset yang berguna di dalam industri. Inti dari Revolusi Industri itu sendiri adalah melakukan optimasi terutama dalam kegiatan produksi di dalam industri dan hingga saat ini berbagai macam upaya telah dikembangkan untuk mencapai optimasi.

Optimasi merupakan momok penting dalam setiap kegiatan industri. Optimasi bisa dilakukan di setiap bidang dalam industri dan optimasi yang dilakukan bidang produksi menjadi sangat penting mengingat kegiatan produksi merupakan kegiatan vital dari suatu industri. Kegiatan ini menjadi jantung suatu industri agar industri tersebut dapat berjalan dengan lancar.

Optimasi produksi dapat dikategorikan sebagai permasalahan Program Linier atau *Linear Programming* (LP). Salah satu metode yang dapat digunakan untuk menyelesaikan permasalahan Program Linier adalah Algoritma Simpleks. Algoritma Simpleks merupakan metode pertama yang digunakan untuk menyelesaikan LP dan merupakan salah satu metode yang efisien dalam memecahkan permasalahan Program Linier. Metode ini digunakan pertama kali oleh Angkatan Laut Amerika atau *American Air Force* selama Perang Dunia ke-2 dan kemudian dikembangkan sehingga dapat digunakan untuk menyelesaikan permasalahan optimasi dalam dunia industri. Hingga saat ini, Algoritma Simpleks menjadi metode yang paling populer yang sering digunakan untuk menyelesaikan permasalahan LP (Junior & Lins, 2005).

Hingga saat ini, berbagai penelitian dilakukan untuk mengembangkan Algoritma Simpleks. Banyak penelitian mengenai Algoritma Simpleks ini dilakukan dengan tujuan untuk mengurangi jumlah iterasi yang dilakukan dalam

melakukan perhitungan untuk mendapatkan pemecahan pada kasus LP. Stojkovic & Stanimirovic melakukan penelitian untuk mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) yang terbukti dapat mengurangi jumlah iterasi dalam menyelesaikan permasalahan LP. Hal ini tertuang dalam jurnal internasional yang mereka buat pada tahun 2001 yang berjudul *Two Direct Methods in Linear Programming*.

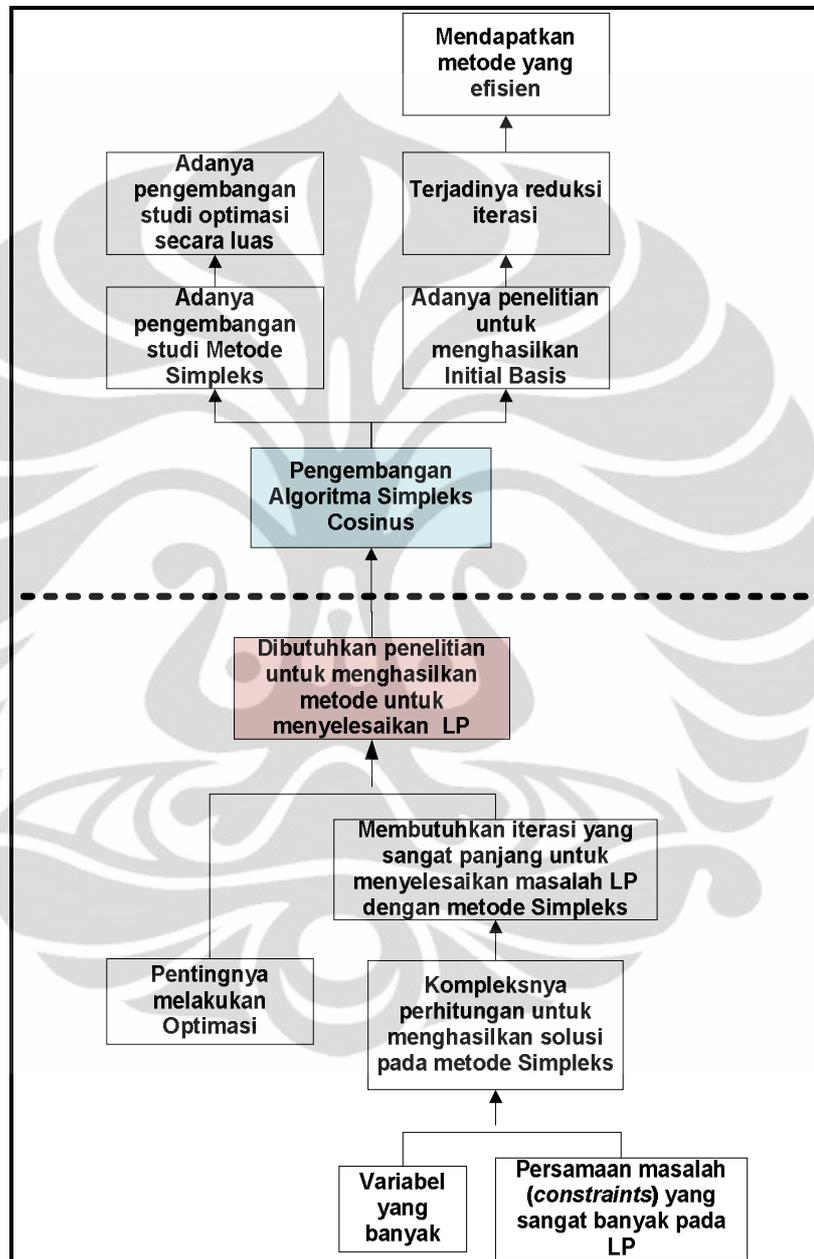
Dalam beberapa penelitian, nilai Cosinus dari sudut yang dibentuk antara *Constraints Function* (Fungsi Kendala) dengan *Objective Function* (Fungsi Tujuan) telah dijadikan sebagai nilai parameter untuk mendapatkan suatu algoritma yang cepat dalam menyelesaikan permasalahan LP (*faster-polynomial algorithm*). *Cosine Simplex Algorithm* merupakan salah satu metode yang dikembangkan berdasarkan kriteria Cosinus ini. Metode *Cosine Simpleks Algorithm* diperkenalkan pada tahun 2006 oleh Corley et al dalam jurnal yang mereka buat. Tujuan dari metode ini adalah mereduksi jumlah iterasi pada Algoritma Simpleks dan jumlah perhitungan dalam setiap iterasi. Dalam *Cosine Simpleks Algorithm*, Kriteria Cosinus digunakan untuk mengurutkan *Constraints Function* (Fungsi Kendala) yang menyebabkan tujuan dari metode ini tercapai.

Kedua hal inilah yang melatarbelakangi studi ini untuk melakukan penelitian lebih mendalam bagi metode *Cosine Simpleks Algorithm* dalam menyelesaikan permasalahan LP. Studi ini menitikberatkan pada pencarian algoritma untuk mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) dengan menggabungkan Kriteria Cosinus dalam proses pencarian BFS tersebut untuk menyelesaikan permasalahan *Linear Programming* (LP). Cara yang digunakan pada studi ini untuk mendapatkan BFS adalah dengan menggunakan metode Eliminasi Gaussian.

1.2 Diagram Keterkaitan Masalah

Berdasarkan latar belakang masalah di atas, maka dapat dibuat Diagram Keterkaitan Masalah yang menampilkan permasalahan secara visual dan sistematis. Diagram ini menjelaskan latar belakang studi yang dilakukan dan pengembangan lebih lanjut dari hasil yang didapat dari studi ini. Dari diagram

tersebut, diharapkan bahwa penelitian ini menjadi inspirasi untuk penelitian lebih lanjut mengenai Algoritma Simpleks secara khusus, dan juga penelitian di bidang optimasi secara umum. Diagram Keterkaitan Masalah untuk studi ini dapat dilihat dengan jelas pada Gambar 1.1



Gambar 1.1 Diagram Keterkaitan Masalah Penelitian

1.3 Rumusan Masalah

Permasalahan yang menjadi fokus penelitian dalam studi ini adalah pengembangan metode *Cosine Simpleks Algorithm* dalam menyelesaikan LP.

Metode pengembangan yang dilakukan adalah dengan mencari pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) sebagai titik awal dalam perhitungan untuk mendapatkan pemecahan yang optimal pada LP. Metode yang digunakan untuk mendapatkan BFS pada studi ini adalah dengan menggunakan metode Eliminasi Gaussian serta menggabungkannya dengan Kriteria Cosinus.

1.4 Tujuan Penelitian

Berdasarkan latar belakang dan diagram keterkaitan masalah yang telah dijelaskan sebelumnya, maka tujuan dari penelitian ini adalah mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) dengan menggunakan metode Eliminasi Gaussian dan menggabungkannya dengan Kriteria Cosinus dalam menyelesaikan kasus LP. BFS yang didapatkan akan dijadikan titik awal pada perhitungan dengan menggunakan Algoritma Simpleks. Sehingga diharapkan dapat mengurangi jumlah iterasi yang dilakukan dalam menyelesaikan permasalahan LP.

1.5 Batasan Penelitian

Dalam penelitian ini dilakukan pembatasan permasalahan agar pelaksanaan serta hasil yang akan diperoleh sesuai dengan tujuan pelaksanaannya. Berikut ini adalah batasan penelitian dalam penelitian ini:

1. Penelitian ini menggunakan Kriteria Cosinus dan Eliminasi Gaussian untuk mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) yang diharapkan dapat mengurangi iterasi dalam menyelesaikan permasalahan LP.
2. Data dan kasus yang digunakan berasal dari penelitian sebelumnya yang telah dipublikasikan. Data yang dipakai berasal dari penelitian tentang Algoritma Simpleks yang dilakukan oleh Stojkovic & Stanimirovic dan juga penelitian H.W. Corley et al tentang penggunaan kriteria Cosinus dalam menggunakan Algoritma Simpleks.
3. Pengujian metode dilakukan pada perangkat lunak Microsoft Excel yang sebelumnya telah dipasang Macro untuk melakukan Eliminasi Gaussian untuk mempermudah melakukan komputasi.

1.6 Metodologi Penelitian

Dalam melaksanakan penelitian ini, digunakan suatu metodologi agar sistematis. Metodologi penelitian yang digunakan dalam penelitian ini secara sistematis dijelaskan pada Gambar 1.2 dengan uraian sebagai berikut:

1. Perumusan ide-ide topik penelitian dan mengidentifikasi permasalahan
Pada tahap pertama, dilakukan pencarian tema-tema yang menarik untuk diangkat, baik dari pencarian pada situs-situs internet, jurnal, maupun buku.
2. Studi literatur dasar teori penelitian
Dilakukan studi literatur teori-teori yang menjadi dasar dalam pelaksanaan penelitian. Landasan teori yang terkait dengan penelitian ini adalah Program Linier, *Simplex Algorithm*, dan *Cosine Simplex Algorithm*.
3. Perumusan masalah
Berdasarkan identifikasi masalah dan studi literatur teori, dapat dirumuskan masalah dalam penelitian ini, yaitu melakukan pengembangan metode *Cosine Simplex Algorithm* dalam menyelesaikan LP. Metode pengembangan yang dilakukan adalah dengan mencari pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) sebagai titik awal dengan menggunakan metode Eliminasi Gaussian dan menggabungkannya dengan Kriteria Cosinus untuk mendapatkan pemecahan yang optimum pada LP.
4. Penentuan tujuan penelitian
Tujuan penelitian ini dapat didefinisikan berdasarkan permasalahan yang ada. Adapun, tujuan penelitian ini adalah mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) dengan menggunakan metode Eliminasi Gaussian dan menggabungkannya dengan Kriteria Cosinus dalam menyelesaikan kasus LP yang pada akhirnya diharapkan dapat mengurangi jumlah iterasi yang dilakukan dalam menyelesaikan permasalahan LP. BFS yang didapatkan akan

dijadikan titik awal pada perhitungan dengan menggunakan Algoritma Simpleks.

5. Pengumpulan data

Proses pengumpulan data dalam penelitian ini didapatkan melalui studi literatur dari jurnal dan beberapa contoh kasus pada penelitian serupa yang pernah dilakukan.

6. Pengembangan metode untuk mendapatkan BFS

Pada tahap ini, dilakukan studi literatur dari buku dan jurnal terkait untuk menemukan metode untuk mendapatkan BFS. Setelah melakukan studi literatur, didapatkan bahwa metode Eliminasi Gaussian akan digunakan untuk mendapatkan BFS dalam penelitian ini.

7. Pengembangan algoritma

Tahap ini dilakukan untuk mendapatkan algoritma yang dapat digunakan untuk menyelesaikan permasalahan LP.

8. Pengujian algoritma

Proses pengujian algoritma dilakukan dengan cara menyelesaikan kasus LP yang terdapat pada jurnal terkait. Hal ini dilakukan untuk menguji apakah algoritma yang dihasilkan dapat digunakan untuk menyelesaikan berbagai macam kasus LP.

9. Menganalisa hasil pengolahan data

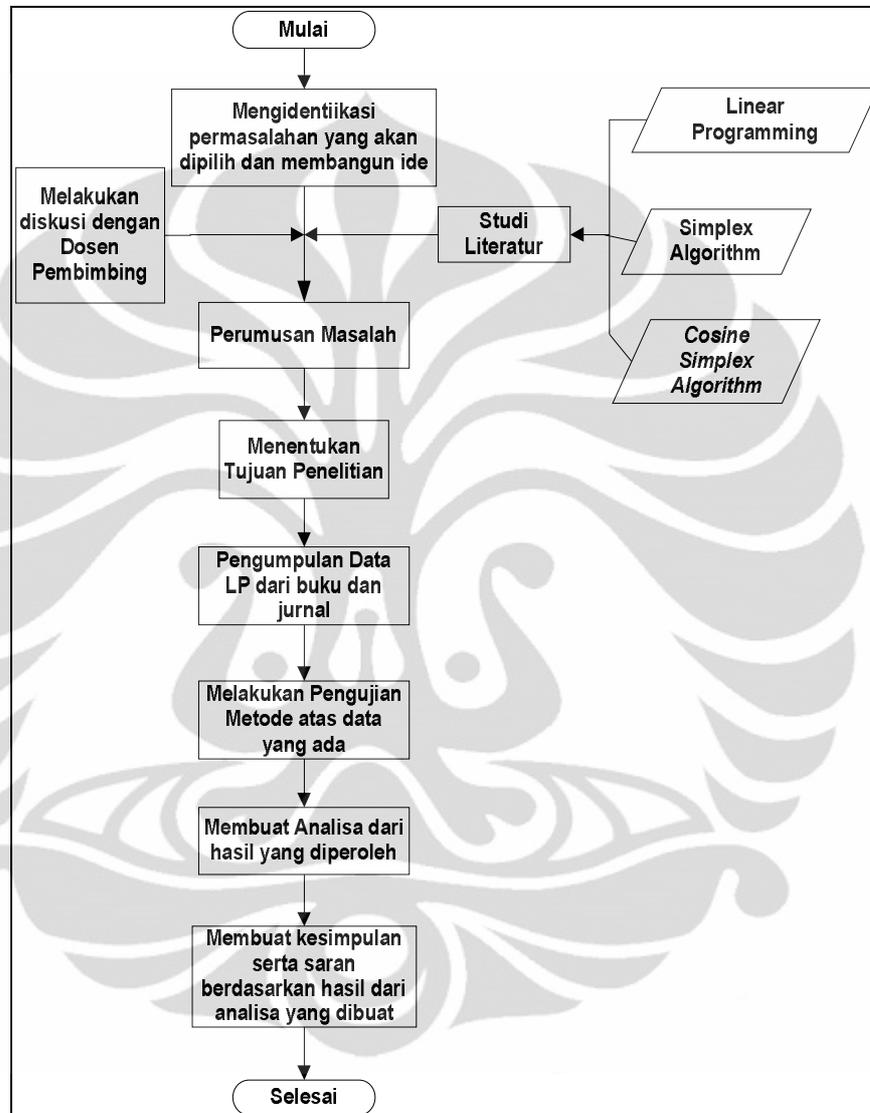
Pada tahap ini, akan dibahas mengenai karakteristik algoritma hasil pengembangan yang telah digunakan untuk menyelesaikan data kasus yang ada.

10. Menarik kesimpulan

Dalam tahapan ini akan dihasilkan kesimpulan mengenai keseluruhan penelitian. Kesimpulan dari penelitian ini merupakan ringkasan dari hasil pengolahan data dan analisa yang telah dilakukan sebelumnya.

1.7 Diagram Alir Metodologi Penelitian

Untuk menggambarkan secara sistematis dan terstruktur metodologi yang dilakukan pada penelitian ini, maka dibuat diagram alir metodologi penelitian yang dapat dilihat pada Gambar 1.2.



Gambar 1.2 Diagram Alir Metodologi Penelitian

1.8 Sistematika Penulisan

Penulisan penelitian ini dibagi menjadi 5 Bab untuk mempermudah dalam memahami penelitian yang dilakukan. Berikut ini adalah penjelasan tiap bab dalam skripsi ini.

Bab pertama adalah pendahuluan. Bab pendahuluan berisikan penjelasan mengenai hal-hal yang mendasari penelitian dan isi dari skripsi yang dibuat. Pada Bab ini akan menjelaskan mengenai latar belakang permasalahan, diagram keterkaitan masalah, perumusan masalah, tujuan penelitian, batasan permasalahan, metodologi penelitian, dan sistematika penulisan.

Bab kedua adalah dasar teori. Pada bab ini, penulis memaparkan teori-teori yang digunakan dan juga berkaitan dengan penelitian ini. Dasar teori ini menjadi landasan dasar dalam melakukan penelitian. Teori-teori yang digunakan diperoleh dari tinjauan pustaka baik berupa buku, jurnal, artikel, informasi yang terdapat di internet dengan sumber yang jelas, serta dari penelitian-penelitian yang pernah dilakukan sebelumnya yang berkaitan dengan permasalahan. Dasar teori yang digunakan dalam penelitian ini meliputi *Linear Programming (LP)*, *Simplex Algorithm*, dan *Cosine Simplex Algorithm*.

Bab ketiga adalah pengumpulan data. Bab ini berisikan pengumpulan data yang dibutuhkan dalam melakukan penelitian ini. Data yang digunakan dalam penelitian kali ini adalah data sekunder yaitu data yang berasal dari penelitian sebelumnya. Data yang digunakan adalah berupa contoh kasus atau soal LP yang akan digunakan untuk menguji *Cosine Simplex Algorithm*.

Bab keempat adalah pengolahan data dan analisa. Di dalam bab ini akan dibahas langkah pengerjaan permasalahan dan juga analisa mengenai hasil yang didapatkan. Proses perencanaan dan pengerjaan kasus yang digunakan untuk mengerjakan permasalahan LP dengan menggunakan algoritma yang dihasilkan akan dipaparkan pada bab ini. Hasil yang didapat, kemudian akan diperiksa apakah layak digunakan untuk menyelesaikan kasus dan akan dibandingkan hasilnya dengan hasil yang didapat pada penelitian sebelumnya.

Bab kelima yang merupakan bab terakhir berisikan kesimpulan dan saran. Kesimpulan dan saran dibuat penulis berdasarkan hasil yang didapat dari penelitian yang dilakukan. Kesimpulan memaparkan hasil dari penelitian yang dilakukan sedangkan saran dibuat penulis sebagai opini pribadi dan catatan dalam mengerjakan penelitian untuk bahan masukan dalam penelitian lebih lanjut yang mungkin akan dilakukan.

BAB II

DASAR TEORI

2.1 Programa Linear

Programa Linier adalah model yang dapat digunakan untuk menentukan bagaimana mencapai tujuan, yang akan memenuhi semua kebutuhan dasar untuk situasi dari suatu permasalahan dengan mengoptimalkan pemakaian sumber yang terbatas. Programa Linier berlaku untuk model optimasi dimana tujuan dan fungsi kendala terhubung secara linier. Programa Linier digunakan dalam berbagai aplikasi, termasuk pertanian, industri, transportasi, ekonomi, sistem kesehatan, perilaku ilmu sosial, dan militer. Programa Linier menawarkan suatu komputasi algoritma efisien untuk masalah dengan ribuan kendala dan variabel.

