

**PENYELESAIAN PERMASALAHAN *HYBRID FLOW*
SHOP SCHEDULING UNTUK MEMINIMASI
MAKESPAN DENGAN ALGORITMA *GREY WOLF*
*OPTIMIZER***

SKRIPSI

Diajukan untuk memenuhi salah satu syarat guna mencapai gelar
Sarjana dalam bidang ilmu Teknik Industri

Disusun oleh :

Nama: Agnes Monica

NPM: 2013610167



**PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KATOLIK PARAHYANGAN
BANDUNG
2017**

**FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KATOLIK PARAHYANGAN
BANDUNG**



Nama : Agnes Monica
NPM : 2013610167
Jurusan : Teknik Industri
Judul Skripsi : **PENYELESAIAN PERMASALAHAN *HYBRID FLOW SHOP SCHEDULING* UNTUK MEMINIMASI *MAKESPAN* DENGAN ALGORITMA *GREY WOLF OPTIMIZER***

TANDA PERSETUJUAN SKRIPSI

Bandung, Juli 2017

Ketua Jurusan Teknik Industri

(Dr. Carles Sitompul, S.T., M.T., M.I.M.)

Pembimbing

(Alfian, S.T., M.T.)



Program Studi Teknik Industri
Fakultas Teknologi Industri
Universitas Katolik Parahyangan



Pernyataan Tidak Mencontek atau Melakukan Tindakan Plagiat

Saya yang bertanda tangan dibawah ini,

Nama : Agnes Monica
NPM : 2013610167

dengan ini menyatakan bahwa Skripsi dengan judul:

**"PENYELESAIAN PERMASALAHAN HYBRID FLOW SHOP SCHEDULING
UNTUK MEMINIMASI MAKESPAN DENGAN ALGORITMA GREY WOLF
OPTIMIZER"**

adalah hasil pekerjaan saya dan seluruh ide, pendapat atau materi dari sumber lain telah dikutip dengan cara penulisan referensi yang sesuai.

Pernyataan ini saya buat dengan sebenar-benarnya dan jika pernyataan ini tidak sesuai dengan kenyataan, maka saya bersedia menanggung sanksi yang akan dikenakan kepada saya.

Bandung, Juli 2017

Agnes Monica
NPM: 2013610167

ABSTRAK

Hybrid Flow Shop Scheduling (HFS) adalah sebuah sistem penjadwalan yang terdiri atas berbagai tahapan (*stage*) proses produksi dan material/pekerjaan (*job*) yang akan di proses dalam aliran yang searah dimana terdapat minimum satu tahapan yang memiliki mesin identik yang disusun secara paralel. Permasalahan HFS bersifat *non polynomial-hard* (NP-Hard). Hal ini dikarenakan semakin banyak material, jumlah tahapan yang akan dikerjakan, dan jumlah mesin, maka semakin banyak kombinasi urutan pekerjaan sehingga waktu komputasi yang dibutuhkan akan semakin lama. Permasalahan HFS dapat diselesaikan dengan menggunakan metode eksak namun metode tersebut dinilai sudah tidak efisien dari segi waktu komputasi. Oleh karena itu, perlunya metode lain untuk dapat menyelesaikan permasalahan HFS. Salah satu metode yang dapat menyelesaikan permasalahan dengan *np-hard* adalah dengan menggunakan metode metaheuristik.

Pada penelitian ini, permasalahan HFS akan diselesaikan dengan algoritma *Grey Wolf Optimizer* (GWO). GWO merupakan sebuah algoritma metaheuristik yang meniru perilaku serigala dalam proses perburuan mangsa. Pada algoritma GWO, solusi terbaik dianalogikan sebagai mangsa yang akan diburu oleh serigala. Serigala akan berpindah posisi untuk mencari mangsa dengan berpatokan pada serigala hierarkis yakni serigala alpha, beta, dan delta.

Algoritma GWO diimplementasikan ke dalam 21 kasus yang terdiri atas 16 *easy problems* dan 5 *hard problems* dengan menggunakan 8 kombinasi parameter dan 20 kali replikasi. Algoritma GWO memiliki 3 parameter yang diuji dimana masing-masing parameter memiliki 2 *level*. Eksperimen dilakukan dengan menggunakan teknik *Full Factorial Completely Randomized Design* (CRD) untuk mencari pengaruh masing-masing parameter, yakni parameter pergerakan mangsa (n_a), parameter jumlah serigala (n), dan jumlah iterasi (T). Pada penelitian ini juga dilakukan perbandingan performansi yang dihasilkan algoritma GWO dengan algoritma pembandingan yaitu algoritma *improved discrete artificial bee colony* (iDABC). Hasil implementasi menunjukkan performansi algoritma GWO sama baiknya dengan iDABC pada *easy problems*. Akan tetapi, algoritma GWO belum dapat menghasilkan solusi yang lebih baik dibandingkan dengan iDABC pada *hard problems* kecuali pada kasus ke-19.

ABSTRACT

Hybrid Flow Shop Scheduling (HFS) is a scheduling system consisting of various stages of production processes and materials/jobs that will be processed in a unidirectional stream where there is a minimum of one stage that has identical machines arranged in parallel. The problem of HFS is non polynomial-hard (NP-Hard). It is because the more jobs, the number of stages to be done, and the number of machines, the more the combination of job sequences so the computation time required will be longer. HFS problems can be solved by using exact methods but they are considered inefficient in solving HFS problems in terms of computation time. Therefore, the other methods is needed to solve HFS problems. One method that can solve np-hard problems is to use metaheuristic methods.

In this research, Grey Wolf Optimizer (GWO) will be used to solve HFS scheduling. GWO is a metaheuristic algorithm that mimics the hunting behavior of grey wolves. The best solution will be modeled as prey. Grey Wolves will move to hunt the prey according to the movement of the alpha, beta, and delta wolves.

The GWO algorithm is implemented into 21 cases, consists of 16 easy problems and 5 hard problems using 8 combination parameters and 20 replication times. The GWO algorithm has 3 parameters tested where each parameter has 2 levels. The experiment was performed by using Full Factorial Completely Randomized Design (CRD) technique to find the influence of each parameter, ie the parameter of prey movement (na), wolf number parameter (n), and number of iteration (T). This research is also done by comparing the performance generated by GWO algorithm with comparative algorithm that is improved discrete artificial bee colony (IDABC) algorithm. The result of the implementation shows that the performance of GWO algorithm is as well as IDABC in solving easy problems problems. However, the GWO algorithm has not been able to produce a better solution than IDABC to the hard problems except for case 19.

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat dan rahmat yang diberikan kepada penulis dalam menyelesaikan penelitian dan juga tugas akhir laporan skripsi dengan judul Penyelesaian Permasalahan *Hybrid Flow Shop Scheduling* untuk Meminimasi *Makespan* dengan Algoritma *Grey Wolf Optimizer*. Penulis mengucapkan banyak terima kasih kepada orang tua penulis yang selalu memberikan dukungan moral dan motivasi dalam menyelesaikan laporan skripsi ini. Dalam proses pengerjaan dan penyelesaian laporan skripsi ini, penulis juga banyak mendapat dukungan dan bantuan baik secara langsung maupun tidak langsung dari banyak pihak. Oleh karena itu, tidak lupa penulis juga mengucapkan banyak terima kasih kepada:

1. Bapak Alfian, S.T., M.T. selaku dosen pembimbing skripsi yang selalu memberikan dukungan dan motivasi dengan penuh kesabaran dalam proses pengerjaan laporan skripsi ini.
2. Bapak Sani Susanto, Ph.D. selaku dosen penguji proposal skripsi yang telah memberikan masukan dan perbaikan terhadap proposal skripsi.
3. Ibu Titi Iswari, S.T., M.Sc., M.B.A. selaku dosen penguji proposal skripsi yang telah memberikan masukan dan perbaikan terhadap proposal skripsi.
4. Bapak Daniel Siswanto, S.T., M.T. selaku dosen penguji skripsi yang telah memberikan masukan dan perbaikan terhadap proposal skripsi.
5. Ibu Titi Iswari, S.T., M.Sc., M.B.A. selaku dosen penguji skripsi yang telah memberikan masukan dan perbaikan terhadap proposal skripsi.
6. Bapak Marihot Nainggolan, S.T., M.T., M.S. selaku dosen kepala laboratorium Komputer FTI yang telah memberikan izin atas penggunaan ruangan Laboratorium Komputer untuk proses pengambilan data implementasi algoritma pada skripsi ini.
7. Ricky Nugraha Tendi yang selalu memberikan dukungan dan bersedia membantu penulis dalam pengerjaan skripsi ini.
8. Maria Natalie Saujana dan Floretta Meilianty Tano selalu teman baik penulis yang telah memberikan dukungan dan motivasi selama pengerjaan skripsi ini selesai.

9. Andre Yanto Gunawan, S.T. dan Ricky Soetono, S.T. selaku sesama penulis laporan skripsi mengenai algoritma *Grey Wolf Optimizer* yang bersedia memberikan bantuan terhadap kelancaran laporan skripsi ini.
10. Deva, Adrianus, Felix, dan Arnold selaku sesama penulis laporan skripsi algoritma yang telah memberikan masukan dan motivasi dalam pembuatan laporan skripsi ini.
11. Teman-teman komsel gereja yang telah memberikan dukungan, motivasi, serta doa terhadap kelancaran skripsi ini.
12. Teman-teman rekan asisten Simulasi Sistem dan Pemrograman Komputer.
13. Teman-teman kelas A.
14. Teman-teman Teknik Industri angkatan 2013.
15. Pihak-pihak lainnya yang telah membantu penulis baik secara langsung maupun tidak langsung yang tidak dapat disebutkan satu per satu.

