

OPTIMASI PENJADWALAN PROYEK KONSTRUKSI MENGGUNAKAN KOMBINASI CRITICAL PATH METHOD DAN ALGORITMA PARTICLE SWARM OPTIMIZATION

Ruspandi Reynaldi Benedictus, Marthin Dody Sumayouw, Arthur Harris Thambas, Grace Yoyce Malingkas

*Program Studi Magister Teknik Sipil Universitas Sam Ratulangi, Manado, Indonesia
Email: ruspandireynaldi@gmail.com*

ABSTRAK

Sebuah proyek merupakan aktivitas yang terjadi dalam durasi waktu tertentu dengan keterbatasan sumber daya, dan bertujuan untuk menghasilkan produk sesuai standar kualitas. Oleh karena itu, penjadwalan proyek yang tepat sangat diperlukan. Salah satu metode untuk mengatasi permasalahan penjadwalan proyek adalah Critical Path Method (CPM), namun CPM hanya mempertimbangkan batasan waktu. Algoritma Particle Swarm Optimization (PSO) yang merupakan salah satu contoh metode metaheuristik, pada penelitian ini diimplementasikan untuk mengoptimalkan penyeimbangan biaya mingguan proyek. Berdasarkan data RAB beserta durasi dan kegiatan pendahulu, dihitung waktu ES, EF, LS, LF dan *float* setiap kegiatan dengan CPM dibantu aplikasi Microsoft Project dan POM-QM for Windows 5. Hasil tersebut kemudian diproses lebih lanjut menggunakan PSO yang ditulis dalam bahasa pemrograman Python untuk mencari kombinasi waktu dimulainya kegiatan pada jalur non-kritis untuk mendapatkan nilai rata-rata penyimpangan biaya mingguan terkecil. Berdasarkan hasil penelitian, kombinasi waktu dimulainya kegiatan menggunakan PSO dapat memberikan nilai yang lebih optimum dari pada hanya menggunakan CPM saja, dimana PSO memberikan nilai rata-rata penyimpangan biaya mingguan Rp125.283.677, sedangkan CPM *Earliest* Rp 146.197.206 dan CPM *Latest* Rp144.526.841.

Kata kunci: *Particle swarm optimization, optimalisasi penjadwalan, proyek konstruksi, Critical Path Method*

OPTIMIZATION OF CONSTRUCTION PROJECT SCHEDULING USING A COMBINATION OF CRITICAL PATH METHOD AND PARTICLE SWARM OPTMIZATION ALGORITHM

ABSTRACT

A project is an activity carried out within a specific period, with limited resources, aiming to produce a product that meets quality standards. Therefore, accurate project scheduling is crucial. One method for solving project scheduling issues is the Critical Path Method (CPM); however, CPM only considers time constraints. In this study, the Particle Swarm Optimization (PSO) algorithm, a metaheuristic method, is implemented to optimize the balancing of weekly project costs. Based on Bill of Quantities (BOQ) data, along with the duration and predecessor activities, the ES, EF, LS, LF and float times for each activity are calculated using CPM, aided by Microsoft Project and POM-QM for Windows 5. These results are then further processed using PSO, written in Python, to find the optimal combination of start times for activities on non-critical paths to achieve the smallest average weekly cost deviation. According to the results, the combination of start times using PSO provides more optimal values compared to using CPM alone, with PSO yielding an average weekly cost deviation of IDR125.283.677, while CPM *Earliest* and CPM *Latest* yield Rp 146.197.206 and IDR144.526.841, respectively.

Keywords: *Particle swarm optimization, scheduling optimization, construction project, Critical Path Method*

1. PENDAHULUAN

Sebuah proyek merupakan aktivitas yang terjadi dalam durasi waktu tertentu dengan keterbatasan sumber daya, dan bertujuan untuk menghasilkan produk atau hasil kerja yang memenuhi standar kualitas yang jelas. Oleh karena itu, perencanaan proyek yang tepat sangat diperlukan agar proyek dapat diselesaikan tanpa melanggar batasan-batasan tersebut (Soeharto, 1999). CPM atau *Critical Path Method* merupakan salah satu contoh metode yang biasa digunakan untuk menyelesaikan permasalahan penjadwalan proyek. Namun, metode ini hanya mempertimbangkan batasan waktu dan tidak mengakomodasi batasan lainnya seperti biaya. (Zhou et al., 2013). Penyusunan jadwal proyek dengan memperhatikan batasan waktu dan biaya apabila disusun menggunakan metode matematis akan menjadi hal yang kompleks (Cholissodin, 2016).

Ada beberapa teknik yang dikembangkan untuk penyelesaian masalah *Resource-Constrained Project Scheduling Problem* (RCPSP) ini, diantaranya adalah metode metaheuristik, dimana metode metaheuristik merupakan teknik yang dipakai untuk mencari solusi terbaik dari masalah optimasi kombinatorial dan biasanya diadaptasi dari perilaku alam (Zhou et al., 2013). Contoh dari metode metaheuristik ini adalah Algoritma Genetika, *Ant Colony Optimization*, *Particle Swarm Optimization* (PSO) dan *Cuckoo Search*. Metode metaheuristik ini bisa digabungkan dengan metode lain, seperti contohnya penggabungan Algoritma Genetika dengan CPM (Arifudin, 2011).

PSO sendiri merupakan metode optimasi yang berdasarkan kawanan (*swarm*) yang menirukan perilaku sosial hewan seperti burung yang saling bekerja sama dan belajar dari pengalaman anggota lain untuk mencari jalan menuju sumber makanan (Wang et al., 2018). Pada penelitian ini, penulis menggunakan kombinasi metode CPM dan PSO untuk mengoptimalkan penjadwalan proyek konstruksi sehingga bisa menyeimbangkan biaya mingguan proyek.