Linier digunakan untuk menggambarkan suatu hubungan langsung yang proporsional antara dua variabel atau lebih dan *Programming* mengacu pada penggunaan teknik matematis tertentu untuk mendapatkan pemecahan terbaik yang mungkin dilakukan atas suatu masalah yang melibatkan sumber daya yang terbatas. Ada empat karakteristik utama yang harus dimiliki oleh suatu permasalahan program linier, yaitu:

1. Tujuan yang ingin dicapai dapat digambarkan sebagai suatu fungsi persamaan
2. Alternatif lintasan tindakan (*alternative course of action*) dimana salah satunya akan mencapai tujuan
3. Suplai sumber daya terbatas
4. Semua fungsi dan kendala dapat diekspresikan dalam suatu bentuk persamaan matematis

Pemecahan permasalahan dalam program linier dimulai dengan identifikasi gambaran keseluruhan ruang lingkup permasalahan. Identifikasi gambaran keseluruhan ruang lingkup permasalahan dimulai dari identifikasi tujuan yang diinginkan, sumber daya yang tersedia untuk mencapai tujuan tersebut, kebutuhan dan kendala-kendala yang mungkin muncul, dan semua data

relevan yang terkait dengan semua aspek ruang lingkup permasalahan. Data-data yang ada dikonversikan ke dalam model matematis untuk menjabarkan hubungan yang relevan antara tujuan dan batasannya.

Prosesi konversi data menjadi model matematis merupakan bagian yang tersulit dalam menggunakan program linier. Untuk itu diperlukan langkah-langkah yang tepat dalam merumuskan permasalahan menjadi model program linier. Berikut ini adalah urutan proses dalam perumusan permasalahan menjadi model program linier:

1. Identifikasi variabel keputusan
2. Identifikasi koefisien fungsi tujuan
3. Membuat kombinasi linier dari variabel keputusan dan fungsi tujuan
4. Identifikasi ketersediaan sumber daya
5. Menghubungkan ketersediaan sumber daya dengan pemanfaatan sumber daya (pembuatan kendala model)
6. Membuat kombinasi linier dari kendala dan variabel keputusan

Model program linier terdiri dari tiga bagian, yaitu:

1. Variabel keputusan

Solusi optimal ingin dicari oleh model dengan memenuhi semua kendala yang ada. Variabel keputusan umumnya berbentuk $X_1, X_2, X_3, \dots, X_n$. Variabel itu sendiri harus berbentuk non negatif (≥ 0)

2. Fungsi tujuan

Fungsi tujuan berupa persamaan matematis yang ingin dioptimalkan untuk memperoleh solusi yang optimal dengan cara memaksimalkan atau meminimalkan fungsi. Fungsi tujuan dapat berbentuk:

- a. Meminimalkan : $Z(X) = \sum_{i=0}^n C_i X_i$
- b. Memaksimalkan : $Z(X) = \sum_{i=0}^n C_i X_i$

3. Kendala

Kendala merupakan unsur-unsur yang membatasi nilai yang harus dipenuhi oleh variabel keputusan yang dapat diambil. Hubungan antara ruas kiri dan ruas kanan dapat berbentuk $<$, $>$, \leq , \geq , dan $=$.

Secara umum, bentuk model program linier adalah sebagai berikut:

1. Fungsi Tujuan Maksimum / Minimum $Z (X_1, X_2, \dots, X_n) = C_1X_1 + C_2X_2 + \dots + C_nX_n$
2. Kendala $a_{i1}X_1 + a_{i2}X_2 + \dots + a_{in}X_n = b_i \quad i = 1, 2, \dots, m$
 $\left. \begin{array}{l} \leq \\ \geq \end{array} \right\}$
3. Pembatasan ($X_j \geq 0$) $j = 1, 2, \dots, m$

2.2 Model Simplex

Model simplex adalah suatu model penyelesaian program linier dengan variabel keputusan lebih dari 2 buah. Untuk mengembangkan sebuah metode pemecahan umum, masalah program linier harus ditempatkan dalam format standar dengan kriteria sebagai berikut ini:

1. Semua batasan berbentuk persamaan
2. Semua variabel adalah non-negatif
3. Fungsi tujuan dapat berupa memaksimalkan atau meminimalkan

2.2.1 Gagasan Keseluruhan tentang Metode Simpleks

Pada permasalahan *linier programming* solusi optimum selalu berkaitan dengan titik ekstrim atau titik sudut dari ruang pemecahan. Hal ini dibuktikan pada permasalahan LP dengan menggunakan metode grafik. Gagasan ini dengan tepat mengatur pengembangan metode simpleks. Pada intinya apa yang dilakukan metode simpleks adalah menerjemahkan definisi geometris dari titik ekstrim menjadi definisi aljabar.

Metode simpleks mengharuskan agar tiap batasan ditempatkan dalam bentuk standar yang khusus dimana semua batasan diekspresikan sebagai

persamaan dengan menambahkan variabel *slack* dan *surplus* sesuai dengan kendala. Jenis konversi ini umumnya menghasilkan sekelompok persamaan dimana jumlah variabel adalah lebih besar dibandingkan jumlah persamaan, yang umumnya berarti bahwa persamaan-persamaan tersebut menghasilkan sejumlah. Pemecahan yang tidak terbatas (dibandingkan dengan ruang pemecahan secara grafik). Titik ekstrim dari ruang ini dapat diidentifikasi secara aljabar sebagai pemecahan dasar (*basic solution*) dari sistem persamaan simultan tersebut. Dari teori aljabar linear, sebuah pemecahan dasar diperoleh dengan menetapkan beberapa variabel yang sebanyak selisih antara jumlah total variabel dengan jumlah total persamaan memiliki nilai sama dengan nol dan lalu memecahkan variabel sisanya, dengan ketentuan bahwa kondisi tersebut menghasilkan satu pemecahan yang unik. Pada intinya transisi dari prosedur grafik ke prosedur aljabar bergantung pada teori bahwa titik ekstrim merupakan pemecahan dasar.

Dengan tidak adanya ruang pemecahan grafik untuk menentukan arah titik optimum, diperlukan sebuah prosedur yang mengidentifikasi pemecahan-pemecahan dasar yang dapat membawa fungsi tujuan pada nilai yang optimum. Metode simpleks mengidentifikasi satu pemecahan dasar awal dan kemudian bergerak secara sistematis ke pemecahan dasar lainnya yang memiliki potensi untuk memperbaiki nilai fungsi tujuan. Pada akhirnya pemecahan dasar yang bersesuaian dengan nilai optimum akan diidentifikasi dan proses perhitungan berakhir. Jadi metode simpleks merupakan prosedur perhitungan yang berulang (iteratif) dimana setiap pengulangan (iterasi) berkaitan dengan satu pemecahan dasar. Metode simpleks menghasilkan beberapa pemecahan dasar secara berurutan yang pada akhirnya akan mengarahkan fungsi tujuan pada titik ekstrim optimum.

Sifat dari permasalahan *linier programming* yang dapat diselesaikan dengan menggunakan metode simpleks adalah sebagai berikut:

1. Semua variabel adalah non negatif (fungsi kendala dasar)
2. Fungsi tujuan dapat berupa maksimalisasi atau minimalisasi
3. Semua batasan adalah persamaan (dengan sisi kanan yang non negatif jika model tersebut dipecahkan dengan metode Simpleks Primal)

2.2.2 Batasan

Suatu fungsi kendala yang berjenis \leq dapat di konversi menjadi sebuah persamaan dengan menambahkan variabel *slack* ke sisi kiri batasan tersebut.

Contohnya:

$$X_1 + 2X_2 \leq 6$$

Diubah menjadi persamaan dengan proses dibawah ini:

$$X_1 + 2X_2 + S_1 = 6, S_1 \geq 0$$

Untuk fungsi kendala yang berjenis \geq dapat di konversi dengan menambahkan variabel surplus dari sisi kiri batasan tersebut. Contohnya:

$$X_1 + 2X_2 \geq 6$$

Diubah menjadi persamaan dengan proses dibawah ini:

$$X_1 + 2X_2 + S_1 = 6, S_1 \leq 0$$

Metode simpleks pada umumnya digunakan untuk menyelesaikan permasalahan dengan fungsi tujuan yang bersifat memaksimalkan. Namun pada kenyataannya permasalahan LP terdapat kasus dengan fungsi tujuan meminimalisasi. Oleh karena itu pada kasus ini fungsi tujuan harus diubah menjadi memaksimalkan dengan cara mengalikan fungsi tujuan dengan -1 . Sehingga fungsi tujuan tersebut bersifat memaksimalkan nilai dari negatif persamaan fungsi tujuan tersebut.

2.2.3 Pemecahan Dasar

Pertimbangkan model LP standar dengan m persamaan dan n variabel yang tidak diketahui. Pemecahan dasar yang berkaitan ditentukan dengan menetapkan $n - m$ variabel sama dengan nol dan kemudian memecahkan m persamaan dengan n variabel sisanya, asalkan terdapat pemecahan yang dihasilkan dan pemecahan itu unik. Pada contoh permasalahan LP yang memiliki $m = 2$ dan $n = 4$, sebuah pemecahan dasar berkaitan dengan $n - m = 4 - 2 = 2$ variabel adalah nol. Ini berarti bahwa sekelompok persamaan tersebut dapat memiliki $n! / m! (n-m)! = 4! / 2! 2! = 6$ pemecahan dasar yang mungkin. Dikatakan pemecahan dasar yang mungkin karena beberapa kombinasi

kemungkinan tidak menghasilkan pemecahan dasar sama sekali. Hal ini terjadi pada persamaan yang sejajar. Karena pada persamaan tersebut tidak dihasilkan suatu pemecahan (tidak konsisten).

2.2.4 Proses Metode Simpleks Primal

Metode Simpleks merupakan metode yang populer dalam memecahkan permasalahan LP. Metode Simpleks Primal dimulai dari satu pemecahan dasar (titik ekstrim) dan berlanjut untuk berulang melalui pemecahan dasar yang layak berikutnya sampai titik optimum fungsi tujuan tercapai. Pada metode ini pemecahan dasar awal yang layak yang dipilih adalah titik O dimana semua variabel bernilai nol. Kemudian Metode Simpleks Primal akan menentukan variabel yang keluar dari sistem ini dengan cara melihat nilai koefisien variabel pada persamaan fungsi tujuan yang terbesar. Setelah mendapatkan variabel keluar, langkah selanjutnya adalah menentukan fungsi kendala yang membatasi pergerakan fungsi tujuan untuk mendapatkan nilai yang optimum. Langkah ini dengan cara melihat perbandingan antara nilai sisi kanan dengan koefisien variabel keluar di setiap fungsi kendala. Lalu dipilih kendala yang memiliki perbandingan terkecil karena kendala inilah yang membatasi pergerakan fungsi tujuan. Lihat contoh dibawah ini:

$$\text{Maksimumkan } z = 4x_1 + x_2$$

$$\text{Dengan batasan } 3x_1 + x_2 \leq 3$$

$$4x_1 + 3x_2 \leq 6$$

$$x_1 + 2x_2 \leq 4$$

Pada permasalahan tersebut yang akan menjadi variabel keluar adalah X_1 karena pada fungsi tujuan variabel X_1 memiliki koefisien paling besar diantara koefisien variabel-variabel lainnya. Kemudian buat rasio antara sisi kanan dan koefisien X_1 di setiap kendala. Pada contoh di atas rasio kendala 1 adalah $3 / 3 = 1$, rasio kendala dua adalah $6 / 4 = 1,5$, rasio kendala ketiga adalah $4 / 1 = 4$. Dari rasio-rasio tersebut maka kendala yang membatasi pergerakan fungsi tujuan adalah kendala satu karena kendala ini memiliki nilai rasio terkecil. Jika dibuat dalam persamaan aljabar, penyelesaian soal di atas dengan menggunakan metode Simpleks dapat dilihat pada tabel dibawah ini:

Tabel 2.1 Contoh Penyelesaian Kasus Menggunakan Metode Simpleks

	z	x1	x2	Rhs	Ratio
z	1	-4	-1		
1	0	3	1	3	1
2	0	4	3	6	1,5
3	0	1	2	4	4
z	1	0	0,333	4	
1	0	1	0,333	1	
2	0	0	1,667	2	
3	0	0	1,667	3	

2.3 Kriteria Cosinus

Pada pengembangan lebih lanjut mengenai Metode Simpleks, Kriteria Cosinus telah digunakan sebagai parameter untuk menentukan kendala masuk pada perhitungan menggunakan Metode Simpleks. Metode ini dikenalkan oleh Corley et al pada jurnal yang berjudul *The Cosine Simplex Algorithm* tahun 2006. Kriteria Cosinus yang dimaksud adalah menggunakan nilai cosinus dari sudut yang dibentuk oleh Fungsi Kendala dengan Fungsi Tujuan dan kemudian diurutkan berdasarkan nilai cosinus terbesar hingga terkecil. Urutan ini akan dijadikan parameter untuk memilih Fungsi Kendala yang akan digunakan sebagai kendala masuk pada perhitungan Metode Simpleks.

Kriteria cosinus ini muncul berdasarkan teori bahwa semakin besar nilai cosinus atas sudut yang dibentuk antara Fungsi Kendala dengan Fungsi Tujuan memiliki kemungkinan untuk menggeser Fungsi Tujuan ke arah optimalitas sehingga diharapkan Fungsi Kendala ini menjadi kendala yang dapat menghasilkan nilai yang optimum bagi Fungsi Tujuan. Hal ini ditunjukkan oleh Jurnal *The Cosine Simplex Algorithm* yang dibuat oleh Corley et al.

BAB III

PENGUMPULAN DATA

Penelitian yang akan dilakukan bertujuan untuk mendapatkan algoritma yang dapat digunakan untuk menyelesaikan permasalahan Program Linier (LP). Oleh karena itu, data-data yang digunakan dalam penelitian ini bersifat variatif untuk menguji seberapa jauh algoritma yang diperoleh dapat digunakan untuk menyelesaikan permasalahan Program Linier (LP). Data-data pada penelitian ini didapatkan dari beberapa jurnal yang membahas mengenai Algoritma Simpleks, yaitu data yang berasal dari jurnal yang dibuat oleh Corley et al dengan judul *The Cosine Simplex Algorithm* dan juga jurnal yang dibuat oleh Stojkovic & Stanimirovic dengan judul *Two Methods in Linear Programming*. Kasus-kasus LP dari kedua jurnal tersebut penulis gunakan sebagai data penelitian karena kasus-kasus LP tersebut memiliki karakteristik tersendiri sehingga dapat digunakan untuk menguji sejauh mana algoritma yang dihasilkan dapat menyelesaikan permasalahan LP.

3.1 Data Kasus

Pada bagian ini akan dibahas mengenai data-data kasus permasalahan LP yang diperoleh dari jurnal-jurnal tersebut. Berikut ini adalah data-data kasus tersebut:

- Kasus 1

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimalkan } Z = X_1 + X_2$$

Fungsi Kendala (*Constraint Function*)

$$0,001X_1 + 0,111X_2 \leq 1$$

$$0,2X_1 + 0,143X_2 \leq 1$$

$$0,204X_1 + 0,009X_2 \leq 1$$

$$X_1, X_2 \geq 0$$

- Kasus 2

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimalkan } Z = 321X_1 + 287X_2$$

Fungsi Kendala (*Constraint Function*)

$$1,134X_1 + X_2 \leq 1511,8$$

$$0,98X_1 + X_2 \leq 1400,2$$

$$4X_1 + 3X_2 \leq 5000$$

$$0,855X_1 + X_2 \leq 1315,9$$

$$3,036X_1 + 2X_2 \leq 3636,6$$

$$3X_1 + 4X_2 \leq 5000$$

$$3,464X_1 + 2X_2 \leq 4000$$

$$1,974X_1 + 3X_2 \leq 3592,2$$

$$1,984X_1 + X_2 \leq 2222,2$$

$$0,577X_1 + X_2 \leq 1154,7$$

$$2,291X_1 + X_2 \leq 2500$$

$$3X_1 + X_2 \leq 2856,9$$

$$0,504X_1 + X_2 \leq 1119,8$$

$$0,872X_1 + 2X_2 \leq 2182,2$$

$$3,873X_1 + X_2 \leq 4000$$

$$1,122X_1 + 3X_2 \leq 3202,8$$

$$0,314X_1 + X_2 \leq 1048,2$$

$$1,032X_1 + 4X_2 \leq 4130,2$$

$$1,02X_1 + 5X_2 \leq 5103$$

$$X_1, X_2 \geq 0$$

- Kasus 3

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimalkan } Z = -X_1 - X_2 + 20X_3$$

Fungsi Kendala (*Constraint Function*)

$$X_1 + X_2 + 20X_3 \leq 120$$

$$-X_1 + X_2 + X_3 \leq 4$$

$$X_1 - X_2 + X_3 \leq 5$$

$$X_1, X_2, X_3 \geq 0$$

- Kasus 4

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = -X_1 - X_2 + 20X_3$$

Fungsi Kendala (*Constraint Function*)

$$X_1 - X_2 + 20X_3 \leq 20$$

$$-X_1 + X_2 + X_3 \leq 4$$

$$X_1 - X_2 + X_3 \leq 5$$

$$X_1, X_2, X_3 \geq 0$$

- Kasus 5

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 0,75X_1 - 150X_2 + 0,02X_3 - 6X_4$$

Fungsi Kendala (*Constraint Function*)

$$0,25X_1 - 60X_2 - 0,04X_3 + 9X_4 \leq 0$$

$$0,5X_1 - 90X_2 - 0,02X_3 + 3X_4 \leq 0$$

$$X_3 \leq 1$$

$$X_1, X_2, X_3, X_4 \geq 0$$

- Kasus 6

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 4X_1 + 5X_2 + 9X_3 + 11X_4$$

Fungsi Kendala (*Constraint Function*)

$$3X_1 + 5X_2 + 10X_3 + 15X_4 \leq 100$$

$$X_1 + X_2 + X_3 + X_4 \leq 15$$

$$7X_1 + 5X_2 + 3X_3 + 2X_4 \leq 120$$

$$X_1, X_2, X_3, X_4 \geq 0$$

- Kasus 7

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 11X_1 + 18X_2 + 29X_3 - 0,8X_4$$

Fungsi Kendala (*Constraint Function*)

$$\begin{aligned} X_1 + 4X_2 + 6X_3 + 2X_4 &\leq 60 \\ 6X_1 + 5X_2 + 8X_3 + 5X_4 &\leq 102 \\ -0,5X_1 + 0,2X_2 - 0,333X_3 - 0,5X_4 &\leq 1 \\ 4X_1 + X_2 + 5X_3 + 9X_4 &\leq 91 \\ 2X_1 + 5X_2 + 7X_3 + 7X_4 &\leq 110 \\ X_1, X_2, X_3, X_4 &\geq 0 \end{aligned}$$

- Kasus 8

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 11X_1 + 18X_2 + 29X_3 + 28X_4$$

Fungsi Kendala (*Constraint Function*)

$$\begin{aligned} X_1 + 4X_2 + 6X_3 + 2X_4 &\leq 60 \\ 6X_1 + 5X_2 + 8X_3 + 5X_4 &\leq 102 \\ -0,5X_1 + 0,2X_2 - 0,333X_3 - 0,5X_4 &\leq 1 \\ 4X_1 + X_2 + 5X_3 + 9X_4 &\leq 91 \\ 2X_1 + 5X_2 + 7X_3 + 7X_4 &\leq 110 \\ 0,143X_1 - 0,333X_2 - 0,436X_3 + 0,294X_4 &\leq 1 \\ X_1, X_2, X_3, X_4 &\geq 0 \end{aligned}$$

