

BAB III

METODOLOGI PENELITIAN

Bab ini membahas mengenai deskripsi masalah, tahapan penelitian, dan tahapan penyelesaian.

3.1 Deskripsi Masalah

Penelitian ini meneliti masalah penjadwalan perkuliahan di Prodi Matematika dan Prodi Pendidikan Matematika S1 FPMIPA UPI Semester Genap Tahun Akademik 2023/2024. Penjadwalan perkuliahan di Prodi Matematika dan Prodi Pendidikan Matematika S1 FPMIPA UPI masih dilakukan secara semimanual, belum ada metode penjadwalan khusus yang digunakan. Selain itu, pembuatan jadwal perkuliahan membutuhkan waktu yang cukup lama sehingga dibutuhkan cara yang dapat membantu mempercepat proses pembuatan jadwal perkuliahan.

Masalah penjadwalan perkuliahan yang akan disusun pada penelitian ini adalah masalah memasang mata kuliah, dosen pengampu, dan kelas yang diampu, terhadap hari, waktu, dan ruangan yang tersedia. Penjadwalan perkuliahan ini disusun untuk mengatur jadwal perkuliahan selama satu pekan. Penyusunan jadwal perkuliahan akan dibuat dengan mengikutsertakan kedua dosen mata kuliah, mata kuliah nonpraktikum, dan mata kuliah praktikum. Dalam penelitian ini penyusunan jadwal perkuliahan akan menggunakan Algoritma *Ant Colony Optimization* (ACO).

3.2 Tahapan Penelitian

Penelitian ini dilakukan melalui tahapan sebagai berikut:

1. Studi Pustaka

Studi Pustaka pada penelitian ini dilakukan dengan cara mempelajari teori-teori yang berkaitan dengan masalah penjadwalan, graf, dan Algoritma ACO dari berbagai sumber berupa jurnal nasional dan internasional, buku, serta skripsi.

2. Pengumpulan Data

Data penelitian yang digunakan terdiri dari data mata kuliah, dosen, kelas, hari, slot waktu, dan ruangan Semester Genap Tahun Akademik 2023/2024 di Prodi Matematika dan Prodi Pendidikan Matematika S1 FPMIPA UPI.

3. Pemodelan

Pada tahap ini model optimisasi untuk masalah penjadwalan perkuliahan dibangun dengan terlebih dahulu mendefinisikan himpunan dan parameter, serta menentukan variabel keputusan, fungsi kendala, dan fungsi tujuan.

4. Penyelesaian Model

Model yang telah dibangun akan diselesaikan menggunakan Algoritma ACO.

5. Validasi

Validasi akan dilakukan dengan tujuan untuk mengecek apakah model dan teknik penyelesaiannya sudah valid atau belum. Validasi dilakukan dengan cara membandingkan solusi optimal yang diperoleh dari implementasi ACO secara komputasi dengan solusi optimal hasil perhitungan manual dari suatu kasus penjadwalan berukuran kecil. Jika kedua solusi adalah sama atau solusi hasil komputasi lebih baik maka model dan teknik penyelesaiannya telah valid, tetapi jika kedua solusinya tidak sama dan solusi hasil komputasi tidak lebih baik maka tahapan akan diulang dari konstruksi model.

6. Implementasi

Jika model dan teknik penyelesaian telah valid maka model dan teknik penyelesaian akan diimplementasikan untuk menyelesaikan masalah penjadwalan perkuliahan di Prodi Matematika dan Prodi Pendidikan Matematika S1 FPMIPA UPI. Selanjutnya, akan dianalisis hasil kinerja dari Algoritma ACO.

7. Penarikan Kesimpulan

Kesimpulan akan diambil berdasarkan hasil dari implementasi Algoritma ACO dalam menyelesaikan penjadwalan perkuliahan di Prodi Matematika dan Prodi Pendidikan Matematika S1 FPMIPA UPI.

3.3 Tahapan Penyelesaian Menggunakan Algoritma *Ant Colony Optimization*

Tahapan penyusunan jadwal perkuliahan menggunakan Algoritma ACO adalah sebagai berikut (Anugrah, 2011):

1. Merepresentasikan Masalah Dalam Bentuk Graf

Masalah penjadwalan akan terlebih dahulu direpresentasikan sebagai graf, di mana setiap simpul pada graf mewakili satu kegiatan perkuliahan dan sisi yang menghubungkan dua simpul diberi nilai penalti yang mencerminkan pelanggaran yang terjadi antara dua kegiatan perkuliahan yang direpresentasikan oleh dua simpul tersebut.

