

**PEMODELAN SISTEM PERSEDIAAN (s,S) DENGAN METODA
PARTICLE SWARM OPTIMIZATION DUA ESELON UNTUK
MENENTUKAN JUMLAH PESANAN OPTIMAL MENGGUNAKAN
SOFWARE MATLAB**

(STUDI KASUS : PT. XYZ KOTA BANDUNG)

TUGAS AKHIR

**Karya tulis sebagai salah satu syarat
untuk memperoleh gelar Sarjana Teknik dari
Program Studi Teknik Industri
Fakultas Teknik Universitas Pasundan**

Oleh

SURYA DALIMAN

NRP : 193010028



**PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN
2025**

**PEMODELAN SISTEM PERSEDIAAN (s,S) DENGAN METODA
PARTICLE SWARM OPTIMIZATION DUA ESELON UNTUK
MENENTUKAN JUMLAH PESANAN OPTIMAL MENGGUNAKAN
SOFWARE MATLAB**

(STUDI KASUS : PT. XYZ KOTA BANDUNG)

SURYA DALIMAN

NRP : 193010028

ABSTRAK

PT. XYZ merupakan perusahaan distributor yang bergerak dalam pendistribusian produk ke berbagai ritel dengan kebutuhan permintaan yang beragam. Permasalahan utama yang dihadapi perusahaan adalah tingginya jumlah persediaan di gudang pusat maupun ritel yang menyebabkan terjadinya penumpukan barang, biaya penyimpanan yang meningkat, serta ketidakefisienan dalam proses pemesanan. Kondisi ini berdampak pada membesarnya total biaya persediaan dan menurunnya efektivitas rantai pasok. Oleh karena itu, diperlukan metode optimisasi yang mampu memberikan solusi efektif dalam penentuan jumlah persediaan yang optimal agar biaya dan tingkat persediaan dapat diminimalkan. Penelitian ini bertujuan untuk meminimalkan jumlah persediaan dan total biaya persediaan baik di gudang pusat maupun ritel dengan menerapkan metode Particle Swarm Optimization (PSO). Metode PSO digunakan untuk mencari nilai optimal dari jumlah pemesanan (order quantity) dan titik pemesanan kembali (reorder point) sehingga diperoleh kombinasi kebijakan persediaan yang paling efisien. Penelitian ini dibagi menjadi dua fokus utama, yaitu meminimalkan jumlah persediaan pada setiap tingkat distribusi serta menentukan biaya persediaan yang paling optimal.

Hasil penelitian menunjukkan bahwa penerapan metode PSO mampu memberikan pengurangan terhadap jumlah persediaan yang berlebihan serta menurunkan total biaya persediaan dibandingkan kondisi eksisting. Dengan demikian, metode PSO dapat menjadi alternatif solusi yang efektif dalam pengambilan keputusan persediaan bagi PT. XYZ guna meningkatkan efisiensi operasional dan performa rantai pasok perusahaan.

Kata kunci: Persediaan, Particle Swarm Optimization, Gudang Pusat, Ritel, Biaya Persediaan, Optimisasi.

MODELING OF INVENTORY SYSTEM (s,S) WITH TWO-ECHELON PARTICLE SWARM OPTIMIZATION METHOD TO DETERMINE OPTIMAL ORDER QUANTITY USING MATLAB SOFTWARE

(CASE STUDY: PT. XYZ, BANDUNG)

SURYA DALIMAN

NRP: 193010028

ABSTRACT

PT. XYZ is a distribution company engaged in the distribution of products to various retailers with diverse demand requirements. The main problem faced by the company is the high amount of inventory in the central warehouse and retailers, which causes stockpiling, increased storage costs, and inefficiency in the ordering process. This condition has an impact on the increase in total inventory costs and a decrease in the effectiveness of the supply chain. Therefore, an optimization method is needed that can provide an effective solution in determining the optimal inventory level so that costs and inventory levels can be minimized. This study aims to minimize inventory levels and total inventory costs in both the central warehouse and retail outlets by applying the Particle Swarm Optimization (PSO) method. The PSO method is used to find the optimal value of the order quantity and reorder point so that the most efficient combination of inventory policies can be obtained. This study is divided into two main focuses, namely minimizing inventory levels at each distribution level and determining the most optimal inventory costs.

The results of the study indicate that the application of the PSO method can reduce excess inventory and lower total inventory costs compared to existing conditions. Thus, the PSO method can be an effective alternative solution in inventory decision-making for PT. XYZ to improve operational efficiency and the company's supply chain performance.

Keywords: Inventory, Particle Swarm Optimization, Central Warehouse, Retail, Inventory Costs, Optimization.

