

BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Penyusunan tinjauan pustaka dilakukan untuk memperoleh informasi yang relevan dari studi terdahulu yang telah dilakukan, tidak hanya untuk memperoleh informasi saja tinjauan pustaka membantu mengembangkan konsep berpikir dalam melakukan penelitian. Tinjauan pustaka berguna untuk memperoleh gambaran dan pengetahuan untuk mempertegas arah penelitian. Berdasarkan dengan permasalahan yang diteliti yaitu penjadwalan mata kuliah, kata kunci yang digunakan untuk memperoleh tinjauan pustaka yaitu penjadwalan mata kuliah, *timetabling management*, *timetabling*, *university timetabling problem*, dan *university timetabling course problem* (UTCP). Tinjauan pustaka diperoleh dari beberapa sumber seperti *ResearchGate*, *ProQuest*, *Google Scholar*, dan *Repository Universitas Atma Jaya Yogyakarta*.

2.1.1. Penelitian Terdahulu

Penelitian yang dilakukan Latumahina (2021) yaitu permasalahan sistem penjadwalan mata kuliah yang dilakukan secara manual dan membutuhkan waktu yang cukup lama dikarenakan dalam menyusun jadwal terdapat beberapa batasan yang harus diperhatikan ketika menyusun jadwal mata kuliah. Batasan tersebut menjadi dua kategori yaitu *hard constraint* dan *soft constraint*. *Hard constraint* ditetapkan berdasarkan ketentuan yang berlaku pada Program Studi Sarjana Teknik Industri terdapat 9 *hard constraint* berupa pengaturan waktu mengajar dosen, mata kuliah (reguler, paket, dan praktikum), ruangan (kelas dan laboratorium), dan kelas internasional, sedangkan *soft constraint* dari penelitian yang dilakukan yaitu kelas mata kuliah yang menggunakan ruangan laboratorium sebisa mungkin dijadwalkan pada sesi empat dalam setiap lima hari pengajaran dalam seminggu. Metode yang digunakan untuk permasalahan tersebut adalah *Modified Hybrid Genetic Algorithm* (MHGA) ini merupakan kombinasi dari dua algoritma yang dimodifikasi yaitu algoritma *Largest Color Degree* (LCD) dan *Genetic Algorithm* (GA) menyesuaikan *hard constraint* dan *soft constraint* yang telah ditetapkan sebelumnya. Algoritma LCD yang dimodifikasi berperan saat fase konstruksi dan algoritma genetika berperan saat fase optimasi. Fase konstruksi yaitu membentuk jadwal awal yang dipastikan tidak melanggar *hard constraint*,

sedangkan fase optimasi bertujuan untuk memperoleh susunan jadwal terbaik dengan meminimalkan *soft constraint*.

Penelitian yang dilakukan Sari (2018) yaitu mengoptimasi penjadwalan mata kuliah pada Program Studi Teknologi Informasi Universitas Lambung Mangkurat, masih dilakukan secara manual yang membutuhkan waktu 1 hingga 2 hari dikarenakan kesulitan saat pengalokasian mata kuliah, dosen, ruangan agar tidak terjadi bentrok. Permasalahan tersebut diuraikan dalam bentuk faktor-faktor yang berpengaruh dalam penyusunan jadwal yang terdiri dari waktu mengajar dosen, pengaturan ruangan, batas waktu perkuliahan yang diperbolehkan dan yang tidak diperbolehkan, dan pengaturan mata kuliah sesuai semesternya. Sari (2018) menggunakan metode algoritma genetika untuk menyelesaikan permasalahan tersebut yang dimana dilakukan dengan persiapan data jadwal perkuliahan kemudian masuk pada tahapan algoritma genetika yaitu tahapan inisialisasi populasi awal, tahapan *selection* atau pemilihan individu terbaik yang nantinya akan menjadi induk, tahapan *crossover* atau kawin silang, tahapan mutasi, dan yang terakhir yaitu output berupa jadwal perkuliahan. Berdasarkan dari penelitian yang dilakukan diperoleh penyusunan jadwal hanya diperlukan waktu sekitar 14.7 menit serta mendapatkan kombinasi terbaik antara mata kuliah, dosen yang mengajar, mahasiswa yang mengambil mata kuliah, dan ruangan yang tersedia.

Kusmarna dkk (2015) melakukan penelitian pada permasalahan penjadwalan mata kuliah yang ada pada Program Studi Informatika Universitas Islam Negeri Sultan Syarif Kasim Riau yaitu beberapa batasan tidak terpenuhi pada saat penyusunan jadwal mata kuliah seperti keterbatasan ruangan. Terdapat 13 aturan yang diterapkan pada saat penyusunan penjadwalan mata kuliah pada UIN Suska Riau mulai dari pengaturan mengajar dosen, ruangan, mata kuliah, dan waktu-waktu perkuliahan tertentu yang diperbolehkan dan tidak diperbolehkan. Metode yang digunakan untuk permasalahan tersebut yaitu *particle swarm optimization* (PSO) yang terdiri dari 3 tahap yaitu pembangkitan posisi serta kecepatan partikel, *update* kecepatan (*velocity update*), dan *update* posisi (*position update*).

Suhandi dkk (2023) melakukan penelitian pada permasalahan penjadwalan mata kuliah Program Studi Teknik Industri Universitas Kristen Maranatha Bandung yaitu ketersediaan waktu mengajar yang ada dan satuan kredit semester yang tidak terpisah. Waktu mengajar yang telah ditetapkan oleh universitas yaitu 32 jam kerja dalam satu minggu, untuk penyampaian perkuliahan berlangsung dua atau tiga

kali selama 50 menit setiap minggunya dan terkadang dalam seminggu terdapat perkuliahan selama 100 menit atau 150 menit untuk mata kuliah 2 atau 3 SKS yang tidak boleh dipisah permasalahan tersebut diperjelas dengan 16 kendala yang dibuat dalam bentuk formulasi matematika yang akan diselesaikan menggunakan metode *Integer Linear Programming* (ILP) dengan bantuan *software* CLPEx. Penelitian yang dilakukan Zaulir dkk (2022) mengimplementasikan model matematika untuk menyelesaikan permasalahan penjadwalan mata kuliah yang ada di *Malaysian Public University*. Zaulir dkk (2021) mengklasifikasikan permasalahan yang ada dalam bentuk daftar keterbatasan yang harus dipenuhi dalam penyusunan jadwal yang terdiri dari 24 batasan berupa mata kuliah (reguler dan praktikum), slot waktu, jumlah ruangan, mahasiswa, jam perkuliahan (yang diperbolehkan dan tidak diperbolehkan), dan lain-lain, kemudian batasan dirumuskan dalam persamaan matematika yang akan diselesaikan menggunakan metode *mixed integer linear programming* (MILP) dengan bantuan *software* CPLEX.

