



**PENYELESAIAN RESOURCE-CONSTRAINED
PROJECT SCHEDULING PROBLEM
MENGUNAKAN ALGORITMA CAT SWARM
OPTIMIZATION**

Skripsi

disajikan sebagai salah satu syarat
untuk memperoleh gelar Sarjana Sains
Program Studi Matematika

UNNES
oleh
UNIVERSITAS NEGERI SEMARANG
Eka Retna Fitriyani

4111413020

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI SEMARANG**

2017

PERNYATAAN


Saya menyatakan bahwa skripsi ini bebas plagiat, kecuali yang secara tertulis dirujuk dalam skripsi ini dan disebutkan dalam daftar pustaka. Apabila di kemudian hari terbukti terdapat plagiat dalam skripsi ini, maka saya bersedia menerima sanksi sesuai ketentuan perundang-undangan.



UNN
UNIVERSITAS NEGERI

Semarang, Desember 2017




Eka Retna Fitriyani
NIM 4111413020

PENGESAHAN

Skripsi yang berjudul

Penyelesaian Resource-Constrained Project Scheduling Problem
Menggunakan Algoritma Cat Swarm Optimization

Disusun oleh

Eka Retna Fitriyani

NIM 4111413020

Telah dipertahankan di hadapan sidang Panitia Ujian Skripsi FMIPA UNNES pada
tanggal 12 Desember 2017.



Prof. Dr. Zaenuri, S. E., M. Si., Akt.
NIP. 196412231988031001

Sekretaris

Drs. Arief Agoestanto, M. Si.
NIP. 196807221993031005

Ketua Penguji

Dr. Tri Sri Noor Asih, S.Si., M.Si.
NIP. 197706142008122002

Anggota Penguji/Pembimbing I

Dr. Isnaini Rosyida, S.Si.,
NIP. 197302191998022001

Anggota Penguji/Pembimbing II

Drs. Mashuri, M.Si.
NIP. 196708101992031003

MOTTO

- ❖ Barang siapa yang bersungguh sungguh akan mendapatkannya.
- ❖ Sekali anda mengerjakan sesuatu, jangan takut gagal dan jangan tinggalkan itu.
Orang-orang yang bekerja dengan ketulusan hati adalah mereka yang paling bahagia (Chanakya)
- ❖ Tak akan lari gunung dikejar.

PERSEMBAHAN

- Ibu dan Bapak tercinta atas kasih sayang, doa, dan dukungannya.
- Adikku Tersayang, yang selalu menyemangati.
- Reno A. Nugroho yang selalu menyertakan semangat dan canda tawanya.
- Sahabat-sahabat baikku Abriana Santi, Agnes Fitriana, Nunik Sutrisni, Yuliyana Fathonah. yang selalu membantu dan memberi dorongan bagi aku.
- Teman-teman Matematika Angkatan 2013.

KATA PENGANTAR

Puji Syukur kehadiran Allah SWT atas segala rahmat dan hidayah-Nya. sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penyelesaian Resource-Constrained Project Scheduling Problem Menggunakan Algoritma Cat Swarm Optimization” Penyelesaian skripsi ini dimaksudkan untuk melengkapi persyaratan agar memperoleh gelar Sarjana Sains Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.

Sehubungan dengan pelaksanaan penelitian sampai tersusunnya skripsi ini, dengan rasa rendah hati disampaikan rasa terima kasih yang sebesar-besarnya kepada yang terhormat:

1. Prof. Dr. Fathur Rokhman, M.Hum., Rektor Universitas Negeri Semarang.
2. Prof. Dr. Zaenuri, S.E., M.Si., Akt., Dekan FMIPA Universitas Negeri Semarang.
3. Drs. Arief Agoestanto, M.Si., Ketua Jurusan Matematika FMIPA Universitas Negeri Semarang.
4. Drs. Mashuri, M.Si., selaku Ketua Prodi Matematika, Dosen Pembimbing II yang telah memberikan bimbingan, pengarahan, nasehat, saran, dan dorongan selama penyusunan skripsi ini.
5. Dr. Isnaini Rosyida S.Si., M.Si. selaku Dosen Pembimbing I yang telah memberikan bimbingan, pengarahan, nasehat, saran, dan dorongan selama penyusunan skripsi ini.
6. Dr. Tri Sri Noor Asih S.Si., M.Si. selaku Dosen Penguji yang telah memberikan penilaian dan saran dalam perbaikan skripsi ini.

7. Staf Dosen Matematika dan Staf Tata Usaha Universitas Negeri Semarang yang telah membekali dengan berbagai ilmu selama mengikuti perkuliahan sampai akhir penulisan skripsi ini.
8. Ibu, Bapak dan adikku tercinta, yang senantiasa memberikan dukungan dan doa yang tiada putusnya.
9. I. W. Radika A. yang senantiasa membantu dan memberikan dukungan dalam menyelesaikan skripsi ini.
10. Reno A. Nugroho yang senantiasa membantu dan memberikan semangat hingga selesainya skripsi.
11. Teman-teman Matematika angkatan 2013 yang berjuang bersama untuk mewujudkan cita-cita.
12. Semua pihak yang tidak dapat disebutkan satu per satu yang telah memberikan bantuan.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat banyak kekurangan. Oleh karena itu, penulis mengharapkan saran dan kritik yang membangun dari pembaca. Semoga amal baik dari semua pihak yang terlibat mendapat pahala yang berlipat dari Allah SWT. Amin.

Semarang, Desember 2017

Penulis

ABSTRAK

Firiyani, Eka R.. 2017. *Penyelesaian Resource-Constrained Project Scheduling Problem Menggunakan Algoritma Cat Swarm Optimization*. Skripsi, Matematika, Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Semarang. Pembimbing I: Dr. Isnaini Rosyida S.Si., M.Si., Pembimbing II: Drs. Mashuri, M.Si.

Kata Kunci: RCPSP, Cat Swarm Optimization, penjadwalan proyek.

Penjadwalan kegiatan proyek merupakan hal yang penting karena menentukan berhasil tidaknya suatu proyek. *Resource-Constrained Project Scheduling Problem* (RCPSP) adalah masalah penjadwalan proyek yang harus memenuhi urutan pengerjaan kegiatan (*precedence constraint*) dan memepertimbangkan sumber daya yang digunakan pada setiap kegiatan agar tidak melebihi kapasitas sumber daya yang tersedia (*resource constraints*).

Penelitian ini membahas mengenai penyelesaian *Resource-Constrained Project Scheduling Problem* (RCPSP) dengan algoritma Cat Swarm Optimization. Tujuan penelitian ini adalah mengetahui susunan penjadwalan proyek yang memiliki waktu minimal dengan *precedence constraint* dan *resource constraint* untuk seluruh aktivitas proyek. Penelitian ini menggunakan algoritma Cat Swarm Optimization (CSO) dalam menyelesaikan permasalahan yang ada dengan bantuan *software* Matlab.

Pada penelitian ini dapat disimpulkan: 1) Penerapan Algoritma Cat Swarm Optimization dalam permasalahan penjadwalan proyek adalah a) Membuat solusi RCPSP yang valid dan dihitung *maskepannya*; b) Bangkitkan sebanyak N kucing, termasuk kucing dengan solusi valid; c) Evaluasi kucing sesuai dengan posisi tiap kegiatan; d) Berdasarkan N kucing yang diperoleh hitung nilai *maskepannya*; e) Pindahkan kucing sesuai dengan MR dalam *seeking mode* dan *tracing mode*; f) Evaluasi kembali posisi setiap kucing; g) Akhiri algoritma dengan mengambil solusi yang memiliki *fitness* terkecil. Menambahkan tahap a, c, dan f pada algoritma CSO membuat solusi yang diperoleh lebih optimal dengan *maskepan* minimum. 2) Algoritma Cat Swarm Optimization dengan bantuan *software* Matlab telah diterapkan pada 2 kasus RCPSP, yaitu : a) Kasus RCPSP dengan solusi valid yang memiliki nilai *maskepan* yang masih jauh dari nilai *maskepan* terkecil yang bisa diperoleh menghasilkan solusi yang berbeda-beda setiap kali program dijalankan sehingga diperlukan beberapa kali simulasi untuk memastikan solusi yang didapatkan benar-benar memiliki *maskepan* terkecil; b) Bila solusi valid yang dibuat memiliki *maskepan* yang telah mendekati *maskepan* terkecil yang bisa diperoleh, maka program akan menghasilkan solusi yang sama setiap program dijalankan.

DAFTAR ISI

| | |
|---------------------------------|------|
| HALAMAN JUDUL | i |
| PERNYATAAN..... | ii |
| PENGESAHAN | iii |
| MOTTO DAN PERSEMBAHAN | iv |
| KATA PENGANTAR | v |
| ABSTRAK | vii |
| DAFTAR ISI | viii |
| DAFTAR TABEL..... | x |
| DAFTAR GAMBAR | xi |
| DAFTAR LAMPIRAN..... | xiii |
| BAB 1 PENDAHULUAN | |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 5 |
| 1.3 Pembatas Masalah | 5 |
| 1.4 Tujuan Penelitian | 6 |
| 1.5 Manfaat Penelitian | 6 |
| 1.6 Sistematika Penulisan | 6 |
| BAB 2 LANDASAN TEORI | |
| 2.1 Riset Operasi | 9 |
| 2.2 Proyek | 12 |
| 2.3 Manajemen proyek | 17 |

| | | |
|-------------------------|--|----|
| 2.4 | Resource Constrained Project Scheduling Problem (RCPSP)..... | 21 |
| 2.5 | Algoritma Cat Swarm Optimization (CSO)..... | 25 |
| 2.6 | Program Simulasi..... | 32 |
| BAB 3 METODE PENELITIAN | | |
| 3.1 | Masalah..... | 36 |
| 3.2 | Studi Pustaka..... | 36 |
| 3.3 | Algoritma CSO | 36 |
| 3.4 | Penerapan Penyelesaian Permasalahan dengan Algoritma CSO.. | 37 |
| 3.5 | Penarikan Kesimpulan | 38 |
| BAB 4 PEMBAHASAN | | |
| 4.1 | Hasil Penelitian | 39 |
| 4.1.1. | Penerapan Algoritma CSO dalam Menyelesaikan Permasalahan Penjadwalan Proyek | 40 |
| 4.1.2. | Simulasi Penjadwalan Proyek yang Optimal pada Beberapa Kasus RCPSP | 58 |
| 4.2 | Pembahasan | 71 |
| BAB 5 PENUTUP | | |
| 5.1 | Simpulan | 75 |
| 5.2 | Saran | 76 |
| DAFTAR PUSTAKA | | 77 |
| LAMPIRAN..... | | 79 |