**PEMODELAN SISTEM PERSEDIAAN (S,S) DENGAN METODA
PARTICLE SWARM OPTIMIZATION DUA ESELON UNTUK
MENENTUKAN JUMLAH PESANAN OPTIMAL MENGGUNAKAN
SOFTWARE MATLAB**

(STUDI KASUS : PT. XYZ KOTA BANDUNG)

SURYA DALIMAN

NRP : 193010028

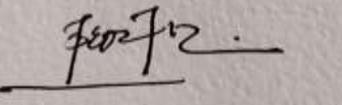
Menyetujui

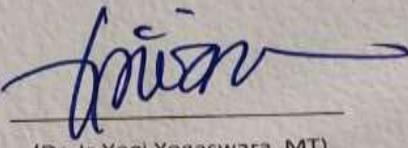
Tim Pembimbing

Tanggal 29 - 12 - 2025

Pembimbing

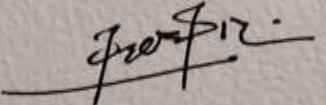
Penelaah


(Dr. Ir. M. Nurman Helmi, DEA)


(Dr. Ir. Yogi Yogaswara, MT)

Mengetahui,

Ketua Program Studi


(Dr. Ir. M. Nurman Helmi, DEA)

PEDOMAN PENGGUNAAN TUGAS AKHIR

Tugas Akhir Sarjana yang tidak dipublikasikan terdaftar dan tersedia di Perpustakan Universitas Pasundan, dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada pengarang dengan mengikuti aturan HaKI yang berlaku di Universitas Pasundan. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin pengarang dan harus disertai dengan kebiasaan ilmiah untuk menyebutkan sumbernya.

Memperbanyak atau menerbitkan sebagian atau seluruh Tugas Akhir haruslah seizin Program Studi Teknik Industri, Fakultas Teknik, Universitas Pasundan.

PERNYATAAN

Dengan ini Saya Menyatakan bahwa Judul Tugas Akhir :

PEMODELAN SISTEM PERSEDIAAN (s,S) DENGAN METODA PARTICLE SWARM OPTIMIZATION DUA ESELON UNTUK MENENTUKAN JUMLAH PESANAN OPTIMAL MENGGUNAKAN SOFWARE MATLAB

(STUDI KASUS : PT. XYZ KOTA BANDUNG)

Adalah hasil kerja saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya dengan cara penulisan referensi yang sesuai. Pernyataan ini saya buat dengan sebenar-benarnya dan jika pernyataan ini tidak sesuai dengan kenyataan maka saya bersedia menanggung sanksi yang akan dikenakan sesuai dengan kentuan berlaku.

Bandung, 2025

Surya Daliman

NRP : 193010028

Dipersembahkan

Kepada kedua orangtuaku, Istriku , kakaku dan orang-orang tersayang

Yang telah memberikan dukungan, do'a serta bantuan.

KATA PENGANTAR

Puji Syukur dipanjatkan kepada Allah SWT, berkat Rahmat dan Hidayahnya akhirnya penyusunan Laporan Tugas Akhir ini dapat diselesaikan tepat pada waktunya. Pada kesempatan ini disampaikan ucapan terima kasih yang tiada terhingga kepada:

1. Allah SWT yang telah mengizinkan penulis menyelesaikan Laporan Tugas Akhir sampai selesai.
2. Kedua orang tua, yang telah memberikan dukungan moril maupun materi dan dukungan penuh serta doa kepada penulis.
3. Bapak Dr. Ir. Muhammad Nurman Helmi, DEA. Selaku dosen pembimbing Tugas Akhir yang telah bersedia meluangkan waktu, memberi bimbingan, saran dan bantuan kepada penulis dalam menyelesaikan Laporan Tugas Akhir ini.
4. Bapak Dr. Ir. Yogi Yogaswara, MT, selaku dosen penelaah atas bimbingan dan Koordinator Tugas Akhir Program Studi Teknik Industri Universitas Pasundan, saran dan nasihat sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini.
5. Bapak Dr. Ir. Muhammad Nurman Helmi, DEA. selaku dosen wali yang senantiasa memberikan arahan dan bimbingan selama proses perkuliahan sampai Tugas Akhir ini.
6. Seluruh pihak yang tidak dapat disebutkan satu persatu yang telah membantu sehingga penulis dapat menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kata sempurna dikarnakan terbatasnya pengalaman dan pengetahuan yang dimiliki penulis, oleh karna itu, penulis mengharapkan segala bentuk saran serta masukan bahkan keritikan yang membangun dari berbagai pihak. Semoga Tugas Akhir ini dapat bermanfaat bagi para pembaca dan semua pihak.