Sunarni dkk (2017) melakukan Penelitian mengoptimasi penjadwalan mata kuliah pada Program Studi Teknik Industri Universitas Katolik Musi Charitas Palembang. Penyusunan jadwal mata kuliah mengalami beberapa kendala diantaranya yaitu dosen dan mahasiswa pada semester yang sama dijadwalkan pada slot waktu yang sama sehingga terjadi bentrok mata kuliah, dan dosen yang tidak ingin dijadwalkan pada waktu tertentu. Permasalahan ini diselesaikan dengan menggunakan metode pewarnaan *graf* dengan algoritma *wech powell*. Penggunaan metode perwanaaan *graf* ini dilakukan dengan membuat 3 skenario pengkondisian yang dimana tiap skenario ini disusun dengan mempertimbangkan mata kuliah, ruangan, beban sks, dan slot waktu yang tersedia.

Penelitian yang dilakukan Setiawan dkk (2015) yaitu permasalahan penjadwalan mata kuliah pada Fakultas Sains dan Teknologi Universitas Ma Chung yang dilakukan secara manual dengan sistem *resource sharing* yang membuat rumit dan ruangan serta dosen yang dimiliki sangat terbatas. Permasalahan tersebut diselesaikan dengan menggunakan metode *firefly algorithm* (algoritma kunang-kunang) yang dilakukan dengan beberapa tahapan yaitu representasi kunang-kunang, inisialisasi kunang-kunang, kualitas kunang-kunang, posisi kunang-kunang, dan kondisi terminasi. Representasi kunang-kunang yaitu langkah merepresentasikan jadwal yang terdiri dari banyaknya slot waktu yang terbagi dalam 4 dimensi yaitu id sesi mata kuliah, Id ruangan, kapasitas ruangan, dan jenis

ruangan. Insialisasi kunang-kunang yaitu melakukan penempatan 4 dimensi sebelumnya berdasarkan kebutuhan dan sebisa mungkin meminimasi pelanggaran *constraint*. Kualitas kunang-kunang adalah tahapan menghitung intensitas cahaya kunang-kunang, intensitas cahaya yang dimaksud yaitu nilai *objective function* atau fungsi *fitness*, semakin besar nilai *objective function*nya maka semakin baik kualitas jadwal yang dihasilkan. Posisi kunang-kunang mewakili semua id sesi mata kuliah pada slot waktu yang nantinya akan berotasi berdasarkan slot region waktu. Kondisi terminasi merupakan proses iterasi keseluruhan algoritma kunang-kunang, iterasi akan berhenti ketika nilai penalti bernilai 0 atau jumlah iterasi telah mencapai nilai tertentu yang telah ditetapkan sebelumnya.

Khader dkk (2018) melakukan penelitian terhadap permasalahan penjadwalan mata kuliah yang ada pada Program Studi Informatika Institut Teknologi Nasional Bandung. Penjadwalan mata kuliah pada Program Studi Informatika masih dilakukan secara manual yang membutuhkan waktu lama, dan terkadang terdapat jadwal mata kuliah yang bentrok baik dari ruangan, dosen, dan slot waktu. Permasalahan tersebut diselesaikan dengan menggunakan algoritma *greedy* yang dimana permasalahan tersebut dideskripsikan dalam bentuk batasan-batasan yaitu *hard constraint* dan *soft constraint*. Terdapat 7 *hard constraint* yang digunakan untuk mengeksekusi algoritma *greedy* sedangkan tidak dijelaskan *soft constraint* yang digunakan. Proses penjadwalan mata kuliah yang dilakukan dengan menggunakan algoritma *greedy* terdiri dari 6 proses yaitu pembentukan jadwal mata kuliah awal, pengurutan nilai konflik terbesar dari mata kuliah yang ada, memilih nilai konflik yang paling tinggi dari setiap mata kuliah, menentukan slot waktu yang mempunyai nilai terbaik dengan algoritma *greedy*, setelah diketemukan slot waktu terbaik maka mata kuliah tersebut dihapus dari jadwal mata kuliah awal sedangkan jika slot waktu tidak ditemukan maka mata kuliah tersebut disimpan pada mata kuliah tidak terjadwal, dan proses yang terakhir menyimpan jadwal mata kuliah yang telah memiliki slot waktu tersebut pada *database*.

Penelitian yang dilakukan Wahyuningsih (2015) yaitu membuat rancangan sistem penjadwalan mata kuliah pada Sekolah Manajemen Informatika dan Komputer (STMIK) Atma Luhur Pangkal Pinang. Terdapat beberapa kendala dalam penyusunan jadwal mata kuliah pada STMIK Atma Luhur Pangkal Pinang yaitu kesulitan mengatur waktu, tempat, dan mata kuliah agar tidak terjadinya duplikat mata kuliah pada waktu yang sama, kesulitan mengatur jadwal dosen mengajar agar tidak terjadinya dosen mengajar 2 mata kuliah pada waktu yang sama, dan masih terdapat kendala lainnya yang rumit dalam pengaturan jadwal perkuliahan dan membutuhkan waktu yang lama dalam mengatur jadwal perkuliahan tersebut. Permasalahan tersebut dideskripsikan dalam bentuk batasan yang terbagi menjadi 2 yaitu *hard constraint* dan *soft constraint*. Terdapat 7 *hard constraint* dan sebuah *soft constraint* dari permasalahan yang ada, *hard constraint* dan *soft constraint* tersebut akan diselesaikan dengan menggunakan algoritma *Max Min Ant System*. Jadwal yang dihasilkan dari algoritma *Max Min Ant System* ini hilangnya jadwal bentrok dari mengajar dosen, mata kuliah, dan waktu perkuliahan.