Tak ada gading yang tak retak, begitu pula sama halnya dengan laporan skripsi ini yang masih memiliki ketidaksempurnaan. Penulis dengan senang hati menerima kritik dan saran terhadap laporan skripsi ini sehingga dapat menjadikan laporan skripsi yang lebih baik. Akhir kata, semoga laporan skripsi ini memiliki manfaat bagi pembaca dan pihak-pihak lainnya.

Bandung, 11 Juli 2017

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR TABEL	vii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	I-1
I.1 Latar Belakang Masalah.....	I-1
I.2 Identifikasi dan Rumusan Masalah.....	I-6
I.3 Batasan Masalah dan Asumsi	I-13
I.4 Tujuan Penelitian	I-14
I.5 Manfaat Penelitian	I-14
I.6 Metodologi Penelitan.....	I-15
I.7 Sistematika Penulisan	I-17
BAB II TINJAUAN PUSTAKA	II-1
II.1 Definisi Penjadwalan	II-1
II.2 Klasifikasi Penjadwalan	II-4
II.3 Penjadwalan <i>Hybrid Flow Shop</i>	II-5
II.4 Model Matematis untuk Penjadwalan <i>Hybrid Flow Shop</i>	II-6
II.5 Model Eksak dalam Penjadwalan <i>Hybrid Flow Shop</i>	II-9
II.6 Metode Heuristik dalam Penjadwalan <i>Hybrid Flow Shop</i>	II-10
II.7 Metaheuristik.....	II-10
II.8 Analogi Model <i>Grey Wolf Optimizer</i>	II-12
II.9 Persamaan Matematis <i>GWO Algorithm</i>	II-15
II.9.1 Model Matematis Kegiatan Mengitari Mangsa	II-15
II.9.2 Model Matematis Kegiatan Berburu.....	II-15
II.9.3 Model Matematis Kegiatan Menyerang	

Mangsa (Eksplorasi)	II-19
II.9.4 Model Matematis Kegiatan Mencari	
Mangsa (Eksplorasi).....	II-19
II.10 <i>Design of Experiment</i>	II-22

BAB III PERANCANGAN ALGORITMAIII-1

III.1 Proses <i>Encoding</i> dan Penugasan Pekerjaan dalam Pembentukan Posisi Awal Serigala	III-1
III.2 Hierarki Serigala dalam Permasalahan <i>Hybrid Flow Shop</i>	III-2
III.3 Perancangan Algoritma <i>Grey Wolf Optimizer</i>	III-3
III.3.1 Notasi Algoritma <i>Grey Wolf Optimizer</i>	III-3
III.3.2 Algoritma Utama <i>Grey Wolf Optimizer</i>	III-5
III.3.3 Inisialisasi Posisi Awal Serigala (Algoritma A).....	III-8
III.3.4 Menghitung Nilai <i>Fitness</i> Serigala (Algoritma B)	III-14
III.3.5 Mengubah Hierarki Serigala (Algoritma C)	III-21
III.3.6 Perhitungan Parameter Serigala (Algoritma D)	III-24
III.3.7 Algoritma Perhitungan Parameter C.....	III-28
III.3.8 Algoritma Perhitungan Parameter A.....	III-30
III.3.9 Algoritma Perhitungan Nilai D	III-31
III.3.10 Mengubah Posisi Serigala untuk Iterasi Selanjutnya (Algoritma E).....	III-32
III.4 Verifikasi dan Validasi Algoritma <i>Grey Wolf Optimizer</i> untuk Permasalahan HFS	II-40
III.4.1 Verifikasi Algoritma <i>Grey Wolf Optimizer</i>	III-40
III.4.2 Validasi Algoritma <i>Grey Wolf Optimizer</i>	III-63

BAB IV IMPLEMENTASI ALGORITMAIV-1

IV.1 Verifikasi dan Validasi Program.....	IV-1
IV.1.1 Verifikasi Program <i>Grey Wolf Optimizer</i>	IV-2
IV.1.2 Validasi Program <i>Grey Wolf Optimizer</i>	IV-9
IV.2 Penentuan Parameter <i>Grey Wolf Optimizer</i>	IV-14
IV.3 Rekapitulasi Data Kasus Hipotetik Permasalahan HFS.....	IV-18
IV.4 Hasil Implementasi Algoritma pada Data Kasus Hipotetik	IV-21
IV.4.1 Hasil Kasus 1	IV-22

IV.4.2	Hasil Kasus 2	IV-22
IV.4.3	Hasil Kasus 3	IV-23
IV.4.4	Hasil Kasus 4	IV-24
IV.4.5	Hasil Kasus 5	IV-25
IV.4.6	Hasil Kasus 6	IV-25
IV.4.7	Hasil Kasus 7	IV-26
IV.4.8	Hasil Kasus 8	IV-27
IV.4.9	Hasil Kasus 9	IV-28
IV.4.10	Hasil Kasus 10	IV-29
IV.4.11	Hasil Kasus 11	IV-29
IV.4.12	Hasil Kasus 12	IV-30
IV.4.13	Hasil Kasus 13	IV-31
IV.4.14	Hasil Kasus 14	IV-32
IV.4.15	Hasil Kasus 15	IV-32
IV.4.16	Hasil Kasus 16	IV-33
IV.4.17	Hasil Kasus 17	IV-34
IV.4.18	Hasil Kasus 18	IV-35
IV.4.19	Hasil Kasus 19	IV-35
IV.4.20	Hasil Kasus 20	IV-36
IV.4.21	Hasil Kasus 21	IV-37
IV.5	Pengujian ANOVA pada Kasus Hipotetik	IV-38
IV.5.1	Hasil Pengujian ANOVA Kasus 3	IV-39
IV.5.2	Hasil Pengujian ANOVA Kasus 4	IV-40
IV.5.3	Hasil Pengujian ANOVA Kasus 5	IV-41
IV.5.4	Hasil Pengujian ANOVA Kasus 7	IV-42
IV.5.5	Hasil Pengujian ANOVA Kasus 8	IV-43
IV.5.6	Hasil Pengujian ANOVA Kasus 9	IV-44
IV.5.7	Hasil Pengujian ANOVA Kasus 11	IV-45
IV.5.8	Hasil Pengujian ANOVA Kasus 12	IV-46
IV.5.9	Hasil Pengujian ANOVA Kasus 14	IV-47
IV.5.10	Hasil Pengujian ANOVA Kasus 16	IV-48

IV.5.11 Hasil Pengujian ANOVA Kasus 17	IV-49
IV.5.12 Hasil Pengujian ANOVA Kasus 18	IV-50
IV.5.13 Hasil Pengujian ANOVA Kasus 19	IV-51
IV.5.14 Hasil Pengujian ANOVA Kasus 20	IV-52
IV.5.15 Hasil Pengujian ANOVA Kasus 21	IV-53
IV.6 Rekapitulasi Hasil Pengujian ANOVA.....	IV-55
IV.7 Perbandingan Algoritma GWO dengan Algoritma Pembanding	IV-56

BAB V ANALISIS V-1

V.1 Analisis Metode <i>Encoding</i> pada Algoritma GWO.....	V-1
V.2 Analisis Proses Perpindahan Serigala.....	V-4
V.3 Analisis Pemilihan Parameter Algoritma GWO	V-7
V.4 Analisis Penentuan Nilai Parameter Algoritma GWO.....	V-8
V.5 Analisis Pengaruh Nilai Parameter Algoritma GWO	V-10
V.5.1 Analisis Pengaruh Parameter T terhadap Performansi Algoritma GWO	IV-11
V.5.2 Analisis Pengaruh Parameter n terhadap Performansi Algoritma GWO	IV-12
V.5.3 Analisis Pengaruh Parameter a terhadap Performansi Algoritma GWO	IV-13
V.5.4 Analisis Pengaruh Parameter na dan T terhadap Performansi Algoritma GWO	IV-14
V.5.5 Analisis Pengaruh Parameter na dan n terhadap Performansi Algoritma GWO	IV-15
V.5.6 Analisis Pengaruh Parameter T dan n terhadap Performansi Algoritma GWO	IV-17
V.5.7 Analisis Pengaruh Parameter na, T, dan n terhadap Performansi Algoritma GWO	IV-18
V.6 Analisis Perbandingan Performansi Algoritma GWO dengan Algoritma Pembanding	V-19