2. PENJADWALAN PROYEK

Proyek merupakan rangkaian aktivitas yang terjadi dalam durasi waktu tertentu dan dalam keterbatasan sumber daya guna menghasilkan produk akhir yang telah ditentukan (Hafnidar, 2016). Sebuah proyek dapat dianggap sebagai sebuah rangkaian aktifitas dan pekerjaan yang (Kerzner and Saladis, 2009):

1. Mempunyai tujuan yang spesifik untuk diselesaikan menurut spesifikasi tertentu
2. Mempunyai tanggal mulai dan selesai yang ditentukan
3. Mempunyai batasan pendanaan
4. Mengonsumsi sumber daya manusia dan non-manusia
5. Bersifat multifungsional

Manajemen proyek adalah proses mengorganisasikan, merencanakan, memimpin, dan mengendalikan sumber daya perusahaan untuk tercapainya tujuan jangka pendek yang telah ditetapkan. Sementara itu, perencanaan proyek didefinisikan sebagai proses pemilihan metode dan urutan pekerjaan yang akan diterapkan pada sebuah proyek dari berbagai cara dan urutan yang mungkin untuk menyelesaikan pekerjaan tersebut (Mubarak, 2020). Perencanaan proyek melibatkan serangkaian langkah yang bertujuan untuk memastikan proyek selesai sesuai dengan jadwal yang telah ditetapkan. Langkah-langkah ini mencakup penentuan komponen kegiatan, pengaturan urutan pelaksanaan setiap komponen, serta estimasi waktu penyelesaian masing-masing komponen pekerjaan tersebut (Soeharto, 1999).

Salah satu optimasi dalam penjadwalan proyek adalah penyeimbangan biaya. Biaya mingguan dianggap optimal jika rata-rata deviasi biaya mingguan terhadap nilai rata-rata biaya mingguan adalah minimal. Berikut ini adalah fungsi objektif untuk memperoleh biaya mingguan minimal:

$$\text{Min } B = \frac{1}{T} \sum_{i=1}^T |b_i - \bar{b}| \quad (1)$$

dimana: \square = penyimpangan biaya harian rata-rata;

T = durasi total pengerjaan proyek;

b_i = biaya proyek mingguan (yang terjadi pada minggu ke- i); dan

\bar{b} = rata-rata biaya proyek mingguan.

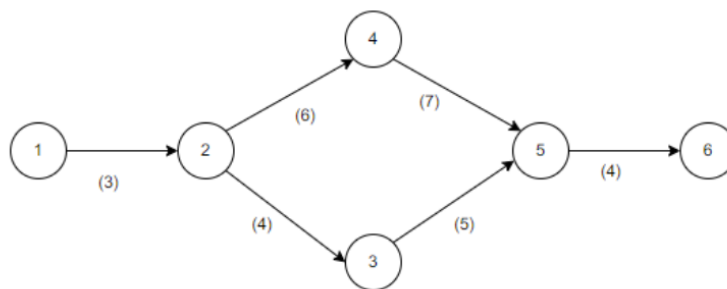
Biaya mingguan suatu proyek dapat dikalkulasi dengan menjumlahkan biaya mingguan untuk setiap aktivitas yang dilakukan pada minggu yang sama. Biaya rata-rata mingguan suatu proyek dapat dihitung dengan membagi total biaya proyek dengan total durasi proyek.

$$b_i = \sum_i \frac{b_{ki}}{T} \quad (2)$$

dimana: b_i = biaya proyek mingguan (pada minggu ke-i);
 b_{ki} = biaya aktivitas kegiatan yang terjadi pada minggu ke-i; dan
 T = durasi total pengerjaan proyek.

a. *Critical Path Method (CPM)*

CPM merupakan salah satu versi analisis jaringan kerja yang pertama kali digunakan oleh tim dari perusahaan Du-Pont untuk menekan biaya proyek pembuatan pabrik kimia hingga sebesar 10% dari biaya rencana semula (Hafnidar, 2016). Perusahaan tersebut mengembangkan sistem pengendalian manajemen untuk mengatur berbagi kegiatan yang saling terkait dalam masalah desain, konstruksi dan *maintenance*. Biaya yang diperlukan dapat diminimalkan dengan cara mengoptimasi durasi terselesaikannya suatu kegiatan. Metode CPM menggunakan grafik sebagai alat representasi, menggunakan diagram panah dan lingkaran serta prinsip ketergantungan logis dasar untuk mengatur urutan kegiatan.



Gambar 1. Contoh diagram jaringan proyek

Terminologi umum dalam CPM adalah sebagai berikut (Soeharto, 1999):

1. Waktu mulai paling awal (ES) suatu kegiatan dihitung berdasarkan waktu paling awal dari kegiatan pendahulunya.
2. Waktu selesai paling awal suatu kegiatan (EF), jika hanya terdapat satu aktivitas pendahulu, EF aktivitas tersebut sama dengan ES aktivitas penerus.
3. Waktu paling akhir suatu kegiatan dapat dimulai tanpa menunda waktu penyelesaian proyek (LS).
4. Waktu paling akhir suatu kegiatan harus diselesaikan tanpa menunda waktu penyelesaian proyek secara keseluruhan (LF).

Elemen kunci dari CPM adalah jalur kritis. Ini mewakili kumpulan aktivitas dengan total waktu terlama dan menentukan waktu terpendek untuk menyelesaikan proyek. Keterlambatan pada jalur kritis dapat mengakibatkan tertundanya keseluruhan proyek.

CPM melakukan beberapa langkah perhitungan: perhitungan maju, perhitungan mundur, dan perhitungan waktu *float* kegiatan. Tujuan perhitungan maju adalah untuk menentukan waktu paling awal suatu kegiatan dapat dimulai dan diakhiri. Sedangkan, perhitungan mundur adalah perhitungan yang digunakan untuk menentukan waktu atau tanggal paling lambat suatu kegiatan dapat dimulai dan diakhiri tanpa mempengaruhi waktu penyelesaian proyek secara keseluruhan.

Sifat umum jalur kritis, yaitu:

1. Untuk kegiatan pertama: $ES = LS = 0$ atau $E(1) = L(1) = 0$.
2. Untuk kegiatan terakhir atau terminal: $LF = EF$.
3. *Float total*: $TF = 0$.