Saifullah & Hermawan (2017) melakukan penelitian terhadap permasalahan penjadwalan mata kuliah pada Fakultas Sains dan Teknologi Universitas Teknologi Yogyakarta yaitu penjadwalan mata kuliah yang tidak efektif serta sering terjadinya jadwal yang bentrok dan membutuhkan waktu yang lama untuk direvisi kembali, untuk mengatasi hal tersebut Saifullah & Hermawan (2017) merancang sebuah sistem informasi penjadwalan mata kuliah dengan menggunakan algoritma *Steepest Ascent Hill Climbing* yang dilakukan dengan langkah-langkah yaitu perencanaan dan seleksi sistem, analisis sistem, desain sistem menggunakan *Entity Relationship Diagram* (ERD), dan implementasi sistem. Berdasarkan dari penelitian tersebut diperoleh sistem penjadwalan mata kuliah yang tidak memerlukan waktu yang lama lagi serta output jadwal yang dihasilkan mampu meminimalisir jadwal yang bentrok.

Kumar (2016) melakukan penelitian pada *Institute Chatered of Financial Analysis of India* (ICFAI) dengan permasalahan perebutan mata kuliah pada beberapa Fakultas yang ada pada ICFAI. Permasalahan tersebut diselesaikan dengan menggunakan metode *Analytical Hierarchy Process* (AHP) yang dilakukan dengan mengembangkan model dari permasalahan menggunakan notasi matematika, mengembangkan model hiearaki dengan menggunakan skala 1-9 mulai dari *Equally Preferred* sampai *Extremely Preferred*, kemudian dilanjutkan dengan *Pair-Wise Comparison Matrix*, pemberian *Random Index*, *Constancy Checking*, dan langkah terakhir yaitu mengembangkan tingkatan prioritas. Berdasarkan penelitian tersebut diperoleh hasil bahwa Fakultas yang memperoleh mata kuliah tersebut berdasarkan nilai prioritas tertinggi.

Penelitian yang dilakukan Kalu dkk (2018) yaitu mengembangkan teknik atau cara untuk menghandel jadwal mata kuliah pada *Department of Electrical/Electronic and Computer Engineering University of Uyo Akwa Ibom State Nigeria* yang terdapat bentrok dari jadwal mengajar dosen, jumlah dan kapasitas ruangan, dan slot waktu yang tersedia. Kalu, C. dkk. (2018) mengembangkan sebuah prosedur penjadwalan mata kuliah berdasarkan masalah tersebut dengan menggunakan metode *Total Rank Score* (TRS) dan *Rank Weight* (RW). TRS dikhususkan untuk mata kuliah, jumlah ruangan beserta kapasitas, dan slot waktu, sedangkan RW digunakan untuk mengatur jadwal mengajar dosen. Berdasarkan penelitian di tersebut jadwal bentrok terminimalisir dengan penggunaan TRS dan bentrok terhadap jadwal mengajar dosen terminimalisir dengan baik dengan menggunakan RW untuk jadwal dosen.

Yudha & Elfatiha (2021) melakukan penelitian pada penjadwalan ruang kelas pada Institut Bisnis Muhammadiyah Bekasi, Yudha dan Elfatiha menemukan permasalahan yaitu terjadinya benturan ruang kelas pada saat kuliah perdana dilakukan yang menyebabkan mahasiswa kecewa terhadap kampus dan hal ini memberikan kesan yang buruk terhadap mahasiswa, setelah ditelusuri lebih lanjut didapatkan bahwa saat ini terjadi kesulitan saat mengatur dan mengelola pembagian ruangan kelas karena pengerjaannya masih dilakukan secara manual dan ruang perkuliahan yang minim, berangkat dari permasalahan tersebut Yudha dan Elfatiha melakukan penelitian dan mencoba memberikan solusi dengan tujuan tidak kesulitan dalam pengelolaan ruangan kelas yang dilakukan saat ini dan tidak terjadi lagi tabrakan ruangan kelas, Yudha dan Elfatiha merancang sebuah sistem informasi penjadwalan ruangan kelas untuk IBM Bekasi, perancangan sistem informasi tersebut menggunakan metode *waterfall*, hasil yang didapatkan dari penelitian ini berupa sebuah sistem informasi dalam bentuk *website* yang dimana mempermudah dalam pengelolaan dan pengaturan ruangan kelas serta pihak admin yang menggunakan sistem informasi ini dapat mengontrol sistem pengelolaan jadwal ruangan kelas secara *real time*.

Seluruh penelitian terdahulu yang sudah dijelaskan sebelumnya kemudian direkap yang dapat dilihat pada Tabel 2.1 sebagai berikut.

Tabel 2.1. Rekapitan Tinjauan Pustaka

Rekapitan Tinjauan Pustaka							
No	Penulis	Objek	Judul	Masalah yang Diteliti	Tujuan Studi/ <i>Critical Succes Factor</i>	Metode yang Digunakan	Hasil
1	Latumahina (2021)	Program Studi Sarjana Teknik Industri Universitas Atma Jaya Yogyakarta	Penjadwalan Mata Kuliah Otomatis Menggunakan Modified Hybrid Genetic Algorithm	Sistem penjadwalan yang manual menghabiskan waktu yang lama untuk menghasilkan jadwal yang feasible, terlebih lagi mencapai jadwal yang optimal bagi sistem.	Membuat sistem penjadwalan otomatis yang menghasilkan jadwal optimal secara cepat dan memenuhi batasan yang ada.	MHGA (<i>Modified Hybrid Genetic Algorithm</i>)	Penyusunan jadwal mata kuliah yang cepat dan menghasilkan beberapa alternatif jadwal <i>feasible</i>
2	Sunarni dkk (2017)	Program Studi Sarjana Teknik Industri Fakultas Sains dan Teknologi Universitas Katolik Misi Charitas Palembang	Optimasi Penjadwalan Mata Kuliah Menggunakan Pewarnaan Graf	Mengoptimalkan sistem penjadwalan kuliah di prodi TI FST UKMC.	Pengembangan sistem penjadwalan mata kuliah dapat menghindari adanya bentrok dan terpenuhinya permintaan dosen untuk tidak dijadwalkan pada slot waktu tertentu.	Algoritma <i>Largest Degree Ordering</i>	Menghasilkan jadwal yang optimal, yakni diperoleh jadwal yang tidak bentrok mahasiswa/dosen/ruang dan juga permintaan dosen untuk tidak dijadwalkan pada slot waktu tertentu dapat dipenuhi.
3	Suhandi dkk (2023)	Program Studi Teknik Industri Universitas Kristen Maranatha	Penjadwalan Mata Kuliah dengan Mempertimbangkan Ketersediaan Waktu Pengajar dan Satuan Kredit Semester yang Tidak Terpisah	Penjadwalan mata kuliah dengan ketersediaan waktu pengajar terbatas dan bobot SKS yang bervariasi pada prodi TI Universitas Kristen Maranatha.	Mengoptimalkan penjadwalan mata kuliah dengan meminimasi ketidaksesuaian jadwal yang dihasilkan dengan ketersediaan waktu para pengajar serta mempertimbangkan sistem kredit yang tidak dapat dipisahkan dalam setiap minggunya.	<i>Integer Linear Programming (ILP)</i>	Jadwal setiap mata kuliah tidak ada yang terpisah dan sesuai bobot jumlah slot waktu yang dibutuhkan setiap minggunya dan setiap pengajar hanya mengajar satu mata kuliah di slot waktu yang sama.