DAFTAR TABEL

| Tabel | Halaman |
|--|---------|
| 4.1. Kombinasi Kombinasi Parameter yang Digunakan untuk Perhitungan Manual | 46 |
| 4.2. Hasil Kombinasi Parameter Terbaik untuk Simulasi | 61 |
| 4.3. Hasil Simulasi 1 Algoritma CSO pada Data ke-1 | 63 |
| 4.4. Hasil Simulasi 2 Algoritma CSO pada Data ke-1 | 63 |
| 4.5. Hasil Simulasi 1 Algoritma CSO pada Data ke-2 | 64 |
| 4.6. Hasil Simulasi 2 Algoritma CSO pada Data ke-2 | 65 |
| 4.7. Hasil Simulasi Algoritma CSO pada Data ke-3.1 | 67 |
| 4.8. Hasil Simulasi Algoritma CSO pada Data ke-3.2 | 68 |
| 4.9. Hasil Simulasi Algoritma CSO pada Data ke-3.3 | 70 |



DAFTAR GAMBAR

| Gambar | Halaman |
|---|---------|
| 2.1 Hubungan Keperluan Sumber Daya Terhadap Waktu dalam Siklus Proyek..... | 15 |
| 2.2 Hubungan Kegiatan Proyek dalam Jaringan Proyek..... | 18 |
| 2.3 Proyek dengan <i>General Structure</i> | 19 |
| 2.4 Proyek dengan <i>Serial Structure</i> | 19 |
| 2.5 Visualisasi RCPSP | 23 |
| 2.6 Visualisasi Bentuk Jadwal..... | 24 |
| 2.7 <i>Flowchart</i> | 31 |
| 3.1 Alur Kerja..... | 35 |
| 4.1 Visualisasi RCPSP Data 1 dengan 6 Kegiatan 1 Jenis Sumber Daya | 41 |
| 4.2 Visualisasi Bentuk Jadwal Solusi Valid Data ke-1 | 42 |
| 4.3 Tampilan Gui Matlab | 58 |
| 4.4 Tampilan GUI Matlab Hasil Simulasi | 59 |
| 4.5 Visualisasi Bentuk Jadwal Hasil Simulasi Data 1..... | 62 |
| 4.6 Visualisasi RCPSP Data 2 dengan 7 Kegiatan 1 Jenis Sumber Daya | 64 |
| 4.7 Visualisasi Bentuk Jadwal Hasil Simulasi Data Ke-2..... | 66 |
| 4.8 Visualisasi RCPSP Data 3 dengan 25 kegiatan 3 Jenis Sumber Daya | 66 |

| | | |
|------|--|----|
| 4.9 | Visualisasi Bentuk Jadwal Hasil Simulasi Data Ke-3.1..... | 68 |
| 4.10 | Visualisasi Bentuk Jadwal Hasil Simulasi Data Ke-3.2..... | 69 |
| 4.11 | Visualisasi Bentuk Jadwal Hasil Simulasi Data Ke-3.3 | 71 |



DAFTAR LAMPIRAN

| Lampiran | Halaman |
|---|---------|
| 1. Data 3 yang Telah Dipisahkan Berdasarkan Sumber Daya | 79 |
| 2. Perhitungan Manual Data 2 | 81 |
| 3. Visualisasi Bentuk Jadwal Solusi Valid Data 2 | 92 |
| 4. Visualisasi Bentuk Jadwal Solusi Valid Data 3 | 92 |
| 5. Hasil Perhitungan Menggunakan Kombinasi Parameter yang Berbeda-Beda..... | 94 |
| 6. Hasil Simulasi dengan Solusi Valid lain dari Data 1 | 96 |
| 7. Simulasi Program Data 1 | 97 |
| 8. Simulasi Program Data 2 | 101 |
| 9. Simulasi Program Data 3.1 | 105 |
| 10. Simulasi Program Data 3.2 | 107 |
| 11. Simulasi Program Data 3.3 | 109 |
| 12. <i>Source Code</i> | 111 |

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Proyek merupakan kegiatan yang berlangsung dalam jangka waktu yang terbatas dengan mengalokasikan sumber daya tertentu dan dimaksudkan untuk menghasilkan produk atau *deliverable* yang kriteria mutunya telah digariskan dengan jelas (Soeharto,1999). Majunya peradapan manusia semakin membuat proyek yang dikerjakan besar dan kompleks yang melibatkan penggunaan bahan-bahan (material), tenaga kerja, dan teknologi yang semakin canggih. Proyek pada umumnya memiliki batas waktu (*deadline*), artinya proyek harus diselesaikan sebelum atau tepat pada waktu yang telah ditentukan.

Sebelum proyek dilaksanakan ada beberapa tahap pengelompokan yang meliputi perencanaan, penjadwalan, dan pengkoordinasian. Dari ketiga tahap tersebut tahap penjadwalan dan perencanaan merupakan tahapan yang menentukan berhasil tidaknya suatu proyek karena penjadwalan adalah tahap ketergantungan antar aktivitas yang membangun proyek secara keseluruhan (Arifudin. 2011). Pemecahan masalah penjadwalan yang baik merupakan salah satu faktor keberhasilan pelaksanaan sebuah proyek tepat pada waktunya yang merupakan tujuan yang penting baik bagi pemilik proyek maupun kontraktor.

Pendekatan penjadwalan proyek secara tradisional seperti *Critical Path Method* (CPM) dan *Program Evaluation and Review Technique* (PERT) hanya berfokus pada hubungan ketergantungan antar aktivitas dengan menggunakan asumsi ketersediaan sumber daya yang tidak terbatas. Pada kenyataannya dalam pengerjaan suatu proyek sumber daya yang tersedia sangatlah terbatas guna meminimalisasi pengeluaran sehingga manajer proyek sering sulit untuk melakukan penjadwalan proyek.

Masalah penjadwalan aktivitas-aktivitas pada proyek dengan kendala sumber daya terbatas sering disebut dengan *Resource Constrained Project Scheduling Problem* (RCPSP). Setiap aktivitas memiliki durasi aktivitas dan jumlah kebutuhan sumber daya masing-masing. RCPSP dapat didefinisikan sebagai masalah optimasi kombinatorial yang sulit untuk dipecahkan. Berbagai metode seperti metode heuristik dan metode metaheuristik telah dicoba untuk menyelesaikan kasus RCPSP.

Metode analitik sering kali mengadopsi model-model matematika seperti *integer programming* (Talbot, 1982) dan *dynamic programming* (Gavish, 1991) untuk mencari solusi yang optimal. Namun demikian, metode analitis tidak memungkinkan untuk diterapkan apabila masalah yang menjadi obyek penelitian berukuran sangat besar dan sangat kompleks (Lee, 1996). Penerapan metode analitik B&B pada optimalisasi penjadwalan proyek juga tidak efisien sehingga berdampak pada keterlambatan penyelesaian pembangunan proyek (Widyawati, 2014).

Metode heuristik digunakan untuk menyelesaikan pencarian solusi RCPSP secara optimal dan efisien. Metode heuristik yang ada menggunakan aturan prioritas seperti *Shortest Activity Duration* (SAD), *Minimum Late Finish Time* (MILFT), atau *Minimum Total Float* (MITF) untuk menentukan aktivitas mana yang akan dijadwalkan terlebih dahulu. Namun demikian, tidak ada aturan prioritas yang mendominasi yang lain atau menunjukkan hasil yang lebih baik dari pada yang lain secara konsisten (Davis, 1975).

Seiring berjalannya waktu, telah banyak penelitian yang menggunakan metode metaheuristik untuk menyelesaikan *Resource Constrained Project Scheduling Problem* (RCPSP) seperti Algoritma *Harmony Search* (Putra, 2013), Algoritma *Cross Entropy* (Krisnawati, 2014), Algoritma Genetika (Chan, 1996), dan Algoritma *Particle Swarm Optimization* (Zhang, 2006). Algoritma *Harmony Search* harus dilakukan dengan tujuh langkah di antaranya: inisialisasi masalah, memasukan data RCPSP, inisialisasi parameter Algoritma *Harmony Search*, Inisialisasi *harmony memory*, membangkitkan vektor solusi baru, meng-update *harmony memory*, mengecek criteria pemberhentian. Dibandingkan dengan Algoritma CPM, solusi Algoritma *Harmony Search* tidak selalu merupakan solusi terbaik, karena pencarian solusi dilakukan secara random, sehingga solusi yang diperoleh sangat beragam dan memerlukan waktu yang lebih lama yang bergantung pada kriteria pemberhentian yang dipilih. Algoritma *Cross Entropy* melibatkan prosedur iterasi, dimana tiap iterasi dapat dipecahkan menjadi dua fase yaitu pembangkitan sampel random dan pembaharuan

parameter. Penggunaan jumlah aktivitas proyek 30-60 memperlihatkan bahwa Algoritma *Cross Entropy* mempunyai performansi yang sama dengan penyelesaian menggunakan Algoritma *Differential Evolution*. Algoritma Genetika melakukan pencarian solusi yang optimal dari kromosom-kromosom yang merepresentasikan jadwal dimana kromosom-kromosom tersebut diproduksi melalui *cross-over* dan *mutation*. Mekanisme *updating* kromosom membuat Algoritma Genetika mampu keluar dari solusi yang bersifat lokal optimal. Oleh karena itu, Algoritma Genetika lebih unggul dari pada metode analitis dan metode heuristik. Namun demikian, terdapat kekurangan dalam performansi Algoritma Genetika yaitu proses konvergensi yang lambat. Sedangkan performansi *Particle Swarm Optimization* menunjukkan kemampuan untuk mencari optimum global dan lebih efisien dari pada Algoritma Genetika karena fiturnya (Zhang, 2006).