Float time adalah jumlah waktu maksimum suatu kegiatan dapat ditunda tanpa mempengaruhi jadwal pelaksanaan proyek secara keseluruhan. *Float total* berlaku untuk semua aktivitas dalam jalur tertentu. Oleh karena itu, jika suatu kegiatan menggunakan waktu total *float*, penggunaannya juga akan mengurangi sisa *float total* yang tersedia untuk aktivitas lain dalam jalur yang sama.

b. *Particle Swarm Optimization (PSO)*

Teknik metaheuristik adalah teknik yang digunakan untuk penyelesaian permasalahan optimasi kombinatorial, yang diadaptasi dari perilaku alam. Melalui pengamatan aktivitas kelompok makhluk hidup yang memiliki ciri khas unik, dapat dilihat bahwa perilaku ini mampu membentuk kecerdasan alami. Beberapa elemen

yang dapat diadaptasi ke dalam sistem cerdas berbasis sintesis dari alam adalah sebagai berikut (Imam Cholissodin, 2016):

1. Kecerdasan alami (sosial dan personal)
2. Berkoloni (berpasang-pasangan)
3. Pencarian makanan
4. Regenerasi

Swarm intelligence atau kecerdasan kelompok adalah cabang ilmu dalam sistem cerdas yang menghubungkan sistem alami dan buatan. Konsepnya adalah sekelompok individu yang berkoordinasi dengan menggunakan kontrol yang terdesentralisasi (kecerdasan sosial) dan terorganisasi sendiri (kecerdasan pribadi). Metode metaheuristik yang populer seperti algoritma genetika, optimasi koloni semut, dan *partikel swarm optimization* digunakan untuk meniru proses alam guna mencapai solusi optimal.

PSO (*Particle Swarm Optimization*) adalah salah satu metode metaheuristik yang umum digunakan dalam penyelesaian masalah penjadwalan proyek. Algoritma ini pertama kali dikembangkan oleh Ebenhart dan Kennedy pada tahun 1995. PSO diadopsi dari perilaku suatu kawanan, burung misalnya, dimana terdapat seekor burung yang merupakan individu dan dalam algoritma PSO disebut sebagai partikel, yang selain dipengaruhi kecerdasannya sendiri, juga dipengaruhi perilaku kelompoknya. Sekawanan burung yang terbang diatas suatu area perlu menemukan tempat untuk mendarat, dimana ini merupakan masalah yang kompleks, karena bergantung pada beberapa faktor, seperti memaksimalkan ketersediaan makanan dan meminimalkan risiko kehadiran predator. Setiap anggota burung mencari tempat optimal untuk mendarat dengan menyeimbangkan pengalaman individu dan pengetahuan kelompok.

Setiap anggota burung direpresentasikan sebagai sebuah partikel, dan setiap partikel memiliki dua atribut: posisi dan kecepatan. Setiap partikel bergerak dalam ruang tertentu dan mengingat posisi optimal yang telah dicapai sejauh ini dalam kaitannya dengan sumber makanan atau nilai fungsi tujuan. Partikel-partikel tersebut lantas berbagi informasinya atau posisi optimalnya dengan partikel lain dalam kelompok, dan mereka menyesuaikan posisi serta kecepatannya berdasarkan informasi yang diterimanya tentang posisi optimalnya (Santosa, n.d.).

PSO dimulai dengan sebuah populasi berukuran P partikel acak. Pada setiap iterasi, nilai terbaik yang dicapai suatu partikel (Pbest) dan nilai terbaik yang dicapai semua partikel atau kelompok itu (Gbest) ditentukan melalui evaluasi nilai *fitness*. Sebuah partikel bergerak ke posisi yang baru dan mendapatkan nilai solusi baru yang dituntun oleh kecepatan (Lo et al., 2008). Mempertimbangkan kawanan dengan P partikel, mempunyai vektor posisi dan kecepatan dengan jumlah anggota sebanyak jumlah kegiatan. Vektor-vektor ini diperbarui menurut persamaan berikut.

$$V_{ij}^{t+1} = \omega V_{ij}^t + c_1 r_1^t (pbest_{ij} - X_{ij}^t) + c_2 r_2^t (gbest_j - X_{ij}^t) \quad (3)$$

dimana: $i = 1, 2, \dots, P$ jumlah partikel;
 $j = 1, 2, \dots, n$ jumlah anggota dalam suatu partikel;
 t = iterasi sekarang;
 V = kecepatan partikel;
 ω = bobot inerti;
 c_1 = konstanta kognisi individu;
 c_2 = konstanta pembelajaran sosial; dan
 r_1 dan r_2 = parameter acak rentang 0-1 untuk menghindari konvergensi prematur; dan
 X = posisi partikel

$$X_{ij}^{t+1} = X_{ij}^t + V_{ij}^{t+1} \quad (4)$$

dimana: $i = 1, 2, \dots, P$ jumlah partikel;
 $j = 1, 2, \dots, n$ jumlah anggota dalam suatu partikel;
 t = iterasi sekarang;
 V = kecepatan partikel; dan
 X = posisi partikel.

Berikut adalah langkah-langkah dalam algoritma PSO untuk menemukan nilai minimal dan semua posisi vektor diuji berdasarkan fungsi fitness $f(x)$ (Seixas Gomes De Almeida and Leite, n.d.):

1. Inisialisasi
 - a. Untuk setiap partikel i dalam sebuah kawanan berukuran P:
 - i. Inisialisasi X_i secara acak

- ii. Inisialisasi V_i secara acak
- iii. Evaluasi nilai fitness $f(X_i)$
- iv. Inisialisasi $Pbest_j$ dengan sebuah kopi dari X_i
- b. Inisialisasi $Gbest$ dengan sebuah kopi dari X_i dengan nilai fitness terbaik
- 2. Ulangi hingga kriteria pemberhentian terpenuhi:
 - a. Untuk setiap partikel i:
 - i. Perbarui X_i^t dan V_i^t berdasarkan persamaan (3) dan (4)
 - ii. Evaluasi nilai fitness $f(X_i^t)$
 - iii. $Pbest_j$ adalah X_i^t jika $f(Pbest_j) < f(X_i^t)$
 - iv. $Gbest$ adalah X_i^t jika $f(Gbest) < f(X_i^t)$

3. METODE

a. Pengambilan data

Data yang dipergunakan dalam penelitian ini adalah data RAB proyek pembangunan Gedung Pendidikan Fakultas MIPA kampus Universitas Sam Ratulangi, Kota Manado. Pada tabel RAB (Tabel 1) juga dilengkapi data kegiatan pendahulu dan durasi pengerjaan setiap kegiatan.