Bandung, 2025

Surya Daliman

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
PEDOMAN PENGGUNAAN TUGAS AKHIR	iv
PERNYATAAN	v
KATA PENGANTAR	vii
DAFTAR ISI	viii
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
LAMPIRAN	xv
BAB I Pendahuluan	I-1
I.1 Latar Belakang	I-1
I.2 Rumusan Masalah	I-5
I.3 Tujuan Penelitian	I-5
I.4 Manfaat Penelitian	I-5
I.5 Asumsi dan Batasan Masalah	I-6
I.5.1 Asumsi	I-6
I.5.2 Batasan Masalah	I-6
I.6 Sistematika Penulisan.....	I-6
BAB II Tinjauan Pustaka dan Landasan Teori	II-1
II.1 Supplay Chain Management.....	II-1
II.1.1 Komponen Utama Supplay Chain	II-5

II.1.2 Tujuan Supplay Chain	II-6
II.1.3 Manfaat Supplay Chain	II-7
II.2 Manajemen Persediaan.....	II-8
II.2.1 Inventori Control	II-8
II.2.2 Jenis-jenis Persediaan	II-11
II.2.3 Biaya Dalam Persediaan	II-13
II.2.4 Pengelolaan Persediaan.....	II-14
II.2.5 Model Persediaan	II-16
II.3 Economic Order Quantity (EOQ)	II-20
II.3.1 Persediaan Dalam EOQ	II-23
II.4 Reorder Point (Titik Pemesanan Kembali)	II-26
II.5 Metaheuristik	II-27
II.5.1 Jenis-jenis Metaheuristik	II-29
II.6 Particle Swarm Optimization (PSO)	II-32
II.6.1 Prosedur Algoritma Particle Swarm Optimization.....	II-40
II.6.2 Inisialisasi Populasi	II-41
II.6.3 Personal Best <i>pit</i>	II-41
II.6.4 Global Best (<i>Gt</i>)	II-41
II.6.5 Update Velocity <i>vijt</i>	II-42
II.6.6 Interia Weight.....	II-43
II.6.7 Cognitive Dan Social Parameter C1 C2	II-43
II.7 Penelitian Terdahulu	II-43
BAB III Usulan Pemecahan Masalah.....	III-1
III.1 Metodologi Penelitian.....	III-1

III.2 Model Pemecahan Masalah.....	III-2
III.3 Studi Literatur.....	III-4
III.4 Identifikasi Masalah.....	III-4
III.5 Tujuan Pemecahan Masalah.....	III-5
III.6 Pengumpulan Data.....	III-5
III.7 Pengolahan Data.....	III-5
III.7.1 Optimisasi Model Persediaan s,S Menggunakan Particle Swarm Optimization (PSO).....	III-6
III.8 Analisa dan Pembahasan.....	III-11
III.9 Kesimpulan dan Saran	III-11
BAB IV Pengumpulan dan Pengolahan Data	IV-1
IV.1 Pengumpulan Data	IV-1
IV.2 Gambaran Umum Perusahaan.....	IV-1
IV.2.1 Sejarah PT.XYZ Hijab Fashion.....	IV-1
IV.2.2 Profil Perusahaan	IV-2
IV.2.3 Data Permintaan.....	IV-3
IV.2.4 Data Lead Time	IV-5
IV.2.5 Biaya Pemesanan	IV-6
IV.2.6 Biaya Penyimpanan Produk.....	IV-7
IV.2.7 Data Harga Produk.....	IV-8
IV.3 Pengolahan Data.....	IV-8
IV.3.1 Optimisasi Model Persediaan s,S Menggunakan Particle Swarm Optimization	IV-8
IV.3.2 Model Persediaan s,S	IV-9
IV.3.3 Model Particle Swarm Optimization.....	IV-19

BAB V Analisa dan Pembahasan	V-1
V.1 Analisa	V-1
V.1.1 Analisa Perbandingan Hasil Pengolahan Data.....	V-1
V.2 Pembahasan	V-3
BAB VI Kesimpulan dan Saran	VI-1
VI.1 Kesimpulan	VI-1
VI.2 Saran	VI-2

DAFTAR TABEL

Tabel II. 1Penelitian Terdahulu	II-51
Tabel IV. 1 Data Permintaan (Demand) Tingkat Gudang Pusat	IV-4
Tabel IV. 2Data Permintaan (Demand) Tingkat Ritel	IV-4
Tabel IV. 3 Lead Time Tingkat Ritel.....	IV-5
Tabel IV. 4 Biaya Pemesanan Tingkat Gudang Pusat	IV-7
Tabel V. 1 Perbandingan Hasil Perhitungan Manual dan optimisasi Gudang pusat	V-2
Tabel V. 2 Perbandingan Hasil Perhitungan Manual dan optimisasi Ritel	V-2
Tabel V. 3 Hasil Perbandingan Total Inventory Cost (TIC)	V-6