Davis (1972) menyatakan bahwa terdapat kesamaan yang kuat antara masalah penjadwalan proyek dengan *job shop*. *Job shop* merupakan permasalahan penjadwalan operasi-operasi mesin dengan waktu penyelesaian seluruh pekerjaan (*maskepan*) yang minimum. Pada penelitian terdahulu telah digunakan Algoritma *Cat Swarm Optimization* (CSO) untuk menyelesaikan masalah *job shop* (Bouzidi, 2014). Data yang digunakan bervariasi dari 6 sampai 30 *job* dan 4 sampai 15 mesin. Hasil dari penelitian ini adalah algoritma CSO dapat menghasilkan solusi yang baik dilihat dari waktu yang diperlukan dan solusi terbaik yang dicapai.

Pada pengalokasian *sink node* di lingkungan sensor jaringan nirkabel, algoritma CSO dimanfaatkan untuk mengetahui lokasi terbaik menempatkan *sink node* sehingga dapat mengurangi konsumsi energi dan memperpanjang masa pakai jaringan secara efisien (Tsai, 2016). Penerapan algoritma CSO dengan beberapa penyempurnaan pada permasalahan penjadwalan proyek yang dilakukan oleh Xu (2012) menunjukkan bahwa algoritma CSO memiliki kinerja yang rata-rata lebih baik dibandingkan algoritma lainnya.

Algoritma CSO pertama kali diperkenalkan oleh Shu-Chuan Chu dan Pei-Wei Tsai (Taiwan) pada tahun 2006. Dasar pemikiran algoritma ini adalah memanfaatkan kombinasi perilaku kucing yaitu *seeking mode* dan *tracing mode* untuk memecahkan masalah optimasi. *Seeking mode* menggambarkan keadaan kucing pada saat beristirahat, melihat kondisi sekitarnya mencari posisi berikutnya untuk bergerak. *Tracing mode* menggambarkan keadaan ketika kucing sedang mengikuti jejak targetnya.

Dari enam tes yang dilakukan oleh Shu-Chuan Chu dan Pei-Wei Tsai (2006), menunjukkan kinerja Algoritma *Cat Swarm Optimization* lebih efektif dibandingkan dengan Algoritma *Particle Swarm Optimization* dan Algoritma *Particle Swarm Optimization with weighting*. Algoritma CSO dapat menghasilkan klasifikasi yang lebih baik dalam hal jumlah iterasi yang dibutuhkan untuk mencapai titik optimal dan memiliki tingkat akurasi yang lebih baik (Dhanasaputra, 2010)

Penelitian ini membahas tentang masalah *Resource Constrained Project Scheduling Problem* (RCPSP) menggunakan Algoritma *Cat Swarm*

Optimization (CSO). Berdasarkan uraian tersebut penulis mengambil judul “Penyelesaian *Resource Constrained Project Scheduling Problem* (RCPSP) menggunakan Algoritma *Cat Swarm Optimization*”

1.2 Rumusan Masalah

1. Bagaimana penerapan Algoritma *Cat Swarm Optimization* (CSO) dalam menyelesaikan permasalahan penjadwalan proyek?
2. Bagaimana simulasi penjadwalan proyek yang optimal pada beberapa kasus RCPSP?

1.3 Pembatasan Masalah

1. Durasi aktivitas diketahui tetap dan diukur dengan satuan hari.
2. Tidak adanya tindakan pencegahan aktivitas.
3. Jumlah dan tipe sumber daya yang dibutuhkan untuk setiap aktivitas diketahui tetap.
4. *Mode* yang digunakan adalah *mode* tunggal.
5. Dalam penelitian ini yang dioptimalkan adalah waktu penyelesaian aktivitas.

1.4 Tujuan Penelitian

1. Mengetahui penerapan penggunaan Algoritma *Cat Swarm Optimization* untuk penjadwalan proyek.
2. Mengetahui susunan penjadwalan yang optimal pada beberapa kasus RCPSP.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini sebagai berikut:

1.5.1. Bagi penulis

1. Dapat mengetahui penggunaan Algoritma Cat Swarm Optimization.
2. Menambah wawasan *tentang Resource constrained project scheduling problem*.

1.5.2. Bagi pembaca

1. Pembaca diharapkan dapat mengetahui manfaat terapan ilmu matematika dalam kehidupan sehari-hari.
2. Pembaca diharapkan dapat menerapkan terapan-terapan ilmu matematika dalam kehidupan sehari-hari, khususnya Algoritma *Cat Swarm Optimization*.

1.6 Sistematika Penulisan

Secara garis besar skripsi ini terdiri dari tiga bagian, yaitu bagian awal, bagian isi, bagian akhir.

Bagian awal berisi tentang halaman judul, abstrak, halaman pengesahan, motto dan persembahan, kata pengantar, daftar isi, daftar tabel, daftar gambar, dan daftar lampiran.

Bagian isi terdiri dari lima bab yaitu:

1. BAB I

Pada bab I berisi tentang latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika skripsi.

2. BAB II

Bab II berisi tentang landasan teori yang mendasari pemecahan dari permasalahan yang disajikan. Pada bab ini dibagi menjadi beberapa subbab, yaitu : riset operasi, proyek, manajemen proyek, *Resource Constrained Project Scheduling Problem (RCPS)*, Algoritma *Cat Swarm Optimization (CSO)*, program simulasi.

3. BAB III

Bab ini berisi tentang masalah, studi pustaka, Algoritma *Cat Swarm Optimization*, penerapan Penerapan algoritma CSO untuk beberapa kasus permasalahan RCPS, penarikan kesimpulan.

4. BAB IV

Dalam bab ini dibagi menjadi dua subbab yaitu hasil penelitian dan pembahasan. Hasil penelitian berisi tentang penyelesaian beberapa kasus masalah penjadwalan proyek menggunakan Algoritma *Cat Swarm Optimization*. Pada pembahasan berisi tentang analisis penjadwalan pada beberapa kasus dengan Algoritma *Cat Swarm Optimization*.

5. BAB V

Dalam bab ini berisi simpulan serta saran yang berkaitan dengan penelitian.

Pada bagian akhir skripsi, berisi daftar pustaka dan lampiran-lampiran yang mendukung skripsi ini.

BAB II

LANDASAN TEORI

2.1. Riset Operasi

Riset operasi banyak diterapkan dalam menyelesaikan masalah-masalah manajemen untuk meningkatkan produktivitas dan efisiensi. Riset operasi atau *Operations Research* (OR) terdiri dari kata *operations* dan *research*. *Operations* adalah tindakan-tindakan yang diterapkan pada beberapa masalah atau hipotesa, sedangkan *research* adalah suatu proses yang terorganisasi dalam mencari kebenaran akan masalah atau hipotesa.

Riset operasi merupakan peralatan manajemen yang menyatukan ilmu pengetahuan matematika dan logika dalam rangka memecahkan masalah-masalah yang dihadapi sehari-hari sehingga akhirnya permasalahan tersebut dapat dipecahkan secara optimal (Subagyo, 1999:4).

Pada dasarnya riset operasi merupakan penerapan metode-metode ilmiah terhadap masalah-masalah rumit yang muncul dalam pengarahannya dan pengelolaan dari suatu sistem besar manusia, mesin, bahan, uang dalam industri, bisnis, pemerintahan, dan pertahanan. Pendekatan ini bertujuan membentuk suatu model ilmiah dari sistem, menggabungkan ukuran-ukuran faktor-faktor seperti kesempatan dan resiko, untuk meramalkan dan membandingkan hasil-hasil dari beberapa keputusan menentukan kebijakan dan tindakan secara ilmiah (Mulyono, 2002:2).

2.1.1. Model-Model Riset Operasi

Model adalah gambaran ideal dari suatu (dunia) nyata sehingga sifatnya kompleks dapat disederhanakan (Dimiyati, 2004:3). Ada beberapa jenis yang biasanya digunakan diantaranya ialah:

1. Model Ikonik

Merupakan model tiruan fisik seperti bentuk aslinya dengan skala yang berbeda, seperti model gabungan, model pesawat, dan lain-lain.

2. Model Analog

Merupakan model fisik tetapi tidak memiliki bentuk yang mirip dengan yang dimodelkan, seperti gambar pengukuran temperatur.

3. Model Simbolik

Merupakan model yang menggunakan simbol-simbol huruf, angka bentuk, atau gambar yang menyajikan karakteristik dan properti dari suatu sistem jaringan kerja, *flowchart*, dan lain-lain.

4. Model Simulasi

Merupakan model yang meniru tingkah laku sistem dengan mempelajari interaksi komponen-komponen.

5. Model Heuristik

Merupakan metode pencarian didasarkan atas intuisi atau aturan-aturan empiris untuk memperoleh solusi yang lebih baik dari pada solusi yang telah dicapai.

2.1.2. Langkah-Langkah Penyelesaian Riset Operasi

Aminudin (2005: 6) menyatakan bahwa didalam pemecahan suatu masalah riset operasi, dibutuhkan langkah-langkah prosedur analisis untuk optimasi, yaitu:

1. Definisi Masalah

Pada langkah pertama ini terdiri dari tiga unsur utama yang harus diidentifikasi, yaitu :

- a. Fungsi tujuan, yaitu penetapan tujuan untuk membantu mengarahkan upaya memenuhi tujuan yang dicapai.
- b. Fungsi batasan/kendala, yaitu batasan-batasan yang mempengaruhi persoalan terhadap tujuan yang akan dicapai.
- c. Variabel keputusan, yaitu variabel-variabel yang mempengaruhi persoalan dan pengambilan keputusan.

2. Pengembangan Masalah

Pada langkah kedua ini, yang harus dilakukan adalah mengumpulkan data untuk menaksir besaran parameter yang berpengaruh terhadap persoalan yang dihadapi. Hasil taksiran ini digunakan untuk membangun dan mengevaluasi model matematis dan persoalannya.

3. Pemecahan Model

Pada formulasi ini digunakan model analitis yaitu, model matematis yang menghasilkan persamaan, sehingga dicapai pemecahan yang optimum. Pemecah menggunakan metode yang bersesuaian dengan model yang terbentuk.