Tabel 1. Data penelitian.

Kode	Deskripsi	Durasi (Minggu)	Kegiatan Pendahulu	Biaya (Rp)
A	Pekerjaan persiapan	3	-	152.371.776,26
B	Pekerjaan tanah, pasir dan pondasi	4	A	1.068.226.884,47
C	Struktur Lantai 1	3	B	894.755.369,05
...
R	Pembersihan Akhir	1	P,Q,N,O,H	15.000.000

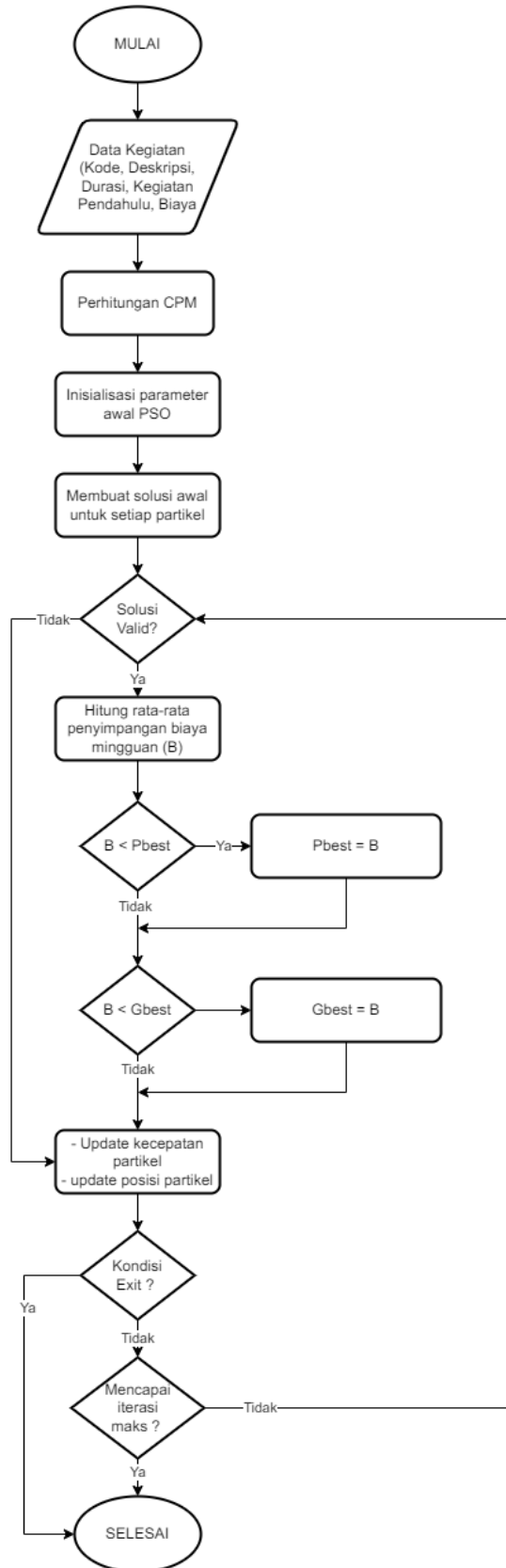
b. Critical Path Method (CPM)

Menggunakan CPM, akan dihitung waktu ES, LS, EF dan LF serta waktu *float* masing-masing kegiatan. Proses ini menggunakan *software* Microsoft Project dan POM-QM for Windows 5. Pada Gambar 2 terlihat bahwa kegiatan kode H, J, N, O adalah kegiatan yang bisa dioptimalkan lebih lanjut menggunakan PSO, karena kegiatan-kegiatan tersebut berada dalam jalur non-kritis (nilai *float* > 0) sehingga waktu mulainya kegiatan-kegiatan ini masih bisa diubah.

Activity	Activity time	Early Start	Early Finish	Late Start	Late Finish	Slack
Project	34					
A	3	0	3	0	3	0
B	4	3	7	3	7	0
C	3	7	10	7	10	0
D	5	10	15	10	15	0
E	4	15	19	15	19	0
F	5	19	24	19	24	0
G	3	24	27	24	27	0
H	1	27	28	32	33	5
I	3	27	30	27	30	0
J	3	24	27	28	31	4
K	1	30	31	30	31	0
L	1	30	31	30	31	0
M	1	31	32	31	32	0
N	1	30	31	32	33	2
O	1	30	31	32	33	2
P	1	32	33	32	33	0
Q	1	32	33	32	33	0
R	1	33	34	33	34	0

Gambar 2. Hasil akhir perhitungan ES,EF,LS,LF dan *float* (slack)

c. Particle Swarm Optimization (PSO)