DAFTAR GAMBAR

Gambar II.1 Konsep Rantai Pasok (Supply Chain)	II-3
Gambar II.2 Komponen Utama Supply Chain	II-5
Gambar II.3 Grafik Continuous Review, Fixed Order Quantity (s, Q)	II-9
Gambar II.4 Grafik Continuous Review, Order-Up-to-Level (s, S).....	II-10
Gambar II.5 Periodic Review, Order-Up-to-Level (T, S).....	II-11
Gambar II.6 Sistem Dua Eselon	II-18
Gambar II.7 System Persediaan Multieselon	II-19
Gambar II.8 Grafik Persediaan dalam Model EOQ.....	II-21
Gambar II.9 Grafik Siklus Persediaan	II-25
Gambar II.10 Kurva Biaya Persediaan	II-26
Gambar II.11 Grafik Siklus Persediaan Stokastik	II-26
Gambar II.12 Algoritma PSO.....	II-37
Gambar II.13 Diagram alir algoritma PSO	II-40
Gambar III. 1 Metodologi Penelitian	III-1
Gambar III. 2 Flowchart Langkah-langkah Pemecahan Masalah	III-3
Gambar IV. 1 Jaringan Pasokan Dua Eselon	IV-8
Gambar IV. 2 Deklarasi Variabel Model s,S.....	IV-10
Gambar IV. 3 Perhitungan Biaya Nilai Tambah	IV-11
Gambar IV. 4 Deklarasi Nilai n Setiap Ritel.....	IV-15
Gambar IV. 5 Deklarasi Perhitungan Jumlah Pesan Optimal (Qw dan Qr).....	IV-16
Gambar IV. 6 Deklarasi Perhitungan Reorder Point dan Tingkat Persediaan Ritel.....	IV-18
Gambar IV. 7 Deklarasi Perhitungan Total Inventory Cost	IV-19
Gambar IV. 8 Deklarasi Inisialisasi Parameter Awal	IV-21
Gambar IV. 9 Deklarasi Inisialisasi Partikel	IV-23
Gambar IV. 10 Deklarasi Randomize Position dan Valecity	IV-24
Gambar IV. 11 Deklarasi Hitung pBest dan gBest	IV-26
Gambar IV. 12 Deklarasi Update Valocity dan Position	IV-27

Gambar IV. 13 Deklarasi Evaluasi Fungsi Objektif dan Fitness.....	IV-29
Gambar IV. 14 Deklarasi Update pBest dan gBest	IV-31
Gambar IV. 15 Deklarasi Reorder Point	IV-36
Gambar IV. 16 Grafik Iteration Jumlah Optimal Tingkat Gudang Pusat	IV-36
Gambar IV. 17 Grafik Iteration Reorder Point Tingkat Gudang Pusat	IV-37
Gambar IV. 18 Hasil Optimisasi PSO di Tingkat Gudang Pusat	IV-38
Gambar IV. 19 Hasil Optimisasi PSO di Tingkat Ritel	IV-38
Gambar IV. 20 Grafik Iteration Jumlah Optimal Total Inventory Cost.....	IV-39
Gambar V. 1 Grafik Perbandingan Kondisi Eksisting dan Optimasi.....	V-5

DAFTAR LAMPIRAN

- Lampiran 1 hasil optimisasi pso jumlah optimal di tingkat ritel..... L-1
Lampiran 2 hasil optimisasi pso reorder poin di tingkat ritel L-9
Lampiran 3 perhitungan manual Economic Order Quantity..... L-17

DAFTAR SINGKATAN DAN LAMBANG

Lambang	Keterangan
C_{ow}	biaya pemesanan per pesanan untuk gudang.
C_{or}	biaya pemesanan per pesanan untuk pengecer.
C_w	harga satuan barang di Gudang.
C_r	harga satuan barang di Ritel.
C_w^c	biaya penambahan Nilai di Gudang
C_r^c	biaya penambahan Nilai di Ritel
i	nilai tercatat per tahun
s_w	titik pemesanan ulang di Gudang
s_r	titik pemesanan ulang di Ritel
Q_w	jumlah pesanan untuk Gudang.
Q_r	jumlah pesanan untuk Ritel.
D_w	permintaan waktu tunggu di Gudang
D_r	permintaan waktu tunggu di Ritel
L	waktu tunggu
HK	Hari Kerja
I_w^c	tingkat persediaan Eselon Gudang Pusat
I_r^c	tingkat persediaan Eselon Ritel