- Kasus 9

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = X_4$$

Fungsi Kendala (*Constraint Function*)

$$\begin{aligned} X_1 &\geq 0,25 \\ X_1 &\leq 1 \\ 0,25X_1 - X_2 &\leq 0 \\ 0,25X_1 + X_2 &\leq 1 \\ 0,25X_2 - X_3 &\leq 0 \\ 0,25X_2 + X_3 &\leq 1 \\ 0,25X_3 - X_4 &\leq 0 \\ 0,25X_3 + X_4 &\leq 1 \end{aligned}$$

$$X_1, X_2, X_3, X_4 \geq 0$$

- Kasus 10

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 2X_1 + 3X_2 + X_3 + 5X_4$$

Fungsi Kendala (*Constraint Function*)

$$0,1X_1 + 0,067X_2 + 0,083X_3 + 0,067X_4 \leq 1$$

$$0,083X_1 + 0,1X_2 + 0,111X_3 + 0,091X_4 \leq 1$$

$$0,125X_1 + 0,167X_2 + 0,2X_3 + 0,111X_4 \leq 1$$

$$0,0834X_1 + 0,063X_2 + 0,071X_3 + 0,05X_4 \leq 1$$

$$0,045X_1 + 0,067X_2 + 0,083X_3 + 0,033X_4 \leq 1$$

$$-0,333X_1 + 0,25X_2 - 0,5X_3 + 0,2X_4 \leq -1$$

$$X_1, X_2, X_3, X_4 \geq 0$$

- Kasus 11

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = X_1 + X_2 + 2X_3 + 5X_4 + 2X_5$$

Fungsi Kendala (*Constraint Function*)

$$30,4X_1 + 12X_2 + 16,5X_3 + 30X_4 + 16,3X_5 \leq 2$$

$$60X_1 - 15,1X_2 - 20X_3 - 25,6X_4 + 40X_5 \leq 5$$

$$48,7X_1 + 24X_2 + 80,2X_3 + 40X_4 + 96,5X_5 \leq 8$$

$$18,7X_1 + 15X_2 + 18,7X_3 + 30X_4 + 18,7X_5 \leq 3$$

$$X_1, X_2, X_3, X_4, X_5 \geq 0$$

- Kasus 12

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 2X_1 + 0,5X_2 + 4X_3 + X_4 + 3X_5 + 5X_6$$

Fungsi Kendala (*Constraint Function*)

$$20X_1 + 6X_2 + 32,3X_3 - 6X_4 + 24X_5 + 60,5X_6 \leq 2$$

$$32X_1 - 6X_2 - 32X_3 - 4X_4 + 48,3X_5 + 160,6X_6 \leq 4$$

$$20X_1 + 7,5X_2 + 100X_3 + 5X_4 + 90,7X_5 + 50X_6 \leq 5$$

$$24X_1 + 15X_2 + 72,7X_3 + 12X_4 + 54X_5 + 120,6X_6 \leq 6$$

$$X_1, X_2, X_3, X_4, X_5 \geq 0$$

- Kasus 13

Fungsi Tujuan (*Objective Function*)

$$\text{Maksimumkan } Z = 4X_1 + 2X_2 + X_3$$

Fungsi Kendala (*Constraint Function*)

$$8X_1 + 4X_2 + X_3 \leq 125$$

$$4X_1 + X_2 \leq 25$$

$$X_1 \leq 5$$

$$X_1, X_2, X_3 \geq 0$$

Kasus-kasus di atas adalah kasus-kasus yang dipilih karena memiliki kondisi tertentu yang dapat menguji algoritma pada penelitian ini. Setiap kasus memiliki keunikan tersendiri yang dapat melihat karakteristik dari algoritma yang akan diuji. Selain itu, pada kasus di atas terdapat kasus Klee-Minty. Kasus tersebut dapat dilihat pada kasus 5 yang didapatkan dari jurnal yang dibuat oleh Corley et al

Kasus Klee-Minty adalah suatu kasus buruk pada LP dimana Algoritma Simpleks membutuhkan iterasi yang cukup banyak untuk mendapatkan nilai optimal walaupun variabel dan fungsi kendala yang terdapat pada kasus ini tergolong tidak banyak. Oleh karena itu, kasus Klee-Minty sangat baik untuk melihat sejauh mana suatu algoritma dapat menyelesaikan suatu permasalahan LP dengan efisien (vanderbei, 2001). Pada penelitian ini, penulis mencoba memecahkan kasus Klee-Minty tersebut dengan menggunakan algoritma yang didapat sehingga dapat mengetahui apakah algoritma tersebut lebih efisien dalam menyelesaikan permasalahan LP.

BAB IV

PENGOLAHAN DATA DAN ANALISA

Pada bagian ini, akan dibahas mengenai algoritma hasil pengembangan beserta dengan langkah-langkah pengerjaan kasus menggunakan algoritma tersebut, dan juga penyelesaian data kasus yang telah ditunjukkan pada Bab 3 disertai dengan analisa pada setiap kasus tersebut.

4.1 Algoritma Eliminasi Gaussian – Kriteria Cosinus

Pengolahan data kasus pada penelitian ini dilakukan untuk menguji sejauh mana algoritma yang diajukan dapat menyelesaikan permasalahan LP. Algoritma yang dikembangkan menitik beratkan pada proses untuk mendapatkan BFS yang kemudian akan digunakan pada proses perhitungan Algoritma Simpleks. Proses pencarian BFS pada algoritma yang dikembangkan dalam penelitian ini melibatkan metode Eliminasi Gaussian yang sebelumnya dilakukan proses penyaringan kendala-kendala yang ada dengan menggunakan Kriteria Cosinus.

Pada dasarnya, perhitungan pada Algoritma Simpleks adalah perhitungan aljabar dengan mengidentifikasi titik ekstrim secara aljabar (Taha, 1993 #7). Solusi pada permasalahan pada Program Linier selalu berada pada titik ekstrim yang dibentuk oleh fungsi kendala dan Algoritma Simpleks mengidentifikasi titik ekstrim ini secara aljabar. Algoritma Simpleks pun melakukan perhitungan dalam menyelesaikan permasalahan Program Linier secara aljabar. Sehingga metode Eliminasi Gaussian dapat digunakan untuk mendapatkan nilai titik ekstrim pada permasalahan Program Linier karena metode ini adalah metode aljabar yang digunakan untuk mengetahui titik temu (titik ekstrim pada Program Linier) dari persamaan linier. Oleh karena itu algoritma yang dikembangkan pada penelitian ini menggunakan metode Eliminasi Gaussian untuk mendapatkan nilai

titik ekstrim yang kemudian akan digunakan sebagai pemecahan dasar yang layak atau *Basic Feasible Solution*.

Terdapat 2 jenis fungsi pada setiap kasus Program Linier, yaitu fungsi tujuan dan fungsi kendala. Fungsi Kendala terdiri beberapa fungsi yang akan membatasi fungsi tujuan untuk mendapatkan hasil yang optimal. Sedangkan Fungsi Tujuan hanya memiliki satu buah fungsi. Terdapat 2 macam Fungsi Tujuan, yaitu:

- Fungsi Tujuan Memaksimalkan
Pada kasus Program Linier dengan jenis ini, solusi yang diperoleh adalah solusi yang dapat memberikan nilai paling maksimal yang dapat dihasilkan oleh fungsi-fungsi kendala.
- Fungsi Tujuan Meminimalkan
Pada kasus Program Linier dengan jenis ini, solusi yang diperoleh adalah solusi yang dapat memberikan nilai paling minimal yang dapat dihasilkan oleh fungsi-fungsi kendala.

Pada algoritma yang akan diuji, akan digunakan Kriteria Cosinus. Kriteria Cosinus ini berfungsi untuk menyaring kendala-kendala yang memiliki kemungkinan akan menyebabkan fungsi tujuan bernilai optimal. Kriteria Cosinus akan mengurutkan kendala yang ada berdasarkan nilai cosinus sudut yang dibentuk antara fungsi kendala dengan fungsi tujuan. Kriteria Cosinus ini mengasumsikan bahwa fungsi kendala yang memiliki nilai cosinus tertinggi akan membawa fungsi tujuan ke solusi yang optimal. Nilai cosinus tinggi menandakan bahwa sudut yang dibentuk antara fungsi kendala dengan fungsi tujuan semakin kecil. Hal ini menunjukkan bahwa fungsi kendala hampir sejajar dengan fungsi tujuan. Ketika fungsi kendala hampir sejajar dengan fungsi tujuan, kendala ini akan menggerakkan fungsi tujuan sejauh mungkin sehingga hasil dari solusi yang diperoleh adalah optimal (Corley et al, 2006).

Secara umum, permasalahan pada Program Linier dapat ditunjukkan pada model matematis di bawah ini

$$\text{Fungsi Tujuan} \quad : Z = c^T x$$

$$\text{Fungsi Kendala} \quad : a_{rx} \leq b_r, x \geq 0$$

Untuk mendapatkan nilai cosinus sudut yang dibentuk antara fungsi kendala dengan fungsi tujuan, digunakan formulasi sebagai berikut:

$$\cos \theta = \frac{a_i^T c}{\|a_i\| \|c\|}$$

Setelah mendapatkan nilai cosinus dari sudut yang dibentuk oleh setiap kendala dengan fungsi tujuan menggunakan (1), kendala akan diurutkan mulai dari kendala dengan nilai cosinus terbesar hingga terkecil. Urutan kendala ini kemudian akan digunakan untuk menentukan kendala yang akan digunakan pada proses Eliminasi Gaussian. Hal ini dilakukan guna menyaring kendala-kendala yang ada karena pada kasus Program Linier, fungsi kendala berjumlah sangat banyak. Metode ini sangat berguna untuk mendapatkan kendala yang tepat.

Jumlah kendala yang akan diambil untuk perhitungan Eliminasi Gaussian adalah sebanyak jumlah variabel yang ada. Hal ini dilakukan karena jika jumlah persamaan kurang dari jumlah variabel, maka tidak ada solusi yang memenuhi kondisi tersebut. Apabila jumlah kendala yang ada lebih sedikit dibandingkan jumlah variabel, maka masukkan kendala dasar (setiap variabel bernilai positif atau ≥ 0) kedalam perhitungan agar proses Eliminasi Gaussian dapat dijalankan.

Selain kasus yang memiliki jumlah persamaan kurang dari jumlah variabel, Eliminasi Gaussian juga tidak dapat menghasilkan solusi untuk kasus yang memiliki persamaan yang sejajar. Karena kasus tersebut tidak menghasilkan solusi pada aljabar linier. Persamaan-persamaan yang sejajar ditandai dengan ratio pada koefisien setiap variabelnya sama. Ketika hal ini ditemukan pada Fungsi Kendala yang dipilih untuk proses Eliminasi Gaussian pada Algoritma Eliminasi Gaussian – Kriteria Cosinus, maka ganti salah satu Fungsi Kendala tersebut dengan Fungsi Kendala berikutnya pada urutan nilai Cosinus hingga tidak didapatkan Fungsi Kendala yang sejajar.

Setelah didapatkan sejumlah kendala, lakukan Eliminasi Gaussian. Hasil dari Eliminasi Gaussian kemudian disubstitusikan ke dalam setiap persamaan kendala untuk mengetahui apakah hasil tersebut merupakan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS). Apabila terdapat fungsi kendala yang tidak terpenuhi, jadikan fungsi kendala tersebut menjadi kendala yang akan dimasukkan ke dalam proses Eliminasi Gaussian menggantikan kendala yang

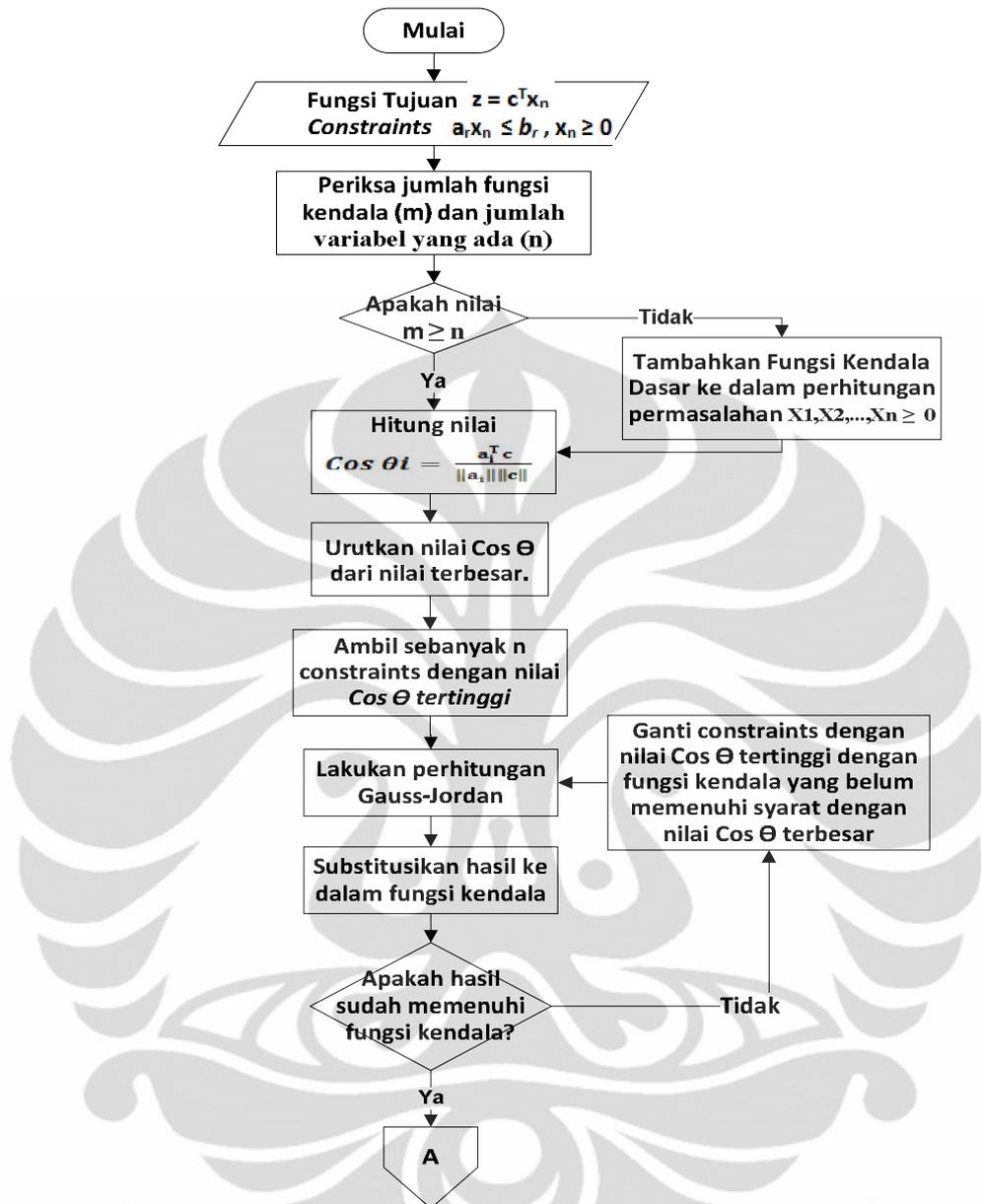
memiliki nilai cosinus terbesar. Kemudian lakukan pemeriksaan kembali apakah hasil yang didapat sudah memenuhi semua fungsi kendala dengan cara mensubstitusikan hasil tersebut pada semua fungsi kendala. Lakukan proses tersebut hingga didapatkan hasil yang dapat memenuhi semua fungsi kendala. Apabila hal tersebut sudah terpenuhi, maka hasil yang didapat merupakan BFS. Hentikan proses ini.

Langkah selanjutnya adalah membuat Tabel Simpleks pada kondisi BFS tercapai. Proses pembentukan Tabel Simpleks dilakukan dengan melakukan operasi baris sehingga setiap kolom memiliki Satu Utama. Untuk melakukan pengecekan, lihat nilai Z. Nilai Z pada Tabel Simpleks harus sesuai dengan nilai Z pada saat hasil Eliminasi Gaussian disubstitusikan pada Fungsi Tujuan.

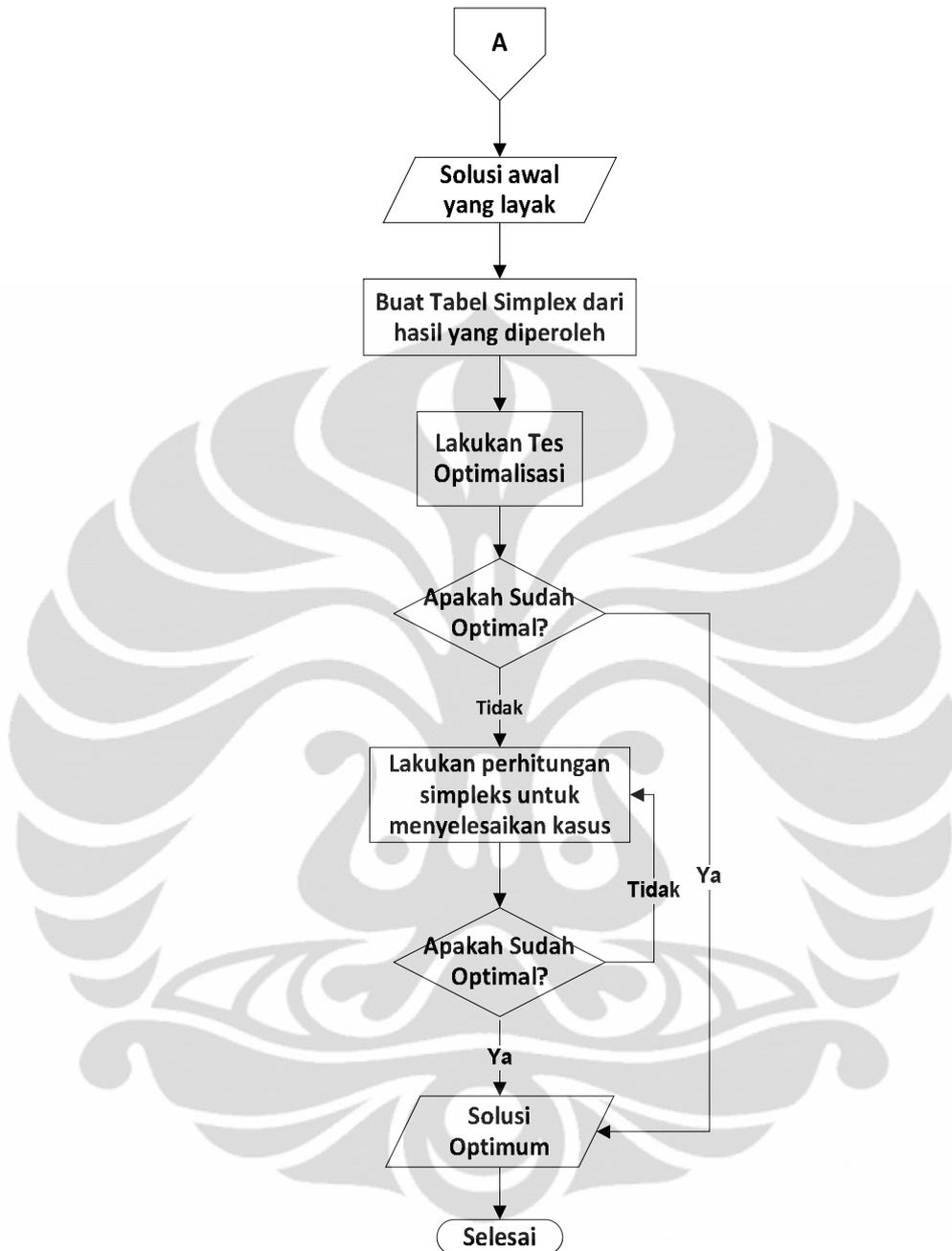
Setelah Tabel Simplex sudah terbentuk, periksa apakah Tabel Simpleks tersebut telah mencapai nilai optimal. Hal ini dilakukan dengan melihat entri pada baris pertama (Fungsi Tujuan). Apabila ditemukan entri pada baris pertama bernilai negatif, hal ini menunjukkan bahwa kondisi tersebut belum optimal. Lakukan Algoritma Simpleks hingga didapatkan kondisi yang optimal. Algoritma Simpleks yang digunakan adalah Algoritma Simpleks Primal. Namun pada proses pengerjaannya, terjadi kemungkinan untuk menggunakan Algoritma Simpleks Dual untuk mendapatkan nilai Fungsi Tujuan yang optimal. Algoritma Simpleks Dual dilakukan apabila Tabel Simpleks sudah mencapai optimalitas, namun variabelnya bernilai negatif.