Gambar 3. Diagram alir algoritma PSO

Setelah mendapatkan data perhitungan dari CPM, selanjutnya dibuat program komputer menggunakan bahasa pemrograman Python untuk menjalankan algoritma PSO berdasarkan diagram alir pada Gambar 3. Di tahap awal PSO, akan dibangkitkan sejumlah partikel sebanyak 50. Artinya ada sebanyak 50 kemungkinan solusi dari penjadwalan proyek konstruksi ini. Pada Gambar 4 dapat dilihat representasi solusi yang dibuat dalam matriks partikel. Setiap partikel berisi posisi atau minggu dimulainya kegiatan. Pada iterasi pertama, solusi dibangkitkan secara acak yang masih dalam kurun waktu *float* setiap kegiatan.

```

['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 31, 'I': 27, 'J': 27, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 31, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 28, 'I': 27, 'J': 24, 'K': 30, 'L': 30, 'M': 31, 'N': 30, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 31, 'I': 27, 'J': 28, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 32, 'I': 27, 'J': 26, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 31, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 30, 'I': 27, 'J': 26, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 32, 'I': 27, 'J': 24, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 28, 'I': 27, 'J': 27, 'K': 30, 'L': 30, 'M': 31, 'N': 30, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 29, 'I': 27, 'J': 24, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 27, 'I': 27, 'J': 28, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 27, 'I': 27, 'J': 27, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 29, 'I': 27, 'J': 24, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 31, 'I': 27, 'J': 26, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 31, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 32, 'I': 27, 'J': 26, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 30, 'I': 27, 'J': 26, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 32, 'I': 27, 'J': 27, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 28, 'I': 27, 'J': 24, 'K': 30, 'L': 30, 'M': 31, 'N': 31, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 29, 'I': 27, 'J': 28, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 31, 'I': 27, 'J': 28, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 30, 'I': 27, 'J': 25, 'K': 30, 'L': 30, 'M': 31, 'N': 30, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 30, 'I': 27, 'J': 28, 'K': 30, 'L': 30, 'M': 31, 'N': 32, 'O': 30, 'P': 32, 'Q': 32, 'R': 33]
['A': 0, 'B': 3, 'C': 7, 'D': 10, 'E': 15, 'F': 19, 'G': 24, 'H': 30, 'I': 27, 'J': 27, 'K': 30, 'L': 30, 'M': 31, 'N': 30, 'O': 32, 'P': 32, 'Q': 32, 'R': 33]
    
```

Gambar 4. Representasi partikel memuat solusi jadwal kegiatan

Ditahap selanjutnya, akan dihitung biaya mingguan untuk setiap minggu yang ada (total 34 minggu) menggunakan Persamaan 2. Setelah itu, dihitung penyimpangan biaya mingguan dan rata-ratanya dengan Persamaan 1. Hasil perhitungan bisa dilihat pada Gambar 5.

```

[0, 3, 7, 10, 15, 19, 24, 32, 27, 28, 30, 30, 31, 32, 32, 32, 32, 33]
Rata - Rata Biaya Mingguan : Rp 371,663,193
Minggu ke - 1 : Biaya : Rp 50,790,592; Penyimpangan Biaya Mingguan : Rp 320,872,601
Minggu ke - 2 : Biaya : Rp 50,790,592; Penyimpangan Biaya Mingguan : Rp 320,872,601
Minggu ke - 3 : Biaya : Rp 50,790,592; Penyimpangan Biaya Mingguan : Rp 320,872,601
Minggu ke - 4 : Biaya : Rp 267,056,721; Penyimpangan Biaya Mingguan : Rp 104,606,472
Minggu ke - 5 : Biaya : Rp 267,056,721; Penyimpangan Biaya Mingguan : Rp 104,606,472
Minggu ke - 6 : Biaya : Rp 267,056,721; Penyimpangan Biaya Mingguan : Rp 104,606,472
Minggu ke - 7 : Biaya : Rp 267,056,721; Penyimpangan Biaya Mingguan : Rp 104,606,472
Minggu ke - 8 : Biaya : Rp 298,251,790; Penyimpangan Biaya Mingguan : Rp 73,411,403
Minggu ke - 9 : Biaya : Rp 298,251,790; Penyimpangan Biaya Mingguan : Rp 73,411,403
Minggu ke - 10 : Biaya : Rp 298,251,790; Penyimpangan Biaya Mingguan : Rp 73,411,403
Minggu ke - 11 : Biaya : Rp 421,070,213; Penyimpangan Biaya Mingguan : Rp 49,407,020
Minggu ke - 12 : Biaya : Rp 421,070,213; Penyimpangan Biaya Mingguan : Rp 49,407,020
Minggu ke - 13 : Biaya : Rp 421,070,213; Penyimpangan Biaya Mingguan : Rp 49,407,020
Minggu ke - 14 : Biaya : Rp 421,070,213; Penyimpangan Biaya Mingguan : Rp 49,407,020
Minggu ke - 15 : Biaya : Rp 421,070,213; Penyimpangan Biaya Mingguan : Rp 49,407,020
Minggu ke - 16 : Biaya : Rp 495,433,674; Penyimpangan Biaya Mingguan : Rp 123,770,481
Minggu ke - 17 : Biaya : Rp 495,433,674; Penyimpangan Biaya Mingguan : Rp 123,770,481
Minggu ke - 18 : Biaya : Rp 495,433,674; Penyimpangan Biaya Mingguan : Rp 123,770,481
Minggu ke - 19 : Biaya : Rp 495,433,674; Penyimpangan Biaya Mingguan : Rp 123,770,481
Minggu ke - 20 : Biaya : Rp 337,989,422; Penyimpangan Biaya Mingguan : Rp 33,673,771
Minggu ke - 21 : Biaya : Rp 337,989,422; Penyimpangan Biaya Mingguan : Rp 33,673,771
Minggu ke - 22 : Biaya : Rp 337,989,422; Penyimpangan Biaya Mingguan : Rp 33,673,771
Minggu ke - 23 : Biaya : Rp 337,989,422; Penyimpangan Biaya Mingguan : Rp 33,673,771
Minggu ke - 24 : Biaya : Rp 337,989,422; Penyimpangan Biaya Mingguan : Rp 33,673,771
Minggu ke - 25 : Biaya : Rp 271,534,201; Penyimpangan Biaya Mingguan : Rp 100,128,992
Minggu ke - 26 : Biaya : Rp 271,534,201; Penyimpangan Biaya Mingguan : Rp 100,128,992
Minggu ke - 27 : Biaya : Rp 271,534,201; Penyimpangan Biaya Mingguan : Rp 100,128,992
Minggu ke - 28 : Biaya : Rp 384,615,409; Penyimpangan Biaya Mingguan : Rp 12,952,216
Minggu ke - 29 : Biaya : Rp 803,616,103; Penyimpangan Biaya Mingguan : Rp 431,952,910
Minggu ke - 30 : Biaya : Rp 803,616,103; Penyimpangan Biaya Mingguan : Rp 431,952,910
Minggu ke - 31 : Biaya : Rp 1,091,006,441; Penyimpangan Biaya Mingguan : Rp 719,343,248
Minggu ke - 32 : Biaya : Rp 341,403,830; Penyimpangan Biaya Mingguan : Rp 30,259,363
Minggu ke - 33 : Biaya : Rp 490,301,174; Penyimpangan Biaya Mingguan : Rp 118,637,981
Minggu ke - 34 : Biaya : Rp 15,000,000; Penyimpangan Biaya Mingguan : Rp 356,663,193
Rata - Rata Penyimpangan Biaya Mingguan : Rp 144,526,841
    
```