BAB VI KESIMPULAN DAN SARANVI-1

VI.1 Kesimpulan.....	VI-1
----------------------	------

VI.2 SaranVI-3

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR RIWAYAT HIDUP

DAFTAR TABEL

Tabel I.1	Waktu Proses untuk 3 <i>Job</i> dan 2 <i>Stage</i>	I-3
Tabel I.2	Persentase Kasus pada Permasalahan Hybrid Flow Shop	I-6
Tabel II.1	Tipe Data untuk <i>Single Factor Experiment</i>	II-25
Tabel III.1	Solusi Awal Permasalahan HFS dengan Model Matematis	III-1
Tabel III.2	Ilustrasi <i>Encoding</i> untuk Posisi Awal Serigala	III-9
Tabel III.3	Posisi Awal Serigala Berdasarkan Hasil Urut Bilangan <i>Random</i>	III-9
Tabel III.4	Matriks Waktu Proses <i>Job</i> I pada <i>Stage</i> j.....	III-41
Tabel III.5	Jumlah Mesin Identik pada <i>Stage</i> ke-j.....	III-41
Tabel IV.1	Kombinasi Parameter yang Diuji pada Kasus Hipotetik.....	IV-18
Tabel IV.2	Keterangan Notasi Kasus Hipotetik untuk <i>Layout</i> Mesin	IV-19
Tabel IV.3	Rekapitulasi Data <i>Layout</i> Mesin Kasus Hipotetik <i>Easy Problems</i>	IV-19
Tabel IV.4	Rekapitulasi Data <i>Layout</i> Mesin Kasus Hipotetik (<i>Hard Problems</i>).....	IV-20
Tabel IV.5	Rekapitulasi Urutan Pengambilan Data Secara Acak.....	IV-20
Tabel IV.6	Rekapitulasi Hasil Implementasi Kasus 1	IV-22
Tabel IV.7	Rekapitulasi Hasil Implementasi Kasus 2.....	IV-23
Tabel IV.8	Rekapitulasi Hasil Implementasi Kasus 3.....	IV-23
Tabel IV.9	Rekapitulasi Hasil Implementasi Kasus 4.....	IV-24
Tabel IV.10	Rekapitulasi Hasil Implementasi Kasus 5.....	IV-25
Tabel IV.11	Rekapitulasi Hasil Implementasi Kasus 6.....	IV-26
Tabel IV.12	Rekapitulasi Hasil Implementasi Kasus 7	IV-26
Tabel IV.13	Rekapitulasi Hasil Implementasi Kasus 8.....	IV-27
Tabel IV.14	Rekapitulasi Hasil Implementasi Kasus 9.....	IV-28
Tabel IV.15	Rekapitulasi Hasil Implementasi Kasus 10.....	IV-29
Tabel IV.16	Rekapitulasi Hasil Implementasi Kasus 11	IV-30
Tabel IV.17	Rekapitulasi Hasil Implementasi Kasus 12.....	IV-30
Tabel IV.18	Rekapitulasi Hasil Implementasi Kasus 13.....	IV-31
Tabel IV.19	Rekapitulasi Hasil Implementasi Kasus 14.....	IV-32

Tabel IV.20	Rekapitulasi Hasil Implementasi Kasus 15	IV-33
Tabel IV.21	Rekapitulasi Hasil Implementasi Kasus 16	IV-33
Tabel IV.22	Rekapitulasi Hasil Implementasi Kasus 17	IV-34
Tabel IV.23	Rekapitulasi Hasil Implementasi Kasus 18	IV-35
Tabel IV.24	Rekapitulasi Hasil Implementasi Kasus 19	IV-36
Tabel IV.25	Rekapitulasi Hasil Implementasi Kasus 20	IV-36
Tabel IV.26	Rekapitulasi Hasil Implementasi Kasus 21	IV-37
Tabel IV.27	Rekapitulasi Hasil Pengujian ANOVA.....	IV-55
Tabel IV.28	Hasil Perbandingan Kasus Mudah Algoritma GWO dengan Algoritma IDABC	IV-57
Tabel IV.29	Hasil Perbandingan Kasus Susah Algoritma GWO dengan Algoritma IDABC	IV-58
Tabel V.1	Perbedaan Metode <i>Encoding random</i> dan Permutasi	V-2
Tabel V.2	Rekapitulasi Pengaruh Parameter T terhadap Performansi.....	V-11
Tabel V.3	Rekapitulasi Pengaruh Parameter n terhadap Performansi.....	V-12
Tabel V.4	Rekapitulasi Pengaruh Parameter na terhadap Performansi....	V-13
Tabel V.5	Rekapitulasi Pengaruh Interaksi Parameter na dan T terhadap Performansi.....	V-14
Tabel V.6	Rekapitulasi Pengaruh Interaksi Parameter na dan n terhadap Performansi.....	V-16
Tabel V.7	Rekapitulasi Pengaruh Interaksi Parameter T dan n terhadap Performansi.....	V-17
Tabel V.8	Rekapitulasi Pengaruh Interaksi Parameter na, T, dan n terhadap Performansi.....	V-18
Tabel V.9	Rekapitulasi Perhitungan <i>Deviation</i> Kasus Mudah pada Algoritma GWO	V-20
Tabel V.10	Rekapitulasi Perhitungan <i>Deviation</i> Kasus Susah pada Algoritma GWO	V-21

DAFTAR GAMBAR

Gambar I.1	Alternatif 1 Model Penjadwalan pada Contoh Kasus HFS	I-3
Gambar I.2	Alternatif 2 pada Contoh Kasus Penjadwalan HFS	I-4
Gambar I.3	Peningkatan Jumlah Permasalahan <i>Hybrid Flow Shop</i>	I-5
Gambar I.4	Metodologi Penelitian.....	I-16
Gambar II.1	Klasifikasi Penjadwalan dalam Industri	II-4
Gambar II.2	Ilustrasi Pengalokasian <i>Job</i> pada <i>Stage</i>	II-5
Gambar II.3	<i>Gantt Chart</i> untuk Kasus Permasalahan HFS 4 <i>Job</i> dan 3 <i>Stage</i>	II-9
Gambar II.4	Dua Kriteria Pembuatan Metode Metaheuristik	II-11
Gambar II.5	Kompleksitas Permasalahan Optimasi.....	II-11
Gambar II.6	Hierarki Serigala	II-13
Gambar II.7	Tahapan Berburu Serigala	II-14
Gambar II.8	Pergerakan Serigala dalam GWO.....	II-17
Gambar II.9	Posisi 3D Serigala.....	II-17
Gambar II.10	Posisi 2 Dimensi Perburuan Serigala terhadap Mangsa.....	II-18
Gambar II.11	Posisi 2D Serigala.....	II-18
Gambar II.12	Ilustrasi GWO : (a) eksploitasi; (b) eksplorasi.....	II-29
Gambar II.13	<i>Pseudo Code</i> GWO	II-20
Gambar II.14	Formulasi untuk <i>Oneway ANOVA</i>	II-26
Gambar III.1	Contoh Posisi Awal Serigala	III-2
Gambar III.2	Algoritma <i>Grey Wolf Optimizer</i>	III-6
Gambar III.3	Algoritma A	III-11
Gambar III.4	Algoritma B	III-17
Gambar III.5	Algoritma C	III-22
Gambar III.6	Contoh Posisi Serigala pada Iterasi <i>t</i>	III-25
Gambar III.7	Contoh Hasil Perhitungan Nilai <i>D</i>	III-25
Gambar III.8	Contoh Hasil Perhitungan Nilai <i>X</i>	III-26
Gambar III.9	Contoh Hasil Perhitungan Nilai <i>X</i> dan Proses <i>Ranking</i>	III-27
Gambar III.10	Algoritma D	III-28
Gambar III.11	Algoritma Perhitungan Parameter <i>C</i>	III-29

Gambar III.12 Algoritma Perhitungan Parameter A	III-30
Gambar III.13 Algoritma Perhitungan Nilai D	III-31
Gambar III.14 Algoritma E	III-33
Gambar III.15 Validasi Fungsi Tujuan Algoritma	III-63
Gambar III.16 Validasi <i>Constraint</i> Pertama (a) dan Kedua (b) pada Algoritma.....	III-65
Gambar III.17 Validasi <i>Constraint</i> Ketiga pada Algoritma.....	III-66
Gambar III.18 Proses Urutan <i>Job</i> pada <i>Stage</i> Berikutnya.....	III-67
Gambar IV.1 Perancangan Pemrograman Tahap Awal.....	IV-2
Gambar IV.2 Perancangan Pemrograman Algoritma A	IV-3
Gambar IV.3 Perancangan Pemrograman Algoritma B	IV-4
Gambar IV.4 Perancangan Pemrograman Algoritma C	IV-5
Gambar IV.5 Perancangan Pemrograman Pengubahan Nilai a	IV-6
Gambar IV.6 Perancangan Pemrograman Algoritma D.....	IV-6
Gambar IV.7 Perancangan Pemrograman Perhitungan Nilai X dan D.....	IV-8
Gambar IV.8 Perancangan Pemrograman Perhitungan Posisi Serigala Iterasi Selanjutnya.....	IV-8
Gambar IV.9 Contoh Kesalahan dalam Input Jumlah Iterasi	IV-10
Gambar IV.10 Validasi Input Jumlah Iterasi	IV-10
Gambar IV.11 Contoh Kesalahan dalam Input Jumlah Serigala.....	IV-11
Gambar IV.12 Validasi Input Jumlah Serigala	IV-11
Gambar IV.13 Contoh Kesalahan dalam Input Jumlah <i>Job</i>	IV-11
Gambar IV.14 Validasi Input Jumlah <i>Job</i>	IV-12
Gambar IV.15 Contoh Kesalahan dalam Input Jumlah <i>Stage</i>	IV-12
Gambar IV.16 Validasi Input Jumlah <i>Stage</i>	IV-12
Gambar IV.17 Contoh Kesalahan dan Validasi dalam Input Jumlah Mesin	IV-13
Gambar IV.18 Validasi <i>Output</i> Hasil <i>Running</i>	IV-14
Gambar IV.19 Hasil Uji ANOVA Kasus 3	IV-39
Gambar IV.20 Hasil Uji ANOVA Kasus 4	IV-41
Gambar IV.21 Hasil Uji ANOVA Kasus 5	IV-42
Gambar IV.22 Hasil Uji ANOVA Kasus 7	IV-43
Gambar IV.23 Hasil Uji ANOVA Kasus 8	IV-44
Gambar IV.24 Hasil Uji ANOVA Kasus 9	IV-45
Gambar IV.25 Hasil Uji ANOVA Kasus 11	IV-46

Gambar IV.26 Hasil Uji ANOVA Kasus 12	IV-47
Gambar IV.27 Hasil Uji ANOVA Kasus 14	IV-48
Gambar IV.28 Hasil Uji ANOVA Kasus 16	IV-49
Gambar IV.29 Hasil Uji ANOVA Kasus 17	IV-50
Gambar IV.30 Hasil Uji ANOVA Kasus 18	IV-51
Gambar IV.31 Hasil Uji ANOVA Kasus 19	IV-52
Gambar IV.32 Hasil Uji ANOVA Kasus 20	IV-53
Gambar IV.33 Hasil Uji ANOVA Kasus 21	IV-54