Proses ini akan berhenti hingga entri pada Fungsi Tujuan tidak ada yang bernilai negatif. Hal ini menunjukkan bahwa kondisi tersebut telah optimal dan kasus mendapatkan solusi dengan nilai pada Fungsi Tujuan yang optimal.

Langkah-langkah pengerjaan Algoritma Eliminasi Gaussian – Kriteria Cosinus akan ditunjukkan secara sistematis pada Gambar 4.1 untuk mempermudah dalam memahami langkah-langkah dalam pengerjaan permasalahan LP dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus. Gambar 4.1 menggambarkan diagram alir Algoritma Eliminasi Gaussian – Kriteria Cosinus yang digunakan pada penelitian ini.



Gambar 4.1 Diagram Alir Algoritma Eliminasi Gaussian – Kriteria Cosinus



Gambar 4.2 Diagram Alir Eliminasi Gaussian – Kriteria Cosinus (sambungan)

4.1.1 Macro Eliminasi Gaussian

Algoritma hasil pengembangan pada penelitian ini menggunakan metode Eliminasi Gaussian untuk mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS). Pada metode Eliminasi Gaussian, matriks yang digunakan adalah matriks yang diperluas. Maksudnya adalah entri pada kolom terakhir matriks tersebut bukan merupakan koefisien variabel, melainkan nilai

hasil pada persamaan kendala atau disebut sebagai *Right Handed Side* (RHS). Hal ini menyebabkan bentuk matriks menjadi tidak persegi dengan jumlah kolom lebih banyak 1 buah dibandingkan jumlah baris. Sehingga pada proses Eliminasi Gaussian, kolom terakhir ini tidak dilakukan operasi baris untuk mendapatkan nilai Satu Utama.

Eliminasi Gaussian dapat digunakan untuk menyelesaikan permasalahan Aljabar Linier dengan jumlah persamaan \geq jumlah variabel. Jika jumlah persamaan kurang dari jumlah variabel, maka proses Eliminasi Gaussian tidak dapat dilakukan. Pada kenyataannya, kasus seperti ini memang tidak dapat menghasilkan solusi yang memenuhi.

Selain kasus dengan jumlah persamaan kurang dari jumlah variabel yang ada, Eliminasi Gaussian juga tidak dapat digunakan pada kasus yang memiliki persamaan yang sejajar. Kasus ini tidak dapat menghasilkan solusi yang memenuhi. Oleh karena itu, pada perhitungan Eliminasi Gaussian, kedua kasus ini harus dihindari sebisa mungkin.

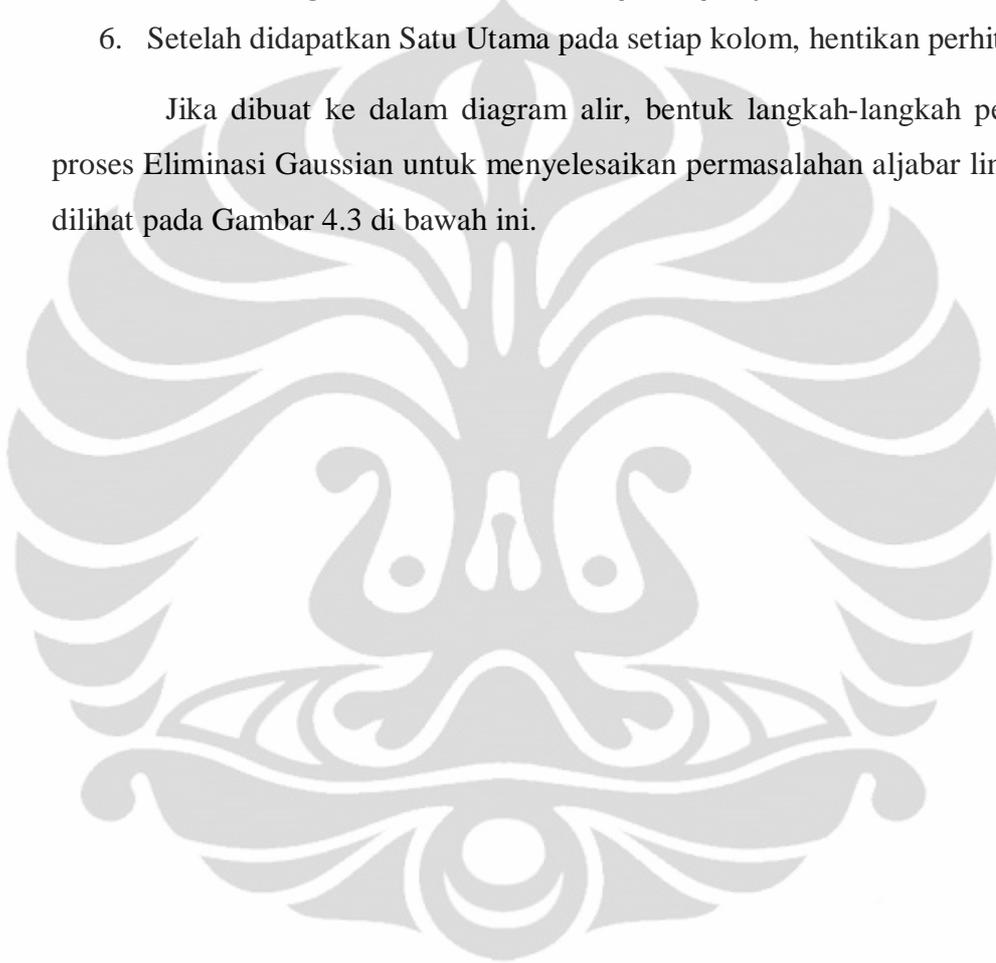
Dalam penelitian ini, pengolahan data dilakukan pada perangkat lunak Microsoft Excel yang telah diintegrasikan dengan macro untuk mempermudah melakukan komputasi. Hal ini dilakukan karena pada Microsoft Excel dapat dilakukan operasi baris dengan mudah sehingga mempermudah Macro yang diintegrasikan mempermudah peneliti dalam mendapatkan hasil dari Eliminasi Gaussian.

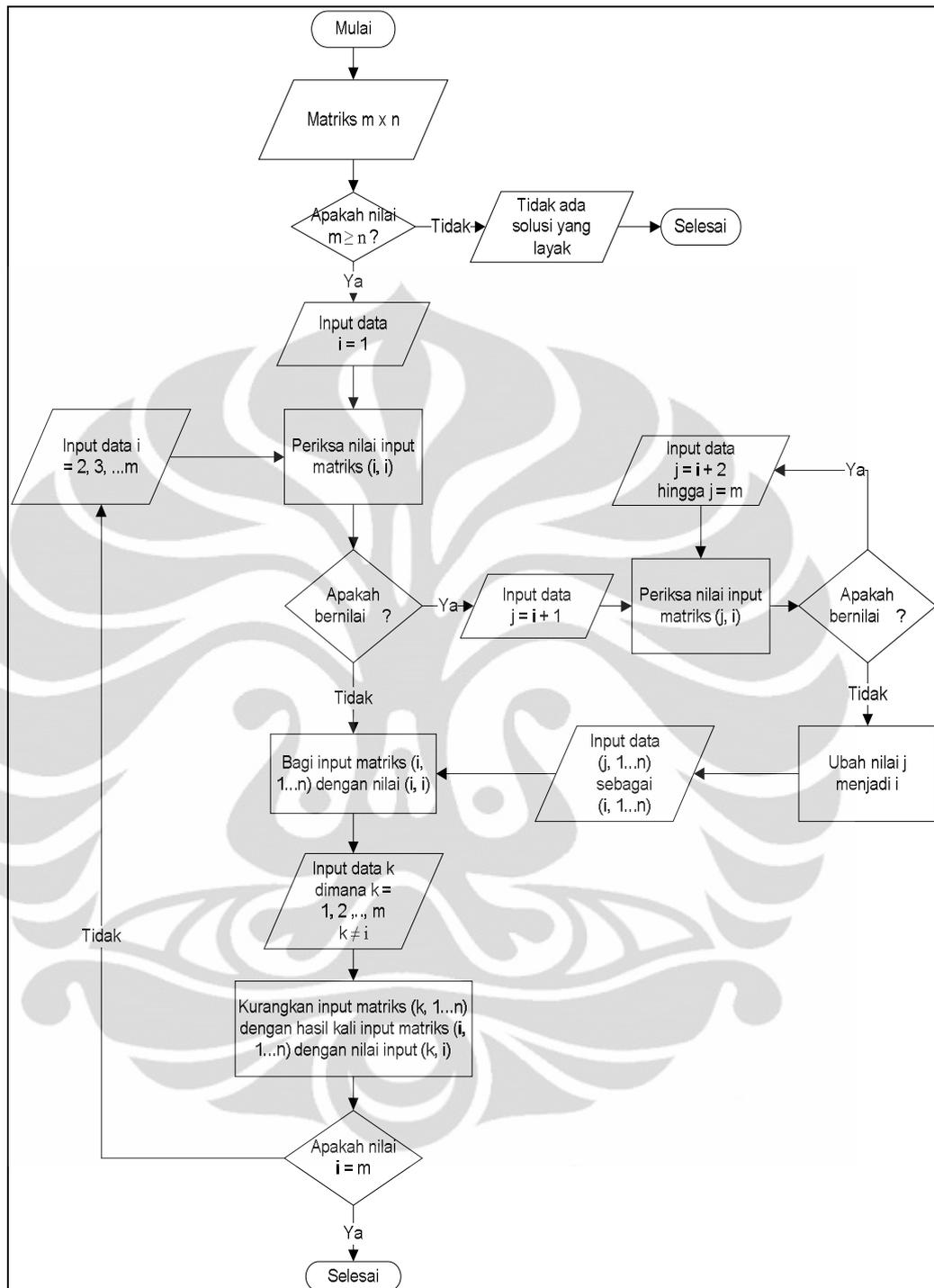
Berikut ini adalah langkah-langkah dalam melakukan metode Eliminasi Gaussian pada Macro di Microsoft Excel yang digunakan untuk menyelesaikan matriks dengan jumlah baris sebanyak m dan jumlah kolom sebanyak n :

1. Periksa apakah nilai $m+1 \geq n$. Jika tidak, maka matriks tersebut tidak memiliki pemecahan dan iterasi dihentikan pada langkah ini.
2. Periksa apakah entri pada baris ke- i kolom ke- $j \neq 0$ dengan nilai $i = j$ dimana $i = 1,2,3,\dots,m$ dan $j = 1,2,3,\dots,n-1$. Apabila bernilai 0, periksa pada baris $i + 1$ hingga menemukan baris yang memiliki entri pada kolom ke- j bernilai $\neq 0$. Jika semua entri pada kolom ke- j bernilai 0, hentikan proses Eliminasi Gaussian karena tidak terdapat pemecahan yang layak.

3. Buatlah entri pada kolom ke- j baris ke- i tersebut menjadi bernilai 1. Lakukan operasi pada kolom tersebut terhadap semua entri kolom pada baris tersebut.
4. Bentuklah Satu Utama pada kolom tersebut dengan membuat entri kolom ke-1 pada baris yang lain menjadi 0 dengan melakukan operasi baris.
5. Lakukan langkah 2 untuk nilai i dan j selanjutnya.
6. Setelah didapatkan Satu Utama pada setiap kolom, hentikan perhitungan.

Jika dibuat ke dalam diagram alir, bentuk langkah-langkah pengerjaan proses Eliminasi Gaussian untuk menyelesaikan permasalahan aljabar linier dapat dilihat pada Gambar 4.3 di bawah ini.





Gambar 4.3 Diagram Alir Macro Eliminasi Gaussian

4.2 Pengolahan Data dan Analisa Kasus

Pada bagian ini, akan dipaparkan pengolahan data kasus menggunakan algoritma Eliminasi Gaussian – Kriteria Cosinus hingga dihasilkan suatu solusi yang dapat memberikan nilai optimum pada fungsi tujuan kasus tersebut. Pada

setiap kasus akan dipaparkan proses penyelesaian menggunakan algoritma Eliminasi Gaussian – Kriteria Cosinus beserta analisa kasus tersebut berdasarkan proses pengerjaan dan hasil yang didapatkan.

4.2.1 Pengolahan Data dan Analisa Kasus 1

Tabel 4.1 Tabel Permasalahan Kasus 1

	X1	X2	Rhs
z	1	1	
1	0,001	0,111111	1
2	0,2	0,014286	1
3	0,20408163	0,009009	1
a	-1	0	0
b	0	-1	0

Kasus 1 ini merupakan contoh permasalahan Program Linier yang sederhana. Pada kasus ini terdapat 3 fungsi kendala dengan 2 variabel. Berikut ini adalah tabel perhitungan nilai cosinus kasus 1.

Tabel 4.2 Tabel Perhitungan Nilai Cosinus Kasus 1

	X1	X2	Rhs	A.Z	IAI	cos
z	1	1			1,414214	
2	0,2	0,014286	1	0,2142857	0,20051	0,7556891
3	0,20408163	0,009009	1	0,2130906	0,20428	0,7376031
1	0,001	0,111111	1	0,1121111	0,111116	0,7134418
a	-1	0	0	-1	1	-0,7071068
b	0	-1	0	-1	1	-0,7071068

Berdasarkan algoritma Eliminasi Gaussian – Kriteria Cosinus, kendala yang akan dipilih untuk proses Eliminasi Gaussian adalah kendala 2 dan 3. Hasil yang didapat dengan melakukan Eliminasi Gaussian pada kedua kendala ini adalah $X_1 = 4,74$ dan $X_2 = 3,367$. Setelah mendapatkan solusi dari proses Eliminasi Gaussian, solusi ini kemudian disubstitusikan kedalam fungsi kendala. Lalu periksa apakah solusi ini telah memenuhi semua fungsi kendala. Apabila terdapat kendala yang tidak terpenuhi, lakukan Eliminasi Gaussian kembali. Namun apabila semua fungsi kendala terpenuhi, maka solusi tersebut merupakan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS).

Pada kasus ini, solusi yang diperoleh pada proses Eliminasi Gaussian telah memenuhi semua fungsi kendala. Oleh karena itu nilai $X_1 = 4,74$ dan $X_2 = 3,67$ merupakan BFS dan solusi ini menghasilkan nilai $Z = 8,4$.

Setelah mendapatkan BFS, kemudian buat Tabel Simpleks dari kondisi saat BFS tersebut. Namun sebelumnya buat Tabel Simplex pada kondisi awal. Berikut ini adalah Tabel Simplex pada kondisi awal kasus

Tabel 4.3 Tabel Simpleks Kondisi Awal Kasus 1

	Z	X1	X2	S1	S2	S3	S4	S5	RHS
Z	1	-1	-1	0	0	0	0	0	0
2	0	0,2	0,014286	0	1	0	0	0	1
3	0	0,20408163	0,009009	0	0	1	0	0	1
1	0	0,001	0,111111	1	0	0	0	0	1

Pada Tabel 4.3 diatas, baris yang diberi warna adalah kendala yang digunakan dalam proses Eliminasi Gaussian. Untuk membuat Tabel Simpleks saat kondisi BFS tercapai, lakukan proses Eliminasi Gaussian untuk baris yang diberi warna. Lakukan operasi baris pada proses Eliminasi Gaussian pada baris lainnya. Hasil akhir dari proses ini merupakan Tabel Simpleks pada saat BFS tercapai. Pada tabel 4.4 di bawah ini menunjukkan Tabel Simpleks saat kondisi tersebut.

Tabel 4.4 Tabel Simpleks Kasus 1 Ketika BFS Tercapai

	Z	X1	X2	S1	S2	S3	S4	S5	RHS
Z	1	0	0	0	175,1651	-166,762	0	0	8,403302
2	0	1	0	0	-8,089623	12,82783	0	0	4,738208
3	0	0	1	0	183,25473	-179,59	0	0	3,665095
1	0	0	0	1	-20,35355	19,94158	0	0	0,588029

Dapat dilihat pada Tabel 4.4 diatas bahwa nilai $X_1 = 4,74$; $X_2 = 3,67$; dengan $Z = 8,4$. Nilai ini sudah sesuai dengan kondisi saat BFS tercapai. Hal ini menandakan bahwa Tabel Simpleks tersebut adalah Tabel Simpleks saat kondisi BFS tercapai.

Langkah selanjutnya adalah melakukan pengecekan apakah nilai Z yang diperoleh saat BFS tercapai sudah optimum (Tes Optimalitas atau *Optimality Test*). Cara untuk melakukan Tes Optimalitas ini adalah dengan melihat koefisien pada Fungsi Tujuan Z , pada Tabel Simpleks di atas ditunjukkan pada baris kedua. Kondisi Optimum dicapai apabila semua entri pada baris ini bernilai positif.

Namun jika hal sebaliknya yang terjadi, hal tersebut menandakan bahwa Tabel Simpleks belum mencapai optimalitas dan perlu dilakukan komputasi Simpleks Primal untuk mendapatkan nilai Z yang optimum.

Pada Tabel Simpleks saat kondisi BFS tercapai di atas, masih terdapat entri pada baris Fungsi Tujuan Z yang bernilai negatif. Hal ini menandakan kondisi ini belum mencapai tingkat optimalitas. Langkah selanjutnya adalah melakukan Algoritma Simpleks Primal untuk mencapai kondisi optimalitas pada kasus ini. Proses pada komputasi Simpleks Primal ini dapat dilihat pada Tabel 4.5 di bawah ini.

Tabel 4.5 Tabel Simpleks Iterasi ke-1 Kasus 1

	Z	X1	X2	S1	S2	S3	S4	S5	RHS	
Z	1	0	0	0	175,1651	-166,762	0	0	8,403302	-0,050391
2	0	1	0	0	-8,089623	12,82783	0	0	4,738208	0,3693694
3	0	0	1	0	183,25473	-179,59	0	0	3,665095	-0,020408
1	0	0	0	1	-20,35355	19,94158	0	0	0,588029	0,0294876

Tabel 4.6 Tabel Simpleks Hasil Iterasi ke-1

Z	1	-2,793E-15	0	8,362519	4,9581874	0	0	0	13,32071	Z
2	0	1	0	-0,64327	5,0032164	0	0	0	4,359946	X1
3	0	-3,007E-15	1	9,005789	-0,045029	0	0	0	8,96076	X2
1	0	-1,675E-17	0	0,050146	-1,020659	1	0	0	0,029488	
a	0	-2,148E-16	-1,4E-17	0,643271	-5,003216	0	1	0	-4,35995	
b	0	3,0073E-15	0	-9,00579	0,0450289	0	0	1	-8,96076	

Tabel 4.6 menunjukkan hasil dari iterasi ke-1 pada Algoritma Simpleks yang dilakukan. Tabel Simpleks yang ditunjukkan pada Tabel 4.6 sudah mencapai optimalitas. Hal ini dapat dilihat pada entri-entri yang terdapat pada Fungsi Tujuan Z yang bernilai positif. Oleh karena itu, Kasus 1 mencapai tingkat optimalitas pada saat nilai variabel $X1 = 4,36$ dan $X2 = 8,96$ dengan nilai $Z = 13,32$. Nilai ini juga ditunjukkan oleh *solver* yang terdapat pada *Microsoft Excel* ketika digunakan untuk menyelesaikan Kasus 1.

Total iterasi yang dilakukan untuk menyelesaikan Kasus 1 dengan menggunakan algoritma pada penelitian ini adalah 2, dengan rincian 1 iterasi pada proses pencarian nilai BFS, dan 1 iterasi pada proses Simpleks Primal. Algoritma Simpleks Original menghasilkan nilai yang sama dengan 3 kali iterasi.

