

ABSTRAK

Penjadwalan merupakan rencana pengaturan urutan kerja serta pengalokasian sumber baik waktu maupun fasilitas untuk setiap operasi yang harus diselesaikan. Secara garis besarnya, penjadwalan adalah bagian yang strategis dari proses perencanaan dan pengorganisasian waktu dan sumber daya. Konsep dari penjadwalan umum digunakan dalam perencanaan produksi, penjadwalan mesin, dan juga kasus yang dijadikan fokus pada penelitian ini, yakni pembuatan jadwal pelajaran untuk sekolah. Agar aktivitas belajar mengajar dapat dilaksanakan dengan baik dan teratur, sebuah jadwal pelajaran yang bebas bentrok akan sangat dibutuhkan. Semakin besarnya sebuah sekolah, maka semakin banyak pula jumlah kelas, jumlah guru, dan jumlah murid yang harus diajar untuk dijadikan pertimbangan dalam pembuatan sebuah jadwal pelajaran. Dengan kondisi tersebut, pembuatan sebuah jadwal pelajaran yang bebas bentrok akan memakan waktu yang lama karena kompleksitasnya yang lebih tinggi. Permasalahan inilah yang menjadi ide pembuatan Aplikasi Penjadwalan Pelajaran Sekolah Menengah Pertama. Aplikasi ini menggunakan sebuah algoritma optimasi yang bernama *Artificial Bee Colony*. *Artificial Bee Colony* adalah algoritma yang masih tergolong baru. Dalam pengimplementasiannya, algoritma ini memodelkan tingkah laku cerdas lebah madu dalam pencarian sumber makanan/sumber nektar, yakni eksplorasi, evaluasi, seleksi, dan eksploitasi, berdasarkan komunikasi dan aktivitas antar agen lebahnya. Eksplorasi adalah tahap di mana solusi atau untuk kasus ini adalah jadwal pelajaran dibangkitkan, evaluasi adalah tahap di mana tiap-tiap jadwal pelajaran dievaluasi kualitas atau *fitnessnya*, seleksi adalah tahap penyaringan solusi berdasarkan *fitness* masing-masing jadwal yang didapatkan dari evaluasi, sedangkan eksploitasi adalah tahap pendayagunaan jadwal pelajaran yang terpilih. Dari pengujian yang dilakukan, didapatkan hasil bahwa eksploitasi berbasis metode *Random* memiliki waktu penyelesaian eksekusi yang lebih cepat dibandingkan eksploitasi berdasarkan metode *Sequential*. Sedangkan pengujian dengan metode *Sequential* mampu menghasilkan jadwal pelajaran dengan nilai *fitness* yang sama dengan eksploitasi berdasarkan metode *Random* namun dengan jumlah generasi yang lebih sedikit.

Kata kunci: Jadwal pelajaran, algoritma *Artificial Bee Colony*, *fitness*.

ABSTRACT

Scheduling is a planning for work order with allocation of time and resources to finish each assigned operations. Basically, scheduling is a strategic element from planning process that maintains the use of time and resources. The concept of scheduling itself is usually used for work planning, machine scheduling, and one case which serves as the main focus of this research, is school timetabling. In order for teaching and learning activity to be executed neatly and properly, a school timetable or lesson schedule that is accurate and possessing as minimal flaws as possible is highly necessary. Along with the growth of the school, the increase number of classes, teachers, and students need to be taken into consideration when making a proper school timetable. With those constraints, making a good school timetable will take more time due to its higher complexity. This highlighted problem is the main idea behind the construction of this Scheduling Application for a Junior High School. The application utilizes an optimization algorithm called the Artificial Bee Colony. Artificial Bee Colony itself is considered still brand-new to-date. In the implementation, the algorithm represents the clever behaviors of honey bees during their attempts to find favorable food sources, consisted by exploration, evaluation, selection, and exploitation, through communications and activities between each active agent. Exploration is an operator which solutions or school timetables for the case, are generated, evaluation is an operator which all of the solutions are being evaluated to earn the fitness, selection is an operator which purpose is to filter all of the timetables according to its fitness, and exploitation is an operator which is used to describe the process of improving the quality of selected timetables. According to the research's results, it's proven that the exploitation based from Random method is consuming lesser time than the exploitation based from Sequential method. On the other hand, exploitation based from Sequential method is capable to produce a timetable that's as good as the one made by the exploitation based from Random method, but with less generations or loops needed.

Keywords: School timetable, Artificial Bee Colony algorithm, fitness.