2. Inisialisasi Parameter ACO

Parameter awal diinisialisasi untuk menetapkan nilai-nilai yang diperlukan agar algoritma dapat berjalan. Penentuan nilai parameter awal dapat dengan bebas ditentukan, tetapi agar algoritma dapat berjalan dengan baik, penentuan nilai parameter awal dapat mengacu pada percobaan-percobaan yang sebelumnya telah dilakukan oleh para peneliti (Anugrah, 2011). Parameter-parameter awal yang diinisialisasi adalah intensitas feromon antar simpul (τ_{ij}), jarak antar simpul (d_{ij}), tetapan lintasan semut (Q), tetapan pengendali intensitas feromon (α), tetapan pengendali visibilitas (β), tetapan penguapan feromon (ρ), banyak semut (m), dan jumlah iterasi maksimum (NCmax).

3. Pembentukan Lintasan

Lintasan dikonstruksi melalui tahapan-tahapan sebagai berikut:

- a. Pemilihan Rute Perjalanan

Setiap semut akan menentukan rute perjalanannya sendiri sehingga masing-masing semut akan memiliki lintasan yang berbeda. Pemilihan simpul yang akan dikunjungi oleh setiap semut didasarkan pada perhitungan peluang menggunakan persamaan (2.1). Setiap simpul yang telah dikunjungi semut akan disimpan pada tabu list masing-masing semut.

- b. Perhitungan Panjang Lintasan

Panjang lintasan dari setiap semut akan dihitung untuk menentukan lintasan terpendek yang nantinya akan diubah menjadi jadwal. Panjang lintasan setiap semut dihitung menggunakan persamaan berikut:

$$L_k = \sum_{i=1}^{N-1} T(i, i+1)$$

di mana L_k adalah panjang lintasan semut k dan $T(i, i+1)$ adalah nilai penalti simpul ke- i dan ke- $i+1$ pada tabu list semut k .

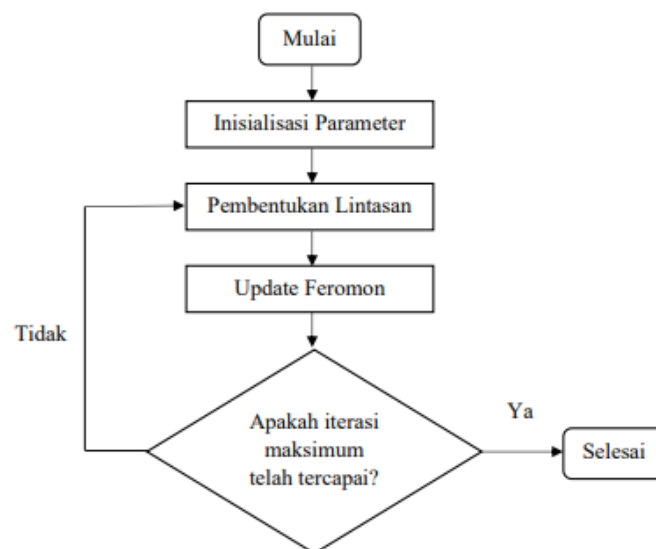
4. *Update Feromon*

Feromon diperbarui setelah menyelesaikan satu iterasi sehingga intensitas feromon antar simpul (τ_{ij}) diperbarui pada setiap iterasi. Nilai intensitas feromon antar simpul (τ_{ij}) untuk iterasi selanjutnya dihitung menggunakan persamaan (2.5).

5. Pengosongan Tabu List

Tabu list dikosongkan sebelum memulai iterasi berikutnya. Setelah tabu list dikosongkan, langkah-langkah diulangi dari pembentukan lintasan. Semua proses dihentikan jika iterasi maksimum (NCmax) yang telah ditentukan pada saat inisialisasi parameter tercapai.

Secara garis besar Algoritma ACO digambarkan dengan diagram pada Gambar 3.1.



Gambar 3.1 *Flowchart* Algoritma ACO

Output dari Algoritma ACO berupa lintasan terpendek. Lintasan terpendek yang dihasilkan akan diubah menjadi jadwal perkuliahan dengan cara menempatkan satu persatu setiap simpul ke jam, hari, dan ruangan yang tersedia.