4.2.2 Pengolahan Data dan Analisa Kasus 2

Tabel 4.7 Tabel Permasalahan Kasus 2

	X1	X2	Rhs
z	321	287	
1	1,02	5	5103
2	1,122	3	3202,8
3	0,577	1	1154,7
4	0,855	1	1315,9
5	3,036	2	3636,6
6	1,984	1	2222,2
7	3,873	1	4000
8	1,032	4	4130,2
9	0,872	2	2182,2
10	1,974	3	3592,2
11	0,98	1	1400,2
12	3,464	2	4000
13	2,291	1	2500
14	0,314	1	1048,2
15	0,504	1	1119,8
16	3	4	5000
17	4	3	5000
18	1,134	1	1511,8
19	3	1	2.857
a	-1	0	0
b	0	-1	0

Kasus 2 adalah contoh kasus yang memiliki 2 variabel dan 19 Fungsi Kendala. Pengolahan kasus ini akan menunjukkan seberapa efisien algoritma pada penelitian ini dalam menyelesaikan kasus dengan jumlah Fungsi Kendala yang banyak.

Tabel 4.8 Tabel Perhitungan Nilai Cosinus Kasus 2

	X1	X2	Rhs	A.Z	IAI	cos
z	321	287			430,5926	
18	1,134	1	1511,8	651,014	1,511938	0,999977
11	0,98	1	1400,2	601,58	1,400143	0,997825
17	4	3	5000	2145	5	0,996301
4	0,855	1	1315,9	561,455	1,315684	0,991053
5	3,036	2	3636,6	1548,556	3,63556	0,989211
16	3	4	5000	2111	5	0,980509
12	3,464	2	4000	1685,944	3,999912	0,978873
10	1,974	3	3592,2	1494,654	3,591194	0,966574
6	1,984	1	2222,2	923,864	2,221769	0,965701
3	0,577	1	1154,7	472,217	1,154525	0,949886
13	2,291	1	2500	1022,411	2,499736	0,949871
19	3	1	2.857	1145,996	2,856742	0,931634
15	0,504	1	1119,8	448,784	1,119829	0,93072
9	0,872	2	2182,2	853,912	2,18183	0,90892
7	3,873	1	4000	1530,233	4,000016	0,888442
2	1,122	3	3202,8	1221,162	3,202949	0,885435
14	0,314	1	1048,2	387,794	1,048139	0,859242
8	1,032	4	4130,2	1479,272	4,130983	0,831626
1	1,02	5	5103	1762,42	5,10298	0,802082
a	-1	0	0	-321	1	0,745484
b	0	-1	0	-287	1	0,666523

Eliminasi Gaussian dilakukan untuk kendala 18 dan 1 dengan hasil akhir nilai $X1 = 724,67$ dan $X2 = 690,02$. Hasil ini sudah mencapai BFS karena semua Fungsi Kendala sudah terpenuhi oleh nilai dari solusi ini. Pada kondisi ini, nilai Z adalah 430.655,997. Langkah selanjutnya adalah membentuk Tabel Simpleks yang ditunjukkan pada Tabel 4.9 dibawah ini.

Tabel 4.9 Tabel Simpleks Kasus 2 Ketika Kondisi BFS Tercapai

	Z	X1	X2	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	Rhs		
Z	1	0	0	258,1	28,95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4E+05	Z
18	0	1	0	6,494	-6,49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	724,7	X1
11	0	0	1	-6,36	7,364	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	690	X2
17	0	0	0	-6,88	3,883	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31,24	
4	0	0	0	0,812	-1,81	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6,284	
5	0	0	0	-6,99	4,987	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	56,45	
16	0	0	0	5,974	-9,97	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	65,9	
12	0	0	0	-9,77	7,766	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	109,7	
10	0	0	0	6,273	-9,27	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	91,64	
6	0	0	0	-6,52	5,519	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	94,43	
3	0	0	0	2,617	-3,62	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	46,54	
13	0	0	0	-8,51	7,513	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	149,8	
19	0	0	0	-11	10,01	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	227,7	
15	0	0	0	3,091	-4,09	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	64,55	
9	0	-0	0	7,065	-9,06	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	170,2	
7	0	0	0	-18,8	17,79	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	503,3	
2	0	0	0	11,81	-14,8	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	319,7	
14	0	0	0	4,325	-5,32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	130,6	
8	0	0	0	18,75	-22,8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	622,3	
1	0	0	0	25,19	-30,2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	913,7	

Dari Tabel Simpleks pada saat BFS tercapai yang ditunjukkan pada Tabel 4.9, dapat dilihat bahwa kondisi ini sudah optimum. Hal ini ditunjukkan dengan nilai semua entri pada Fungsi Tujuan Z bernilai positif. Jadi pada Kasus 2 ini dicapai nilai optimum Fungsi Tujuan Z sebesar 430.655,997. Nilai yang sama juga ditunjukkan ketika kasus ini diselesaikan menggunakan *solver* pada *Microsoft Excel*.

Algoritma Eliminasi Gaussian – Kriteria mendapatkan nilai BFS dengan 1 kali iterasi dan BFS ini merupakan solusi yang menghasilkan nilai optimum pada Fungsi Tujuan Z dari kasus 2. Ketika kasus 2 diselesaikan menggunakan Algoritma Simpleks Original, diperlukan iterasi sebanyak 9 kali. Iterasi yang digunakan jauh lebih banyak dibandingkan iterasi yang digunakan oleh Algoritma Eliminasi Gaussian – Kriteria Cosinus untuk menyelesaikan Kasus 2 ini.

4.2.3 Pengolahan Data dan Analisa Kasus 3

Tabel 4.10 Tabel Permasalahan Kasus 3

	X1	X2	X3	Rhs
z	-1	-1	20	
1	1	1	20	120
2	-1	1	1	4
3	1	-1	1	5
a	-1	0	0	0
b	0	-1	0	0
c	0	0	-1	0

Pada Kasus 3, terdapat 3 Fungsi Kendala dengan 3 variabel. Nilai Optimum dari Fungsi Tujuan Z pada Kasus 3 ini berbanding lurus dengan variabel 3 dan berbanding terbalik dengan variabel 1 dan 2, artinya nilai Z akan semakin besar seiring bertambahnya nilai pada variabel 3 dan akan semakin kecil ketika variabel 1 dan variabel 2 bertambah. Pada Kasus ini, akan dibuktikan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus dapat menyelesaikan permasalahan Program Linier jenis ini.

Tabel 4.11 Tabel Perhitungan Nilai Cosinus Kasus 3

	X1	X2	X3	RHS	A.Z	IAI	COS
z	-1	-1	20			20,049938	
1	1	1	20	120	398	20,049938	0,99005
2	-1	1	1	4	20	1,7320508	0,575912
3	1	-1	1	5	20	1,7320508	0,575912
c	0	0	-1	0	-20	1	0,997509
a	-1	0	0	0	1	1	0,049875
b	0	-1	0	0	1	1	0,049875

Dalam perhitungan nilai Cosinus, Fungsi Kendala yang akan dipakai untuk Eliminasi Gaussian adalah Fungsi Kendala 1, 2, dan 3. Hasil dari Eliminasi Gaussian untuk Fungsi-Fungsi Kendala tersebut adalah $X_1 = 15,25$, $X_2 = 14,75$ dan $X_3 = 4,5$. Solusi ini merupakan BFS karena sudah memenuhi semua Fungsi Kendala yang ada. Langkah selanjutnya adalah membuat Tabel Simpleks pada kondisi ini. Tabel Simpleks yang terbentuk dapat dilihat pada Tabel 4.12.

Tabel 4.12 Tabel Simpleks Kasus 3 Ketika BFS Tercapai

	Z	X1	X2	X3	S1	S2	S3	RHS
Z	1	0	0	0	-1	20	20	60
1	0	1	0	0	0,5	-5,25	-4,75	15,25
2	0	0	1	0	0,5	-4,75	-5,25	14,75
3	0	0	0	1	0	0,5	0,5	4,5

Pada Tabel Simpleks yang terbentuk, dapat dilihat bahwa kondisi ini belum mencapai nilai optimum dengan melakukan Tes Optimalitas. Oleh karena itu, diperlukan perhitungan menggunakan Algoritma Simpleks untuk mendapatkan nilai optimum dari Kasus 3. Berikut ini adalah proses perhitungan Algoritma Simpleks hingga dihasilkan nilai optimum pada Kasus 3.

Tabel 4.13 Tabel Simpleks Proses Iterasi Kasus 3

	Z	X1	X2	X3	S1	S2	S3	RHS	
Z	1	0	0	0	-1	20	20	60	
1	0	1	0	0	0,5	-5,25	-4,75	15,25	30,5
2	0	0	1	0	0,5	-4,75	-5,25	14,75	29,5
3	0	0	0	1	0	0,5	0,5	4,5	-
Z	1	0	2	0	0	10,5	9,5	89,5	Z
1	0	1	-1	0	0	-0,5	0,5	0,5	X1
2	0	0	2	0	1	-9,5	-10,5	29,5	
3	0	0	0	1	0	0,5	0,5	4,5	X3

Proses Perhitungan Algoritma Simpleks setelah kondisi BFS tercapai memerlukan 1 kali iterasi untuk mencapai nilai optimum $Z = 89,5$ dengan nilai $X_1 = 0,5$, $X_2 = 0$ dan $X_3 = 4,5$. Hasil yang sama juga ditunjukkan ketika mengerjakan Kasus 3 dengan menggunakan Algoritma Simpleks Original dan menghasilkan jumlah iterasi sebanyak 2 buah. Jumlah iterasi yang dilakukan oleh Algoritma Eliminasi Gaussian – Kriteria Cosinus adalah 2 buah; 1 iterasi untuk mendapatkan nilai BFS dan 1 iterasi pada proses perhitungan Algoritma Simpleks.

4.2.4 Pengolahan Data dan Analisa Kasus 4

Tabel 4.14 Tabel Permasalahan Kasus 4

	X1	X2	X3	Rhs
z	-1	-1	20	
1	1	-1	20	20
2	-1	1	1	4
3	1	-1	1	5
4	-1	-1	0	-40
a	-1	0	0	0
b	0	-1	0	0
c	0	0	-1	0

Kasus 4 memiliki 3 buah variabel dan 4 buah Fungsi Kendala. Fungsi Tujuan Z yang digunakan pada Kasus 4 sama dengan Fungsi Tujuan Z pada Kasus 3. Pada bagian ini akan ditunjukkan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus dapat menyelesaikan permasalahan dimana Algoritma Simpleks Original tidak dapat menyelesaikannya.

Tabel 4.15 Tabel Perhitungan Nilai Cosinus Kasus 4

	X1	X2	X3	Rhs	A.Z	IAI	cos
z	-1	-1	20			20,04994	
1	1	-1	20	20	400	20,04994	0,995025
2	-1	1	1	4	20	1,732051	0,575912
3	1	-1	1	5	20	1,732051	0,575912
4	-1	-1	0	-40	2	1,414214	0,070535
c	0	0	-1	0	-20	1	0,997509
a	-1	0	0	0	1	1	0,049875
b	0	-1	0	0	1	1	0,049875

Pada Tabel 4.15 dapat dilihat bahwa Fungsi Kendala 1, 2, dan 3 akan dipilih untuk proses Eliminasi Gaussian. Tetapi karena Fungsi Kendala 1 dan 3 sejajar maka kita akan memilih Fungsi kendala 4 untuk menggantikan Fungsi

Kendala 3 sehingga Fungsi Kendala yang dipilih dalam proses Eliminasi Gaussian adalah Fungsi Kendala 1, 2, dan 4. Hasil dari Eliminasi Gaussian pada Fungsi-Fungsi Kendala ini adalah $X_1=18,57$, $X_2=21,43$ dan $X_3=1,14$ dan solusi ini merupakan kondisi BFS karena hasil yang didapat memenuhi semua Fungsi Kendala. Lalu buat Tabel Simpleks pada kondisi ini.

Tabel 4.16 Tabel Simpleks Kasus 4 Ketika BFS Tercapai

	Z	X1	X2	X3	S1	S2	S3	S4	RHS	
Z	1	0	0	0	-1,053	1	1,0526	0	-24,21	
3	0	1	0	0	0,5263	-0,5	-0,026	0	22,105	42
4	0	0	1	0	-0,526	-0,5	0,0263	0	17,895	-34
1	0	0	0	1	-0,053	0	0,0526	0	0,7895	-15
2	0	0	0	0	1,1053	0	-0,105	1	7,4211	6,7143

Z	1	0	0	0	0	1	0,9524	0,9524	-17,14	Z
3	0	1	0	0	0	-0,5	0,0238	-0,476	18,571	X1
4	0	0	1	0	0	-0,5	-0,024	0,4762	21,429	X2
1	0	0	0	1	0	0	0,0476	0,0476	1,1429	X3
2	0	0	0	0	1	0	-0,095	0,9048	6,7143	

Tabel 4.16 di atas menunjukkan hasil Tabel Simpleks ketika kondisi BFS tercapai. Pada Tabel 4.16 dapat dilihat bahwa nilai semua entri pada baris kesatu (Fungsi Tujuan Z) bernilai positif. Hal ini menunjukkan bahwa kondisi ini sudah optimum. Hasil yang sama juga ditunjukkan ketika menyelesaikan kasus ini dengan menggunakan *solver*. Namun ketika kasus ini diselesaikan menggunakan Algoritma Simpleks Original, terjadi siklus pada proses perhitungannya yang menyebabkan Algoritma Simpleks Original tidak dapat menemukan hasil yang optimum. Siklus yang terjadi dapat dilihat pada Lampiran 4.

4.2.5 Pengolahan Data dan Analisa Kasus 5

Tabel 4.17 Tabel Permasalahan Kasus 5

	X1	X2	X3	X4	Rhs
Z	0,75	-150	0,02	-6	
1	0,25	-60	-0,04	9	0
2	0,5	-90	-0,02	3	0
3	0	0	1	0	1
a	-1	0	0	0	0
b	0	-1	0	0	0
c	0	0	-1	0	0
d	0	0	0	-1	0

.Kasus 5 memiliki 4 variabel dengan 3 Fungsi Kendala. Pada kasus ini akan ditunjukkan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus dapat menyelesaikan kasus yang memiliki jumlah Fungsi Kendala yang lebih sedikit dibandingkan jumlah Variabel yang ada.

Tabel 4.18 Tabel Perhitungan Nilai Cosinus Kasus 5

	X1	X2	X3	X4	Rhs	A.Z	IAI	cos
Z	0,75	-150	0,02	-6			150,1218	
2	0,5	-90	-0,02	3	0	13482,37	90,05138	0,997315
1	0,25	-60	-0,04	9	0	8946,187	60,67177	0,982217
3	0	0	1	0	1	0,02	1	0,000133
b	0	-1	0	0	0	150	1	0,999188
d	0	0	0	-1	0	6	1	0,039968
a	-1	0	0	0	0	-0,75	1	0,004996
c	0	0	-1	0	0	-0,02	1	0,000133

Tabel 4.18 diatas menunjukkan nilai Cosinus dari setiap Fungsi Kendala terhadap Fungsi Tujuan Z. Dari tabel tersebut dapat dilihat bahwa Fungsi Kendala yang dipilih untuk proses Eliminasi Gaussian adalah Fungsi Kendala 2, 1, dan 3. Namun, karena proses Eliminasi Gaussian membutuhkan Fungsi Kendala sejumlah banyaknya variabel, maka kita akan memasukkan Fungsi Kendala dasar pada kasus ini. Fungsi Kendala dasar menyatakan bahwa nilai setiap variabel harus bernilai positif. Dengan adanya tambahan Fungsi Kendala, maka kita dapat melakukan Eliminasi Gaussian untuk mendapatkan BFS. Fungsi Kendala yang dipilih adalah Fungsi Kendala 2, 1, 3, dan b dengan hasil solusi $X1=0,016$ $X2=0$ $X3=1$ dan $X4=0,004$ serta nilai Fungsi Tujuan yang dihasilkan adalah sebesar $Z=0,008$. Hasil dari Eliminasi Gaussian untuk Fungsi-Fungsi Kendala ini sudah memenuhi semua Fungsi Kendala lainnya. Hal ini menunjukkan bahwa solusi yang didapat adalah BFS. Lalu buat Tabel Simpleks ketika kondisi ini terjadi. Tabel Simpleks tersebut dapat dilihat pada Tabel 4.19

Tabel 4.19 Tabel Simpleks Kasus 5 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	RHS
Z	1	0	0	0	0	2,2	-1,4	0,008	36	0	0	0	0,008
2	0	1	0	0	0	2,4	-0,8	0,016	-168	0	0	0	0,016
1	0	0	1	0	0	0	0	0	-1	0	0	0	0
3	0	0	0	1	0	0	0	1	0	0	0	0	1
b	0	0	0	0	1	-0,06667	0,133333	0,004	-2	0	0	0	0,004
d	0	0	0	0	0	-0,06667	0,133333	0,004	-2	1	0	0	0,004
a	0	0	0	6,94E-18	-1,8E-15	2,4	-0,8	0,016	-168	0	1	0	0,016
c	0	0	0	0	0	0	0	1	0	0	0	1	1

Dari Tabel Simpleks yang dihasilkan, dapat dilihat bahwa kondisi ini belum optimum. Oleh karena itu perlu dilakukan iterasi Simpleks untuk mendapatkan hasil yang optimum. Proses iterasi Simpleks Kasus 5 dapat dilihat pada Tabel 4.20

Tabel 20 Tabel Simpleks Proses Iterasi Kasus 5

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	RHS	
Z	1	0	0	0	0	2,2	-1,4	0,008	36	0	0	0	0,008	
2	0	1	0	0	0	2,4	-0,8	0,016	-168	0	0	0	0,016	-0,02
1	0	0	1	0	0	0	0	0	-1	0	0	0	0	-
3	0	0	0	1	0	0	0	1	0	0	0	0	1	-
b	0	0	0	0	1	-0,06667	0,133333	0,004	-2	0	0	0	0,004	0,03
d	0	0	0	0	0	-0,06667	0,133333	0,004	-2	1	0	0	0,004	0,03
a	0	0	0	6,94E-18	-1,8E-15	2,4	-0,8	0,016	-168	0	1	0	0,016	-0,02
c	0	0	0	0	0	0	0	1	0	0	0	1	1	-
Z	1	0	0	0	0	1,5	0	0,05	15	10,5	0	0	0,05	Z
2	0	1	0	0	0	2	0	0,04	-180	6	0	0	0,04	X1
1	0	0	1	0	0	0	0	0	-1	0	0	0	0	X2
3	0	0	0	1	0	0	0	1	0	0	0	0	1	X3
b	0	0	0	0	1	0	0	0	0	-1	0	0	0	X4
d	0	0	0	0	0	-0,5	1	0,03	-15	7,5	0	0	0,03	
a	0	0	0	6,94E-18	-1,8E-15	2	0	0,04	-180	6	1	0	0,04	
c	0	0	0	0	0	0	0	1	0	0	0	1	1	

Setelah melakukan 1 kali iterasi pada proses Simpleks ini, sudah didapatkan nilai optimum untuk Fungsi Tujuan Z sebesar 0,05 dengan nilai variabel $X_1=0,04$ $X_2=0$ $X_3=1$ dan $X_4=0$. Total iterasi yang diperlukan untuk menyelesaikan kasus ini dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus adalah 2 iterasi, dengan 1 iterasi pada proses BFS dan 1 iterasi pada proses Simpleks. Algoritma Simpleks Original menyelesaikan kasus ini dengan 2 iterasi.

4.2.6 Pengolahan Data dan Analisa Kasus 6

Tabel 4.21 Tabel Permasalahan Kasus 6

No.	X1	X2	X3	X4	RHS
Z	4	5	9	11	
1	3	5	10	15	100
2	1	1	1	1	15
3	7	5	3	2	120
a	-1	0	0	0	0
b	0	-1	0	0	0
c	0	0	-1	0	0
d	0	0	0	-1	0

Kasus 6 memiliki 4 variabel dan 3 Fungsi Kendala. Seperti halnya Kasus 5, jumlah Fungsi Kendala yang ada lebih sedikit dibandingkan jumlah

variabelnya. Sehingga pada proses penentuan Fungsi Kendala untuk dimasukkan ke dalam proses Eliminasi Gaussian, diperlukan Fungsi Kendala Dasar.