4. Pengujian Model

Pemecahan atau solusi yang diperoleh diuji apakah model yang dibangun telah menggambarkan keadaan nyata secara akurat. Jika belum model diperbaiki atau dibentuk model yang baru.

5. Implementasi Solusi

Langkah ini menerjemahkan hasil studi atau perhitungan kedalam bahasa sehari-hari agar mudah dimengerti.

2.2. Proyek

Kegiatan proyek dapat diartikan sebagai satu kegiatan sementara yang berlangsung dalam jangka waktu terbatas, dengan alokasi sumber daya tertentu dan dimaksudkan untuk menghasilkan produk atau *deliverable* yang kriteria mutunya telah digariskan dengan jelas (Soeharto, 1999.2).

Menurut Subagya (2000), proyek adalah suatu pekerjaan yang memiliki tanda-tanda khusus sebagai berikut:

1. Waktu mulai dan selesainya sudah direncanakan.
2. Merupakan suatu kesatuan pekerjaan yang dapat dipisahkan dari yang lain.
3. Biasanya volume pekerjaan besar dan hubungan antar aktivitas kompleks.

2.2.1. Ciri-Ciri Proyek

Berdasarkan pengertian proyek di atas, ciri-ciri proyek antara lain :

- a. Memiliki tujuan tertentu berupa hasil kerja akhir.
- b. Sifatnya sementara karena siklus proyek relatif pendek.

- c. Dalam proses pelaksanaannya, proyek dibatasi oleh jadwal, anggaran biaya, dan mutu hasil akhir.
- d. Merupakan kegiatan nonrutin, tidak berulang-ulang.
- e. Keperluan sumber daya berubah, baik macam maupun volumenya.

Sebuah proyek mendefinisikan satu kombinasi kegiatan-kegiatan yang saling berkaitan yang harus dilakukan dalam urutan tertentu sebelum keseluruhan tugas dapat diselesaikan. Kegiatan-kegiatan ini saling berkaitan dalam satu urutan yang logis dalam arti bahwa beberapa kegiatan tidak dapat dimulai sampai kegiatan-kegiatan lainnya diselesaikan. Sebuah kegiatan (aktivitas) dalam sebuah proyek biasanya dipandang sebagai sebuah tugas yang memerlukan waktu sumber daya untuk penyelesaiannya. Pada umumnya, sebuah proyek adalah satu usaha satu kali, yaitu : urutan kegiatan yang sama kemungkinan tidak diulangi di masa mendatang (Arifudin, 2011).

2.2.2. Jenis-Jenis Proyek

Menurut Soeharto (1999), proyek dapat dikelompokkan menjadi :

- a. Proyek *Engineering*-Konstruksi

Terdiri dari pengkajian kelayakan, desain *engineering*, pengadaan, dan konstruksi. Contoh proyek macam ini adalah pembangunan jalan, jembatan, gedung, fasilitas industri, dan lain-lain.

- b. Proyek *Engineering*-Manufaktur

Dimaksudkan untuk membuat produk baru, meliputi pengembangan produk, manufaktur, perakitan, uji coba fungsi dan operasi produk yang

dihasilkan. Contohnya adalah pembuatan ketel uap, generator listrik, mesin pabrik, dan lain-lain.

c. Proyek Penelitian dan Pengembangan

Bertujuan untuk melakukan penelitian dan pengembangan dalam rangka menghasilkan produk tertentu.

d. Proyek Pelayanan Manajemen

Proyek pelayanan manajemen tidak memberikan hasil dalam bentuk fisik, tetapi laporan akhir, misalnya merancang sistem informasi manajemen.

e. Proyek Kapital

Proyek kapital merupakan proyek yang berkaitan dengan penggunaan dana kapital untuk investasi.

f. Proyek Radio-Telekomunikasi

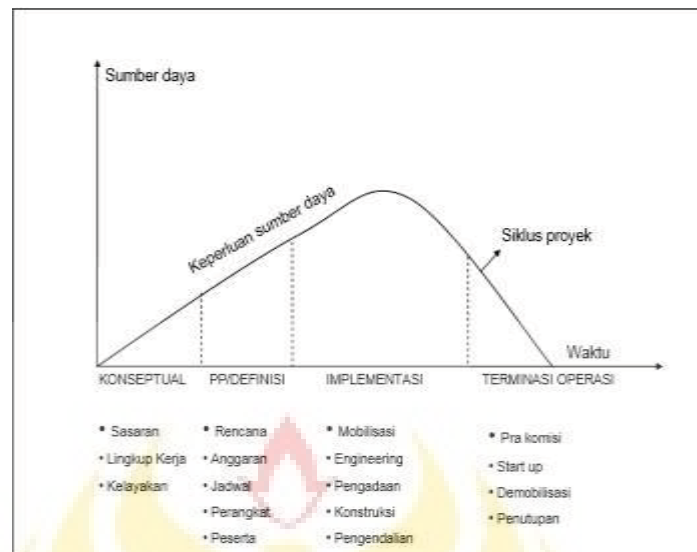
Bertujuan untuk membangun jaringan telekomunikasi yang dapat menjangkau area yang luas dengan biaya minimal.

g. Proyek Konservasi Bio-Diversity

Proyek konservasi bio-diversity merupakan proyek yang berkaitan dengan usaha pelestarian lingkungan.

2.2.3. Tahap Siklus Proyek

Kegiatan-kegiatan dalam sebuah proyek berlangsung dari titik awal, kemudian jenis dan intensitas kegiatannya meningkat hingga ke titik puncak, turun, dan berakhir, seperti ditunjukkan dalam Gambar 2.1. Kegiatan-kegiatan tersebut memerlukan sumber daya yang berupa jam-orang (*man-hour*), dana, material atau peralatan (Soeharto, 1999).



Gambar 2.1. Hubungan Keperluan Sumber Daya Terhadap Waktu dalam Siklus Proyek (Sumber : Manajemen Proyek, Dari Konseptual sampai Operasional, 1999)

Salah satu sistematika penahapan yang disusun oleh PMI (*Project Management Institute*) terdiri dari tahap-tahap konseptual, perencanaan dan pengembangan (PP/Definisi), implementasi, dan terminasi.

a) Tahap Konseptual

Dalam tahap konseptual, dilakukan penyusunan dan perumusan gagasan, analisis pendahuluan, dan pengkajian kelayakan. *Deliverable* akhir pada tahap ini adalah dokumen hasil studi kelayakan.

b) Tahap PP/Definisi

Kegiatan utama dalam tahap PP/Definisi adalah melanjutkan evaluasi hasil kegiatan tahap konseptual, menyiapkan perangkat (berupa data, spesifikasi teknik, *engineering*, dan komersial), menyusun perencanaan

dan membuat keputusan strategis, serta memilih peserta proyek. *Deliverable* akhir pada tahap ini adalah dokumen hasil analisis lanjutan kelayakan proyek, dokumen rencana strategis dan operasional proyek, dokumen anggaran biaya, jadwal induk, dan garis besar kriteria mutu proyek.

c) Tahap Implementasi

Pada umumnya, tahap implementasi terdiri dari kegiatan *design-engineering* yang rinci dari fasilitas yang hendak dibangun, pengadaan material dan peralatan, manufaktur atau pabrikasi, dan instalasi atau konstruksi. *Deliverable* akhir pada tahap ini adalah produk atau instalasi proyek yang telah selesai.

d) Tahap Terminasi

Kegiatan pada tahap terminasi antara lain mempersiapkan instalasi atau produk beroperasi (uji coba), penyelesaian administrasi dan keuangan lainnya. *Deliverable* akhir pada tahap ini adalah instalasi atau produk yang siap beroperasi dan dokumen pernyataan penyelesaian masalah asuransi, klaim, dan jaminan.

e) Tahap Operasi atau *Utilitas*

Dalam tahap ini, kegiatan proyek berhenti dan organisasi operasi mulai bertanggung jawab atas operasi dan pemeliharaan instalasi atau produk hasil proyek.

2.3. Manajemen proyek

Menurut Kerzner dalam Soeharto (1999), manajemen proyek adalah merencanakan, mengorganisir, memimpin, dan mengendalikan sumber daya perusahaan untuk mencapai sarana jangka pendek yang telah ditentukan. Lebih jauh, manajemen proyek menggunakan pendekatan sistem dan hirarki (arus kegiatan) vertikal dan horisontal.

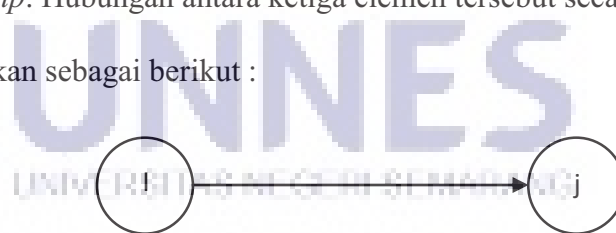
Menurut Soeharto (1999), sistem manajemen proyek bertujuan untuk dapat menjalankan setiap proyek secara efektif dan efisien sehingga dapat memberikan pelayanan maksimal bagi semua pelanggan. Tujuan dari manajemen proyek yaitu bagaimana sumber daya yang tersedia (pekerja, mesin, material dan uang) dapat diaplikasikan dengan baik.

Pada manajemen proyek, sebelum proyek dikerjakan perlu adanya tahap-tahap pengelolaan proyek yang meliputi tahap perencanaan, tahap penjadwalan, dan tahap pengkoordinasian. Dari ketiga tahapan ini, tahap perencanaan dan penjadwalan adalah tahap yang paling menentukan berhasil/tidaknya suatu proyek, karena penjadwalan adalah tahap ketergantungan antar aktivitas yang membangun proyek secara keseluruhan. Penjadwalan sendiri harus disusun secara sistematis dengan penggunaan sumber daya secara efektif dan efisien agar tujuan proyek bisa tercapai secara optimal (Arifudin, 2011). Masalah akan timbul apabila terjadi ketidaksesuaian antara rencana awal dengan realitas yang ada dalam pelaksanaan proyek.