Tabel 2.1. Lanjutan

Rekapan Tinjauan Pustaka							
No	Penulis	Objek	Judul	Masalah yang Diteliti	Tujuan Studi/ <i>Critical Succes Factor</i>	Metode yang Digunakan	Hasil
4	Kusmarna dkk (2015)	Program Studi Teknik Informatika Universitas Islam Negeri Sulta Syarif kasyim Riau	Aplikasi Penjadwalan Mata Kuliah Menggunakan Algoritma <i>Particle Sworm Optimization</i>	Penjadwalan mata kuliah di UIN SUSKA Riau yang tidak optimal karena pemenuhan beberapa <i>constraint</i> seperti keterbatasan ruangan.	Menghasilkan jadwal perkuliahan yang sudah tidak ada bentrok tetapi masih tidak memenuhi dari segi kualitas yaitu jam dimulainya perkuliahan inti yang difokuskan pada jam-jam yang efektif.	<i>Particle Swarm Optimization</i> (PSO)	Menghasilkan jadwal perkuliahan yang sudah tidak terdapat bentrok, namun tidak berhasil memenuhi dari segi kualitas yaitu jam dimulainya perkuliahan yang difokuskan pada jam-jam yang efektif.
5	Zaulir dkk (2022)	<i>Malaysian Public University</i>	<i>A General Mathematical Model for University Courses Timetabling: Implementation to a Public University in Malaysia</i>	Penjadwalan mata kuliah yang memperhatikan banyak faktor yaitu waktu dan ruangan tidak boleh bertabrakan, keterbatasan jam mengajar, dan ketersediaan dosen & ruangan dll.	Menghasilkan jadwal yang mampu memenuhi faktor keterbatasan yang ada.	<i>Mixed Integer Linear Programming</i> (MILP)	Menghasilkan jadwal yang memenuhi sebanyak 24 <i>constraint</i> dalam waktu kurang dari 30 menit.
6	Setiawan dkk (2015)	Fakultas Sains dan Teknologi Universitas Ma Chung	Implementasi Algoritma Kunang-Kunang Untuk Penjadwalan Mata Kuliah Universitas Ma Chung	Penjadwalan mata kuliah pada Fakultas Sains dan Teknologi Universitas Ma Chung masih dilakukan secara manual serta ruangan kelas dan dosen yang terbatas.	Membuat sistem penjadwalan otomatis yang mengampu batasan serta mengoptimasi penjadwalan mata kuliah.	<i>Firefly Algorithm</i> (Algoritma Kunang-kunang)	Menghasilkan jadwal mata kuliah yang tidak bertabrakan serta pengaturan slot waktu dan ruangan yang tidak bertabrakan.

Tabel 2.1. Lanjutan

Rekapan Tinjauan Pustaka							
No	Penulis	Objek	Judul	Masalah yang Diteliti	Tujuan Studi/ <i>Critical Succes Factor</i>	Metode yang Digunakan	Hasil
7	Khader dkk (2018)	Program Studi Informatika Institut Teknologi Nasional Bandung	Penjadwalan Mata Kuliah Menggunakan Algoritma Greedy	Penerapan algoritma greedy untuk menghasilkan jadwal mata kuliah yang tidak bentrok serta kapasitas kelas dan jumlah mahasiswa dapat menyesuaikan serta penerapan algoritma greedy untuk batasan lainnya seperti dosen, mata kuliah, ruangan, dan slot waktu.	Mengimplementasikan algoritma greedy sehingga didapatkan hasil penjadwalan perkuliahan yang tidak memiliki bentrok, kapasitas kelas dan jumlah mahasiswa dapat menyesuaikan.	<i>Greedy Algorithm</i> (Algoritma Greedy)	Menghasilkan jadwal mata kuliah yang tidak bentrok serta batasan serta pengaturan ruangan, dosen, mata kuliah teroptimasi.
8	Wahyuningsih (2015)	Sekolah Manajemen Informatika dan Komputer (STMIK) Atma Luhur Pangkal Pinang	Rancangan Sistem Penjadwalan Akademik Menggunakan Algoritma <i>Max Min Ant System</i> (MMAS)	Kesulitan dalam pengaturan waktu, ruangan, dan mata kuliah agar tidak bentrok, kesulitan dalam pengaturan jadwal mengajar dosen agar tidak bentrok dan kendala lainnya sehingga penyusunan jadwal berlangsung lama.	Memberikan sebuah perancangan sistem penjadwalan akademik pada STMIK Atma Luhur Pangkal Pinang dengan optimal.	Algoritma <i>Max Min Ant System</i> (MMAS)	Menghasilkan sistem penjadwalan yang berupa <i>User Interface</i> (UI) dan jadwal mata kuliah yang dihasilkan tidak bentrok.
9	Saifullah & Hermawan (2017)	Fakultas Sains dan Teknologi Universitas Teknologi Yogyakarta	Pengembangan Sistem Penjadwalan Menggunakan Algoritma <i>Steepest Ascent Hill Climbing</i>	Sistem penjadwalan yang tidak efektif serta masih terjadi jadwal bentrok dan membutuhkan waktu yang lama untuk merevisi jadwal mata kuliah tersebut.	Mengembangkan sistem penjadwalan mata kuliah yang mampu meningkatkan proses belajar serta menghilangkan bentrok pada jadwal yang mengganggu kegiatan belajar mengajar	Algoritma <i>Steepest Ascent Hill Climbing</i>	Mempercepat proses penyusunan jadwal mata kuliah dan tingkat bentrok pada jadwal terminimalisir.