DAFTAR LAMPIRAN

LAMPIRAN A DATA WAKTU PROSES KASUS HIPOTETIK.....	A-1
LAMPIRAN B REKAPITULASI PENGUJIAN PARAMETER	B-1
LAMPIRAN C HASIL IMPLEMENTASI ALGORITMA <i>EASY PROBLEMS</i>	C-1
LAMPIRAN D HASIL IMPLEMENTASI ALGORITMA <i>HARD PROBLEMS</i>	D-1
LAMPIRAN E <i>MAIN EFFECT PLOT</i> PARAMETER	E-1

BAB I

PENDAHULUAN

Pada bab ini berisi mengenai dasar dari penelitian yang meliputi latar belakang permasalahan, identifikasi dan perumusan masalah, batasan dan asumsi masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

I.1 Latar Belakang Masalah

Perkembangan industri saat ini semakin cepat bahkan di berbagai bidang dituntut untuk bergerak cepat. Perkembangan tersebut memunculkan adanya tingkat persaingan antarindustri yang semakin berat. Langkah yang menjadi kunci awal dalam menghadapi persaingan yang semakin berat adalah dengan menentukan sistem perusahaan yang baik dan mampu untuk memenuhi *demand* pelanggan dengan ketepatan waktu yang telah disepakati. Perusahaan harus mampu untuk mengimbangi permintaan konsumen dengan kapasitas produksi yang dimiliki oleh perusahaan tersebut. Ketepatan waktu dalam memenuhi permintaan konsumen tidak lepas dari sistem penjadwalan produksi yang benar. Dengan demikian, sistem penjadwalan yang direncanakan oleh perusahaan merupakan hal yang sangat penting dalam mempengaruhi kelancaran produksi dalam memenuhi permintaan konsumen.

Penjadwalan dalam sistem produksi merupakan suatu kegiatan mengalokasikan sejumlah pekerjaan ke sejumlah sumber daya yang ada (Bedworth & Bailey, 1987). Sistem penjadwalan menjadi hal yang sangat penting bagi perusahaan. Hal ini dikarenakan sistem penjadwalan mempengaruhi performansi dan produktivitas perusahaan untuk memenuhi target permintaan dengan kapasitas yang dimiliki. Sistem penjadwalan yang baik seharusnya memiliki tingkat efisiensi yang tinggi untuk mencapai tingkat utilitas sumber daya perusahaan yang tinggi pula. Dengan tingkat utilisasi yang tinggi, *output* produksi yang dihasilkan dapat meningkat. Sehingga, keuntungan yang diperoleh oleh perusahaan akan semakin meningkat.

Perusahaan harus dapat menentukan sistem penjadwalan yang tepat sesuai dengan situasi persaingan dan strategi yang dijalankan oleh suatu perusahaan. Salah satu sistem penjadwalan dalam bidang industri adalah sistem penjadwalan *flow shop*. Sistem ini dapat diterapkan pada variasi produk yang relatif kecil dan diterapkan pada volume produksi yang tinggi dimana produk akan dialokasikan dengan sejumlah mesin yang dikerjakan secara berurutan (Fogarty, D.W., J.H. Blackstone, dan T.R Hoffman, 1991), sehingga semua produk yang akan diproses dalam 1 arah aliran sesuai dengan jumlah mesin yang dialokasikan.

Seiring dengan perkembangan industri, sistem penjadwalan *flowshop* semakin berkembang dan kompleks. *Hybrid Flow Shop* (HFS) merupakan salah satu penjadwalan *flowshop* dimana permasalahan dalam penjadwalannya sedang meningkat pada masa ini. Masalah-masalah yang sedang meningkat merupakan permasalahan pengalokasian *job* yang akan dikerjakan pada sejumlah proses dan mesin yang tersedia. Semakin banyak *job* yang akan dikerjakan mengakibatkan semakin kompleks penanganan dalam pengalokasian *job* pada mesin yang tersedia. Jika sistem pengalokasian tersebut tidak berjalan dengan lancar dan tepat, maka *output* yang dihasilkan tidak dapat memenuhi *demand* tepat waktu. Hal ini mengakibatkan kerugian bagi perusahaan karena performansi produksi menjadi menurun.

HFS merupakan sebuah sistem yang terdiri atas berbagai tahapan (*stage*) proses produksi dan material yang akan diproses dalam aliran yang searah dimana terdapat minimum satu tahap (*stage*) yang memiliki mesin identik yang disusun secara parallel (Uetake T., Tsubone H., dan Ohba M., 1995).

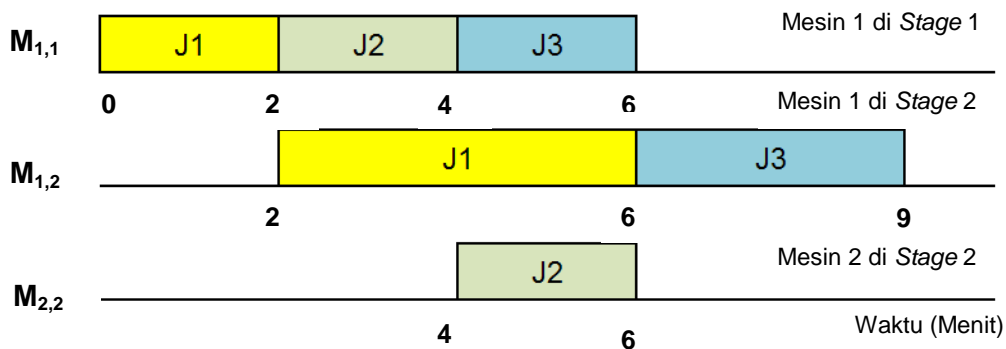
Untuk lebih memahami bagaimana sistem dalam penjadwalan HFS dibuat ilustrasi permasalahan HFS sederhana. Sebuah perusahaan tekstil yang memproduksi pakaian. Perusahaan tersebut harus mengerjakan 3 jenis pakaian atau dapat dikatakan sebagai 3 *job*. Dalam memproduksi pakaian, terdapat beberapa tahapan dalam produksi yang harus dikerjakan secara berurutan yaitu tahap pemotongan (*cutting*) dan tahap penjahitan (*sewing*). Tahapan-tahapan tersebut merupakan *stage* atau proses sehingga dapat dikatakan bahwa perusahaan tersebut memiliki 2 *stage* dalam pembuatan pakaian (*job*). Diketahui bahwa di *stage* 1 terdapat 1 buah mesin sedangkan untuk *stage* 2 terdapat 2 buah mesin. Masing-masing *job* memiliki waktu proses produksi (*processing*

time). Berikut merupakan waktu proses masing-masing *job* untuk masing-masing *stage* (H. Allaoui & A. Artiba, 2004).

Tabel I.1 Waktu Proses untuk 3 *Job* dan 2 *Stage*

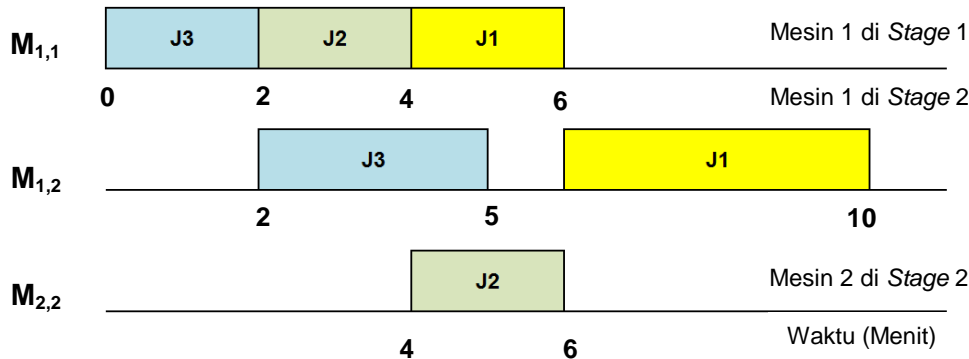
Jenis Job	Waktu proses (menit)	
	Stage 1	Stage 2
Job 1 (J1)	2	4
Job 2 (J2)	2	2
Job 3 (J3)	2	3

Berdasarkan data waktu proses pada Tabel II.1, dapat dilakukan penjadwalan untuk ketiga jenis *job*. Diketahui pada awal proses penjadwalan urutan *job* yang akan dikerjakan untuk *stage* 1 adalah *Job* 1 kemudian *job* 2, dan terakhir adalah *job* 3. Model Penjadwalan tersebut dapat dilihat pada Gambar II.1.



Gambar II.1. Alternatif 1 Model Penjadwalan pada Contoh Kasus HFS

Berdasarkan Gambar I.1, dapat dilihat bahwa dalam penjadwalan HFS menerapkan *First Come First Serve* (FCFS) yang berarti *job* yang memiliki waktu penyelesaian paling kecil di *stage* sebelum akan dijadwalkan terlebih dahulu untuk *stage* berikutnya. Pada model penjadwalan alternatif 1, didapatkan total waktu keseluruhan (*makespan*) untuk menyelesaikan ketiga jenis *job* adalah 9 menit. Hal ini terjadi jika pada *stage* 2, mesin 1 mengerjakan *job* 1 dan *job* 3 sedangkan untuk mesin 2 mengerjakan *job* 2. Namun, terdapat juga alternatif penjadwalan lain jika urutan *job* yang dikerjakan pada *stage* pertama berbeda. Jika urutan *job* yang dikerjakan adalah sebaliknya yaitu *job* 3, kemudian *job* 2, dan *job* 1, maka didapatkan total waktu keseluruhan (*makespan*) yang berbeda pula. Model penjadwalan alternatif kedua pada kasus HFS dapat dilihat pada Gambar I.2.