Lambang	Keterangan
cw	Biaya Satuan Produk Gudang
cr	Biaya Satuan Produk Ritel
ccw	Biaya Nilai Tambah Gudang
ccr	Biaya Nilai Tambah Ritel
cow	Biaya Pesan Gudang
cor_n	Biaya Pesan Ritel ke-1 sampai Ritel ke-n
dw	Demand Gudang
dr_n	Demand Ritel ke-1 sampai Ritel ke-n
qw	Jumlah Pesan Optimal Gudang
qrr_n	Jumlah Pesan Optimal Ritel ke-1 sampai Ritel ke-n
iw	Persedian Gudang
irr_1	Persediaan Ritel ke-1 sampai Ritel ke-n
icw	Tingkat Persediaan Gudang
icr_1	Tingkat Persediaan Ritel ke-1 sampai Ritel ke-n
L_w	Lead Time Gudang
L_r	Lead Time Ritel
s_w	Reorder Point Gudang
s_r	Reorder Point Ritel
v_{ij}^t	velocity yang terletak pada partikel i, job j untuk iterasi 1-1
w^{t-1}	inertia weight pada iterasi 1-1

Lambang	Keterangan
x_{ij}^t	position value yang terletak pada partikel i, job j untuk iterasi t
x_{ij}^{t-1}	position value yang terletak pada partikel i, job j untuk iterasi t-1
p_{ij}^{t-1}	personal best yang terletak pada partikel i, job j untuk iterasi 1-
g_{ij}^{t-1}	global best pada job j untuk iterasi -1
$c_1 c_2$	cognitivedan social parameter
$r_1 r_2$	random uniforam [0,1]
w^t	inertia weight pada iterasi
w^{t-1}	inertia weight pada iterasi 1-1

BAB I Pendahuluan

I.1 Latar Belakang

Continous Review Pixed Order Up-To-Level Dalam era globalisasi dan persaingan yang semakin ketat, perusahaan dihadapkan pada tekanan untuk menjaga efisiensi operasional. Salah satu pondasi utama dalam operasional perusahaan adalah persediaan barang Menurut Handoko (2015). Perusahaan yang bergerak dalam bidang industri baik itu perusahaan besar, menengah, ataupun perusahaan kecil sudah tentu memiliki strategi dalam pengelolaan persediaan. Persediaan adalah barang yang disimpan untuk digunakan atau dijual pada periode mendatang (Kusuma, 2001). manajemen persediaan membantu perusahaan menjaga tingkat persediaan yang tepat, mengoptimalkan penggunaan sumber daya, mengurangi biaya penyimpanan, dan menjaga kelancaran operasional perusahaan secara keseluruhan.

Pengelolaan persediaan yang baik sangat penting untuk menjaga kelancaran operasional dan menghindari kesenjangan antara permintaan dan persediaan produk atau bahan. Seperti yang dijelaskan oleh Dasgupta et al., (2019)Oleh karena itu, manajemen persediaan telah menjadi komponen penting dalam pengambilan keputusan bisnis, terutama dalam menghadapi peningkatan permintaan costumer (Schiff dan Lieber, 1974, Zietsman dan Van Vuuren, 2022). Manajemen persediaan yang efektif bisa mengurangi biaya persediaan (Shirley & Winston, 2004) dan meningkatkan pangsa pasar perusahaan (Atnafu dan Balda, 2018, Maji et al., 2022) dan daya saing (Chan et al., 2017), yang pada akhirnya berkontribusi pada peningkatan kinerja bisnis (Nyabwanga dan Ojera, 2012, Prinotakis dan Argyropoulos, 2018).

Perusahaan lebih mengutamakan rantai persediaan untuk mencapai keunggulan kompetitif yang dapat menunjang kelangsungan bisnisnya. Rantai persediaan juga dikenal sebagai jaringan logistik, yang terdiri dari para penyalur pusat, pabrikasi, gudang, pusat distribusi, dan toko pengecer. Manajemen rantai pasok juga mempertimbangkan fasilitas yang berdampak signifikan dan berperan dalam membuat

produk untuk dapat menyesuaikan dengan kebutuhan pelanggan. Rantai pasokan terdiri dari berbagai pihak yang terlibat baik secara langsung maupun tidak langsung dalam memenuhi permintaan konsumen. Di dalam suatu organisasi, misalnya dalam suatu pabrik manufaktur, rantai pasok meliputi fungsi-fungsi yang menerima dan memenuhi permintaan konsumen. Fungsi tersebut meliputi pengembangan produk baru, pemasaran, operasi, distribusi, keuangan, dan layanan pelanggan (Chopra dan Meindl, 2016). Lebih luas, rantai pasokan tidak hanya terdiri dari pabrik manufaktur dan pemasok saja, tetapi juga meliputi perusahaan penyedia transportasi, gudang, retailer, dan bahkan konsumen itu sendiri.