Tabel 4.22 Tabel Perhitungan Nilai Cosinus Kasus 6

No	X1	X2	X3	X4	RHS	A.Z	IAI	COS
Z	4	5	9	11			15,58846	
1	3	5	10	15	100	292	18,9473	0,988627
3	7	5	3	2	120	102	9,327379	0,701516
2	1	1	1	1	15	29	2	0,930175
d	0	0	0	-1	0	-11	1	0,70565
c	0	0	-1	0	0	-9	1	0,57735
b	0	-1	0	0	0	-5	1	0,32075
a	-1	0	0	0	0	-4	1	0,2566

Pada proses Eliminasi Gaussian, diperlukan 4 kali iterasi untuk mendapatkan nilai BFS. Fungsi Kendala yang dipilih pada setiap iterasi adalah 1,2,3,d; 2,3,d,b; 3,d,b,c; dan d,b,c,2. Pada iterasi keempat, solusi yang didapat adalah $X_1=15$ $X_2=0$ $X_3=0$ $X_4=0$. Solusi ini menghasilkan nilai Fungsi Tujuan Z sebesar 60. Lalu buat Tabel Simpleks pada kondisi BFS tercapai. Tabel Simpleks tersebut dapat dilihat pada Tabel 4.22

Tabel 4.23 Tabel Simpleks Kasus 6 Ketika BFS Tercapai

No.	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	RHS
Z	1	0	0	0	0	0	4	0	0	-1	-5	-7	60
d	0	1	0	0	0	0	1	0	0	1	1	1	15
b	0	0	1	0	0	0	0	0	0	-1	0	0	0
c	0	0	0	1	0	0	0	0	0	0	-1	0	0
2	0	0	0	0	1	0	0	0	0	0	0	-1	0
1	0	0	0	0	0	1	-3	0	0	2	7	12	55
3	0	0	0	0	0	0	-7	1	0	-2	-4	-5	15
a	0	0	0	0	0	0	1	0	1	1	1	1	15

Tabel simpleks yang dihasilkan menunjukkan bahwa kondisi saat BFS tercapai belum optimal. Oleh karena itu diperlukan proses Simpleks untuk mendapatkan nilai optimal. Proses Simpleks yang dilakukan dapat dilihat pada Tabel 4.24.

Tabel 24 Tabel Simpleks proses Iterasi Kasus 6

No.	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	RHS
Z	1	0	0	0	0	0	4	0	0	-1	-5	-7	60
d	0	1	0	0	0	0	1	0	0	1	1	1	15
b	0	0	1	0	0	0	0	0	0	-1	0	0	0
c	0	0	0	1	0	0	0	0	0	0	-1	0	0
2	0	0	0	0	1	0	0	0	0	0	0	-1	0
1	0	0	0	0	0	1	-3	0	0	2	7	12	55
3	0	0	0	0	0	0	-7	1	0	-2	-4	-5	15
a	0	0	0	0	0	0	1	0	1	1	1	1	15

Z	1	0	0	0	0	0,5833333	2,25	0	0	0,166667	-0,916667	0	92,08333	
d	0	1	0	0	0	-0,0833333	1,25	0	0	0,8333333	0,416667	0	10,41667	25
b	0	0	1	0	0	0	0	0	0	-1	0	0	0	#DIV/0!
c	0	0	0	1	0	0	0	0	0	0	-1	0	0	0
2	0	0	0	0	1	0,0833333	-0,25	0	0	0,166667	0,5833333	0	4,583333	7,857143
1	0	0	0	0	0	0,0833333	-0,25	0	0	0,166667	0,5833333	1	4,583333	7,857143
3	0	0	0	0	0	0,4166667	-8,25	1	0	-1,16667	-1,083333	0	37,91667	-35
a	0	0	0	0	0	-0,0833333	1,25	0	1	0,8333333	0,416667	0	10,41667	25

Z	1	0	0	0	1,571429	0,7142857	1,857143	0	0	0,428571	0	0	99,28571	Z
d	0	1	0	0	-0,71429	-0,1428571	1,428571	0	0	0,714286	0	0	7,142857	X1
b	0	0	1	0	0	0	0	0	0	-1	0	0	0	X2
c	0	0	0	1	1,714286	0,1428571	-0,42857	0	0	0,285714	0	0	7,857143	X3
2	0	0	0	0	1,714286	0,1428571	-0,42857	0	0	0,285714	1	0	7,857143	
1	0	0	0	0	-1	0	0	0	0	0	0	1	0	
3	0	0	0	0	1,857143	0,5714286	-8,71429	1	0	-0,85714	0	0	46,42857	
a	0	0	0	0	-0,71429	-0,1428571	1,428571	0	1	0,714286	0	0	7,142857	

Setelah melakukan 2 kali iterasi pada proses Simpleks, didapatkan hasil Fungsi Tujuan Z yang optimal. Nilai Fungsi Z yang optimal ini adalah 99,286 dengan nilai variabel $X_1=7,14$ $X_2=0$ $X_3= 7,86$ dan $X_4=0$. Total iterasi yang diperlukan berjumlah 6 iterasi dengan 4 iterasi pada proses BFS dan 2 iterasi pada proses Simpleks.

Dengan menggunakan Algoritma Simpleks Original, hasil ini diperoleh dengan 3 kali iterasi. Iterasi yang dilakukan pada Algoritma Eliminasi Gaussian – Kriteria Cosinus lebih banyak dibandingkan pada Algoritma Simpleks Original. Iterasi pada Algoritma Eliminasi Gaussian – Kriteria Cosinus banyak terjadi pada proses mendapatkan BFS. Tetapi proses iterasi Simpleks pada Algoritma ini lebih sedikit dibandingkan pada Algoritma Simpleks Original.

4.2.7 Pengolahan Data dan Analisa Kasus 7

Tabel 4.25 Tabel Permasalahan Kasus 7

	X1	X2	X3	X4	Rhs
z	11	18	29	-0,8	
1	1	4	6	2	60
2	6	5	8	5	102
3	-0,5	0,2	-0,3333333	-0,5	1
4	4	1	5	9	91
5	2	5	7	7	110
a	1	0	0	0	0
b	0	1	0	0	0
c	0	0	1	0	0
d	0	0	0	1	0

Kasus ini memiliki 5 Fungsi Kendala dengan 4 variabel. Pada kasus ini tidak diperlukan tambahan Fungsi Kendala Dasar karena jumlah Fungsi Kendala sudah memenuhi persyaratan ketika akan melaksanakan proses Metode Simpleks.

Tabel 4.26 berikut ini adalah tabel perhitungan nilai Cosinus yang akan digunakan untuk mengurutkan kendala untuk dijadikan input pada proses Eliminasi Gaussian.

Tabel 4.26 Tabel Perhitungan Nilai Cosinus Kasus

	X1	X2	X3	X4	Rhs	A.Z	IAI	cos
z	11	18	29	-0,8			35,8697644	
1	1	4	6	2	60	255,4	7,54983444	0,94309391
2	6	5	8	5	102	384	12,2474487	0,87409185
5	2	5	7	7	110	309,4	11,2694277	0,76540258
4	4	1	5	9	91	199,8	11,0905365	0,50224359
3	-0,5	0,2	-0,3333333	-0,5	1	-11,166667	0,80691456	0,38580466
c	0	0	-1	0	0	29	1	0,80848036
b	0	-1	0	0	0	18	1	0,50181539
a	-1	0	0	0	0	11	1	0,30666496
d	0	0	0	-1	0	-0,8	1	0,02230291

Dari tabel 4.26 di atas, Fungsi Kendala yang dipilih untuk menjadi input pada proses Eliminasi Gaussian adalah Fungsi Kendala 1, 2, 5, dan 4. Eliminasi Gaussian untuk Fungsi Kendala ini menghasilkan nilai $X_1=1,48$ $X_2=8,61$ $X_3=1,45$ $X_4=7,69$ dengan nilai Fungsi Tujuan $Z=207,166$. Hasil yang diperoleh pada proses Eliminasi Gaussian ini telah memenuhi semua Fungsi Kendala. Hal ini menandakan bahwa solusi yang didapatkan merupakan BFS. Lalu buat Tabel Simpleks saat kondisi tersebut. Tabel Simpleks yang terbentuk dapat dilihat pada tabel 4.27.

Tabel 4.27 Tabel Simpleks Kasus 7 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	Rhs
z	1	0	0	0	4E-15	5,851	1,9503	0	-0,136	-3,004	207,17
1	0	1	0	0	0	-0,236	0,2548	0	-0,045	-0,057	1,4841
2	0	0	1	0	0	-0,503	0,1656	0	-0,379	0,5127	8,6146
5	0	0	0	1	0	0,5987	-0,134	0	0,2484	-0,395	1,4459
4	0	0	0	0	1	-0,172	-0,057	0	0,035	0,1879	7,6911
3	0	0	0	7E-16	0	0,0964	0,021	1	0,1538	-0,169	4,3466

Tabel 4.27 menunjukkan bahwa nilai yang dicapai oleh BFS belum optimum. Maka perlu dilakukan perhitungan menggunakan Metode Simpleks

untuk mendapatkan nilai yang optimum. Berikut ini adalah Tabel Simpleks untuk perhitungan Metode Simpleks pada kasus 7.

Tabel 4.28 Tabel Simpleks proses Iterasi Kasus 7

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	Rhs	
z	1	0	0	0	4E-15	5,851	1,9503	0	-0,136	-3,004	207,17	
1	0	1	0	0	0	-0,236	0,2548	0	-0,045	-0,057	1,4841	-25,89
2	0	0	1	0	0	-0,503	0,1656	0	-0,379	0,5127	8,6146	16,801
5	0	0	0	1	0	0,5987	-0,134	0	0,2484	-0,395	1,4459	-3,661
4	0	0	0	0	1	-0,172	-0,057	0	0,035	0,1879	7,6911	40,932
3	0	0	0	7E-16	0	0,0964	0,021	1	0,1538	-0,169	4,3466	-25,74
z	1	0	5,8584	0	4E-15	2,9031	2,9205	0	-2,357	0	257,63	
1	0	1	0,1118	0	0	-0,292	0,2733	0	-0,087	0	2,4472	-28,14
2	0	0	1,9503	0	0	-0,981	0,323	0	-0,739	1	16,801	-22,73
5	0	0	0,7702	1	0	0,2112	-0,006	0	-0,043	0	8,0807	-185,9
4	0	0	-0,366	0	1	0,0124	-0,118	0	0,1739	0	4,5342	26,071
3	0	0	0,3294	7E-16	0	-0,069	0,0756	1	0,029	0	7,1843	247,86
z	1	0	0,8929	0	13,55	3,0714	1,3214	0	0	0	319,07	Z
1	0	1	-0,071	0	0,5	-0,286	0,2143	0	0	0	4,7143	X1
2	0	0	0,3929	0	4,25	-0,929	-0,179	0	0	1	36,071	
5	0	0	0,6786	1	0,25	0,2143	-0,036	0	0	0	9,2143	X3
4	0	0	-2,107	0	5,75	0,0714	-0,679	0	1	0	26,071	
3	0	0	0,3905	7E-16	-0,167	-0,071	0,0952	1	0	0	6,4286	

Setelah melakukan iterasi sebanyak 2 kali, didapatkan nilai optimum untuk kasus 7. Nilai optimum Fungsi Tujuan yang dihasilkan adalah 319,07 dengan nilai variabel $X_1=4,71$ $X_2=0$ $X_3=9,21$ dan $X_4=0$. Hasil ini ditunjukkan pula ketika menyelesaikan kasus ini menggunakan *solver*.

Total iterasi yang diperlukan oleh Algoritma Eliminasi Gaussian – Kriteria Cosinus adalah 3 iterasi dengan rincian 1 iterasi pada proses BFS dan 2 iterasi pada proses metode Simpleks. Sedangkan Algoritma Simpleks Original menyelesaikan kasus ini dengan 7 kali iterasi. Nilai ini jauh lebih banyak dibanding jumlah iterasi yang diperlukan oleh Algoritma Eliminasi Gaussian – Kriteria Cosinus. Hal ini menunjukkan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus lebih efisien dibandingkan Algoritma Simpleks Original.

4.2.8 Pengolahan Data dan Analisa Kasus 8

Tabel 4.29 Tabel Permasalahan Kasus 8

	X1	X2	X3	X4	Rhs
z	11	18	29	28	
1	1	4	6	2	60
2	6	5	8	5	102
3	-0,5	0,2	-0,33333	-0,5	1
4	4	1	5	9	91
5	2	5	7	7	110
6	0,142857	-0,33333	-0,43478	0,294118	1
a	1	0	0	0	0
b	0	1	0	0	0
c	0	0	1	0	0
d	0	0	0	1	0

Kasus 8 memiliki 6 Fungsi Kendala dengan jumlah variabel sebanyak 4 buah. Urutan Fungsi Kendala berdasarkan Kriteria Cosinus dapat dilihat pada Tabel 4.30 di bawah ini.

Tabel 4.30 Tabel Perhitungan Nilai Cosinus Kasus 8

	X1	X2	X3	X4	Rhs	A.Z	IAI	cos
z	11	18	29	28			45,49725	
1	1	4	6	2	60	313	7,549834	0,911217
2	6	5	8	5	102	528	12,24745	0,947552
3	-0,5	0,2	-0,33333	-0,5	1	-25,5667	0,806915	0,696404
4	4	1	5	9	91	459	11,09054	0,909651
5	2	5	7	7	110	511	11,26943	0,99663
6	0,142857	-0,33333	-0,43478	0,294118	1	-8,80197	0,638013	-0,30323
a	1	0	0	0	0	11	1	0,241773
b	0	1	0	0	0	18	1	0,395628
c	0	0	1	0	0	29	1	0,637401
d	0	0	0	1	0	28	1	-0,61542

Untuk mendapatkan nilai BFS, diperlukan 1 kali iterasi dengan melakukan Eliminasi Gaussian pada Fungsi Kendala 5, 2, 1, dan 4. Nilai variabel dasar yang didapatkan pada kondisi BFS adalah $X_1 = 1,48$ $X_2 = 8,61$ $X_3 = 1,45$ dan $X_4 = 7,69$. Solusi ini menghasilkan nilai Fungsi Tujuan Z sebesar 428,67. Lalu untuk mengetahui apakah nilai ini sudah optimum, bentuk Tabel Simpleks pada kondisi ini. Tabel Simpleks tersebut dapat dilihat pada Tabel 4.31.

Tabel 4.31 Tabel Simpleks Kasus 8 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	Rhs	
z	1	0	0	0	0	2,408	0,299	0,898	0,873	0	0	428,7	Z
5	0	1	0	0	0	-0,06	0,255	-0,24	-0,04	0	0	1,484	X1
2	0	0	1	0	0	0,513	0,166	-0,5	-0,38	0	0	8,615	X2
1	0	0	0	1	0	-0,39	-0,13	0,599	0,248	0	0	1,446	X3
4	0	0	0	0	1	0,188	-0,06	-0,17	0,035	0	0	7,691	X4
3	0	0	0	0	0	-0,17	0,021	0,096	0,154	1	0	4,347	
6	0	0	0	0	0	-0,05	-0,02	0,177	-0,02	0	1	2,026	

Tabel Simpleks yang dibentuk saat BFS tercapai menunjukkan bahwa kondisi ini telah optimum. Maka nilai BFS $X_1 = 1,48$ $X_2 = 8,61$ $X_3 = 1,45$ dan $X_4 = 7,69$ merupakan solusi optimum dari kasus 8. Hasil yang sama ditunjukkan pula oleh *solver*.

Algoritma Simpleks Original memerlukan iterasi sebanyak 5 kali untuk menghasilkan nilai solusi ini. Sedangkan Algoritma Eliminasi Gaussian – Kriteria Cosinus hanya membutuhkan 1 kali iterasi yaitu iterasi untuk mendapatkan BFS.

4.2.9 Pengolahan Data dan Analisa Kasus 9

Tabel 4.32 Tabel Permasalahan Kasus 9

	X1	X2	X3	X4	Rhs
Z	0	0	0	1	
1	-1	0	0	0	-0,25
2	1	0	0	0	1
3	0,25	-1	0	0	0
4	0,25	1	0	0	1
5	0	0,25	-1	0	0
6	0	0,25	1	0	1
7	0	0	0,25	-1	0
8	0	0	0,25	1	1
a	1	0	0	0	0
b	0	1	0	0	0
c	0	0	1	0	0
d	0	0	0	1	0

Pada kasus ini, terdapat 8 Fungsi Kendala dengan 4 variabel. Urutan Fungsi Kendala berdasarkan nilai Cosinusnya ditunjukkan pada Tabel 4.33 berikut.

Tabel 33 Tabel Perhitungan Nilai Cosinus Kasus 9

	X1	X2	X3	X4	Rhs	A.Z	IAI	cos
Z	0	0	0	1			1	
7	0	0	0,25	-1	0	-1	1,031	0,97
8	0	0	0,25	1	1	1	1,031	0,97
1	-1	0	0	0	-0,25	0	1	0
2	1	0	0	0	1	0	1	0
3	0,25	-1	0	0	0	0	1,031	0
4	0,25	1	0	0	1	0	1,031	0
5	0	0,25	-1	0	0	0	1,031	0
6	0	0,25	1	0	1	0	1,031	0
d	0	0	0	-1	0	-1	1	1
a	-1	0	0	0	0	0	1	0
b	0	-1	0	0	0	0	1	0
c	0	0	-1	0	0	0	1	0

Dari tabel ini, Fungsi Kendala yang dipilih untuk proses Eliminasi Gaussian adalah Fungsi Kendala 7, 8, 1, dan 2. Namun Fungsi Kendala 1 dan 2 sejajar sehingga hanya salah satu dari kendala ini yang dipilih dan sisa kendala yang dibutuhkan diambil secara acak dari Fungsi Kendala yang ada. Pemilihan dilakukan secara acak karena Fungsi Kendala memiliki nilai Cosinus yang sama, yaitu 0. Pada penelitian ini dipilih Fungsi Kendala 7, 8, 2, dan 3 sebagai Fungsi yang akan memasuki proses Eliminasi Gaussian. Setelah melakukan iterasi sebanyak 2 kali dengan Fungsi Kendala yang digunakan adalah Fungsi Kendala 8, 2, 3, dan 6, nilai BFS tercapai dengan solusi $X_1=1$ $X_2=0,25$ $X_3=0,938$ dan $X_4=0,766$. Nilai Fungsi Tujuan yang didapatkan adalah 0,766. Lalu buatlah Tabel Simpleks untuk melakukan Tes Optimalitas.

Tabel 4.34 Tabel Simpleks Kasus 9 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	S8	Rhs
Z	1	0	0	0	0	1	0,016	-0,06	-0,25	0	0	0	0	0,766
8	0	1	0	0	0	0	1	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0,25	-1	0	0	0	0	0	0,25
3	0	0	0	1	0	0	-0,06	0,25	1	0	0	0	0	0,938
6	0	0	0	0	1	1	0,016	-0,06	-0,25	0	0	0	0	0,766
7	0	0	0	0	0	1	0,031	-0,13	-0,5	1	0	0	0	0,531
1	0	0	0	0	0	0	1	0	0	0	1	0	0	0,75
4	0	0	0	0	0	0	-0,5	1	0	0	0	1	0	0,5
5	0	0	0	0	0	0	-0,13	0,5	1	0	0	0	1	0,875

Dari Tabel Simpleks yang dibentuk ketika BFS tercapai menunjukkan bahwa solusi tersebut belum optimum. Untuk mendapatkan solusi yang optimum, lakukan proses Metode Simpleks pada Tabel Simpleks tersebut.