Gambar I.2. Alternatif 2 pada Contoh Kasus Penjadwalan HFS

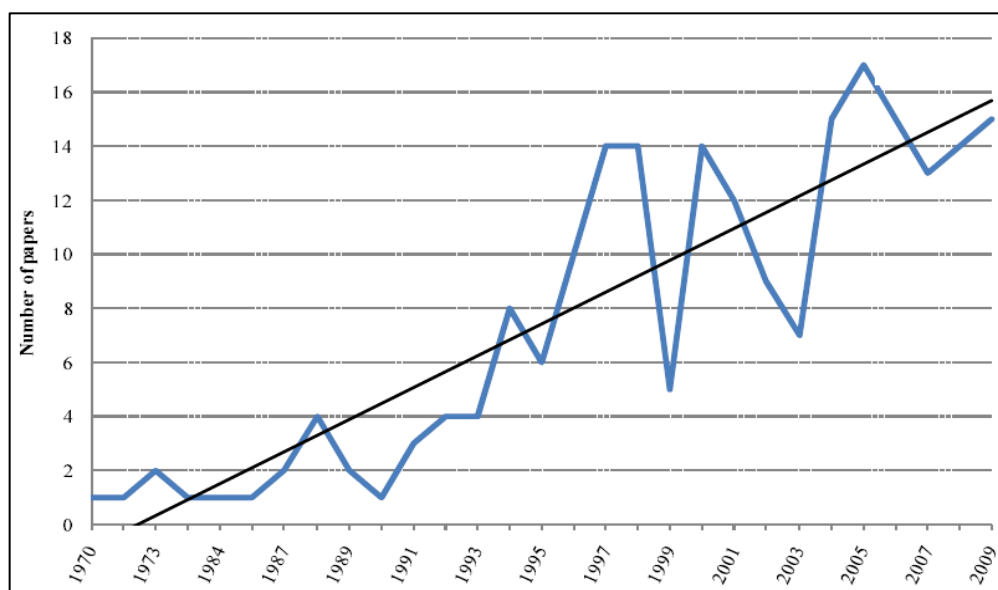
Berdasarkan Gambar I.2, dapat dilihat bahwa *makespan* yang dihasilkan adalah 10 menit jika pada mesin 1 di *stage 2* mengerjakan *job 3* dan *job 1* dan pada mesin 2 mengerjakan *job 2*. Hal ini dikarenakan urutan *job* sangat mempengaruhi total waktu keseluruhan pengerjaan *job*. Perbedaan urutan *job* yang ditentukan untuk *stage* pertama akan menghasilkan total waktu penyelesaian (*makespan*) yang berbeda pula. Sehingga, penentuan urutan *job* harus dilakukan dengan baik dan tepat untuk meningkatkan performansi dalam segi efisiensi waktu dan produktivitas.

Berdasarkan contoh ilustrasi tersebut, dapat dilihat bahwa penjadwalan alternatif 1 lebih baik dibandingkan alternatif 2 karena pada alternatif 1 menghasilkan *makespan* yang lebih kecil jika dibandingkan dengan alternatif 2. Selain itu, jumlah alternatif penjadwalan dapat berjumlah sebanyak $n!m^n$ kemungkinan, dimana n adalah jumlah *job* dan m adalah jumlah mesin di *stage* kedua (Allaoui & Artiba, 2004). Oleh sebab itu, jumlah alternatif dalam menentukan urutan prioritas penjadwalan dapat memiliki banyak kombinasi tergantung pada jumlah *job* dan jumlah mesin yang digunakan.

Sistem penjadwalan dengan HFS banyak diterapkan pada industri manufaktur baja (Rofiq & Santosa, 2012), tekstil (Ruiz & Vazquez-Rodriguez, 2010), elektronik, otomotif, kimia dan kertas. Selain itu, penerapan penjadwalan *hybrid flow shop* juga ditemukan pada bagian produksi dan manufaktur dalam bidang fotografi *film* (Ruiz & Vazquez-Rodriguez, 2010). Salah satu tujuan penting dalam penjadwalan untuk meningkatkan efisiensi dan efektivitas penjadwalan HFS adalah dengan meminimasi *makespan* untuk dapat memenuhi kebutuhan permintaan konsumen tepat waktu. Pada kurun waktu tiga tahun

terakhir, *makespan* telah digunakan sebesar 60% oleh para peneliti (Ruiz & Vazquez-Rodríguez, 2010) sebagai tolak ukur dalam penilaian efisiensi dan efektivitas penjadwalan.

Menurut Ruiz & Vazquez-Rodríguez (2010), diantara 200 jurnal yang diteliti, masalah yang pada umumnya dibahas adalah mengenai permasalahan HFS dengan model kasus yang berbeda-beda. Pada beberapa tahun terakhir, permasalahan penjadwalan HFS mengalami *trend* peningkatan yang dapat dilihat pada Gambar I.3.



Gambar I.3. Peningkatan Jumlah Permasalahan *Hybrid Flow Shop*
(Sumber : Ruiz & Vazquez-Rodríguez, 2010)

Berdasarkan Gambar I.3, terdapat sumbu x sebagai tahun dan sumbu y sebagai jumlah literatur yang membahas permasalahan HFS dan telah dipublikasikan. Berdasarkan gambar tersebut, dapat dilihat bahwa terjadi kenaikan jumlah permasalahan mengenai penjadwalan *hybrid flow shop* terus meningkat pada tahun 1970 hingga tahun 2009 dan memiliki *trend*. Hal ini menunjukkan permasalahan *hybrid flow shop* cenderung akan meningkat terus menerus untuk tahun-tahun berikutnya. Berdasarkan jumlah tingkat permasalahan yang dibahas pada literatur, terdapat persentase tipe kasus pada permasalahan *hybrid flow shop* dengan jumlah *stage* (tahapan) dan jumlah mesin yang digunakan. Persentase mengenai kasus permasalahan pada penjadwalan HFS dapat dilihat pada Tabel I.2 (Ruiz & Vazquez-Rodríguez, 2010).

Tabel I.2. Persentase Kasus pada Permasalahan *Hybrid Flow Shop*

Number of stage	Type of Parallel machines			Total
	Identical	Uniform	Unrelated	
2	25.12	1.86	4.65	31.63
3	4.19	1.4	0	5.59
M	54.41	1.4	6.97	62.78
Total	83.72	4.66	11.62	100

Pada Tabel I.2, dapat dilihat bahwa permasalahan pada penjadwalan *hybrid flow shop* memiliki total persentase sebesar 83,72% untuk kasus HFS dengan mesin identik yang disusun secara paralel, sedangkan untuk mesin *uniform* yang disusun secara paralel adalah sebesar 4,66%, dan 11,62% untuk mesin *unrelated* yang disusun secara paralel. Selain itu, terdapat juga persentase untuk kasus dengan jumlah *stage* (tahapan) yaitu kasus dengan jumlah *stage* sebanyak 2 memiliki persentase sebesar 25,12%, jumlah *stage* sebanyak 3 buah memiliki persentase sebesar 4,19%, dan untuk jumlah *stage* sejumlah *m stage* sebesar 54,41%. Berdasarkan literatur tersebut, dapat disimpulkan bahwa permasalahan *hybrid flow shop* yang paling banyak dibahas adalah pada kasus mesin paralel yang disusun secara identik dengan *m* buah *stage*. Oleh sebab itu, perkembangan masalah HFS terutama pada kasus mesin paralel yang disusun secara identik dengan *m* buah *stage* menjadi latar belakang diangkatnya permasalahan ini untuk diselesaikan dengan lebih baik.

I.2 Identifikasi dan Rumusan Masalah

Sistem penjadwalan sangat mempengaruhi performansi dan tingkat efisiensi hasil produksi dalam suatu perusahaan. Sistem penjadwalan yang baik dan efisien tentunya akan meningkatkan output produk yang dihasilkan oleh perusahaan sehingga keuntungan yang didapatkan dapat semakin meningkat. Oleh sebab itu, tingkat efisiensi dalam perencanaan penjadwalan menjadi faktor penting untuk meningkatkan kelancaran produksi dan efisiensi waktu produksi.

Hybrid Flow Shop (HFS) merupakan permasalahan penjadwalan yang kompleks. Menurut Talbi (2009) dan Rofiq & Santosa (2012), permasalahan HFS merupakan permasalahan NP-Hard. Hal ini dikarenakan semakin banyak *job*, jumlah *stage* (tahapan) yang akan dikerjakan, dan jumlah mesin, maka semakin lama waktu komputasi yang dibutuhkan untuk mendapatkan solusi.

Permasalahan HFS dapat diselesaikan dengan menggunakan metode eksak seperti metode *branch and bound*, *linear programming*, dan *langragian relaxation* (Rofiq & Santosa, 2012). Akan Tetapi, metode tersebut dinilai sudah tidak efisien dalam menyelesaikan permasalahan HFS dari segi hasil dan waktu komputasi. Oleh karena itu, perlunya metode lain untuk dapat menyelesaikan permasalahan HFS.

Salah satu metode yang dapat menyelesaikan permasalahan dengan *np-hard* adalah dengan menggunakan metode metaheuristik (Talbi, 2009). Metode metaheuristik dapat digunakan untuk mendapatkan solusi dalam waktu yang lebih singkat namun tidak menjamin bahwa solusi yang dihasilkan dapat optimal. Beberapa metode metaheuristik yang telah diimplementasikan untuk masalah HFS yakni algoritma *Particle Swarm Optimization* (PSO) (Liao C.J., Tjandradjaja E., dan Chung T.P., 2012), algoritma *hybrid genetic algorithm* (Jun & Park, 2015), algoritma *improved discrete artificial bee colony* (Cui & Gu, 2013), algoritma *immunoglobulin-based artificial immune system* (Chung & Liao, 2013), algoritma *simulated annealing* (Rofiq & Santosa 2012), algoritma *improved cuckoo search* (Marichelvam M.K., Prabaharan, T., dan Yang, X.S, 2012), dan algoritma *immune algorithm approach* (Zandieh M., Ghomy S.M.T.F., dan Hussein S.M.M , 2006).