DAFTAR ISI

PRAKATA	i
ABSTRAK.....	ii
ABSTRACT	iii
DAFTAR ISI	iv
DAFTAR GAMBAR.....	viii
DAFTAR TABEL	x
DAFTAR SIMBOL	xi
DAFTAR KODE PROGRAM	xii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	3
1.5 Sistematika Pembahasan.....	3
BAB II	5
LANDASAN TEORI	5
2.1 <i>Swarm Intelligence</i>	5
2.2 <i>Algoritma Artificial Bee Colony</i>	6
2.3 <i>Random Class</i>	10
BAB III.....	11
ANALISIS DAN DESAIN.....	11
3.1 Analisis	11
3.2 Gambaran Keseluruhan Sistem	12
3.2.1 Rancangan Algoritma ABC.....	12

3.2.1.1	Pemodelan Solusi	12
3.2.1.2	<i>Constraints</i>	14
3.2.1.2.1	<i>Hard Constraints</i>	14
3.2.1.2.2	<i>Soft Constraints</i>	14
3.2.1.3	Inisialisasi Populasi/ <i>Scout Bees Phase</i>	14
3.2.1.4	Fungsi <i>Fitness</i>	15
3.2.1.5	<i>Employed Bees Phase</i>	17
3.2.1.6	<i>Onlooker Bees Phase</i>	17
3.2.1.6.1	Eksplorasi dengan Metode <i>Random</i>	18
3.2.1.6.2	Eksplorasi dengan Metode <i>Sequential</i>	20
3.2.1.7	Perbaikan Jadwal	22
3.2.1.8	Data Input	22
3.3	Desain Perangkat Lunak.....	22
3.3.1	Pemodelan Perangkat Lunak	22
3.3.1.1	<i>Use Case Diagram</i>	23
3.3.1.2	<i>Activity Diagram</i>	24
3.3.1.2.1	Aktivitas Input Parameter	24
3.3.1.2.2	Aktivitas Menyusun Jadwal.....	25
3.3.2	Desain Penyimpanan Data	27
3.3.3	Desain Antarmuka	29
3.3.3.1	<i>Main Form</i>	29
3.3.3.2	<i>Result Form</i>	30
3.3.3.2.1	<i>Tab View All</i>	30
3.3.3.2.2	<i>Tab View by Class</i>	31
3.3.3.3	<i>Form Bentrokan</i>	31
BAB IV	32

PENGEMBANGAN PERANGKAT LUNAK	32
4.1 Implementasi Fungsi Utama	32
4.1.1 Fungsi Inisialisasi Populasi	32
4.1.2 Fungsi Evaluasi <i>Fitness</i>	34
4.1.3 Fungsi <i>Selection by Fitness</i>	36
4.1.4 Fungsi <i>Generate Childs</i>	37
4.1.4.1 <i>Generate Childs</i> dengan Metode <i>Random</i>	37
4.1.4.2 <i>Generate Childs</i> dengan Metode <i>Sequential</i>	40
4.2 Implementasi Antar Muka	43
4.2.1 <i>Main Form</i>	43
4.2.2 <i>Result Form</i>	44
4.2.2.1 <i>Tab View All</i>	44
4.2.2.1 <i>Tab View by Class</i>	45
4.2.3 <i>Form Bentrokan</i>	46
BAB V	47
PENGUJIAN	47
5.1 <i>White Box Testing</i>	47
5.1.1 <i>White Box Testing</i> untuk <i>Method</i> Inisialisasi Populasi	47
5.1.2 <i>White Box Testing</i> untuk <i>Method</i> Evaluasi <i>Fitness</i>	50
5.1.3 <i>White Box Testing</i> untuk <i>Method</i> <i>Selection</i>	53
5.1.4 <i>White Box Testing</i> untuk <i>Method</i> <i>Generate Childs</i>	55
5.2 <i>Black Box Testing</i>	59
5.2.1 <i>Main Form</i>	60
5.2.2 <i>Result Form</i>	60
5.3 Pengujian Algoritma ABC	61
5.3.1 Pengujian Besar Populasi dan Jumlah Generasi	61

5.3.2 Pengujian Perkembangan Generasi dengan Fungsi ABC.....	62
5.3.2.1 Pengujian Eksploitasi dengan Metode <i>Random</i>	65
5.3.2.2 Pengujian Eksploitasi dengan Metode <i>Sequential</i>	67
BAB VI.....	69
KESIMPULAN DAN SARAN	69
6.1 Simpulan.....	69
6.2 Saran	70