Tabel 2.1. Lanjutan

Rekapan Tinjauan Pustaka							
No	Penulis	Objek	Judul	Masalah yang Diteliti	Tujuan Studi/ <i>Critical Succes Factor</i>	Metode yang Digunakan	Hasil
10	Kumar (2016)	<i>Institute of Chatered Financial Analysis of India (ICFAI University)</i>	<i>Solving University Course Timetabling Using Analytical Hierarchy Process (AHP)</i>	Terjadinya perebutan sebuah mata kuliah untuk beberapa fakultas sehingga fakultas yang seharusnya mendapat mata kuliah tersebut tidak mendapat mata kuliah tersebut.	Menentukan pengalokasian mata kuliah yang terbaik agar fakultas menerima mata kuliah yang sesuai.	<i>Analytical Hierarchy Process (AHP)</i>	Mata kuliah teralokasikan dengan baik yang dimana dilakukan berdasarkan perhitungan skala prioritas menggunakan metode AHP.
11	Kalu dkk (2018)	<i>Department of Electrical/Electronic and Computer Engineering University of Uyo Akwa Ibom State Nigeria</i>	<i>Development of Mechanism for Handling Conflicts and Constraints in University Timetable Management System</i>	Terdapat jadwal mata kuliah bentrok dari sisi jadwal dosen, jumlah dan kapasitas ruangan, dan slot waktu.	Menghandel jadwal mata kuliah yang bentrok serta pengaturan jadwal dosen, jumlah dan kapasitas ruangan, dan slot waktu yang sesuai.	<i>Total Rank Score (TRS) & Rank Weight (RW)</i>	Menghasilkan jadwal mata kuliah yang dimana bentrok pada jadwal terminimalisir serta pengaturan dosen dan ruangan tidak bentrok.
12	Sari dkk (2018)	Program Studi Teknologi Informasi Universitas Lambung Mangkurat	Optimasi Penjadwalan Mata Kuliah Menggunakan Metode Algoritma Genetika Dengan Teknik <i>Tournament Selection</i>	Pembuatan jadwal masih secara manual dengan menggunakan <i>Microsoft Office Excel</i> , Sehingga mengakibatkan waktu pekerjaan yang cukup lama dengan kisaran waktu sekitar 1 - 2 hari dari proses rapat sampai proses pembuatan jadwal selesai dan terdapat kesulitan dalam pengalokasian mata kuliah, dosen, mahasiswa yang mengambil mata kuliah beserta ruangnya agar tidak terjadi bentrok.	Merancang suatu sistem yang dapat membuat atau menyusun jadwal perkuliahan secara teroptimasi.	<i>Genetic Algorithm (Algoritma Genetika)</i>	Proses pembuatan atau penyusunan jadwal untuk perkuliahan, yaitu hanya diperlukan waktu sekitar 14.7 menit serta mendapatkan kombinasi terbaik antara mata kuliah, dosen yang mengajar, mahasiswa yang mengambil mata kuliah, dan ruangan yang tersedia.

Tabel 2.1. Lanjutan

Rekapan Tinjauan Pustaka							
No	Penulis	Objek	Judul	Masalah yang Diteliti	Tujuan Studi/ <i>Critical Succes Factor</i>	Metode yang Digunakan	Hasil
13	Yudha & Elfatiha (2021)	Institut Bisnis Muhammadiyah Bekasi	Sistem Informasi Penjadwalan Ruang Kelas Perkuliahan Berbasis <i>Web</i> Menggunakan <i>Waterfall</i> Model Pada Institut Bisnis Muhammadiyah Bekasi	Terjadinya penggunaan ruang kelas yang bertabrakan saat kuliah perdana yang menyebabkan kekecewaan dan membuat kesan awal yang buruk dari mahasiswa terhadap kampus.	Mempermudah dan mempercepat penjadwalan ruang kelas dengan merancang sistem informasi perkuliahan berbasis web.	<i>Waterfall</i>	Sistem informasi penjadwalan menghasilkan jadwal ruang kelas dan tidak terjadinya tabrakan ruang kelas.

Berdasarkan dari tinjauan pustaka yang sudah dilakukan yang dapat dilihat pada Tabel 2.1 didapatkan bahwa penelitian mengenai penyusunan jadwal mata kuliah berfokus kepada permasalahan umum yang muncul pada saat penyusunan jadwal mata kuliah terjadi seperti jadwal bertabrakan dari sisi dosen, kelas, ruangan, dan slot waktu, berbeda dengan penelitian kali ini yaitu terdapat beberapa permasalahan didalam proses penyusunan jadwal mata kuliah yang dilakukan, khususnya pada batasan jadwal yang akan diselesaikan yaitu terdapatnya batasan pada jadwal berupa preferensi jadwal mengajar dosen yang juga harus disesuaikan pada jadwal yang akan disusun, hal ini yang menjadi keunikan dari penelitian kali ini.

Jika diteliti lebih lanjut dari penelitian yang sudah dilakukan sebelumnya mengenai penjadwalan mata kuliah yang dapat dilihat pada Tabel 2.1 sebagian besar penelitian terdahulu tersebut hanya menyoroti bagaimana sebuah solusi yang dirancang dapat menyusun jadwal tanpa terjadi benturan dari data yang diberikan, tetapi jarang ditemukan penelitian yang mencoba mengangkat pembahasan mengenai preferensi mengajar dosen dan maksimalisasi utilitas penggunaan ruangan terhadap kapasitas sebuah kelas dari mata kuliah, hal tersebut menjadi kesenjangan atau *gap* yang ditemukan dari penelitian sebelumnya sehingga penelitian ini bisa menjadi kontribusi dalam mengisi kesenjangan tersebut.