Berikut merupakan rekapitulasi posisi penelitian yang berkaitan dengan permasalahan HFS. Rekapitulasi posisi penelitian yang berkaitan dengan permasalahan HFS dapat dilihat pada Tabel I.3.

Tabel I.3 Rekapitulasi Posisi Penelitian Permasalahan *Hybrid Flow Shop*

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Celah Penelitian
1.	Hamid Allaoui dan Abdelhakim Altiba	<i>Scheduling Two-stage Hybrid Flow Shop with Availability Constraint</i> (2004)	Penelitian dilakukan pada <i>two stage Hybrid Flow Shop</i> dimana <i>stage</i> pertama terdiri atas 1 mesin dan <i>stage</i> kedua terdiri atas m-mesin. Hasil performansi dibandingkan dengan <i>Branch & Bound</i> untuk <i>easy problems</i> dan LPT heuristic untuk <i>hard problems</i> .	Perlu penelitian lebih lanjut dengan menerapkan algoritma metaheuristik ke dalam permasalahan <i>Hybrid Flow Shop Scheduling</i> yang lebih kompleks.

(lanjut)

Tabel I.3 Rekapitulasi Posisi Penelitian Permasalahan *Hybrid Flow Shop* (Lanjutan)

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Celah Penelitian
2.	Jacques Carlier dan Emmanuel Neron	<i>An Exact Method for Solving The Multi-processor FlowShop.</i> (2000)	Penelitian dilakukan dengan merancang sebuah metode untuk menyelesaikan permasalahan <i>multi processor FlowShop</i> dimana hasil penelitian menunjukkan bahwa metode yang digunakan cukup efisien dan dapat menyelesaikan <i>hard problems</i> .	Diperlukan penelitian lebih lanjut dengan menggunakan metode lain yang memiliki performansi yang lebih baik untuk menyelesaikan permasalahan <i>m-machine flowshop</i> .
3.	Tsui-Ping Chung dan Ching-Jong Liao	<i>An Immunoglobulin-based Artificial Immune System for Solving The Hybrid Flow Shop Problem</i> (2013)	Penelitian dilakukan dengan menggunakan algoritma <i>Immunoglobulin-based Artificial Immune System</i> (IAIS). Hasil performansi IAIS lebih efisien dibandingkan dengan algoritma AIS.	Waktu komputasional meningkat seiring dengan meningkatnya tingkat kompleksitas permasalahan. Diperlukan adanya penelitian lebih lanjut dalam menerapkan IAIS untuk permasalahan <i>single machine, parallel machines</i> , dan <i>flowshop</i> dengan adanya penambahan <i>constraint</i> dan fungsi tujuan objektif lain.
4.	Zhe-Chui dan Xingsheng Gu	<i>An Improved Artificial Bee Colony Algorithm to Minimize The Makespan on Hybrid Flow Shop Problems</i> (2013)	Penelitian dilakukan dengan mengembangkan algoritma <i>Artificial Bee Colony</i> (ABC) menjadi algoritma IDABC. Hasil performansi IDABC lebih unggul, efektif, dan efisien jika dibandingkan dengan algoritma AIS, PSO, dan DABC.	Perlu dilakukan penerapan algoritma pada jenis permasalahan <i>scheduling</i> lain, seperti <i>stochastic scheduling</i> dan <i>multi-objective scheduling Hybrid Flowshop</i> .
5.	Ching-Jong Liao, Evi Tjandradjaja, dan Tsui-Ping Chung	<i>An Approach Using Particle Swarm Optimization and Bottleneck Heuristic to Solve Hybrid Flow Shop Scheduling Problem</i> (2012)	Penelitian dilakukan dengan menggabungkan algoritma PSO dengan SA pada <i>local optimization</i> . Hasil perbandingan PSO-SA lebih baik dibandingkan dengan PSO dan AIS.	Diperlukan adanya penelitian lebih lanjut dengan menyelesaikan jenis permasalahan <i>scheduling</i> lain.

(lanjut)

Tabel I.3 Rekapitulasi Posisi Penelitian Permasalahan *Hybrid Flow Shop* (Lanjutan)

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Celah Penelitian
6.	M.K. Marichelva m, T. Prabaharan , dan X.S. Yang	<i>Improved Cuckoo Search Algorithm for Hybrid Flow Shop Scheduling Problems to Minimize Makespan</i> (2012)	Algoritma ICS memiliki performansi yang lebih baik jika dibandingka dengan GA, ACO, SA, dan PSO.	Perlu dilakukan adanya penelitian lebih lanjut dengan menyelesaikan permasalahan HFS dengan mesin non-identik pada masing-masing <i>stage</i> . Selain itu, dapat dilakukan penelitian dengan menyelesaikan permasalahan <i>scheduling</i> dengan adanya <i>setup time</i> dan <i>transportation time</i> .
7.	Ainur Rofiq dan Budi Santosa	Pengembangan Algoritma <i>Simulated Annealing</i> pada Permasalahan <i>Hybrid Flow Shop Scheduling</i> untuk Minimasi <i>Makespan</i> dan Total <i>Tardiness</i> (2012)	Algoritma SA memiliki performansi yang cukup baik seiring dengan meningkatnya kompleksitas permasalahan.	Hasil performansi algoritma SA tidak dilakukan perbandingan dengan Algoritma lainnya.
8.	Ruben Ruiz dan Jose Antonio Vasquez-Rodriguez	<i>The hybrid flow shop scheduling problem</i> (2010)	Hasil penelitian terhadap survey 200 jurnal disimpulkan bahwa permasalahan <i>Hybrid flowshop scheduling</i> sedang meningkat dan banyak diterapkan pada industri saat ini. Tujuan ukuran performansi yang banyak digunakan adalah untuk meminimasi <i>makespan</i> yaitu sebesar 60%.	Belum terdapat algoritma terbaik yang dapat menyelesaikan permasalahan HFS sehingga perlu dilakukan penelitian lebih lanjut dengan menggunakan algoritma lain dalam permasalahan HFS terutama untuk <i>n stage</i> dan <i>m</i> mesin.
9.	M. Zandieh, S.M.T. Fatemi Ghomi, dan S.M. Moattar Husseini	<i>An Immune Algorithm Approach to Hybrid Flow Shops Scheduling with Sequence Dependent Setup Time</i> (2006)	Algoritma IA memiliki performansi yang lebih baik dibandingkan dengan algoritma GA di beberapa kasus.	Diperlukan adanya penelitian lebih lanjut untuk menyelesaikan permasalahan <i>hybrid flow shop with dependent setup times</i> dan permasalahan <i>permutation flowshop scheduling</i> .

(lanjut)

Tabel I.3 Rekapitulasi Posisi Penelitian Permasalahan *Hybrid Flow Shop* (Lanjutan)

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Celah Penelitian
10.	Sungbum Jun dan Jinwoo Park	<i>A Hybrid Genetic Algorithm for The Hybrid Flow Shop Scheduling Problem with Nighttime Work and Simultaneous Work Constraint : A Case Study form A Transformer Industry. Journal of Experts System with Applications.</i> (2015)	Hasil penelitian dilakukan dengan menerapkan algoritma GA ke permasalahan <i>hybrid flow shop</i> dengan <i>constraint</i> pada studi kasus. Algoritma GA dapat diterapkan pada permasalahan <i>real</i> .	Perlu dilakukan penerapan algoritma pada permasalahan <i>multi-objective hybrid flow shop</i> dengan mengkombinasikan tujuan minimasi <i>makespan dan tardiness</i> .

Salah satu metode metaheuristik baru dikembangkan oleh Mirjalili S., Mirjalili S.M., dan Lewis A. (2014) bernama *Grey Wolf Optimizer*. Metode metaheuristik ini bersifat *particle swarm* dan meniru pergerakan serigala dalam mencari mangsa. Algoritma ini telah berhasil diimplementasikan pada beberapa permasalahan *Capacitated Vehicle Routing Problem* (Korayem L., Khorsid M., Kassem S. S., 2014), *Knapsack Problem* (Wu H. S., F. M. Zhang, Zhan R. J., Wang. S., dan C. Zhang, 2014), *Two-stage Assembly Flowshop Scheduling Problem with Release Time* (Komaki & Kayvanfar, 2015), *Capacitated Vehicle Routing Problem with Time Windows* (Soetono, J. R., 2016), dan *cell manufacturing* (Gunawan, A.Y., 2016).

Berikut merupakan rekapitulasi posisi penelitian yang berkaitan dengan algoritma GWO. Rekapitulasi posisi penelitian yang berkaitan dengan Algoritma GWO dapat dilihat pada Tabel I.4.

Tabel I.4 Rekapitulasi Posisi Penelitian Algoritma GWO

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Kekurangan Penelitian
1.	Andre Yanto Gunawan	Penerapan <i>Grey Wolf Optimizer</i> untuk Menyelesaikan Permasalahan <i>Cell Manufacturing</i> (2016)	Algoritma GWO memiliki performansi yang lebih rendah dibandingkan dengan algoritma <i>Max-Min Ant Grouping Genetic</i> . Namun, memiliki performansi yang lebih tinggi/baik jika dibandingkan dengan algoritma <i>Intelligent Water Drops</i> dan algoritma <i>Firefly</i> .	Perlu dilakukan penelitian lebih lanjut untuk menyelesaikan permasalahan lain, seperti <i>Capacitated Vehicle Routing Problems</i> dan <i>Shortest Path Problems</i> .