Manajemen rantai pasok dapat diartikan sebagai usaha mengkoordinasikan dan mengintegrasikan sejumlah aktivitas yang terkait dengan produk, di dalam rantai pasok untuk meningkatkan efisiensi operasi, kualitas, dan layanan pelanggan agar dapat memperoleh keunggulan bersaing yang berkelanjutan bagi semua organisasi yang berkolaborasi (Wisner, Tan, dan Leong; 2009). Oleh karenanya, agar manajemen rantai pasok berhasil, perusahaan yang terlibat di dalamnya harus bekerja sama untuk berbagi informasi misalnya terkait dengan peramalan permintaan konsumen, perencanaan produksi, perubahan kapasitas, strategi pemasaran baru, pengembangan produk atau jasa baru, pemanfaatan teknologi baru, perencanaan pembelian, tanggal pengiriman, dan hal-hal lain yang berpengaruh terhadap rencana pembelian, produksi, dan distribusi.

Rantai pasok pada dasarnya terdiri dari beberapa elemen, diantaranya pemasok (supplier), pusat manufaktur, gudang, pusat distribusi, sistem transportasi, ritel, dan konsumen. Salah satu aspek yang penting dalam rantai pasokan adalah integrasi dan koordinasi dari semua aktivitas yang terjadi di dalam rantai pasok, suatu keputusan yang diambil akan berpengaruh langsung terhadap seluruh rantai pasok. Suatu perusahaan harus mengelola rantai pasokan sebagai satu entitas. Tujuan dari manajemen rantai pasokan adalah mencapai biaya yang minimum dan tingkat pelayanan yang maksimum. Manajemen rantai pasokan mempertimbangkan semua fasilitas yang berpengaruh terhadap produk yang dihasilkan dan biaya yang diperlukan dalam memenuhi kebutuhan konsumen. Aktivitas aktivitas tersebut meliputi pembelian dan outsourcing activities

ditambah dengan fungsi-fungsi lain yang akan meningkatkan hubungan antara pemasok dan distributor.

Dalam era globalisasi dan persaingan yang semakin ketat, perusahaan dihadapkan pada tekanan untuk menjaga efisiensi operasional. Salah satu pondasi utama dalam operasional perusahaan adalah persediaan barang Menurut Handoko (2015). Perusahaan yang bergerak dalam bidang industri baik itu perusahaan besar, menengah, ataupun perusahaan kecil sudah tentu memiliki strategi dalam pengelolaan persediaan. Persediaan adalah barang yang disimpan untuk digunakan atau dijual pada periode mendatang (Kusuma, 2001). manajemen persediaan membantu perusahaan menjaga tingkat persediaan yang tepat, mengoptimalkan penggunaan sumber daya, mengurangi biaya penyimpanan, dan menjaga kelancaran operasional perusahaan secara keseluruhan.

Pengelolaan persediaan yang baik sangat penting untuk menjaga kelancaran operasional dan menghindari kesenjangan antara permintaan dan persediaan produk atau bahan. Seperti yang dijelaskan oleh Dasgupta et al., (2019) Oleh karena itu, manajemen persediaan telah menjadi komponen penting dalam pengambilan keputusan bisnis, terutama dalam menghadapi peningkatan permintaan costumer (Schiff dan Lieber, 1974, Zietsman dan Van Vuuren, 2022). Manajemen persediaan yang efektif bisa mengurangi biaya persediaan (Shirley & Winston, 2004) dan meningkatkan pangsa pasar perusahaan (Atnafu dan Balda, 2018, Maji et al., 2022) dan daya saing (Chan et al., 2017), yang pada akhirnya berkontribusi pada peningkatan kinerja bisnis (Nyabwanga dan Ojera, 2012, Prinotakis dan Argyropoulos, 2018).

Perusahaan lebih mengutamakan rantai persediaan untuk mencapai keunggulan kompetitif yang dapat menunjang kelangsungan bisnisnya. Rantai persediaan juga dikenal sebagai jaringan logistik, yang terdiri dari para penyalur pusat, pabrikasi, gudang, pusat distribusi, dan toko pengecer. Manajemen rantai pasok juga mempertimbangkan fasilitas yang berdampak signifikan dan berperan dalam membuat produk untuk dapat menyesuaikan dengan kebutuhan pelanggan. Rantai pasokan terdiri dari berbagai pihak yang terlibat baik secara langsung maupun tidak langsung dalam

memenuhi permintaan konsumen. Di dalam suatu organisasi, misalnya dalam suatu pabrik manufaktur, rantai pasok meliputi fungsi-fungsi yang menerima dan memenuhi permintaan konsumen. Fungsi tersebut meliputi pengembangan produk baru, pemasaran, operasi, distribusi, keuangan, dan layanan pelanggan (Chopra dan Meindl, 2016). Lebih luas, rantai pasokan tidak hanya terdiri dari pabrik manufaktur dan pemasok saja, tetapi juga meliputi perusahaan penyedia transportasi, gudang, retailer, dan bahkan konsumen itu sendiri.