2.2. Dasar Teori

2.2.1. University Course Timetabling Problem (UCTP)

Permasalahan penjadwalan mata kuliah universitas atau yang sering disebut *University Course Timetabling Problem* (UCTP) merupakan sebuah permasalahan penjadwalan yang bersifat multi-dimensi karena terdapat dosen sebagai tenaga pengajar dan mahasiswa yang akan dijadwalkan berdasarkan waktu dan mata kuliah tertentu (Carter dan Laporte, 1998). Tidak hanya termasuk pada permasalahan yang bersifat multi-dimensi menurut Jain dkk (2010) UCTP dapat dikategorikan sebagai permasalahan penjadwalan NP-hardness (*non deterministic polynomial time hardness*). UCTP terbagi menjadi dua sub-kategori yang pertama *post enrollment based course timetabling* yaitu penjadwalan mata kuliah dalam slot waktu dan ruangan, kedua *curriculum based course timetabling* yaitu penjadwalan kegiatan perkuliahan dilakukan dalam waktu mingguan yang terdiri dari berbagai jenis mata kuliah yang dialokasikan pada ruangan dan slot waktu tertentu.

Penjadwalan mata kuliah pada universitas memiliki banyak *constraints* atau batasan, umunya yang menjadi batasan utama dalam penjadwalan mata kuliah tersebut ialah jumlah ruangan, dosen, mata kuliah, jumlah kelas, dan *timeslots* (slot waktu), dari batasan-batasan yang ada jadwal yang dihasilkan harus terdapat satu mata kuliah pada satu kelas serta ruangan dan pada satu slot waktu (Hambali, 2020). Batasan yang telah disebutkan biasanya diklasifikasikan sebagai *hard constraint* (batasan kaku) yang dimana batasan ini tidak boleh dilanggar (Boonyopakorn, 2017), selain batasan kaku terdapat juga batasan lunak yang biasanya disebut *soft constraint*, batasan lunak ini termasuk batasan yang tidak terlalu diprioritaskan untuk diselesaikan dalam artian batasan yang dapat dilanggar namun ketika batasan ini dilanggar dapat menurunkan kualitas jadwal yang dihasilkan (Latumahina, 2017).

Beberapa contoh batasan kaku yang umumnya sering digunakan pada penjadwalan mata kuliah universitas ialah sebagai berikut.

- a. Durasi mata kuliah harus dilakukan pada hari yang sama dan tidak diperbolehkan dibagi menjadi beberapa hari.
- b. Mata kuliah dengan kelas yang sama tidak diperbolehkan pada slot waktu yang sama.
- c. Seorang dosen tidak diperbolehkan mengajar lebih dari satu mata kuliah dalam satu slot waktu.

- d. Suatu mata kuliah tidak diperbolehkan dalam satu slot waktu ketika tidak ada ruangan yang mampu menampung jumlah mahasiswa yang mengambil kelas mata kuliah tersebut.

Batasan lunak yang umumnya diterapkan pada penjadwalan mata kuliah universitas ialah sebagai berikut:

- a. Kapasitas ruangan kelas harus lebih dari jumlah mahasiswa yang mengambil kelas mata kuliah tersebut.
- b. Waktu mengajar dosen harus dialokasikan semaksimal mungkin untuk setiap slot waktu yang ada.
- c. Jumlah ruangan kelas yang ada harus di maksimalkan penggunaannya untuk setiap slot waktu yang ada.

2.2.2. Genetic Algorithm (GA)

Genetic algorithm atau biasanya disebut algoritma genetika merupakan sebuah algoritma pencarian dan teknik optimasi (Sari dkk, 2018). Algoritma genetika ditemukan oleh Goldberg dan Holland pada awal tahun 1970 di New York, Amerika Serikat yang dimana algoritma genetika ini terinspirasi dari teori evolusi Darwin. Algoritma genetika dikategorikan sebagai *random search* yang dimana *random search* sangat berguna untuk jenis permasalahan *non-linear*. Pada awalnya algoritma genetika mempunyai prosedur dengan pendekatan heuristik dan tidak dijamin bahwa algoritma genetika mampu menghasilkan solusi yang paling optimal, tetapi seiring berjalannya waktu dan banyaknya penelitian yang dilakukan dengan menggunakan algoritma genetika, algoritma genetika mampu menemukan solusi terbaik untuk beberapa masalah (Yadav dan Sohal, 2017).

Algoritma genetika mempunyai mekanisme pemilihan tersendiri untuk menyeleksi individu dari populasi yang ada untuk dikawinkan atau disilangkan, individu yang dikawinkan atau disilangkan bertujuan untuk memperoleh keturunan baru yang nantinya akan menjadi dasar proses generasi selanjutnya. Mekanisme pemilihan individu yang akan disilangkan pada algoritma genetika yaitu pemilihan individu dengan nilai yang lebih baik dalam sebuah populasi dengan menggunakan teknik tekanan seleksi, semakin tinggi tekanan seleksi yang dipilih maka semakin banyak individu dengan nilai yang lebih baik dalam sebuah populasi dan tekanan seleksi yang tinggi menghasilkan nilai konvergensi yang tinggi, namun sebaliknya jika tekanan seleksi rendah menghasilkan konvergensi yang lambat dan algoritma genetika tidak memerlukan waktu yang lama dalam menentukan solusi optimal,

berdasarkan kondisi tekanan seleksi yang ada dapat dikatakan bahwa tekanan seleksi mendorong algoritma genetika untuk terus meningkatkan nilai *fitness* populasi atas generasi berikutnya (Sari dkk, 2018).

Pada penerapan algoritma genetika khususnya untuk studi kasus penjadwalan mata kuliah terdapat istilah-istilah yang biasanya muncul pada algoritma genetika yang merepresentasikan elemen dan proses penjadwalan mata kuliah yang dapat dilihat pada Tabel 2.2 sebagai berikut.