(lanjut)

Tabel I.4 Rekapitulasi Posisi Penelitian Algoritma GWO (Lanjutan)

No.	Penulis	Judul (Tahun)	Hasil Penelitian	Kekurangan Penelitian
2.	G.M.Komaki dan Vahid Kayvanfar	<i>Grey Wolf Optimizer Algorithm for the Two-stage Assembly Flowshop Scheduling Problem with Release Time.</i> Journal (2015)	Penelitian dilakukan pada <i>two stage Flow Shop</i> dimana <i>stage</i> pertama terdiri atas <i>m</i> mesin paralel identik dan <i>stage</i> kedua terdiri atas <i>single machine</i> . Hasil performansi GWO lebih baik dibandingkan dengan PSO dan GA.	Diperlukan penelitian lebih lanjut untuk menyelesaikan permasalahan <i>two stage Flow Shop</i> dengan fungsi objektif tujuan lain seperti meminimasi total <i>tardiness</i> dan lain sebagainya. Selain itu, perlu dilakukan penelitian dengan algoritma GWO dengan menyelesaikan kasus permasalahan <i>flowshop scheduling</i> lebih kompleks dengan membandingkan hasil performansi dengan <i>Branch and Bound method</i> ,
3.	L.Korayem, M. khorsid, dan S.S. Kassem	<i>Using Grey Wolf Algorithm to Solve the Capacitated Vehicle Routing Problem</i> (2014)	Penelitian dilakukan dengan mengembangkan <i>K-means</i> dengan GWO dalam menyelesaikan CVRP. Deviasi yang dihasilkan 3.5% dengan algoritma pembanding.	Perlu dilakukan penelitian lebih lanjut untuk menyelesaikan permasalahan VRP dan CVRP dengan variasi yang berbeda.
4.	Johanes Ricky Soetono	Penerapan <i>Grey Wolf Optimizer</i> pada Penyelesaian <i>Capacitated Vehicle Routing Problem with Time Windows</i> (2016)	Performansi algoritma GWO unggul dari algoritma <i>Viral Systems</i> sebanyak 4 dari 6 kasus yang dibandingkan. Sedangkan, perbandingan performansi algoritma GWO dengan BAT tidak dapat dilakukan karena adanya perbedaan interpretasi penyelesaian masalah	Perlu dilakukan penelitian lebih lanjut untuk menerapkan algoritma GWO pada permasalahan lain.

Berdasarkan Tabel I.3 dan I.4, didapatkan kesimpulan terkait dengan celah penelitian bahwa diperlukan adanya penerapan algoritma metaheuristik untuk permasalahan *Hybrid Flowshop Scheduling* yang lebih kompleks dengan permasalahan *n-stage* dan *m*-mesin dimana salah satu algoritma metaheuristik yaitu algoritma *Grey Wolf Optimizer*.

Berdasarkan hasil penelitian yang telah dilakukan oleh Mirjalili S., et al. (2014), algoritma *Grey Wolf Optimizer* dapat bersaing bahkan menghasilkan solusi yang lebih baik ketika dibandingkan dengan beberapa algoritma. Algoritma

Grey Wolf Optimizer sendiri dapat diterapkan pada permasalahan yang memiliki batasan masalah seperti permasalahan nyata maupun permasalahan yang tidak memiliki batasan masalah, seperti permasalahan fungsi teoretis. Menurut Mirjalili S., Mirjalili S.M., dan Lewis A (2014), algoritma *Grey Wolf Optimizer* yang bersifat *particle swarm* memiliki performansi yang sangat baik dalam menyelesaikan permasalahan nyata seperti permasalahan *optical engineering* dan permasalahan teoretis seperti persamaan *unimodal* (permasalahan yang memiliki nilai optimum tunggal).

Beberapa algoritma yang bersifat *particle swarm* telah diterapkan dalam permasalahan penjadwalan *hybrid flow shop*. Beberapa algoritma yang telah diterapkan adalah algoritma *improved discrete artificial bee colony* (Cui & Gu, 2013) dan algoritma *Particle Swarm Optimization* (Liao C.J., Tjandradjaja E., dan Chung T.P., 2012). Pada penelitian ini, algoritma *Grey Wolf Optimizer* akan diterapkan dalam menyelesaikan permasalahan HFS untuk melihat tingkat performansi dalam pencarian solusi. Penerapan algoritma *Grey Wolf Optimizer* dilakukan untuk menyelesaikan permasalahan HFS dikarenakan algoritma *grey wolf* telah diimplementasikan dalam menyelesaikan permasalahan sistem penjadwalan seperti *two-stage Assembly Flowshop Scheduling Problem with Release Time* dan memiliki performansi yang lebih baik dibandingkan dengan algoritma PSO (Allahverdi & Al-Anzi, 2006b) dan algoritma *Cloud Theory based Simulated Annealing* (CSA) (Torabzadeh & Zandieh, 2010) dari segi perhitungan *lower bound* (Komaki & Kayvanfar, 2015). *Lower bound* merupakan nilai minimum yang dapat dicapai secara teoritis dari solusi untuk sebuah kasus. Menurut Komaki & Kayvanfar (2015), algoritma *grey wolf* juga dapat diterapkan dalam menyelesaikan permasalahan kombinatorial seperti penjadwalan dan memiliki rata-rata deviasi (deviasi antara solusi optimal dengan *lower bound*) yang sangat baik jika dibandingkan dengan algoritma lainnya. Dengan demikian, diharapkan algoritma *grey wolf optimizer* juga mampu memberikan performansi yang baik dalam menyelesaikan permasalahan sistem penjadwalan HFS.

Performansi algoritma *Grey Wolf Optimizer* pun sangat mungkin dipengaruhi oleh nilai parameter yang digunakan. Parameter yang diuji dalam permasalahan HFS yaitu jumlah iterasi dan parameter yang terdapat pada algoritma *grey wolf optimizer*. Selain itu, hasil performansi pada algoritma *Grey Wolf Optimizer* perlu dievaluasi untuk melihat tingkat efisiensi dari

performansinya. Untuk mengevaluasi hasil performansi algoritma *Grey Wolf Optimizer* perlu dilakukan *benchmark* dengan hasil performansi pada algoritma yang telah diimplementasikan pada permasalahan HFS. Pada penelitian ini, hasil performansi algoritma *Grey Wolf Optimizer* akan dibandingkan dengan algoritma *improved discrete artificial bee colony* (Cui & Gu, 2013). Dari penelitian yang telah dilakukan, algoritma *improved discrete artificial bee colony* juga bersifat *particle swarm* dan terinspirasi oleh tingkah laku lebah dalam mencari makanannya. Hasil dari penelitian tersebut menunjukkan bahwa algoritma *improved discrete artificial bee colony* memiliki hasil yang lebih efisien dari segi waktu komputasi dan dapat menghasilkan solusi yang lebih baik dibandingkan dengan beberapa algoritma seperti algoritma *Particle Swarm Optimization* dan algoritma *Artificial Immune System*. Oleh sebab itu, akan sangat baik untuk mengetahui sejauh mana performansi rancangan algoritma *Grey Wolf Optimizer* dibandingkan dengan algoritma *improved discrete artificial bee colony*.

Dari uraian yang telah disampaikan maka permasalahan dalam penelitian ini dapat dirumuskan sebagai berikut.

1. Bagaimana penerapan algoritma *Grey Wolf Optimizer* untuk minimasi *makespan* dalam menyelesaikan permasalahan penjadwalan *Hybrid Flow Shop*?
2. Apakah parameter-parameter algoritma *Grey Wolf Optimizer* berpengaruh signifikan terhadap performansi algoritma dalam menyelesaikan permasalahan *Hybrid Flow Shop* untuk minimasi *makespan*?
3. Bagaimana hasil performansi algoritma *Grey Wolf Optimizer* dibandingkan dengan algoritma *improved discrete artificial bee colony* (Cui & Gu, 2013) dalam menyelesaikan permasalahan penjadwalan HFS untuk minimasi *makespan*?

I.3 Batasan Masalah dan Asumsi

Pembatasan masalah pada penelitian penerapan algoritma *Grey Wolf Optimizer* pada permasalahan penjadwalan HFS bertujuan agar penelitian ini dapat dilakukan dengan lebih mudah dan lebih terfokus. Batasan masalah yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Implementasi algoritma disesuaikan dengan kasus yang digunakan oleh

benchmark.

2. Data waktu proses dan *layout* mesin yang digunakan merupakan data deterministik dan diketahui di awal.
3. Mesin yang tersusun secara parallel adalah mesin yang identik.
Selain menggunakan batasan masalah, penelitian ini juga menggunakan beberapa asumsi. Asumsi-asumsi yang digunakan adalah sebagai berikut.
 1. *Job* yang akan dikerjakan saling bebas dan diproses pada waktu (t) sama dengan nol.
 2. Waktu perpindahan pada *job* dan waktu *setup* mesin termasuk dalam waktu proses (*processing time*).
 3. Tidak terdapat gangguan pada saat *job* diproses seperti mesin *break down* pada saat proses berjalan.
 4. Mesin yang akan bekerja selalu tersedia dan dapat dijalankan secara kontinu pada saat memproses sebuah *job*.

I.4 Tujuan Penelitian

Tujuan dalam penelitian yang dilakukan berdasarkan identifikasi dan perumusan masalah adalah sebagai berikut.

1. Menerapkan algoritma *Grey Wolf Optimizer* dalam menyelesaikan permasalahan penjadwalan HFS untuk minimasi *makespan*.
2. Mengetahui ada tidaknya signifikan pengaruh parameter terhadap performansi *Grey Wolf Optimizer* dalam menyelesaikan permasalahan penjadwalan HFS untuk minimasi *makespan*.
3. Membandingkan hasil performansi algoritma *improved discrete artificial bee colony* (Cui & Gu, 2013) dalam menyelesaikan permasalahan penjadwalan HFS untuk minimasi *makespan*.

I.5 Manfaat Penelitian

Manfaat penelitian yang ingin dicapai dari penelitian ini adalah sebagai berikut.