Penjadwalan proyek adalah pembuatan rencana pelaksanaan setiap kegiatan di dalam suatu proyek dengan mengoptimalkan efisiensi pemakaian waktu dan sumber daya yang tersedia, tetapi kesesuaian presedensi di antara kegiatan tetap dipenuhi (Arifudin, 2011).

Upaya untuk merencanakan dan menjadwalkan proyek dengan analisis jaringan pertama kali dikembangkan oleh Henry L. Gantt pada periode perang dunia I menggunakan metode *Gantt Chart*. Pada tahun 1950 dengan perkembangan teknologi teknik penjadwalan proyek mulai dikembangkan. Untuk meminimalkan resiko pada proyek diperkenalkan teknik baru yaitu jaringan (*network*) seperti CPM (*Critical Path Method*) dan PERT (*Program Evaluation and Review Technique*) mulai dikembangkan.

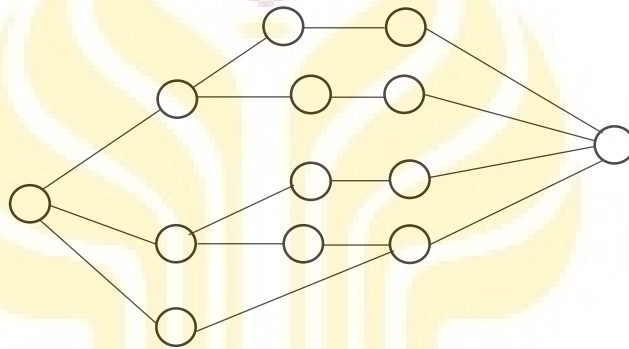
Pada dasarnya suatu jaringan proyek dimulai dari satu kejadian awal, dan diakhiri dengan satu kejadian akhir. Suatu jaringan proyek pada prinsipnya terdiri dari 3 elemen utama, yaitu aktivitas, *event*, dan *precedence relationship*. Hubungan antara ketiga elemen tersebut secara sederhana dapat digambarkan sebagai berikut :



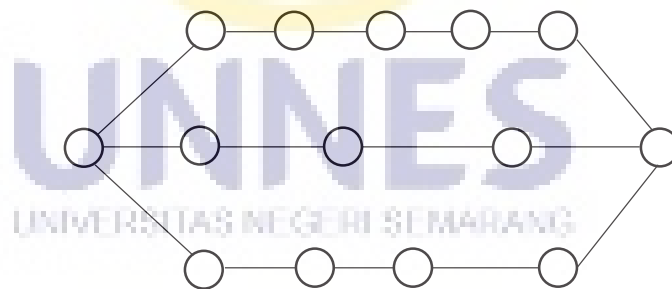
Gambar 2.2 Hubungan kegiatan proyek dalam jaringan proyek

Dari Gambar 2.2, kedua node (lingkaran) *i* dan *j* menggambarkan kegiatan (Aktivitas) penyusunan proyek. *Event* digambarkan dengan anak panah yang menghubungkan antara 2 kegiatan proyek. Kegiatan *i* merupakan kegiatan yang harus dikerjakan terlebih dahulu sebelum kegiatan *j*. Kegiatan

j tidak dapat dikerjakan apabila kegiatan i belum selesai dikerjakan. Hubungan ketergantungan antara i dan j disebut dengan *Precedence relationship*. Suatu node dapat menjadi tempat saat mulainya satu atau lebih kegiatan, dan dapat menjadi tempat saat berakhirnya satu atau lebih kegiatan. Secara umum hubungan antara 2 kegiatan atau lebih yang digambarkan dalam jaringan proyek dapat dibagi menjadi proyek dengan *general structure* dan proyek dengan *serial structure*.



Gambar 2.3 proyek dengan *general structure*



Gambar 2.4 proyek dengan *serial structure*

Penyusunan dan penjadwalan proyek dengan menggunakan teknik jaringan proyek harus menyusun *network planning* dan *network diagram* terlebih dahulu. *Network planning* merupakan suatu model perencanaan penyelenggaraan proyek yang berisi informasi-informasi mengenai kegiatan-

kegiatan yang ada, macam dan jumlah sumber daya yang diperlukan, dan jadwal pelaksanaannya. Sedangkan *network diagram* adalah visualisasi proyek berdasarkan *network planning* yang berupa jaringan kerja, jaringan kerja itu sendiri berisi lintasan-lintasan kegiatan dan urutan kejadian yang terjadi selama penyelenggaraan proyek.

Masalah penjadwalan proyek dapat dilihat dari beberapa sudut pandang seperti tujuannya, bentuk-bentuk sumber daya yang diperlukan pada setiap aktivitas proyek, dan ada tidaknya kondisi *preemption*. Berdasarkan bisa atau tidaknya sumber daya diperbarui, terdapat dua jenis sumber daya yang digunakan oleh kegiatan proyek yaitu sumber daya yang dapat diperbarui dan sumber daya yang tidak dapat diperbarui. Sumber daya yang dapat diperbarui merupakan sumber daya yang dapat digunakan kembali setelah suatu kegiatan proyek selesai misalnya tenaga kerja. Sedangkan sumber daya yang tidak dapat diperbarui adalah sumber daya yang tidak dapat digunakan setelah diselesaikannya kegiatan proyek misalnya uang dan material. Kondisi *preemption* berarti kondisi dimana kegiatan yang sedang berjalan dapat dihentikan dengan alasan tertentu. Sedangkan kondisi *non preemption* berarti kondisi dimana kegiatan yang sedang berjalan tidak dapat dihentikan.

Penjadwalan proyek dapat dibagi menjadi 2 kondisi umum, yaitu penjadwalan proyek dengan kondisi ketersediaan sumber daya tak terbatas dan penjadwalan proyek dengan ketersediaan sumber daya dalam jumlah terbatas (*resource constraint project scheduling problem* – RCPSP).

Penjadwalan proyek dengan jumlah sumber daya yang tak terbatas merupakan penjadwalan proyek yang tidak mempertimbangkan jumlah sumber daya yang tersedia mencukupi atau tidak, penjadwalan proyek ini hanya harus memenuhi urutan pengerjaan kegiatan (*precedence constraint*). Pada umumnya penjadwalan proyek yang seperti ini hanya bertujuan meminimalkan waktu penyelesaian kegiatan proyek secara menyeluruh. Namun penerapan penjadwalan yang tak terbatas sumber dayanya tidak secara luas digunakan. Penjadwalan proyek dengan ketersediaan sumber daya terbatas adalah penjadwalan proyek yang harus memenuhi urutan pengerjaan kegiatan (*precedence constraint*) dan mempertimbangkan sumber daya yang digunakan pada setiap kegiatan agar tidak melebihi kapasitas sumber daya yang tersedia (*resource constraints*).

2.4. Resource Constrained Project Scheduling Problem (RCPSP)

RCPSP terdiri dari dua variasi yaitu *Single-Mode Resource-Constrained Project Scheduling Problem* (SMRCPSP) dan *Multi-Mode Resource-Constrained Project Scheduling Problem* (MMRCPSP). Pada SMRCPSP, setiap aktivitas proyek memiliki mode eksekusi tunggal. Durasi aktivitas dan kebutuhan terhadap suatu set sumber daya diasumsikan tetap. Fungsi tujuan secara umum pada SMRCPSP adalah meminimasi durasi proyek. Sedangkan pada MMRCPSP, setiap aktivitas proyek dapat dieksekusi dalam beberapa mode. Setiap mode menggambarkan suatu set sumber daya yang diperlukan oleh aktivitas proyek dan durasi aktivitas proyek tersebut.

Fungsi tujuan secara umum pada MMRCPSP adalah meminimasi durasi proyek.

Markle (2002) menguraikan RCPSP secara matematis sebagai berikut: $J = \{0, \dots, n + 1\}$ yang menyatakan sebagai himpunan kegiatan sebuah proyek. Himpunan Q adalah himpunan dari q jenis sumber daya. Kapasitas sumber daya $R_i > 0$ dari jenis sumber daya $i \in Q$. Setiap kegiatan $j \in J$ memiliki durasi waktu p dan membutuhkan sumber daya $r_{j,1}, \dots, r_{j,q}$, dimana $r_{j,i}$ adalah sumber daya yang dibutuhkan dengan jenis i per satuan waktu saat kegiatan j dijadwalkan.

Himpunan dari kegiatan yang harus dikerjakan langsung sebelum kegiatan j dilakukan (P_j). Sedangkan (S_j) merupakan himpunan semua kegiatan yang dilakukan setelah kegiatan j berakhir. Kegiatan 0 adalah satu-satunya kegiatan yang tidak memiliki kegiatan sebelumnya dan kegiatan $n + 1$ adalah satu-satunya kegiatan yang tidak memiliki kegiatan setelahnya. Kegiatan awal dan kegiatan akhir tidak memiliki sumber daya yang harus terpenuhi (syarat sumber daya) dan durasi kegiatannya nol.

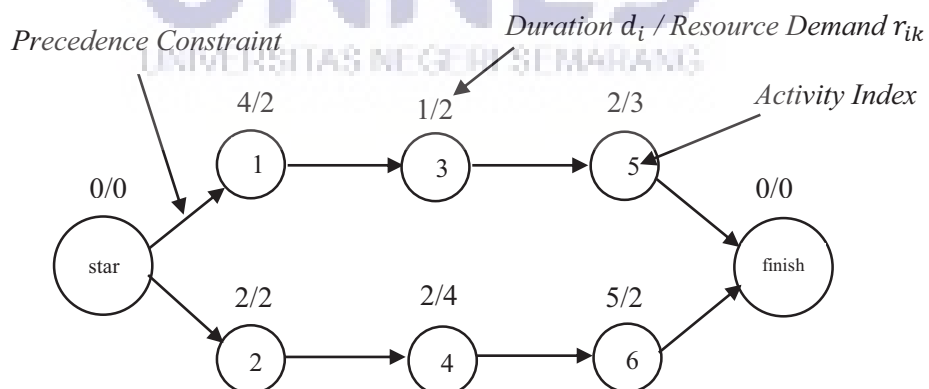
Jadwal proyek dipresentasikan oleh vektor $(s_0, s_1, \dots, s_{n+1})$ dimana s_j merupakan waktu di mulainya dari kegiatan j , maka $f_j = s_j + p_j$ adalah waktu selesainya. Untuk jadwal, waktu mulai adalah waktu minimum $\min\{s_j | j \in J\}$ dan waktu selesai adalah waktu maksimum $\max\{s_j + p_j | j \in J\}$ untuk semua kegiatan. Jarak antara waktu mulai dan waktu selesai merupakan waktu penyelesaian proyek yang sering disebut *maskepan* proyek.