Tabel 2.2. Representasi Elemen Jadwal Pada Algoritma Genetika

Elemen	Penjelasan
Gen	Gen adalah representasi dari sebuah kelas kuliah, misalkan {matematika diskrit, senin 1, R3103, Ibu Tuti}.
Kromosom	Kromosom adalah representasi dari gabungan beberapa gen biasanya gabungan dalam bentuk sebuah semester.
Populasi	Populasi adalah representasi sebuah alternatif jadwal dalam bentuk kumpulan yang banyak.
Generasi	Generasi adalah representasi dari proses iterasi pada algoritma genetika untuk mencari solusi optimal.
<i>Offspring</i>	<i>Offspring</i> adalah representasi dari anak kromosom (hasil dari proses persilangan sepasang kromosom) pada algoritma genetika.
Individu	Sebuah solusi jadwal dalam suatu populasi

Terdapat 6 istilah elemen yang biasanya disebutkan pada penjelasan proses dengan algoritma genetika, selanjutnya pada algoritma genetika terdapat beberapa komponen yang dijelaskan sebagai berikut.

Puspaningrum dkk (2013) menyebutkan terdapat 6 komponen utama dalam algoritma genetika yaitu sebagai berikut.

a. *Technical Code*

Technical code atau teknik pengkodean untuk mengkodekan gen dari kromosom yang dimana teknik pengkodean ini untuk membangkitkan populasi awal yang akan menjadi calon solusi dari sebuah masalah kedalam kromosom, yang dimana terdiri dari gen dan kromosom, gen yang dimaksud ialah bagian dari kromosom dan direpresentasikan dalam bentuk tipe data seperti *string*, *bit*, *array*, dan lain-lain yang nantinya akan digunakan pada operator genetika.

b. *Generate Initial Population*

Generate initial population yaitu proses membangkitkan individu secara acak dan jumlah populasi tergantung dari masalah yang akan diselesaikan yang selanjutnya akan diinisialisasi terhadap kromosom yang ada pada populasi tersebut yang dilakukan secara acak.

c. *Evaluation Function*

Evaluation function atau fungsi evaluasi yaitu proses mengevaluasi setiap individu dengan memperhatikan nilai *fitness*, semakin tinggi nilai *fitness* pada kromosom maka individu tersebut akan dipertahankan sebaliknya jika semakin rendah nilai *fitness* maka individu tersebut akan diganti dengan yang baru. Fungsi *fitness* menyesuaikan dengan permasalahan yang ada dari representasi yang digunakan.

d. *Selection*

Selection atau proses seleksi digunakan pada algoritma genetika dengan teknik *Tournament Selection* yaitu teknik untuk menyeleksi individu berdasarkan nilai *fitness* tertinggi akan dilanjutkan pada proses selanjutnya.

e. *Genetic operators*

Genetic operator terdiri dari 3 proses utama yaitu *crossover*, *mutation*, dan *regeneration* yang dijelaskan sebagai berikut.

i. *Crossover*

Crossover atau proses persilangan yaitu menggabungkan 2 kromosom *parent* sehingga membentuk kromosom baru (*offspring*), proses *crossover* dilakukan dengan memilih salah satu titik yang dipisah secara acak kemudian mengkombinasikan segmen dari satu induk ke sebelah kiri dari titik yang dipisahkan dengan segmen dari induk lain ke sebelah kanan dari titik yang dipisahkan.

ii. *Mutation*

Mutation atau mutasi merupakan proses merubah satu atau lebih beberapa gen dari suatu kromosom yang berfungsi untuk menggantikan gen yang hilang dari populasi yang diakibatkan proses seleksi.

iii. *Regeneration*

Regeneration atau regenerasi adalah lanjutan dari proses mutasi yang dimana kromosom yang baru akan dimasukkan kedalam populasi awal untuk mengganti beberapa kromosom lainnya, pergantian dilakukan dengan mengganti seluruh populasi awal dengan kromosom terbaru dari hasil proses

mutasi sebelumnya dengan memilih kromosom dengan *fitness* terkecil pada populasi sebelumnya (Latumahina, 2017).

f. *Control Parameters*

Parameter kontrol pada algoritma genetika dibagi menjadi 2 yaitu *probability crossover* dan *probability mutation*.