1. Menerapkan algoritma *Grey Wolf Optimizer* dalam menyelesaikan permasalahan penjadwalan *hybrid flow shop* untuk minimasi *makespan*.
2. Menambah penggunaan algoritma *Grey Wolf Optimizer* dalam penelitian dalam metode metaheuristik.

3. Meningkatkan wawasan pembaca berhubungan dengan algoritma *Grey Wolf Optimizer* terutama dalam menyelesaikan permasalahan *hybrid flow shop* untuk minimasi *makespan*.

I.6 Metodologi Penelitian

Dalam suatu penelitian terdapat metodologi penelitian yang digunakan. Tahapan metodologi yang akan digunakan dalam penelitian ini yang dapat dilihat pada Gambar I.4.

1. Studi Literatur

Tahapan ini berisi pengumpulan sumber-sumber literatur yang berhubungan dengan topik permasalahan yang akan dibahas. Sumber literatur yang digunakan berhubungan dengan permasalahan penjadwalan *hybrid flow shop* (HFS) dan mengenai algoritma *Grey Wolf Optimizer*

2. Identifikasi dan Perumusan Masalah

Tahapan ini berisi permasalahan yang muncul dan akan diselesaikan dalam penelitian ini. Di akhir tahap ini, akan dirumuskan masalah-masalah yang akan dibahas berkaitan dengan permasalahan *hybrid flow shop* dalam penelitian ini.

3. Pembatasan Masalah dan Asumsi

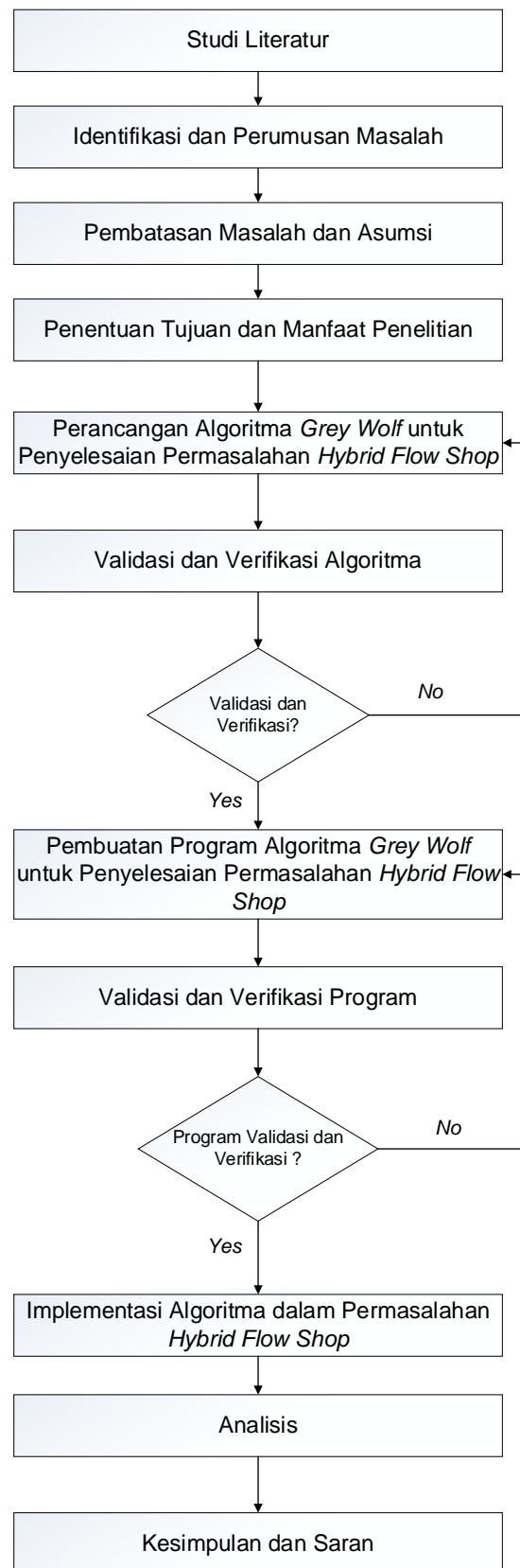
Tahapan ini berisi mengenai pembatasan masalah yang akan dibahas pada penelitian ini. Pembatasan masalah dilakukan agar ruang lingkup dari penelitian ini tidak terlalu luas. Penentuan asumsi digunakan untuk menyederhanakan penelitian ini.

4. Penentuan Tujuan dan Manfaat Penelitian

Tahapan ini berisi perumusan tujuan dan manfaat penelitian. Perumusan tujuan dan manfaat penelitian diperlukan agar penelitian yang dilakukan memiliki tujuan yang jelas.

5. Perancangan Algoritma *Grey Wolf Optimizer* untuk Permasalahan *Hybrid Flow Shop*

Tahapan ini berisi proses perancangan algoritma *Grey Wolf Optimizer* dalam menyelesaikan permasalahan penjadwalan *hybrid flow shop* (HFS). Setelah proses perancangan selesai, dilakukan perancangan program sesuai dengan algoritma GWO dalam permasalahan HFS.



Gambar I.4. Metodologi Penelitian

6. Validasi dan verifikasi algoritma

Tahapan ini berisi proses validasi dan verifikasi hasil rancangan algoritma dalam penyelesaian permasalahan *Hybrid Flow Shop*. Verifikasi dilakukan untuk memastikan bahwa tahapan-tahapan algoritma yang dibuat sesuai dengan algoritma GWO dan permasalahan HFS. Sedangkan validasi dilakukan untuk memastikan apakah hasil perhitungan algoritma dalam menyelesaikan permasalahan HFS sudah benar.

7. Pembuatan Program Algoritma *Grey Wolf Optimizer* dalam Penyelesaian Permasalahan *Hybrid Flow Shop*

Pada tahapan ini, dibuat perancangan program dari algoritma *Grey Wolf Optimizer* sesuai dengan perancangan algoritma pada tahapan sebelumnya untuk menyelesaikan permasalahan penjadwalan *hybrid flow shop* (HFS).

8. Validasi dan Verifikasi Program

Tahapan ini berisi proses verifikasi hasil rancangan program. Verifikasi dilakukan untuk memastikan bahwa program yang dirancang telah sesuai dengan rancangan algoritma dalam menyelesaikan permasalahan HFS. Sedangkan validasi dilakukan untuk memastikan bahwa program yang dirancang dapat dijalankan dengan baik.

9. Implementasi Algoritma dalam Permasalahan *Hybrid Flow Shop*

Tahapan ini berisi penggunaan program untuk menyelesaikan *benchmark problem* untuk mendapatkan hasil. Data yang digunakan merupakan data hipotetik yang terdapat pada persoalan penjadwalan HFS pada algoritma *improved discrete artificial bee colony* yang digunakan sebagai *benchmark*. Setelah dilakukan implementasi, maka dilakukan analisis pada tahap berikutnya.

10. Analisis

Tahapan ini berisi analisis dari implementasi algoritma *Grey Wolf Optimizer* dalam permasalahan penjadwalan *hybrid flow shop*. Selain itu, analisis ini akan berisi perbandingan performansi *Grey Wolf Optimizer* dengan algoritma lain yang digunakan untuk menyelesaikan permasalahan yang sama.

11. Kesimpulan dan Saran

Tahapan ini akan berisi kesimpulan dan saran dari hasil penelitian yang dapat digunakan untuk penelitian selanjutnya. Kesimpulan berisi rangkuman dari hasil penelitian sedangkan saran merupakan rekomendasi untuk penelitian selanjutnya.

I.7 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi dasar dari penelitian yang meliputi latar belakang permasalahan, identifikasi dan perumusan masalah, batasan dan asumsi yang digunakan dalam penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan laporan penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori yang akan digunakan sebagai acuan dalam penelitian ini. Teori-teori yang digunakan berasal dari sumber-sumber terkait penelitian yang akan dilakukan dan sebagai dasar pengolahan data. Literatur yang digunakan meliputi definisi penjadwalan, klasifikasi penjadwalan, penjadwalan *Hybrid Flow Shop*, mathematical model untuk *Hybrid Flow Shop*, metode eksak dalam penjadwalan *Hybrid Flow Shop*, penjelasan mengenai metaheuristik, algoritma *Grey Wolf Optimizer*, dan *design of experiment*.

BAB III PERANCANGAN ALGORITMA

Bab ini berisi perancangan dan penerapan algoritma *Grey Wolf Optimizer* dalam menyelesaikan permasalahan *Hybrid Flow Shop Scheduling*. Bab ini berisi penjelasan mengenai proses *encoding* dan penugasan dalam pembentukan posisi awal serigala, proses perancangan algoritma, dan proses verifikasi & validasi algoritma.

BAB IV IMPLEMENTASI ALGORITMA

Bab ini berisi verifikasi dan validasi program, penentuan parameter algoritma, kasus hipotetik yang akan diuji, hasil implementasi algoritma, dan perbandingan performansi algoritma dengan algoritma pembanding. Perbandingan algoritma dilakukan dengan membandingkan nilai performansi hasil algoritma dengan algoritma pembanding yaitu algoritma *improved discrete artificial bee colony*.

BAB V ANALISIS

Bab ini berisi analisis dari penelitian yang telah disusun. Analisis yang dilakukan meliputi analisis metode *encoding* pada algoritma GWO, analisis

perpindahan posisi serigala dalam algoritma GWO, analisis pemilihan parameter dalam pengujian, analisis pengaruh parameter GWO, analisis performansi algoritma GWO berdasarkan perbandingan dengan algoritma IDABC.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran yang didapatkan dari penelitian. Kesimpulan yang didapatkan bertujuan untuk menjawab rumusan masalah yang terdapat pada bab I. Sedangkan saran diberikan untuk penelitian-penelitian berikutnya yang berkaitan dengan algoritma GWO.