Gambar 5. Hasil perhitungan penyimpangan biaya mingguan

Setelah itu, akan dilakukan perhitungan $Pbest$ dan $Gbest$, dimana $Pbest_j$ adalah X_i^t jika $f(Pbest_j) < f(X_i^t)$ dan $Gbest$ adalah X_i^t jika $f(Gbest) < f(X_i^t)$. Nilai $Pbest$ dan $Gbest$ ini akan disimpan untuk perbandingan di perulangan selanjutnya.

Ditahap berikut, menggunakan Persamaan 3 akan dicari nilai kecepatan untuk masing-masing anggota dalam suatu partikel. Representasi dari vektor kecepatan partikel dapat dilihat pada Gambar 6. Terlihat bahwa

hanya kegiatan pada jalur non-kritis yang mempunyai kecepatan, sedangkan kegiatan pada jalur kritis memiliki kecepatan 0. Untuk nilai parameter, pada penelitian ini digunakan nilai $\omega = 0.4$, $c_1 = 1.5$ dan $c_2 = 1.5$.

```
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 2.5324556735355407e-176, 0.0, 0.0, 0.0, 0.0, 0.0, 6.58656275267525e-176, 3.897944280365207e-176, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.0101144502552704e-174, 0.0, 4.6372749717131144e-176, 0.0, 0.0, 0.0, 8.371190160942415e-176, 2.3141224563607597e-176, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.3742573998128029e-176, 0.0, -8.478512264336569e-174, 0.0, 0.0, 0.0, 3.9018388654619516e-176, 1.721282514670394e-176, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.4415557661395945e-175, 0.0, 5.336689316187875e-176, 0.0, 0.0, 0.0, -5.17323351817986e-175, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -2.0281100881005743e-175, 0.0, 0.0, 0.0, 0.0, 2.437079082804757e-176, 0.0, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.886423549694277e-175, 0.0, 2.693064329021668e-175, 0.0, 0.0, 0.0, 3.3011496080726165e-176, -1.107709635461603e-175, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 6.84398901049902e-176, 0.0, 7.652176138596734e-176, 0.0, 0.0, 0.0, 9.861608528927947e-176, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -3.769467189943586e-176, 0.0, 2.4267495656748915e-175, 0.0, 0.0, 0.0, 2.684216557649104e-176, -2.06294925920282e-175, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -8.025787958704063e-175, 0.0, 4.461758202926945e-176, 0.0, 0.0, 0.0, 1.6448950666252116e-176, 8.331521256822879e-177, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -1.0705358939739114e-174, 0.0, -5.6364184685747025e-176, 0.0, 0.0, 0.0, -1.4291512894639232e-174, 8.208601430423557e-177, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.7737474112765284e-175, 0.0, 0.0, 0.0, 0.0, 2.5733452423719856e-174, 1.900829613953381e-176, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.1557547667572962e-175, 0.0, 6.345689989600546e-176, 0.0, 0.0, 0.0, 2.2592916753126655e-176, 8.735801758807737e-177, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -4.6119594127755925e-175, 0.0, 5.947682180195626e-175, 0.0, 0.0, 0.0, 1.797372601249361e-176, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -8.09594525935802e-176, 0.0, 1.26937735425195084e-175, 0.0, 0.0, 0.0, 3.1564630488462927e-176, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 6.541761856305191e-175, 0.0, 7.004312340099037e-174, 0.0, 0.0, 0.0, -4.35183929512794e-176, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.366399981025934e-175, 0.0, 0.0, 0.0, 0.0, 5.26356558069089e-174, 0.0, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -2.592551141318347e-174, 0.0, -3.023262276164226e-176, 0.0, 0.0, 0.0, 1.692120762021559e-176, -1.498441672702017e-175, 0.0, 0.0, 0.0]
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 2.55870297830144e-176, 0.0, -2.746423316327898e-176, 0.0, 0.0, 0.0, -4.566674207535345e-175, 0.0, 0.0, 0.0, 0.0]
```