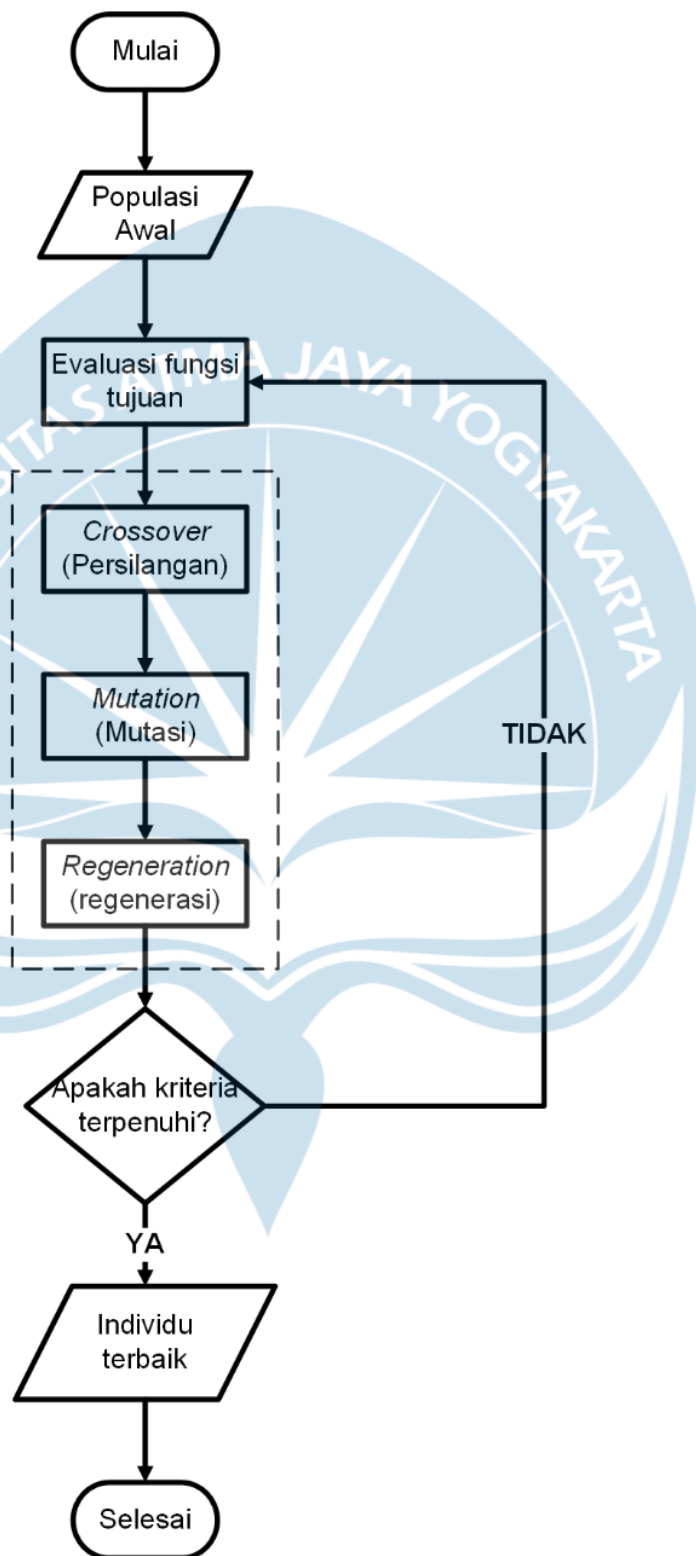
i. *Probability Crossover*

Probabilitas persilangan akan menghandel proses persilangan populasi, struktur individu akan semakin cepat masuk kedalam populasi yang baru jika semakin besar pula nilai probabilitas persilangannya, namun individu yang menjadi kandidat solusi juga dapat hilang pada generasi selanjutnya jika nilai probabilitas persilangan terlalu besar.

ii. *Probability Mutation*

Probabilitas mutasi menghandel proses mutasi setiap generasi dimana probabilitas mutasi lebih kecil dari probabilitas persilangan. Suhartono (2015) menyebutkan bahwa proses mutasi jarang sekali muncul pada proses algoritma genetika

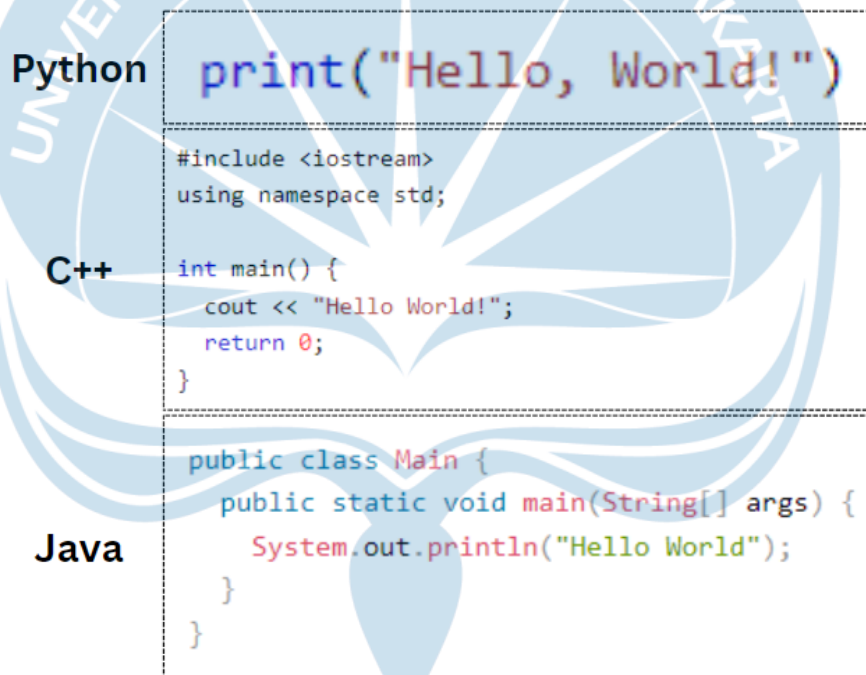
Secara umum proses algoritma genetika dapat digambarkan dalam bentuk diagram alir pada Gambar 2.1 sebagai berikut.



Gambar 2.1. Diagram Alir Algoritma Genetika

2.2.3. Python (Bahasa Pemrograman)

Python merupakan sebuah bahasa pemrograman yang diciptakan oleh Guido Van Rossum pada sekitar tahun 1991. Python termasuk bahasa pemrograman tingkat tinggi atau *high-level* yang dapat digunakan diberbagai bidang atau *general-purpose* dan bersifat *interpreted language* dan *open source* (Thaker dan Shukla, 2020). Dwivedi dan Jaiswal (2021) menyebutkan bahwa python awalnya diciptakan sebagai bahasa pemrograman yang mudah dibaca dan dipahami yang ditandai dengan penggunaan spasi putih (*whitespace*) selain itu penulisan yang mudah dipahami pada bahasa pemrograman python dimaksudkan untuk membantu *software engineer* dalam sebuah proyek pada saat proses penyusunan kode, perbandingan penulisan kode dari bahasa pemrograman python dengan bahasa pemrograman lainnya dapat dilihat pada Gambar 2.2 sebagai berikut.



Bahasa	Kode
Python	<pre>print("Hello, World!")</pre>
C++	<pre>#include <iostream> using namespace std; int main() { cout << "Hello World!"; return 0; }</pre>
Java	<pre>public class Main { public static void main(String[] args) { System.out.println("Hello World"); } }</pre>

Gambar 2.2. Perbandingan Kode “Hello World” Bahasa Pemrograman

Pada praktiknya python digunakan untuk beberapa paradigma pemrograman seperti orientasi objek (*object-oriented*), dan pemrograman fungsional (*functional programming*), tidak hanya itu python mendukung manajemen memori yang bersifat dynamic serta python memiliki modul standar atau yang biasanya disebut *library* dalam skala besar. Python memiliki beberapa kelebihan dan kekurangan, kelebihan yang dimiliki bahasa pemrograman python yaitu *syntax* atau penulisan kode pada python sangat simpel dan mudah dibaca dibandingkan dengan bahasa

pemrograman lainnya seperti C, C++, dan Java. Karena python termasuk bahasa *high-level general purpose* python dapat digunakan untuk beberapa pekerjaan seperti *web development*, *game development*, *machine learning & artificial intelligence*, *embedded systems*, dan lain-lain. Sedangkan kelemahan yang dimiliki python adalah lambat, python termasuk bahasa pemrograman yang lambat ketika proses eksekusi atau pada saat program dijalankan serta alokasi memori penggunaan memori pada python sangat tinggi.