Manajemen rantai pasok dapat diartikan sebagai usaha mengkoordinasikan dan mengintegrasikan sejumlah aktivitas yang terkait dengan produk, di dalam rantai pasok untuk meningkatkan efisiensi operasi, kualitas, dan layanan pelanggan agar dapat memperoleh keunggulan bersaing yang berkelanjutan bagi semua organisasi yang berkolaborasi (Wisner, Tan, dan Leong; 2009). Oleh karenanya, agar manajemen rantai pasok berhasil, perusahaan yang terlibat di dalamnya harus bekerja sama untuk berbagi informasi misalnya terkait dengan peramalan permintaan konsumen, perencanaan produksi, perubahan kapasitas, strategi pemasaran baru, pengembangan produk atau jasa baru, pemanfaatan teknologi baru, perencanaan pembelian, tanggal pengiriman, dan hal-hal lain yang berpengaruh terhadap rencana pembelian, produksi, dan distribusi.

Rantai pasok pada dasarnya terdiri dari beberapa elemen, diantaranya pemasok (supplier), pusat manufaktur, gudang, pusat distribusi, sistem transportasi, retail , dan konsumen. Salah satu aspek yang penting dalam rantai pasokan adalah integrasi dan koordinasi dari semua aktivitas yang terjadi di dalam rantai pasok, suatu keputusan yang diambil akan berpengaruh langsung terhadap seluruh rantai pasok. Suatu perusahaan harus mengelola rantai pasokan sebagai satu entitas. Tujuan dari manajemen rantai pasokan adalah mencapai biaya yang minimum dan tingkat pelayanan yang maksimum. Manajemen rantai pasokan mempertimbangkan semua fasilitas yang berpengaruh terhadap produk yang dihasilkan dan biaya yang diperlukan dalam memenuhi kebutuhan konsumen. Aktivitas aktivitas tersebut meliputi pembelian dan outsourcing activities ditambah dengan fungsi-fungsi lain yang akan meningkatkan hubungan antara pemasok dan distributor.

I.2 Rumusan Masalah

Adapun rumusan masalah berdasarkan latar belakang masalah yang telah disampaikan sebelumnya, maka rumusan masalah pada PT.XYZ Dalam konteks penelitian ini, dibagi menjadi dua aspek utama:

1. Bagaimana meminimalkan jumlah persediaan di Gudang pusat dan di tingkat Ritel dengan optimisasi menggunakan metode Particle Swarm Optimization pada perusahaan PT.XYZ ?
2. Bagaimana menentukan jumlah biaya yang optimal baik di tingkat gudang pusat dan di tingkat ritel dengan menggunakan metode Particle Swarm Optimization pada perusahaan PT.XYZ ?

I.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disampaikan, maka didapatkanlah tujuan pemecahan masalah dari rumusan masalah tersebut, adalah sebagai berikut :

1. Untuk dapat meminimalkan jumlah persediaan di gudang pusat agar tidak terjadi penumpukan barang di tingkat ritel dengan optimisasi menggunakan metode Particle Swarm Optimization (PSO).
2. Untuk dapat memnimalkan biaya persediaan baik di tingkat gudang pusat dan di tingkat ritel optimisasi menggunakan metode Particle Swarm Optimization (PSO).

I.4 Manfaat Penelitian

Adapun manfaat yang diperoleh dari penelitian ini di PT.XYZ, adalah sebagai berikut:

1. Dapat menentukan jumlah persediaan yang optimal di tingkat gudang pusat dan di tingkat ritel di perusahaan PT.XYZ
2. Dapat meminimalkan biaya persediaan di tingkat gudang pusat dan di tingkat ritel di perusahaan PT.XYZ

I.5 Asumsi dan Batasan Masalah

Adapun asumsi dan Batasan masalah yang terdapat dalam penelitian di PT.XYZ adalah sebagai berikut :

I.5.1 Asumsi

Dengan adanya keterbatasan data yang diperoleh maka perlu penetapan data berdasarkan asumsi. Sebagaimana yang akan digunakan dalam pengolahan data terkait permasalahan yang ada didalam perusahaan, berikut asumsi yang diterapkan adalah sebagai berikut :

1. Permintaan ritel bersifat stabil atau dapat diperkirakan berdasarkan data historis
2. Biaya-biaya persediaan (biaya pesan, biaya simpan, biaya kehabisan stok) telah diketahui secara jelas dan tidak berubah selama periode penelitian.
3. Sistem distribusi bersifat satu arah, yaitu dari gudang pusat ke ritel, tanpa adanya aliran barang balik (reverse logistics).
4. Setiap ritel diasumsikan memiliki pola permintaan yang dapat dianalisis secara independen, meskipun tetap terhubung dengan gudang pusat.