Gambar 6. Representasi vektor kecepatan partikel

Tahap selanjutnya akan diperbarui posisi partikel berdasarkan Persamaan 4. Representasi posisi baru partikel dapat dilihat pada Gambar 7. Di setiap perulangan, posisi partikel akan terus di-update dan mendapat waktu mulai baru yang lebih mendekati posisi optimal). Jika terdapat anomali data setelah perbaruan posisi ini, maka partikel itu akan diabaikan dalam perhitungan Pbest dan Gbest.

```
position = [0, 3, 7, 10, 15, 19, 24, 31, 27, 24, 30, 30, 31, 31, 32, 32, 32, 33]
velocity = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -3.0184800853453293, 0.0, 0.0, 0.0, -0.9372237999369921, 2.073239486066926, 0.0, 0.0, 0.0]
best_position = [0, 3, 7, 10, 15, 19, 24, 31, 27, 24, 30, 30, 31, 31, 32, 32, 32, 33]
best_cost = 154187841.64576125
position = [0, 3, 7, 10, 15, 19, 24, 30, 27, 25, 30, 30, 31, 31, 32, 32, 32, 33]
velocity = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.8128750908431955, 0.0, -0.8994450041048793, 0.0, 0.0, 0.0, 0.0, 1.2052763390076775, 0.0, 0.0, 0.0]
best_position = [0, 3, 7, 10, 15, 19, 24, 30, 27, 25, 30, 30, 31, 31, 32, 32, 32, 33]
best_cost = 156197125.13263556
position = [0, 3, 7, 10, 15, 19, 24, 28, 27, 27, 30, 30, 31, 31, 32, 32, 32, 33]
velocity = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.12975938666277775, 0.0, -1.2586613586387585, 0.0, 0.0, 0.0, 0.0, 1.3261681767285842, 0.0, 0.0, 0.0]
best_position = [0, 3, 7, 10, 15, 19, 24, 28, 27, 27, 30, 30, 31, 31, 32, 32, 32, 33]
best_cost = 162087065.8261592
position = [0, 3, 7, 10, 15, 19, 24, 30, 27, 25, 30, 30, 31, 31, 32, 32, 32, 33]
velocity = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05816265641703583, 0.0, -1.696448096252557, 0.0, 0.0, 0.0, 0.0, 6.574493999940538, 1.6211463580698933, 0.0, 0.0, 0.0]
best_position = [0, 3, 7, 10, 15, 19, 24, 30, 27, 25, 30, 30, 31, 31, 32, 32, 32, 33]
best_cost = 163867028.33399656
position = [0, 3, 7, 10, 15, 19, 24, 31, 27, 26, 30, 30, 31, 31, 32, 32, 32, 33]
velocity = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, -0.6467037446726156, 0.0, -1.7311206081274177, 0.0, 0.0, 0.0, -0.6550472958901845, 0.0, 0.0, 0.0, 0.0]
best_position = [0, 3, 7, 10, 15, 19, 24, 31, 27, 26, 30, 30, 31, 31, 32, 32, 32, 33]
best_cost = 144526840.58380623
```

Gambar 7. Representasi posisi baru partikel

Proses di atas akan terus berulang sampai menemui kondisi berhenti. Kondisi berhenti di penelitian ini ada 2, yaitu:

1. Sudah mencapai angka konvergen (selisih 0.00001 dari hasil Gbest sebelumnya) sebanyak 5000 kali berturut-turut.
2. Mencapai jumlah maksimal iterasi 20.000 kali

Setelah proses iterasi berakhir, nilai akhir Gbest yang merupakan nilai rata-rata deviasi biaya mingguan akan dibandingkan dengan perhitungan deviasi biaya rata-rata menggunakan waktu mulai kegiatan berdasarkan ES (CPM *Earliest*) dan LS (CPM *Latest*). Jika hasil akhir dengan menggunakan PSO lebih kecil nilainya daripada CPM *Earliest* dan CPM *Latest*, maka dapat dikatakan metode optimalisasi penyeimbangan biaya dengan menggunakan kombinasi CPM dan PSO bisa memberikan nilai yang lebih optimal daripada CPM saja.

4. HASIL DAN PEMBAHASAN

Berdasarkan hasil optimalisasi menggunakan metode PSO, didapat solusi partikel yang memiliki nilai fitness terbaik. Solusi partikel berbentuk minggu dimulainya masing-masing kegiatan kemudian disandingkan dengan jadwal berdasarkan CPM *Earliest* dan *Latest* dan dapat dilihat pada Gambar 8. Pada gambar tersebut bisa terlihat bahwa kegiatan yang ada pada jalur non-kritis terdapat perbedaan waktu dimulainya kegiatan.

Kemudian, nilai dari penyimpangan tiap minggu untuk ketiga metode tersebut disandingkan dan bisa terlihat pada Gambar 9. Pada minggu ke-25 sampai ke-33 terdapat perbedaan nilai penyimpangan biaya mingguan.

NO	KEGIATAN	DURASI	CPM - Earliest	CPM - Latest	PSO
1	PEKERJAAN PERSIAPAN	3	1	1	1
2	PEKERJAAN TANAH, PASIR DAN PONDASI	4	4	4	4
3	STRUKTUR LANTAI 1	3	8	8	8
4	STRUKTUR LANTAI 2	5	11	11	11
5	STRUKTUR LANTAI 3	4	16	16	16
6	STRUKTUR LANTAI TOP	5	20	20	20
7	PEKERJAAN PASANGAN DAN PLESTERAN	3	25	25	25
8	PEKERJAAN WATERPROOFING	1	28	33	32
9	PEKERJAAN PELAPIS LANTAI DAN DINDING	3	28	28	28
10	PEKERJAAN ATAP, PLAFOND DAN PARTISI GYPSUM	3	25	29	25
11	PEKERJAAN KUSEN, PINTU, JENDELA DAN VENTILASI	1	31	31	31
12	PEKERJAAN ACP DAN RAILLING	1	31	31	31
13	PEKERJAAN FINISHING / PENGECATAN	1	32	32	32
14	PEKERJAAN SANITAIR DAN PLUMBING	1	31	33	33
15	PEKERJAAN ELEKTRIKAL / MEKANIKAL	1	31	33	33
16	PEKERJAAN HURUF ACRYLIC	1	33	33	33
17	PEKERJAAN LOGO	1	33	33	33
18	PEMBERSIHAN AKHIR	1	34	34	34

Gambar 8. Perbandingan waktu mulai kegiatan

MINGGU KE-	CPM - EARLIEST	CPM - LATEST	PSO
1	320.872.601	320.872.601	320.872.601
2	320.872.601	320.872.601	320.872.601
3	320.872.601	320.872.601	320.872.601
4	104.606.472	104.606.472	104.606.472
5	104.606.472	104.606.472	104.606.472
6	104.606.472	104.606.472	104.606.472
7	104.606.472	104.606.472	104.606.472
8	73.411.403	73.411.403	73.411.403
9	73.411.403	73.411.403	73.411.403
10	73.411.403	73.411.403	73.411.403
11	49.407.020	49.407.020	49.407.020
12	49.407.020	49.407.020	49.407.020
13	49.407.020	49.407.020	49.407.020
14	49.407.020	49.407.020	49.407.020
15	49.407.020	49.407.020	49.407.020
16	123.770.481	123.770.481	123.770.481
17	123.770.481	123.770.481	123.770.481
18	123.770.481	123.770.481	123.770.481
19	123.770.481	123.770.481	123.770.481
20	33.673.771	33.673.771	33.673.771
21	33.673.771	33.673.771	33.673.771
22	33.673.771	33.673.771	33.673.771
23	33.673.771	33.673.771	33.673.771
24	33.673.771	33.673.771	33.673.771
25	318.871.702	100.128.992	318.871.702
26	318.871.702	100.128.992	318.871.702
27	318.871.702	100.128.992	318.871.702
28	39.699.013	12.952.216	12.952.216
29	12.952.216	431.952.910	12.952.216
30	12.952.216	431.952.910	12.952.216
31	721.016.930	719.343.248	300.342.553
32	30.259.363	30.259.363	3.512.566
33	328.783.193	118.637.981	91.891.184
34	356.663.193	356.663.193	356.663.193