Penjadwalan dimungkinkan apabila memenuhi batasan sebagai berikut:

1. Kegiatan $j \in J$ tidak boleh dimulai sebelum semua kegiatan pendahulunya selesai, yaitu $s_j \geq s_i + p_i$ untuk setiap $i \in P_j$
2. Batasan sumber daya harus memenuhi untuk setiap unit waktu t dan jumlah keperluan sumber daya dari semua jadwal kegiatan tidak boleh melebihi dari kapasitas sumber daya.

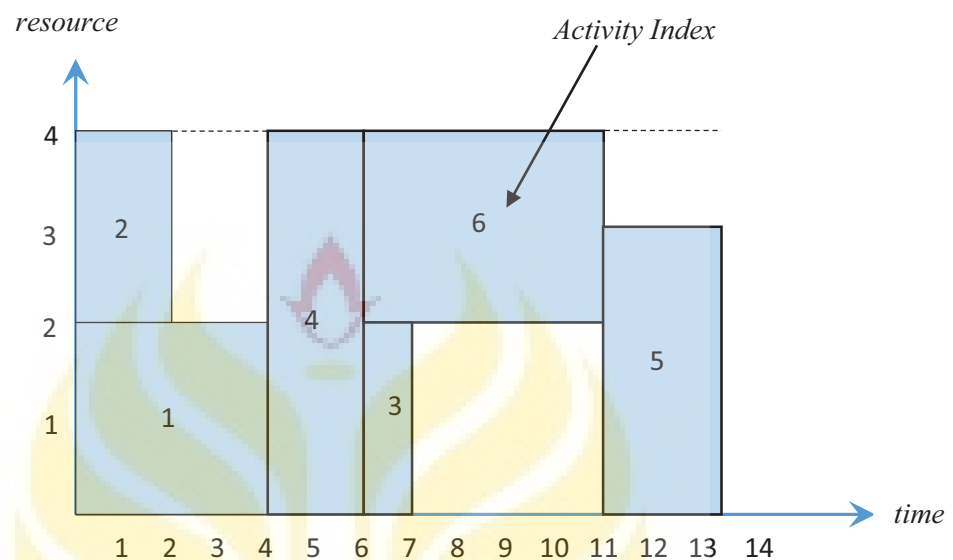
Contoh visualisasi dari RCPSP sebagai berikut (merkle, 2002):

Penggambaran jaringan proyek dari *start* sampai dengan *finish* ditunjukkan pada Gambar 2.5. Dimulai dengan kegiatan *start* dilanjutkan dengan 2 kegiatan sekaligus yaitu kegiatan 1 dan 2, kegiatan 1 membutuhkan waktu selama 4 hari dan 2 sumber daya sedangkan kegiatan 2 membutuhkan waktu selama 2 hari dan 2 sumber daya, begitu pula kegiatan lainnya. Kegiatan 1 dilanjutkan kegiatan 3 dan kegiatan 2 dilanjutkan kegiatan 4. Kegiatan 3 dapat dikerjakan apabila kegiatan 1 telah selesai dan kegiatan 4 dapat dikerjakan apabila kegiatan 2 telah selesai. Kegiatan 5 dapat dikerjakan apabila kegiatan 3 selesai dan kegiatan 6 yang dapat dikerjakan setelah kegiatan 4 selesai.



Gambar 2.5 Visualisasi RCPSP

Apabila sumber daya yang tersedia sebanyak 4 unit, maka salah satu bentuk penjadwalan yang bisa dibuat adalah sebagai berikut



Gambar 2.6 Visualisasi Bentuk Jadwal

Waktu paling akhir dimulainya kegiatan proyek LS_j dan waktu paling akhir berakhirnya kegiatan proyek LF_j dapat dihitung dengan rekursi mundur dari batas atas waktu penyelesaian proyek T . Dimulai dengan $LS_{n+1} = LF_{n+1} = T$ menggambarkan $LF_j = \min\{LS_i | i \in S_j\}$ dan $LS_j = LF_j - p_j$ untuk $j = n, \dots, 0$. Waktu paling awal dimulainya kegiatan proyek ES_j dan waktu paling awal berakhirnya kegiatan proyek EF_j dapat dihitung dengan *forward pass*. Dimulai dengan $ES_0 = EF_0 = 0$ menggambarkan $ES_j = \max\{ES_i | i \in P_j\}$ dan $EF_j = ES_j + p_j$ untuk $j = 1, \dots, n + 1$. Waktu paling awal mulai ES_{n+1} dari pekerjaan $n + 1$ adalah batas bawah minimal *maskepan* dari kemungkinan jadwal.

2.5. Algoritma Cat Swarm Optimization (CSO)

Cat Swarm Optimization (CSO) merupakan salah satu algoritma metaheuristik untuk masalah optimasi kombinatorial. Algoritma CSO pertama kali diperkenalkan oleh Shu-Chuan Chu dan Pei-Wei Tsai (Taiwan) pada tahun 2006. Ketertarikan Shu-Chuan Chu dan Pei-Wei Tsai berawal dari pengamatan terhadap perilaku sekumpulan kucing (*Cat*), namun kucing yang dimaksud bukanlah hanya seekor kucing saja. Kucing termasuk binatang dari *felid* yang memiliki sekitar 32 spesies. Sehingga Shu-Chuan Chu dan Pei-Wei Tsai menggunakan istilah “*Cat*” untuk mewakili spesies kucing dan digambarkan pada algoritma CSO.

Algoritma CSO terbagi menjadi 2 langkah dalam menyelesaikan masalah optimasi, yaitu *Seeking Mode* (SM) yang menggambarkan kucing saat istirahat, melihat sekeliling, menyusun strategi selanjutnya dan *Tracing Mode* (TM) yang menggambarkan kucing saat mengikuti mangsa buruan. Dua sub model *Seeking Mode* dan *Tracing Mode* dikombinasikan dalam satu algoritma melalui *Mixture Ratio* (MR). *Mixture Ratio* adalah suatu parameter yang digunakan untuk membagi jumlah kucing yang masuk pada setiap *mode*. Jika MR sama dengan 0, semua kucing akan dimasukkan pada *seeking mode*. Jika MR sama dengan 1, semua kucing akan dimasukkan ke dalam *tracing mode*. Dengan mengamati perilaku kucing diketahui bahwa kucing menghabiskan sebagian besar waktunya untuk beristirahat, sehingga nilai MR harus mempunyai nilai yang kecil untuk memastikan bahwa kucing

menghabiskan sebagian besar waktunya dalam posisi *seeking mode* (Chu *et al.*, 2006).

Pada penyelesaian masalah optimasi dengan algoritma CSO, kucing dan prilakunya menggambarkan set solusi. Set solusi merupakan urutan acak dari kegiatan J_0 sampai dengan J_{n+1} (Apriana, 2016). Tahapan penting dalam algoritma CSO adalah menentukan berapa banyak kucing (set solusi) yang akan digunakan untuk menyelesaikan masalah. Setiap kucing memiliki posisi yang tersusun dalam dimensi M , kecepatan untuk setiap dimensi, nilai *fitness* yang menunjukkan penyesuaian kucing pada fungsi *fitness* dan bendera untuk menentukan kucing masuk dalam SM atau TM. Solusi akhir adalah satu kucing dengan posisi terbaik. CSO akan menyimpan solusi terbaik hingga akhir iterasi (Chu *et al.*, 2006).

a. *Seeking Mode* (SM)

Seeking Mode merupakan gambaran saat kucing sedang istirahat, melihat sekeliling dan mencari posisi selanjutnya untuk bergerak. *Seeking Mode* dibagi menjadi 4 faktor yaitu *seeking memory pool* (SMP), *seeking range of the select dimention* (SRD), *counts of dimension to change* (CDC), *self-position considering* (SPC) (Chu *et al.*, 2006).

SMP digunakan untuk mendefinisikan ukuran memori pencarian setiap kucing (set solusi) yang menunjukkan titik yang dicari oleh kucing tersebut. SRD menunjukkan rentang perpindahan dari dimensi terpilih. Dalam *seeking mode* ini, jika dimensi diputuskan selisih nilai baru dan lama tidak boleh melebihi suatu rentang yang didefinisikan oleh SRD.

CDC memperlihatkan beberapa dimensi yang akan berubah. SPC merupakan variabel *boolean* (bernilai benar atau salah), yang menetapkan suatu titik yang pernah menjadi posisi kucing akan dijadikan satu kandidat posisi untuk bergerak. Bagaimanapun nilai SPC benar atau salah tidak berpengaruh (Chu *et al.*, 2006).

Langkah- langkah *seeking mode* dapat dideskripsikan dalam 5 tahap sebagai berikut (Chu *et al.*, 2006).

Langkah 1 : Bangkitkan j tiruan dari posisi kucing ke- k , dengan $j = SMP$. Jika nilai SPC benar, maka $j = (SMP - 1)$, kemudian pertahankan posisi saat ini sebagai salah satu kandidat.

Langkah 2 : Untuk setiap tiruan, disesuaikan dengan CDC, tambahkan atau kurangkan SRD persen dari nilai saat ini secara acak dan gantikan nilai yang sebelumnya.

Untuk menghitung jumlah dimensi yang akan dimodifikasi yaitu dengan menggunakan persamaan 2.1.

$$\text{jumlah modifikasi} = CDC \times d \quad (2.1)$$

Selanjutnya menghitung nilai modifikasi dari setiap dimensi yang terpilih dengan menggunakan persamaan 2.2.

$$\text{modifikasi} = x_{i,d} \pm (SRD \times R_d) \quad (2.2)$$

Langkah 3 : Hitung nilai *fitness* (FS) untuk semua titik kandidat. *Fitness* merupakan nilai *maskepan* dari setiap kucing.