Tabel 4.35 Tabel Simpleks Proses Iterasi Kasus 9

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	S7	S8	Rhs	
Z	1	0	0	0	0	1	0,016	-0,06	-0,25	0	0	0	0	0,766	
8	0	1	0	0	0	0	1	0	0	0	0	0	0	1	
2	0	0	1	0	0	0	0,25	-1	0	0	0	0	0	0,25	
3	0	0	0	1	0	0	-0,06	0,25	1	0	0	0	0	0,938	0,938
6	0	0	0	0	1	1	0,016	-0,06	-0,25	0	0	0	0	0,766	-3,06
7	0	0	0	0	0	1	0,031	-0,13	-0,5	1	0	0	0	0,531	-1,06
1	0	0	0	0	0	0	1	0	0	0	1	0	0	0,75	
4	0	0	0	0	0	0	-0,5	1	0	0	0	1	0	0,5	
5	0	0	0	0	0	0	-0,13	0,5	1	0	0	0	1	0,875	0,875
Z	1	0	0	0	0	1	-0,02	0,063	0	0	0	0	0,25	0,984	
8	0	1	0	0	0	0	1	0	0	0	0	0	0	1	1
2	0	0	1	0	0	0	0,25	-1	0	0	0	0	0	0,25	1
3	0	0	0	1	0	0	0,063	-0,25	0	0	0	0	-1	0,063	1
6	0	0	0	0	1	1	-0,02	0,063	0	0	0	0	0,25	0,984	-63
7	0	0	0	0	0	1	-0,03	0,125	0	1	0	0	0,5	0,969	-31
1	0	0	0	0	0	0	1	0	0	0	1	0	0	0,75	0,75
4	0	0	0	0	0	0	-0,5	1	0	0	0	1	0	0,5	-1
5	0	0	0	0	0	0	-0,13	0,5	1	0	0	0	1	0,875	-7
Z	1	0	0	0	0	1	0	0,063	0	0	0,016	0	0,25	0,996	Z
8	0	1	0	0	0	0	0	0	0	0	-1	0	0	0,25	X1
2	0	0	1	0	0	0	0	-1	0	0	-0,25	0	0	0,063	X2
3	0	0	0	1	0	0	0	-0,25	0	0	-0,06	0	-1	0,016	X3
6	0	0	0	0	1	1	0	0,063	0	0	0,016	0	0,25	0,996	X4
7	0	0	0	0	0	1	0	0,125	0	1	0,031	0	0,5	0,992	
1	0	0	0	0	0	0	1	0	0	0	1	0	0	0,75	
4	0	0	0	0	0	0	0	1	0	0	0,5	1	0	0,875	
5	0	0	0	0	0	0	0	0,5	1	0	0,125	0	1	0,969	

Setelah melakukan 2 kali iterasi, didapatkan nilai Optimum untuk Fungsi Z sebesar 0,996 dengan nilai variabel $X_1=0,25$ $X_2=0,06$ $X_3=0,02$ dan $X_4=0,996$ dengan nilai Fungsi Tujuan sebesar 0,996. Total iterasi yang dibutuhkan Algoritma ini berjumlah 4 iterasi, dengan rincian 2 iterasi pada proses Eliminasi Gaussian untuk membentuk BFS dan 2 iterasi sisanya dilakukan pada proses metode Simpleks. Dengan menggunakan Algoritma Simpleks Original, diperlukan 5 kali iterasi untuk menghasilkan nilai Fungsi Z yang optimum.

4.2.10 Pengolahan Data dan Analisa Kasus 10

Tabel 4.36 Tabel Permasalahan Kasus 10

	X1	X2	X3	X4	Rhs
Z	2	3	1	5	
1	0,1	0,06666667	0,083333	0,066667	1
2	0,083333	0,1	0,111111	0,090909	1
3	0,125	0,16666667	0,2	0,111111	1
4	0,083333	0,0625	0,071429	0,05	1
5	0,045455	0,06666667	0,083333	0,033333	1
6	-0,33333	0,25	-0,5	0,2	-1
a	-1	0	0	0	0
b	0	-1	0	0	0
c	0	0	-1	0	0
d	0	0	0	-1	0

Pada Kasus 10, terdapat 6 Fungsi Kendala dengan 4 variabel. Dengan menggunakan Algoritma Simpleks Original, kasus ini diselesaikan dengan jumlah iterasi sebanyak 9 kali. Nilai optimum Fungsi Tujuan Z yang dihasilkan adalah 26,82 dengan nilai variabel $X_1=5,01$ $X_2=0$ $X_3=0$ dan $X_4=3,36$.

Pada kasus ini, akan ditunjukkan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus menghasilkan jumlah iterasi yang lebih sedikit dibandingkan jumlah iterasi yang diperlukan oleh Algoritma Simpleks Original. Berikut adalah langkah proses penyelesaian kasus 10 dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus. Pada tabel 4.37 di bawah, ditunjukkan urutan Fungsi Kendala berdasarkan nilai Cosinus terhadap Fungsi Tujuan Z.

Tabel 4.37 Tabel Perhitungan Nilai Cosinus Kasus 10

	X1	X2	X3	X4	Rhs	A.Z	IAI	cos
Z	2	3	1	5			6,244998	
2	0,083333	0,1	0,111111	0,090909	1	1,032323	0,19379	0,853006
1	0,1	0,06666667	0,083333	0,066667	1	0,816667	0,160728	0,813621
4	0,083333	0,0625	0,071429	0,05	1	0,675595	0,135841	0,796387
3	0,125	0,16666667	0,2	0,111111	1	1,505556	0,309432	0,77911
5	0,045455	0,06666667	0,083333	0,033333	1	0,540909	0,12069	0,717662
6	-0,33333	0,25	-0,5	0,2	-1	0,583333	0,68089	0,137185
d	0	0	0	-1	0	-5	1	0,800641
b	0	-1	0	0	0	-3	1	0,480384
a	-1	0	0	0	0	-2	1	0,320256
c	0	0	-1	0	0	-1	1	0,160128

Proses pencarian BFS memerlukan 2 kali iterasi. Fungsi Kendala yang digunakan pada setiap iterasi adalah Fungsi Kendala (2, 1, 4, 3) dan (1, 4, 3, 6).

Nilai variabel $X_1=19,76$ $X_2=14,26$ $X_3=-10,4$ dan $X_4=-15,9$ serta menghasilkan nilai Fungsi Tujuan Z sebesar $-7,58$. Selanjutnya lakukan tes optimalitas dengan membuat Tabel Simpleks ketika BFS tercapai terlebih dahulu. Tabel Simpleks tersebut ditampilkan pada Tabel 4.38 berikut.

Tabel 4.38 Tabel Simpleks Kasus 10 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	Rhs
z	1	0	0	-0	0	210,9	-223	9,54	5,124	0	0	-7,58
1	0	1	0	0	0	-17,2	47	-10,3	-0,28	0	0	19,76
4	0	0	1	0	0	-59	73,52	0,626	0,926	0	0	14,26
3	0	0	0	1	0	14,6	-34,8	8,758	-1,03	0	0	-10,4
6	0	0	0	0	1	81,52	-101	3,91	0,788	0	0	-15,9
2	0	0	0	0	0	-1,7	1,736	-0,53	-0,03	1	0	0,528
5	0	6E-17	0	0	0	0,779	-0,79	-0,43	0,011	0	1	0,548

Tabel Simpleks yang dibentuk ketika BFS tercapai belum optimum. Oleh karena itu perlu dilakukan iterasi Simpleks untuk menghasilkan nilai Fungsi Tujuan yang optimum. Proses tersebut ditunjukkan pada Tabel 4.39.

Tabel 4.39 Tabel Simpleks Proses Iterasi Kasus 10

	Z	X1	X2	X3	X4	S1	S2	S3	S4	S5	S6	Rhs	
z	1	0	0	-0	0	211	-223	9,54	5,12	0	0	-7,6	
1	0	1	0	0	0	-17	47	-10	-0,3	0	0	19,8	0,42
4	0	0	1	0	0	-59	73,5	0,63	0,93	0	0	14,3	0,19
3	0	0	0	1	0	14,6	-35	8,76	-1	0	0	-10	0,3
6	0	0	0	0	1	81,5	-101	3,91	0,79	0	0	-16	0,16
2	0	0	0	0	0	-1,7	1,74	-0,5	-0	1	0	0,53	0,3
5	0	0	0	0	0	0,78	-0,8	-0,4	0,01	0	1	0,55	-0,7
z	1	0	0	-0	-2,2	30,2	0	0,87	3,38	0	0	27,7	
1	0	1	0	0	0,47	20,9	0	-8,5	0,09	0	0	12,3	26,4
4	0	0	1	0	0,73	0,65	0	3,48	1,5	0	0	2,64	3,6
3	0	0	0	1	-0,3	-14	0	7,4	-1,3	0	0	-4,9	14,2
6	0	0	0	0	-0	-0,8	1	-0	-0	0	0	0,16	-16
2	0	0	0	0	0,02	-0,3	0	-0,5	-0	1	0	0,25	14,7
5	0	0	0	0	-0	0,14	0	-0,5	0	0	1	0,67	-86

	-	-	6,41	-0	-	-2,4	-	1,26	-13	-	-	
z	1	0	3,03	-0	0	32,1	0	11,4	7,93	0	0	35,6
1	0	1	-0,6	0	0	20,5	0	-11	-0,9	0	0	10,6
4	0	0	1,37	0	1	0,89	0	4,77	2,05	0	0	3,6
3	0	0	0,47	1	0	-13	0	9,05	-0,6	0	0	-3,7
6	0	0	0,01	0	0	-0,8	1	0,01	0,01	0	0	0,19
2	0	0	-0	0	0	-0,3	0	-0,5	-0	1	0	0,19
5	0	0	0,01	0	0	0,15	0	-0,4	0,02	0	1	0,7

z	1	0	4,18	2,42	0	0	0	33,3	6,49	0	0	26,8	Z
1	0	1	0,09	1,54	0	0	0	3,22	-1,8	0	0	5,01	X1
4	0	0	1,4	0,07	1	0	0	5,37	2,01	0	0	3,36	X4
3	0	0	-0	-0,1	0	1	0	-0,7	0,04	0	0	0,27	
6	0	0	-0	-0,1	0	0	1	-0,5	0,05	0	0	0,41	
2	0	0	-0	-0	0	0	0	-0,8	-0	1	0	0,28	
5	0	0	0,02	0,01	0	0	0	-0,3	0,01	0	1	0,66	

Pada proses iterasi ketiga, dilakukan proses Simpleks Dual karena pada iterasi ini nilai solusi yang dihasilkan bernilai negatif. Oleh karena itu perlu dilakukan iterasi Simpleks Dual untuk mengatasi masalah ini. Hasil yang diperoleh setelah iterasi Simpleks Dual ini sudah mencapai nilai yang optimum. Nilai Fungsi Tujuan Z yang dihasilkan adalah 26,8 dengan nilai variabel $X_1=5,01$ $X_2=0$ $X_3=0$ dan $X_4=3,36$.

Total iterasi yang dilakukan algoritma ini adalah 5 iterasi dengan rincian 2 iterasi pada proses mendapatkan BFS dan 3 iterasi pada proses Simpleks. Namun, apabila kasus ini diselesaikan dengan menggunakan Algoritma Simpleks Original, diperlukan iterasi sebanyak 9 kali untuk mendapatkan nilai optimum Fungsi Tujuan yang sama.

4.2.11 Pengolahan Data dan Analisa Kasus 11

Tabel 4.40 Tabel Permasalahan Kasus 11

	X1	X2	X3	X4	X5	Rhs
Z	1	1	2	5	2	
1	30,4	12	16,5	30	16,3	2
2	60	-15,1	-20	-25,6	40	5
3	48,7	24	80,2	40	96,5	8
4	18,7	15	18,7	30	18,7	3
a	-1	0	0	0	0	0
b	0	-1	0	0	0	0
c	0	0	-1	0	0	0
d	0	0	0	-1	0	0
e	0	0	0	0	-1	0

Kasus ini menampilkan permasalahan LP dengan jumlah variabel sebanyak 5 dan jumlah Fungsi Kendala sebanyak 4. Metode Simpleks Original menyelesaikan permasalahan ini dengan jumlah iterasi sebanyak 9 kali. Dan pada contoh kasus ini, akan ditunjukkan bahwa Algoritma Eliminasi Gaussian – Kriteria Cosinus dapat menyelesaikan permasalahan ini dengan jumlah iterasi yang jauh lebih sedikit.

Tabel 4.41 Tabel Perhitungan Nilai Cosinus Kasus 11

	X1	X2	X3	X4	X5	Rhs	A.Z	IAI	cos
Z	1	1	2	5	2			5,916	
4	18,7	15	18,7	30	18,7	3	258,5	46,63	0,937
1	30,4	12	16,5	30	16,3	2	258	50,06	0,871
3	48,7	24	80,2	40	96,5	8	626,1	142,4	0,743
2	60	-15,1	-20	-25,6	40	5	-43,1	80,52	0,09
d	0	0	0	-1	0	0	-5	1	0,845
c	0	0	-1	0	0	0	-2	1	0,338
e	0	0	0	0	-1	0	-2	1	0,338
a	-1	0	0	0	0	0	-1	1	0,169
b	0	-1	0	0	0	0	-1	1	0,169

Langkah pertama yang dilakukan untuk menyelesaikan kasus 11 dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus adalah mengurutkan semua Fungsi Kendala. setelah itu pilih Fungsi Kendala dengan nilai Cosinus terbesar sebanyak jumlah variabel untuk dijadikan input pada proses Eliminasi Gaussian. Namun, karena jumlah Fungsi Kendala pada kasus ini berjumlah lebih sedikit dibandingkan jumlah variabel, maka Eliminasi Gaussian tidak dapat dilakukan. Agar proses ini dapat berjalan, masukkan Fungsi Kendala

Dasar (semua variabel bernilai positif) ke dalam perhitungan. Kemudian urutkan pula Fungsi Kendala ini berdasarkan nilai Cosinus terhadap Fungsi Tujuan Z.

Pada proses ini, diperlukan 4 kali iterasi untuk mendapatkan nilai BFS. Pada masing-masing iterasi, Fungsi Kendala yang digunakan pada proses Eliminasi Gaussian adalah (4, 1, 3, 2, d) (3, 2, d, c, a) (2, d, c, a, b) dan (c, a, b, 1, 3). Nilai BFS yang dicapai adalah $X_1=0$ $X_2=0$ $X_3=0$ $X_4=0,03$ dan $X_4=0,07$ dan menghasilkan nilai pada Fungsi Tujuan Z sebesar 0,83. Lalu lakukan tes optimalitas pada kondisi ini dengan sebelumnya membuat Tabel Simpleks terlebih dahulu. Tabel Simpleks tersebut dapat dilihat pada Tabel 4.42.

Tabel 4.42 Tabel Simpleks Kasus 11 Ketika BFS Tercapai

	Z	X1	X2	X3	X4	X5	S1	S2	S3	S4	S5	S6	S7	S8	S9	Rhs
Z	1	0	0	0	0	0	0,19	3,99	0,92	0,18	-0	0	0	0	0	0,28
c	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0
a	0	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0
b	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0,13	0,95	0,34	0,04	-0	0	0	0	0	0,03
3	0	0	0	0	0	1	0,78	0,11	0,11	-0	0,01	0	0	0	0	0,07
4	0	0	0	0	0	0	0,33	-12	2,74	-1	-0	1	0	0	0	0,83
2	0	0	0	0	0	0	-48	80,1	-11	1,81	-0,7	0	1	0	0	2,86
d	0	0	-0	-0	0	0	0,13	0,95	0,34	0,04	-0	0	0	1	0	0,03
e	0	0	0	0	0	0	0,78	0,11	0,11	-0	0,01	0	0	0	1	0,07

Pada Tabel 4.42, dapat dilihat bahwa kondisi ini belum mencapai nilai Fungsi Tujuan yang optimum. Oleh karena itu perlu dilakukan proses Metode Simpleks untuk menghasilkan nilai Fungsi Tujuan yang optimum. Proses Metode Simpleks hingga dihasilkan nilai Fungsi Tujuan yang optimum dapat dilihat pada Tabel

Tabel 4.43 Tabel Simpleks Proses Iterasi Kasus 11

	Z	X1	X2	X3	X4	X5	S1	S2	S3	S4	S5	S6	S7	S8	S9	Rhs
Z	1	0	0	0	0	0	0,19	3,99	0,92	0,18	-0	0	0	0	0	0,28
c	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0
a	0	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0
b	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0,13	0,95	0,34	0,04	-0	0	0	0	0	0,03
3	0	0	0	0	0	1	0,78	0,11	0,11	-0	0,01	0	0	0	0	0,07
4	0	0	0	0	0	0	0,33	-12	2,74	-1	-0	1	0	0	0	0,83
2	0	0	0	0	0	0	-48	80,1	-11	1,81	-0,7	0	1	0	0	2,86
d	0	0	-0	-0	0	0	0,13	0,95	0,34	0,04	-0	0	0	1	0	0,03
e	0	0	0	0	0	0	0,78	0,11	0,11	-0	0,01	0	0	0	1	0,07

Z	1	0	0	0	0	0	0,75	4,07	1	0,17	0	0	0	0	0,72	0,33	Z	
c	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	X1
a	0	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	X2
b	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	X3
1	0	0	0	0	1	0	0,55	1,01	0,4	0,03	0	0	0	0	0,54	0,07	X4	
3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	-1	0	X5	
4	0	0	0	0	0	0	2,2	-12	3	-1	0	1	0	0	2,4	1		
2	0	0	0	0	0	0	-5,9	85,9	-4,9	0,85	0	0	1	0	53,9	6,71		
d	0	0	-0	-0	0	0	0,55	1,01	0,4	0,03	0	0	0	1	0,54	0,07		
e	0	0	0	0	0	0	58,2	8,17	8	-1,3	1	0	0	0	74,8	5,33		

Pada Tabel 4.43 di atas, nilai Fungsi Tujuan yang optimum dihasilkan dengan melakukan iterasi 1 kali pada kondisi BFS. Nilai optimum yang dicapai Fungsi Tujuan Z adalah 0,33 dengan nilai variabel $X_1=0$ $X_2=0$ $X_3=0$ $X_4=0,07$ dan $X_5=0$. Jadi, untuk menyelesaikan kasus ini, Algoritma Eliminasi Gaussian – Kriteria Cosinus hanya memerlukan total iterasi sebanyak 5 kali, dengan rincian 4 kali iterasi pada proses untuk mendapatkan nilai BFS dan 1 kali iterasi pada proses Simpleks.

4.2.12 Pengolahan Data dan Analisa Kasus 12

Tabel 4.44 Tabel Permasalahan Kasus 12

	X1	X2	X3	X4	X5	X6	Rhs
Z	2	0,5	4	1	3	5	
1	20	6	32,3	-6	24	60,5	2
2	32	-6	-32	-4	48,3	160,6	4
3	20	7,5	100	5	90,7	50	5
4	24	15	72,7	12	54	120,6	6
a	-1	0	0	0	0	0	0
b	0	-1	0	0	0	0	0
c	0	0	-1	0	0	0	0
d	0	0	0	-1	0	0	0
e	0	0	0	0	-1	0	0
f	0	0	0	0	0	-1	0

Kasus 12 akan menunjukkan proses penyelesaian permasalahan LP dengan jumlah Fungsi Kendala lebih sedikit dibandingkan jumlah variabel. Kasus ini telah ditunjukkan pada Kasus 11, namun pada Kasus 12, jumlah variabel lebih banyak dibandingkan pada Kasus 11. Pada Kasus 12 ini terdapat 4 Fungsi Kendala dengan 6 variabel. Dengan menggunakan Metode Simpleks Original, Kasus ini diselesaikan dengan jumlah iterasi sebanyak 5 iterasi.