Gambar 9. Perbandingan penyimpangan biaya mingguan

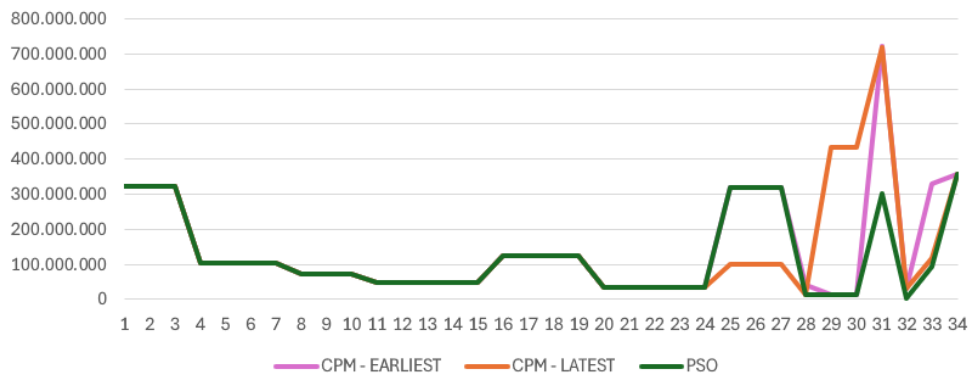
Berdasarkan data yang ada, diambil nilai rata-rata penyimpangan biaya mingguan untuk ketiga metode tersebut. Terlihat bahwa kombinasi metode CPM dan PSO memberikan nilai paling optimal sebesar

Rp125.283.677, dibandingkan dengan CPM *Earliest* dan CPM *Latest*, yaitu Rp146.197.206 dan Rp144.526.841. Perbandingan dapat dilihat pada Tabel 2.

Tabel 2. Perbandingan rata-rata penyimpangan biaya mingguan

Metode	Rata-Rata Penyimpangan Biaya Mingguan
CPM <i>Earliest</i>	Rp146.197.206
CPM <i>Latest</i>	Rp144.526.841
CPM + PSO	Rp125.283.677

Dari perhitungan pada Gambar 9 juga dapat dibuat grafik (Gambar 10) dan bisa terlihat bahwa garis metode CPM + PSO bisa memberikan nilai yang cenderung stabil dan lebih mendekati nilai 0.



Gambar 9. Grafik perbandingan penyimpangan biaya mingguan

Setelah itu dari hasil penjadwalan berdasarkan CPM *Earliest*, *Latest* serta PSO, dibuat Kurva S untuk menggambarkan perkembangan proyek per minggu. Pada kurva S ini terlihat bahwa penjadwalan menggunakan PSO lebih mendekati garis lurus dibandingkan CPM *Earliest* dan *Latest*.



Gambar 10. Kurva S proyek

5. KESIMPULAN

Jadwal proyek konstruksi dapat dioptimasi dengan menggunakan kombinasi CPM dan PSO, dengan nilai akhir Rp.126.857.018, dibandingkan hanya dengan CPM (*Earliest*) Rp.146.197.206 dan CPM (*Latest*) Rp144.526.841. Studi penelitian berikutnya dapat mengoptimalkan aspek lainnya. Selain itu, dapat dipertimbangkan juga pengaruh optimalisasi parameter seperti bobot inerti serta komponen pembelajaran

individu dan sosial. Metode ini juga dapat diterapkan untuk meneliti proyek-proyek lain yang memiliki jumlah kegiatan lebih banyak dan waktu *float* yang lebih besar.

DAFTAR PUSTAKA

- Arifudin, R., 2011, Optimasi Penjadwalan Proyek dengan Penyeimbangan Biaya Menggunakan Kombinasi CPM dan Algoritma Genetika.
- Hafnidar, R., 2016, *Manajemen Proyek Konstruksi*, 1st edn., vol. 1, Deepublish, Yogyakarta.
- Cholissodin, I. & Riyandani, E., 2016, *Buku Ajar Swarm Intelligence*, 1st edn., Malang.
- Kerzner, H., 2009, *Project Management A System Approach to Planning, Scheduling and Controlling*, vol. 10, J. Wiley & Sons.
- Lo, S.-T., Chen, R.-M., Shiau, D.-F. & Wu, C.-L., 2008, Using Particle Swarm Optimization to Solve Resource-constrained Scheduling Problems, *SMCia/08 - Proceedings of the 2008 IEEE Conference on Soft Computing on Industrial Applications*, 38–43.
- Mubarak, S., 2010, *Construction Project Scheduling and Control*, 2nd edn., John Wiley & Sons, Inc., Hoboken.
- Santosa, B., no date, *Tutorial Particle Swarm Optimization*, Surabaya.
- Seixas Gomes De Almeida, B. & Leite, V.C., 2019, *Particle Swarm Optimization: A Powerful Technique for Solving Engineering Problems*, Rio de Janeiro.
- Soeharto, I., 1999, *Manajemen Proyek (Dari Konseptual Sampai Operasional)*, 2nd edn., vol. 1, Erlangga, Jakarta.
- Wang, D., Tan, D. & Liu, L., 2018, Particle swarm optimization algorithm: an overview, *Soft Computing*, 22(2), 387–408.
- Zhou, J., Love, P.E.D., Wang, X., Teo, K.L. & Irani, Z., 2013, A review of methods and algorithms for optimizing construction scheduling, *Journal of the Operational Research Society*, 64(8), 1091–1105.