Langkah 4 : Jika semua nilai *fitness* tidak benar-benar sama, hitung probabilitas terpilih masing-masing titik kandidat dengan

menggunakan persamaan (2.3) sebaliknya atur probabilitas terpilih untuk semua titik sama dengan 1.

$$P_h = \frac{|F S_h - F S_b|}{F S_{max} - F S_{min}}, \text{ dimana } o < h < j \quad (2.3)$$

Jika tujuan fungsi *fitness* adalah untuk menemukan solusi minimal, maka $F S_b = F S_{max}$, sebaliknya $F S_b = F S_{min}$.

Langkah 5 : Secara acak pilih titik untuk bergerak dari titik-titik kandidat, dan pindahkan posisi kucing ke- k .

b. *Tracing Mode* (TM)

Tracing mode adalah model yang menggambarkan keadaan ketika kucing sedang mengikuti jejak targetnya. Sekali kucing memasuki *tracing mode*, kucing tersebut akan bergerak sesuai dengan kecepatannya untuk tiap dimensi (Chu *et al.*, 2006).

Tahapan *tracing mode* dapat dijabarkan dalam 3 langkah sebagai berikut (Chu *et al.*, 2006):

Langkah 1 : Perbarui nilai kecepatan untuk setiap dimensi ($V_{k,d}$) berdasarkan persamaan (2.4)

$$V'_{k,d} = V_{k,d} + r_1 \times C_1 (X_{best,d} - X_{k,d}) \quad (2.4)$$

Langkah 2 : Periksa apakah kecepatan berada dalam rentang kecepatan maksimum. Jika kecepatan yang baru melebihi rentang, tetapkan nilai sama dengan batas.

Langkah 3 : Perbarui posisi kucing ke- k dengan persamaan (2.5)

$$X'_{k,d} = X_{k,d} + V'_{k,d} \quad (2.5)$$

$X_{best,d}$ menyatakan posisi kucing yang memiliki nilai *fitness* terbaik, $X_{k,d}$ menyatakan posisi kucing ke- k pada dimensi ke- d , c_1 adalah konstanta dan r_1 adalah nilai acak dalam rentang $[0,1]$.

Parameter *inertial weight* (w) berguna untuk mengontrol keseimbangan antara kemampuan eksplorasi global dan lokal, serta penurunan kecepatan untuk menghindari stagnasi pada optimum lokal. Jika nilai *inertia weight* terlalu besar akan mengakibatkan posisi kucing berubah terlalu jauh, sehingga gagal untuk menemukan solusi. Sebaliknya, jika nilai *inertia weight* terlalu kecil posisi kucing akan terjebak pada optimum lokal (suyanto, 2010).

CSO dimodifikasi dengan menambahkan parameter baru berupa *inertia weight* agar dapat menyelesaikan permasalahan kucing yang menjahui solusi dan terperangkap pada optimum lokal. Nilai *inertia weight* berubah secara acak dalam *tracing mode*, sehingga kecepatan pada persamaan (2.) menjadi:

$$V'_{k,d} = w \times V_{k,d} + r_1 \times C_1 (X_{best,d} - X_{k,d}) \quad (2.6)$$

Langkah –langkah CSO adalah sebagai berikut (Chu *et al.*, 2006) :

1. Tentukan jumlah N kucing dalam proses
2. Sebarkan secara acak kucing tersebut ke dalam ruang solusi M -dimensi dan tetapkan nilai secara acak yang terdapat didalam barisan kecepatan maksimum, untuk menjadi kecepatan setiap kucing. Lalu pilih sejumlah kucing secara acak dan masukan ke

dalam *tracing mode* sesuai dengan MR, dan yang lainnya masukan ke dalam *seeking mode*. Untuk menghitung jumlah *tracing* yaitu dengan menggunakan persamaan 2.7.

$$\text{jumlah tracing} = MR \times m \quad (2.7)$$

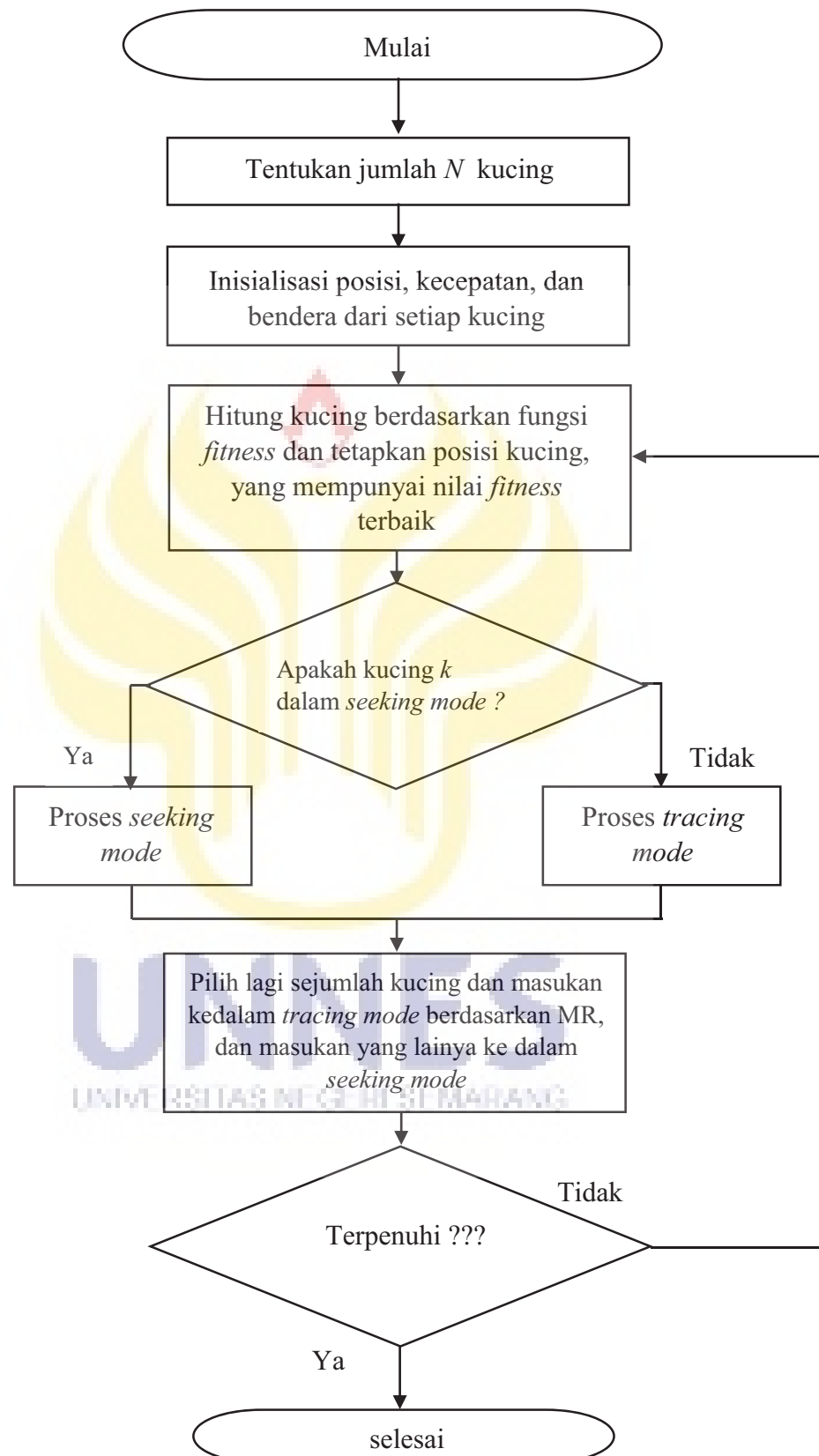
Dan untuk menghitung jumlah *seeking* menggunakan persamaan 2.8.

$$\text{jumlah seeking} = m - \text{jumlah tracing} \quad (2.8)$$

Dimana $m = \text{banyaknya kucing}$

3. Hitung nilai *fitness* dari setiap kucing dengan memasukkan posisi kucing kedalam fungsi *fitness* dan simpan kucing terbaik ke dalam memori. Sebagai catatan bahwa yang perlu disimpan adalah posisi kucing terbaik ($X_{best,d}$) karena mewakili solusi terbaik.
4. Pindahkan kucing-kucing tersebut berdasarkan benderanya, jika kucing ke- k berada dalam *seeking mode*, terapkan kucing tersebut kedalam langkah *seeking mode*, jika sebaliknya terapkan dalam langkah *tracing mode*.
5. Pilih lagi sejumlah kucing dan masukan ke dalam *tracing mode* berdasarkan MR, kemudian masukan kucing lainnya ke dalam *seeking mode*.
6. Perhatikan kriteria pemberhentian. Jika telah terpenuhi, maka hentikan program. Jika sebaliknya maka ulangi langkah (3) sampai dengan (5)

Langkah algoritma CSO secara ringkas disajikan pada Gambar 2.7.



Gambar 2.7 Flowchart (Tsai et al.,2008)

2.6. Program Simulasi

Program komputer (sering kali disebut sebagai perangkat lunak program, atau program saja) merupakan suatu aplikasi yang dibuat dengan menggunakan bahasa program tertentu dan telah ter-install di dalam komputer. Program komputer merupakan contoh perangkat lunak komputer yang menuliskan aksi komputasi yang akan dijalankan oleh komputer. Komputasi ini biasanya dilaksanakan berdasarkan suatu algoritma atau urutan perintah tertentu. Urutan perintah atau algoritma merupakan suatu perangkat yang sudah termasuk dalam program komputer tersebut. Tanpa algoritma tersebut program komputer tak dapat berjalan dengan baik.

Banyak *software* pemrograman perhitungan dan analisis yang digunakan dalam penerapan ilmu matematika terutama riset operasi. Penjadwalan proyek merupakan penerapan dari riset operasi. Software pemrograman yang dapat digunakan dalam penjadwalan proyek dengan algoritma CSO antara lain MATLAB, Visual Basic 6.0, dan C++.

MATLAB merupakan bahasa pemrograman yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti Delphi, Basic maupun C++. MATLAB merupakan bahasa pemrograman seperti komputasi matematik, analisa data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan.