Tabel 4.45 Tabel Perhitungan Nilai Cosinus Kasus 12

	X1	X2	X3	X4	X5	X6	Rhs
Z	2	0,5	4	1	3	5	
1	20	6	32,3	-6	24	60,5	2
2	32	-6	-32	-4	48,3	161	4
3	20	7,5	100	5	90,7	50	5
4	24	15	72,7	12	54	121	6
a	-1	0	0	0	0	0	0
b	0	-1	0	0	0	0	0
c	0	0	-1	0	0	0	0
d	0	0	0	-1	0	0	0
e	0	0	0	0	-1	0	0
f	0	0	0	0	0	-1	0

Untuk menyelesaikan permasalahan LP ini dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus, langkah pertama yang dilakukan adalah membuat urutan Fungsi Kendala. Karena pada kasus ini jumlah Fungsi Kendala lebih sedikit dibandingkan jumlah variabel, maka diperlukan tambahan fungsi agar proses Elminasi Gaussian dapat dilakukan. Tambahan

Fungsi ini didapatkan dengan menambahkan Fungsi Kendala Dasar (setiap variabel bernilai positif).

Nilai BFS diperoleh dengan melakukan 1 kali iterasi. Fungsi Kendala yang digunakan adalah Fungsi Kendala 4, 1, 3, 2, f, dan c. Nilai BFS yang didapatkan adalah $X_1=0,12$ $X_2=0,02$ $X_3=0$ $X_4=0,16$ $X_5=0,02$ dan $X_6=0$ dengan nilai Fungsi Tujuan Z yang dihasilkan solusi ini adalah sebesar 0,46. Setelah itu dilakukan proses tes optimalitas untuk melihat apakah nilai ini merupakan nilai optimum yang dapat dicapai oleh Fungsi Tujuan Z. Tes ini dilakukan dengan membuat Tabel Simpleks pada kondisi BFS terlebih dahulu.

Tabel 4.46 Tabel Simpleks Kasus 12 Ketikas BFS Tercapai

	Z	X1	X2	X3	X4	X5	X6	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Rhs
Z	1	0	0	0	0	0	0	0,09	-0	-0	0,05	8,57	-3,8	0	0	0	0	0,46
4	0	1	-0	0	0	0	0	0,03	0,01	-0	0,03	6,43	-2,1	0	0	0	0	0,12
1	0	0	1	0	0	0	0	0,01	0,07	0	-0,1	-2,9	5,41	0	0	0	0	0,02
3	0	0	0	1	0	0	0	0	0	0	0	0	-1	0	0	0	0	0
2	0	0	0	0	1	0	0	0,06	-0,1	-0	0,03	4,86	-2	0	0	0	0	0,16
f	0	0	0	0	0	1	0	-0	-0	0,02	-0	-0,9	1,23	0	0	0	0	0,02
c	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0
e	0	0	0	0	-0	0	-0	-0	-0	0,02	-0	-0,9	1,23	1	0	0	0	0,02
a	0	0	-0	-0	-0	0	0	0,03	0,01	-0	0,03	6,43	-2,1	0	1	0	0	0,12
d	0	0	-0	0	0	0	0	0,06	-0,1	-0	0,03	4,86	-2	0	0	1	0	0,16
b	0	0	0	0	0	0	-0	0,01	0,07	0	-0,1	-2,9	5,41	0	0	0	1	0,02

Tabel 4.46 di atas adalah Tabel Simpleks yang dihasilkan saat BFS tercapai. Pada tabel tersebut dapat dilihat bahwa solusi yang dihasilkan BFS belum optimum. Kemudian dilanjutkan proses Simpleks untuk mendapatkan nilai optimum Fungsi Tujuan Z. Proses ini ditunjukkan oleh Tabel 4.47.

Tabel 4.47 Tabel Simpleks Proses Iterasi Kasus 12

	Z	X1	X2	X3	X4	X5	X6	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Rhs		
Z	1	0	0	0	0	0	0	0,09	-0	-0	0,05	8,57	-3,8	0	0	0	0	0,46		
4	0	1	0	0	0	0	0	0,03	0,01	-0	0,03	6,43	-2,1	0	0	0	0	0,12	-0,1	
1	0	0	1	0	0	0	0	0,01	0,07	0	-0,1	-2,9	5,41	0	0	0	0	0,02	0	
3	0	0	0	1	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	
2	0	0	0	0	1	0	0	0,06	-0,1	-0	0,03	4,86	-2	0	0	0	0	0,16	-0,1	
f	0	0	0	0	0	1	0	-0	-0	0,02	-0	-0,9	1,23	0	0	0	0	0,02	0,02	
c	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	-	
e	0	0	0	0	-0	0	-0	-0	-0	0,02	-0	-0,9	1,23	1	0	0	0	0,02	0,02	
a	0	0	-0	-0	-0	0	0	0,03	0,01	-0	0,03	6,43	-2,1	0	1	0	0	0,12	-0,1	
d	0	0	-0	0	0	0	0	0,06	-0,1	-0	0,03	4,86	-2	0	0	1	0	0,16	-0,1	
b	0	0	0	0	0	0	-0	0,01	0,07	0	-0,1	-2,9	5,41	0	0	0	1	0,02	0	
Z	1	0	0	0	0	0	-0	0,1	0	-0	0,01	6,5	0	0	0	0	0	0,71	0,47	
4	0	1	0	0	0	0	-0	0,03	0,03	-0	0,01	5,3	0	0	0	0	0	0,39	0,13	-4,1
1	0	-0	1	-0	-0	0	0	0	0	-0	0	0	0	0	0	0	0	-1	-0	8
3	0	0	0	1	0	0	-0	0	0,01	0	-0	-0,5	0	0	0	0	0	0,18	0	6,06
2	0	0	0	0	1	0	-0	0,06	-0,1	-0	0,01	3,75	0	0	0	0	0	0,38	0,16	-6,4
f	0	-0	0	-0	-0	1	0	-0	-0	0,02	0,01	-0,2	0	0	0	0	0	-0,2	0,02	0,81
c	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	0	-
e	0	-0	0	0	-0	0	0	-0	-0	0,02	0,01	-0,2	0	1	0	0	0	-0,2	0,02	0,81
a	0	0	-0	-0	-0	0	0	0,03	0,03	-0	0,01	5,3	0	0	1	0	0	0,39	0,13	-4,1
d	0	0	-0	0	0	0	0	0,06	-0,1	-0	0,01	3,75	0	0	0	1	0	0,38	0,16	-6,4
b	0	0	0	0	0	0	-0	0	0,01	0	-0	-0,5	1	0	0	0	0	0,18	0	6,06
Z	1	0	0	0	0	0	-0	0,08	-0	0	0,02	6,15	0	1,56	0	0	0	0,36	0,49	
4	0	1	0	0	-0	0	-0	0,01	0	0	0,02	4,92	0	1,65	0	0	0	0,01	0,15	58
1	0	-0	1	-0	-0	0	0	-0	-0	0	0	-0	0	0	0	0	0	-1	-0	7,51
3	0	0	-0	1	0	0	-0	0	0,01	0	-0	-0,5	0	-0	0	0	0	0,19	0	0,2
2	0	0	0	0	1	0	-0	0,05	-0,1	0	0,02	3,44	0	1,36	0	0	0	0,07	0,18	-2,1
f	0	-0	-0	-0	0	1	0	0	0	0	0	0	0	-1	0	0	0	0	0	-
c	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	0	-
e	0	-0	0	0	-0	0	0	-0,7	-1	1	0,52	-12	0	53,6	0	0	0	-12	0,81	-0,8
a	0	0	-0	-0	-0	0	0	0,01	0	0	0,02	4,92	0	1,65	1	0	0	0,01	0,15	58
d	0	0	-0	0	-0	0	0	0,05	-0,1	0	0,02	3,44	0	1,36	0	1	0	0,07	0,18	-2,1
b	0	0	-0	0	0	0	-0	0	0,01	0	-0	-0,5	1	-0	0	0	0	0,19	0	0,2
Z	1	0	0	0	0	0	-0	0,08	0	0	0	5,05	2,06	1,5	0	0	0	0,75	0,5	Z
4	0	1	0	0	-0	0	-0	0,01	0	0	0,03	5,02	-0,2	1,66	0	0	0	-0	0,15	X1
1	0	-0	1	-0	-0	0	0	-0	0	0	-0	-0	0	0	0	0	0	-1	-0	X2
3	0	0	0	1	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	X3
2	0	0	0	0	1	0	-0	0,07	0	0	-0,1	0,01	6,45	1,19	0	0	0	1,3	0,2	X4
f	0	-0	-0	-0	0	1	0	0	0	0	0	0	0	-1	0	0	0	0	0	X5
c	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	0	0	X6
e	0	0	0	0	-0	0	-0	-0,5	0	1	-0,3	-50	71,7	51,6	0	0	0	1,5	1	
a	0	0	-0	-0	-0	0	0	0,01	0	0	0,03	5,02	-0,2	1,66	1	0	0	-0	0,15	
d	0	0	-0	0	0	0	-0	0,07	0	0	-0,1	0,01	6,45	1,19	0	1	0	1,3	0,2	
b	0	0	-0	0	0	0	-0	0,23	1	0	-0,8	-40	74,9	-2	0	0	0	14,3	0,2	

Nilai optimum Fungsi Tujuan Z dihasilkan dengan melakukan iterasi Simpleks dari kondisi BFS sebanyak 3 iterasi. Nilai optimum Fungsi Tujuan Z adalah sebesar 0,5 dengan nilai variabel X1=0,15 X2=0 X3=0 X4=0,2 X5=0 dan X6=0. Total iterasi yang diperlukan untuk mencapai nilai optimum Fungsi

Tujuan Z dengan menggunakan Algoritma Eliminasi Gaussian – Kriteria Cosinus adalah 4 iterasi. Nilai ini lebih kecil dibandingkan jumlah iterasi yang diperlukan jika menggunakan Metode Simpleks Original (5 iterasi).

4.2.13 Pengolahan Data dan Analisa Kasus 13

Tabel 4.48 Tabel Permasalahan Kasus 13

	X1	X2	X3	Rhs
z	4	2	1	
1	8	4	1	125
2	4	1	0	25
3	1	0	0	5
a	-1	0	0	0
b	0	-1	0	0
c	0	0	-1	0

Kasus 13 merupakan soal Klee-Minty yang didapat pada jurnal yang dibuat oleh Corley et al. Kasus Klee-Minty adalah contoh permasalahan LP dengan kondisi yang buruk. Dikatakan buruk karena Fungsi Kendala sangat terikat satu sama lain, yang menyebabkan penambahan Fungsi Tujuan Z untuk mendapatkan nilai yang optimum relatif kecil. Hal ini menyebabkan jumlah iterasi yang dibutuhkan Metode Simpleks Original untuk menyelesaikan soal Klee-Minty cukup banyak walaupun jumlah kendala dan variabel tergolong sedikit. Hal ini dapat dilihat pada Kasus 13 ini. Metode Simpleks Original membutuhkan jumlah iterasi sebanyak 7 kali iterasi.

Pada kasus ini akan ditunjukkan bagaimana Algoritma Eliminasi Gaussian – Kriteria Cosinus menyelesaikan soal Klee-Minty. Berikut ini adalah proses Algoritma Eliminasi Gaussian – Kriteria Cosinus dalam menyelesaikan soal Klee-Minty.

Tabel 4.49 Tabel Perhitungan Nilai Cosinus Kasus 13

	X1	X2	X3	Rhs	A.Z	IAI	cos
z	4	2	1			4,5826	
1	8	4	1	125	41	9	0,9941
2	4	1	0	25	18	4,1231	0,9527
3	1	0	0	5	4	1	0,8729
a	-1	0	0	0	-4	1	0,8729
b	0	-1	0	0	-2	1	0,4364
c	0	0	-1	0	-1	1	0,2182

Dengan melakukan Eliminasi Gaussian pada Fungsi Kendala 1, 2, dan 3, didapatkan nilai BFS. Kondisi BFS ini memiliki nilai variabel $X_1=5$ $X_2=5$ dan $X_3=65$ dengan nilai Fungsi Tujuan yang dihasilkan adalah 95.

Tabel 4.50 Tabel Simpleks Kasus 13 Ketika BFS Tercapai

	Z	X1	X2	X3	S1	S2	S3	RHS
Z	1	0	0	0	1	-2	4	95
1	0	1	0	0	0	0	1	5
2	0	0	1	0	0	1	-4	5
3	0	0	0	1	1	-4	8	65

Kondisi BFS ini belum mencapai kondisi optimum. Hal ini ditunjukkan oleh Tabel Simpleks pada Tabel 4.50. kemudian lakukan proses Simpleks hingga dihasilkan nilai Fungsi Tujuan Z yang optimum.

Tabel 4.51 Tabel Simplex Proses Iterasi Kasus 13

	Z	X1	X2	X3	S1	S2	S3	RHS
Z	1	0	0	0	1	-2	4	95
1	0	1	0	0	0	0	1	5
2	0	0	1	0	0	1	-4	5
3	0	0	0	1	1	-4	8	65
								5
								-16,25
Z	1	0	2	0	1	0	-4	105
1	0	1	0	0	0	0	1	5
2	0	0	1	0	0	1	-4	5
3	0	0	4	1	1	0	-8	85
								-1,25
								-10,63
Z	1	4	2	0	1	0	0	125
1	0	1	0	0	0	0	1	5
2	0	4	1	0	0	1	0	25
3	0	8	4	1	1	0	0	125
								Z
								X3

Dengan melakukan 2 kali iterasi pada kondisi BFS sudah didapatkan nilai Fungsi Tujuan Z yang optimum. Nilai Fungsi Tujuan Z yang optimum adalah 125 dengan nilai variabel $X_1=0$ $X_2=0$ dan $X_3=125$. Total iterasi yang diperlukan Algoritma Eliminasi Gaussian – Kriteria Cosinus untuk menyelesaikan kasus 13 (soal Klee-Minty) adalah 3 kali iterasi, dengan rincian 1 kali iterasi pada proses mendapatkan BFS dan 2 kali iterasi pada proses Simpleks. Jumlah iterasi ini lebih sedikit dibandingkan jumlah iterasi yang dilakukan Metode Simpleks Original yaitu sebanyak 7 kali iterasi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian mengenai pengembangan metode Simpleks untuk mendapatkan pemecahan dasar yang layak atau *Basic Feasible Solution* (BFS) dengan menggunakan proses Eliminasi Gaussian – Kriteria Cosinus, dapat disimpulkan bahwa algoritma yang dikembangkan dapat menyelesaikan permasalahan LP. Dalam menyelesaikan kasus yang sama, Algoritma ini membutuhkan iterasi yang relatif lebih sedikit dibandingkan dengan jumlah iterasi yang dibutuhkan oleh Metode Simpleks Original. Dari 13 kasus yang dipilih dalam penelitian ini, metode Eliminasi Gaussian – Kriteria Cosinus menyelesaikan 9 kasus (kasus 1, 2, 4, 7, 8, 10, 11, 12 dan 13) dengan jumlah iterasi lebih sedikit dibandingkan metode Simpleks biasa. Dan 3 kasus (kasus 3, 5 dan 9) membutuhkan iterasi dengan jumlah yang sama dengan jumlah iterasi yang dibutuhkan oleh metode Simpleks biasa. Namun pada kasus 6, metode Eliminasi Gaussian – Kriteria Cosinus membutuhkan iterasi lebih banyak dibandingkan metode Simpleks biasa. Hal ini disebabkan karena iterasi yang digunakan untuk menghasilkan BFS cukup banyak (4 iterasi). Namun jika dibandingkan jumlah iterasi pada proses Simpleksnya saja, iterasi yang dibutuhkan lebih sedikit.

Hasil penelitian tersebut juga telah membuktikan mengenai penelitian yang dilakukan oleh Stojkovic dan Stanimirovic yang tertuang dalam jurnal yang berjudul *Two Direct Methods in Linear Programming*. Dalam jurnal tersebut dikatakan bahwa penggunaan *Basic Feasible Solution* (BFS) sebagai *Initial Basis* dalam tabel simpleks secara signifikan akan mengurangi jumlah iterasi yang dibutuhkan untuk menyelesaikan kasus *Linear Programming* (LP). Hal ini dikarenakan penggunaan BFS sebagai *Initial Basis* pada tabel simpleks memotong pencarian titik ekstrim yang menghasilkan nilai optimum pada Fungsi Tujuan; tidak dimulai dari titik pusat O seperti yang dilakukan oleh metode Simpleks biasa.

5.2 Saran

Dari hasil analisa pada bab 4, ada beberapa saran untuk pengembangan lebih lanjut dari Algoritma yang dikembangkan pada penelitian kali ini, yaitu:

- Proses pencarian BFS pada algoritma yang dikembangkan perlu dilakukan suatu metode penyaringan Fungsi Kendala selain penyaringan lewat Kriteria Cosinus. Hal ini dilakukan agar iterasi yang dilakukan pada proses pencarian BFS efisien.
- Perlu dilakukan perbandingan waktu komputasi menggunakan perangkat lunak yang sesuai agar dapat dilihat waktu komputasi mana yang lebih baik.
- Diperlukan suatu metode untuk membuat Tabel Simplex saat kondisi BFS tercapai untuk mempermudah dan mempercepat perhitungan.

DAFTAR REFERENSI

- Anton, H. dan Rorres, C. Aljabar Linear Elementer Versi Aplikasi Edisi Kedelapan Jilid 1. Jakarta: Erlangga.
- Corley, H.W., Rosenberger, J., Yeh, W.C., dan Sung T.K. (2006). *The cosine simplex algorithm*. Int J Adv Manuf Technol. 27, 1047-1050
- Hu, Jian-Feng. (2007). *A note on "an improved initial basis for the simplex algorithm"*. Computers & Operations Research. 34, 3397-3401.
- Junior, H.V. dan Lins, M.P.S. (2005). *An improved initial basis for the Simplex algorithm*. Computers & Operations Research. 32, 1983-1993.
- Luh, H. dan Tsaih, R. (2002). *An efficient search direction for linear programming problems*. Computers & Operations Research. 29, 195-203.
- Stojkovic, N.V. dan Stanimirovic, P.S. (2001). *Two direct methods in linear programming*. European Journal of Operational Research. 131, 417-439.
- Taha, H.A. (1996). Riset Operassi Edisi Kelima Jilid 1 (*Operations Research*). Jakarta: Binarupa Aksara.
- Vanderbei, R.J. (2001). *Linear Programming: Foundations and Extensions*. USA: Robert J. Vanderbei.

LAMPIRAN 1

Tabel Simplex Proses Iterasi Kasus 4
 Dengan Menggunakan Metode Simpleks

	Z	X1	X2	X3	S1	S2	S3	S4	RHS	
Z	1	1	1	-20	0	0	0	0	0	
1	0	1	-1	20	1	0	0	0	20	1
2	0	-1	1	1	0	1	0	0	4	4
3	0	1	-1	1	0	0	1	0	5	5
4	0	-1	-1	0	0	0	0	1	-40	-

Z	1	2	0	0	1	0	0	0	20
1	0	0,05	-0,05	1	0,05	0	0	0	1
2	0	-1,05	1,05	0	-0,05	1	0	0	3
3	0	0,95	-0,95	0	-0,05	0	1	0	4
4	0	-1	-1	0	0	0	0	1	-40
	-	-2	0	-	-	-	-	0	

Z	1	2	0	0	1	0	0	0	20
1	0	0,1	0	1	0,05	0	0	-0,05	3
2	0	-2,1	0	0	-0,05	1	0	1,05	-39
3	0	1,9	0	0	-0,05	0	1	-0,95	42
4	0	1	1	0	0	0	0	-1	40
	-	-0,95238	-	-	-20	0	-	0	

Z	1	2	0	0	1	0	0	0	20
1	0	0	0	1	0,047619	0,047619	0	0	1,142857
2	0	-2	0	0	-0,04762	0,952381	0	1	-37,1429
3	0	0	0	0	-0,09524	0,904762	1	0	6,714286
4	0	-1	1	0	-0,04762	0,952381	0	0	2,857143
	-	-1	-	-	-21	0	-	0	

Z	1	2	0	0	1	0	0	0	20
1	0	0,1	0	1	0,05	0	0	-0,05	3
2	0	-2,1	0	0	-0,05	1	0	1,05	-39
3	0	1,9	0	0	-0,05	0	1	-0,95	42
4	0	1	1	0	0	0	0	-1	40