DAFTAR GAMBAR

Gambar 2.1 <i>Flowchart</i> dari algoritma ABC (Zhang, et al., 2011)	8
Gambar 3.1 Contoh Pemodelan Solusi.....	13
Gambar 3.2 Contoh Model Solusi untuk Penentuan <i>Fitness</i>	16
Gambar 3.3 Gambaran tahap <i>Employed Bees</i>	17
Gambar 3.4a Jadwal Pelajaran Sebelum Eksploitasi <i>Random</i>	18
Gambar 3.4b Jadwal Pelajaran Setelah Eksploitasi <i>Random</i>	19
Gambar 3.5a Jadwal Pelajaran Sebelum Eksploitasi <i>Sequential</i>	20
Gambar 3.5b Jadwal Pelajaran Setelah Eksploitasi <i>Sequential</i>	20
Gambar 3.5c Jadwal Pelajaran Setelah Eksploitasi <i>Sequential</i> (lanjutan).....	21
Gambar 3.6 <i>Use Case</i>	23
Gambar 3.7 <i>Activity Diagram</i> Input Parameter	24
Gambar 3.8 <i>Activity Diagram</i> Menyusun Jadwal.....	25
Gambar 3.9 Struktur Data Mata Pelajaran.....	27
Gambar 3.10 Struktur Data Guru	28
Gambar 3.11 Struktur Data Kelas.....	28
Gambar 3.12 Struktur Data Solusi Jadwal Pelajaran.....	28
Gambar 3.13 <i>Main Form</i>	29
Gambar 3.14 Rancangan <i>Interface Tab View All</i>	30
Gambar 3.15 Rancangan <i>Interface Tab View by Class</i>	31
Gambar 3.16 Rancangan <i>Interface Form Bentrokan</i>	31
Gambar 4.1 <i>Main Form</i>	43
Gambar 4.2 <i>Tab View All</i>	44
Gambar 4.3 <i>Tab View by Class</i>	45
Gambar 4.4 <i>Form Bentrokan</i>	46
Gambar 5.1 Hasil <i>WhiteBox Testing Method</i> Inisialisasi Populasi	50

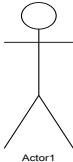


Gambar 5.2 Hasil <i>White Box Testing Method</i> Evaluasi <i>Fitness</i>	53
Gambar 5.3 Hasil <i>White Box Testing Method Selection</i>	55
Gambar 5.4 Hasil <i>White Box Testing Generate Childs</i>	59
Gambar 5.5 Grafik <i>Fitness</i> Inisial Dari Jumlah Populasi yang Berbeda.....	62
Gambar 5.6a Jadwal Pelajaran Terbaik Pada Inisialisasi Awal.....	63
Gambar 5.6b Jadwal Pelajaran Terbaik Pada Akhir Iterasi Fungsi ABC.....	64
Gambar 5.7 Grafik Perkembangan Jadwal per-Generasi Metode <i>Random</i>	66
Gambar 5.8 Grafik Perkembangan Jadwal per-Generasi Metode <i>Sequential</i>	68

DAFTAR TABEL






Tabel 5.1 Pengujian <i>Main Form</i>	60
Tabel 5.2 Pengujian <i>Result Form</i>	61
Tabel 5.3 Pengujian Besar Populasi dan Jumlah Generasi.....	61
Tabel 5.4 Pengujian dengan Fungsi ABC Eksploitasi <i>Random</i> – 3 Kelas	65
Tabel 5.5 Pengujian dengan Fungsi ABC Eksploitasi <i>Random</i> – 6 Kelas	65
Tabel 5.6 Pengujian dengan Fungsi ABC Eksploitasi <i>Random</i> – 15 Kelas	65
Tabel 5.7 Pengujian dengan Fungsi ABC Eksploitasi <i>Random</i> – 21 Kelas	66
Tabel 5.8 Pengujian dengan Fungsi ABC Eksploitasi <i>Sequential</i> – 3 Kelas.....	67
Tabel 5.9 Pengujian dengan Fungsi ABC Eksploitasi <i>Sequential</i> – 6 Kelas.....	67
Tabel 5.10 Pengujian dengan Fungsi ABC Eksploitasi <i>Sequential</i> – 15 Kelas.....	67

DAFTAR SIMBOL

1. Simbol *Use Case Diagram*

Deskripsi	Gambar
<p style="text-align: center;">Actor Pegguna aplikasi</p>	
<p style="text-align: center;">Use Case Proses yang dapat dilakukan oleh aktor</p>	
<p style="text-align: center;">Communication Hubungan antara aktor dan <i>use case</i></p>	

2. Simbol *Activity Diagram*

Deskripsi	Gambar
<p style="text-align: center;">Initial State Titik awal dari diagram</p>	
<p style="text-align: center;">Action State merepresentasikan aktivitas yang terjadi</p>	
<p style="text-align: center;">Decision</p>	
<p style="text-align: center;">Final State Titik akhir dari diagram</p>	
<p style="text-align: center;">Control Flow Panah Aliran</p>	

DAFTAR KODE PROGRAM

Kode Program 4.1 Inialisasi Populasi	33
Kode Program 4.2 Evaluasi <i>Fitness</i>	35
Kode Program 4.3 <i>Selection by Fitness</i>	36
Kode Program 4.4 <i>Generate Childs</i> dengan Metode <i>Random</i>	39
Kode Program 4.5 <i>Generate Childs</i> dengan Metode <i>Sequential</i>	42
Kode Program 5.1 Kode Program <i>WhiteBox Testing Method</i> Inialisasi Populasi	49
Kode Program 5.2 Kode Program Input <i>Method</i> Evaluasi <i>Fitness</i>	51
Kode Program 5.3 Kode Program <i>WhiteBox Testing Method</i> Evaluasi <i>Fitness</i>	52
Kode Program 5.4 Kode Program Input <i>Method Selection</i>	54
Kode Program 5.5 Kode Program <i>WhiteBox Testing Method Selection</i>	54
Kode Program 5.6 Kode Program <i>WhiteBox Testing Method Generate Childs</i>	58