MATLAB hadir dengan membawa warna yang berbeda. Hal ini karena MATLAB membawa keistimewaan dalam fungsi-fungsi matematika, fisika,

statistik, dan visualisasi. MATLAB dikembangkan oleh MathWorks, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek LINPACK dan EISPACK. Saat ini MATLAB memiliki ratusan fungsi yang dapat digunakan sebagai *problem solver* mulai dari simpel sampai masalah kompleks dari berbagai disiplin ilmu (Firmansyah, 2007:1)

Adapun beberapa bagian dari *window* yang terdapat dalam program MATLAB meliputi:

1. *Current Directory*

Bagian dari *window* ini merupakan isi dari direktori kerja saat menggunakan MATLAB.

2. *Command History*

Bagian ini berfungsi untuk menyimpan perintah-perintah apa saja yang sebelumnya dilakukan oleh pengguna terhadap MATLAB.

3. *Command Window*

Bagian ini merupakan tempat untuk menjalankan fungsi, variabel, mendeklarasikan variabel, menjalankan proses-proses serta melihat isi variabel.

4. *Workpace*

Bagian ini berfungsi untuk menampilkan seluruh variabel-variabel yang sedang aktif pada saat pemakaian MATLAB (Firmansyah, 2007).

MATLAB menyediakan fungsi *help* yang berisikan tutorial lengkap mengenai MATLAB dan segala keunggulannya. *User* dapat menjalankan

fungsi ini dengan menekan tombol apa saja pada *toolbar* atau menulis perintah “*helpwin*” pada *command window*. MATLAB juga menyediakan fungsi demos yang berisikan video tutorial MATLAB serta contoh-contoh program yang bisa dibuat dengan MATLAB

Untuk memastikan proses yang sedang berjalan pada MATLAB dapat dilakukan dengan menekan tombol Ctrl+C. Sedangkan untuk keluar dari MATLAB dapat dilakukan dengan menulis perintah *exit* atau *quit* pada *command window* atau dengan menu *exit* pada bagian menu *file* dari *menu bar*

MATLAB hanya memiliki dua jenis tipe data yaitu numerik dan string. Dalam MATLAB setiap variabel akan disimpan dalam bentuk matriks. *User* dapat langsung menulis variabel baru tanpa harus mendeklarasikanya terlebih dahulu pada *command window* (Firmansyah, 2007).

BAB V

PENUTUP

5.1. Simpulan

Berdasarkan pembahasan yang telah dibahas pada bab sebelumnya, dapat diambil simpulan sebagai berikut

1. Penerapan Algoritma Cat Swarm Optimization dalam permasalahan penjadwalan proyek adalah a) Membuat solusi RCPSP yang valid dan dihitung *maskepannya*; b) Bangkitkan sebanyak N kucing, termasuk kucing dengan solusi valid; c) Evaluasi kucing sesuai dengan posisi tiap kegiatan; d) Berdasarkan N kucing yang diperoleh hitung nilai *maskepannya*; e) Pindahkan kucing sesuai dengan MR dalam *seeking mode* dan *tracing mode*; f) Evaluasi kembali posisi setiap kucing; g) Akhiri algoritma dengan mengambil solusi yang memiliki *fitness* terkecil. Menambahkan tahap a, c, dan f pada algoritma CSO membuat solusi yang diperoleh lebih optimal dengan *maskepan* minimum.
2. Algoritma Cat Swarm Optimization dengan bantuan *software* Matlab telah diterapkan pada 2 kasus RCPSP, yaitu : a) Kasus RCPSP dengan solusi valid yang memiliki nilai *maskepan* yang masih jauh dari nilai *maskepan* terkecil yang bisa diperoleh menghasilkan solusi yang berbeda-beda setiap kali program dijalankan sehingga diperlukan

beberapa kali simulasi untuk memastikan solusi yang didapatkan benar-benar memiliki *maskepan* terkecil; b) Bila solusi valid yang dibuat memiliki *maskepan* yang telah mendekati *maskepan* terkecil yang bisa diperoleh, maka program akan menghasilkan solusi yang sama setiap program dijalankan.

5.2. Saran

Berdasarkan kesimpulan yang diperoleh, maka saran yang perlu disampaikan pada penelitian selanjutnya adalah:

1. Menerapkan Algoritma Cat Swarm Optimization pada permasalahan lain yang akan diselesaikan adalah *Multi-Mode Resource Constrained Project Scheduling Problem* (MMRCPSP) karena permasalahan MMRCPSP lebih mendekati kondisi dunia nyata dari pada *Single-Mode Resource Constrained Project Scheduling Problem* (SMRCPSP).
2. Dalam penelitian ini simulasi program dengan data masih dilakukan secara manual dimana data dimasukkan dalam *coding*. Penelitian selanjutnya perlu ada pengembangan program yang lebih praktis dalam penginputan data secara langsung melalui *Excel*, *notepad*, dll.
3. Dalam menyelesaikan kasus *Resource Constrained Project Scheduling Problem* (RCPSP) dapat menerapkan algoritma CSO yang lain, misalnya *Binary Cat Swarm Optimization*, *Average-Inertia Weighted Cat Swarm Optimization*, dan lainnya.

DAFTAR PUSTAKA

- Aminuddin. 2005. *Prinsip-Prinsip Riset Operasi*. Jakarta: Erlangga.
- Arifudin, R. 2011. Optimasi Penjadwalan Proyek dengan Penyeimbang Biaya Menggunakan Kombinasi CPM dan Algoritma Genetika. *Jurnal Masyarakat Informatika*, 2(4):2086-4930.
- Apriana, I.W.R.. 2016. Implementasi Algoritma Cat Swarm Optimization dalam Menyelesaikan Job Shop Scheduling Problem (JJSP). *E-jurnal Matematika*. 5(3):90-97.
- Bouzidi, A. & M.E. Riffi. 2014. Cat Swarm Optimization to Solve Job Shop Scheduling Problem. *IEEE*, 72(2):202-205.
- Chan, W.T., D.K.H. Chua. & G. Kannan. 1996. Construction Resource Scheduling with Genetic Algorithms. *Journal Construction Engineering and Management ASCE*, 122(2):125-132.
- Chu, S.C. & P.W. Tsai. 2006. Cat Swarm Optimization. *Proceedings of the 9th Pacific Rim International Conference on Artificial Intelligence LNAI 4099*, pp. 854-858.
- Dhanasaputra, N. & B. Santosa. 2010. Pengembangan Algoritma Cat Swarm Optimization (CSO) untuk Klasifikasi. *Jurnal Institut Teknologi Sepuluh Noverber*. Diakses tanggal 16 Mei 2017.
- Davis, E.W. 1973. Project Scheduling Under resource Constraints Historical Review and Catagorization of Procedures. *AIIE Transactions*, 5(4): 297-313.
- Davis, E.W., & J.H. Patterson. 1975. A Comparison of Heuristic and Optimum Solutions in Resource-Constrained Project Scheduling. *Manage Science*, 21(8):944-55.
- Dimiyati, T & A. Dimiyati. 2004. *Operation Research (Model-Model Pengambilan Keputusan)*. Bandung : Sinar Baru Algensindo.
- Firmansyah, A. 2007. Dasar-Dasar Pemograman Matlab. IlmuKomputer.com.
- Gavish, B. dan H. Pirkul. 1991. Algorithms for Multi-Resource Generalized Assignment Problem. *Manage Sciencei*, 37(6):695-713.
- Krisnawati, M. 2014. Penyelesaian Permasalahan Penjadwalan Aktivitas Proyek dengan Batasan Sumber Daya Menggunakan Metode Cross Entropy. *Dinamika Rekayasa*, 10(1):1-5.

- Lee, J.K., & Y.D. Kim. 1996. Search heuristics for resource constrained project scheduling. *Jurnal Operation Research Society*, 47(5): 678–689.
- Merkel, D., M. Minddendorf, & H. Schmeck. 2002. Ant Colony Optimization for Resource-Constrained Project Scheduling. *IEEE Transactions on Evolutionary Computation*, 6(4):333-346.
- Molyono, S. 2002. *Riset Operasi*. Jakarta : Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia.
- Putra, R.I. 2013. Penerapan Algoritma Harmony Search pada Resource-Constrained Project Scheduling Problem (RCPSP). *Jurnal Online Universitas Negeri Malang*. Diakses tanggal 20 Mei 2017.
- Soeharto, I. 1999. *Manajemen Proyek (dari Konseptual sampai Operasional), Jilid I, Edisi 2*. Jakarta: Erlangga.
- Subagyo, P., M. Asri, & T.H. Handoko. 1999. *Dasar-dasar Operation Research*. Yogyakarta : Edisi kedua BPFE.
- Suyanto. 2010. *Algoritma Optimasi Deterministik atau Probabilitik*. Yogyakarta: Graha Ilmu.
- Talbot, F.B. 1982. Resource-Constrained Project Scheduling with Time Resource Tradeoffs: The Nonpreemptive Case. *Manage Science*, 28(10):1197–210.
- Tsai, P.W., J.S. Pan, dkk. 2016. Utilizing Cat Swarm Optimization in Allocating the Sink Node in the Wireless Sensor Network Environment. *International Conference on Computing Meansurement Control*, pp. 166-169.
- Tsai, P.W., J.S. Pan, S.M. Chen, B.Y. Liao, & S.P. Hao. 2008. Parallel Cat Swarm Optimization. *International Conference on Machine Learning and Cybernetics*, pp. 3328-3333.
- Widyawati, K., Mashuri, & R. Arifudin. 2014. Analisis Algoritma Branchand Bound Untuk Menyelesaikan Masalah Penjadwalan Proyek Pembangunan Mega Tower. *UNNES Journal Of Mathematics*, 3(1):2460-5859.
- Xu, L. & W. Hu. 2012. Cat swarm Optimization-based schemes for resource-constrained project scheduling. *Applied Mechanics and Materials*. Vols. 220-223. pp. 251-258.
- Zhang, H., H. Li, dan C.M. Tam. 2006. Particle Swarm Optimization for Resource-Constrained Project Scheduling. *International Journal of Project Management*, 24(1): 83– 92.