I.5.2 Batasan Masalah

Adapun Batasan masalah dalam penelitian ini dengan tujuan agar memperoleh pokok pembahasan yang akurat sehingga tidak meluas dari pokok permasalahannya, berikut Batasan masalah pada penelitian ini adalah sebagai berikut :

1. Penelitian ini dilakukan di PT.XYZ yang berlokasi di Bandung
2. Penelitian ini mengenai persediaan dalam sistem dua eselon yang mencakup anatara Gudang pusat dan Ritel untuk mencari jumlah persedian optimal.
3. Penelitian ini mengenai persediaan dalam sistem dua eselon yang mencakup anatara Gudang pusat dan Ritel untuk mencari jumlah biaya pemesanan optimal.

I.6 Sistematika Penulisan

Adapun sistematika penulisan sebagaimana yang diperuntukan pembaca guna mempermudah dalam pembuatan penyusunan dan pembaca pada laporan ini. Maka,

dibuatlah laporan ini berdasarkan sistematika penulisan tugas akhir, adalah sebagai berikut

BAB I PENDAHULUAN

Bab I ini berisikan latar belakang masalah yang terdapat pada PT.XYZ. terdapat juga rumusan masalah, tujuan penelitian, manfaat penelitian, asumsi dan batasan masalah dari penelitian yang dilakukan serta sistematika penulisan laporan.

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

Dalam bab ini berisikan mengenai teori-teori sebagai dasar pendukung pada penelitian ini agar memperoleh hasil penelitian yang terverifikasi yang relevan untuk dijadikan landasan rangka berpikir dalam penyusunan pada laporan penelitian ini. Teori ini Membahas konsep dasar tentang apa itu persediaan, jenis-jenis persediaan, dan peran persediaan dalam rantai pasokan Menjelaskan teori di perhitungan Economic Order Quantity (EOQ), formula, dan bagaimana EOQ digunakan dalam manajemen persediaan mendalam pada penggunaan metode particle swarm optimization PSO dalam konteks manajemen persediaan dua eselon. Selain itu, terdapat pula pembahasan mengenai penelitian terdahulu yang sesuai dengan permasalahan pada penelitian ini tujuan penelitian terdahulu bertujuan untuk referensi dalam mengatasi permasalahan tersebut .

BAB III METODOLOGI PENELITIAN

Dalam bab ini berisikan tentang model usulan pemecahan masalah yang berisikan tahapan-tahapan untuk menentukan jumlah persediaan di tingkat ritel dan gudang pusat, menentukan biaya pesan di setiap ritel dan gudang pusat, menentukan biaya simpan disetiap ritel dan gudang pusat, dan menentukan reorder point pada setiap ritel dan gudang pusat, serta tahapan penjelasan rumus-rumus persedian untuk menentukan menghitung jumlah persediaan yang optimal dengan langkah-langkah pemecahan masalah hingga pada kesimpulan.

BAB IV ANALISIS DATA

Dalam bab ini membahas mengenai pengumpulan data yang diperoleh selama penelitian pada PT.XYZ. data yang diperoleh mengenai jumlah permintaan pada setiap ritel dan demand persediaan pada gudang pusat, biaya pesan di tingkat ritel dan gudang, biaya simpan di tingkat ritel dan gudang yang dikeluarkan PT.XYZ. Selanjutnya data yang telah didapatkan dilakukanlah pengolahan data untuk memperoleh jumlah pemesanan yang optimal, jumlah persediaan yang optimal.

BAB V HASIL DAN PEMBAHASAN

Dalam bab ini tentang analisa dan pembahasan terkait dengan pengolahan data yang telah dilakukan pada bab sebelumnya. Penejelasan yang tersampaikan dalam analisa berkaitan dengan hasil akhir yang diperoleh untuk menentukan jumlah peresediaan yang optimal dan pemesanan yang optimal baik di tingkat gudang dan ritel di PT.XYZ. begitu pula pembahasan

BAB VI PENUTUP

Dalam bab ini membahas tentang kesimpulan sebagai hasil akhir dari berbagai permasalahan dari penelitian serta sebagai jawaban pada perumusan masalah yang telah tersampaikan sebelumnya. Dan saran yang diperuntukan bagi perusahaan sebagai bahan evaluasi agar menjadikan perusahaan lebih baik lagi dalam pengelolaan untuk menentukan jumlah pemesanan dan persediaan yang lebih akurat agar tidak terjadi kekurangan stok ataupun kelebihan stok. Serta saran yang diperuntukan pula bagi peneliti selanjutnya sebagai referensi dan bahan penyempurnaan pada masa yang akan datang.

BAB II Tinjauan Pustaka dan Landasan Teori

II.1 Supplay Chain Management

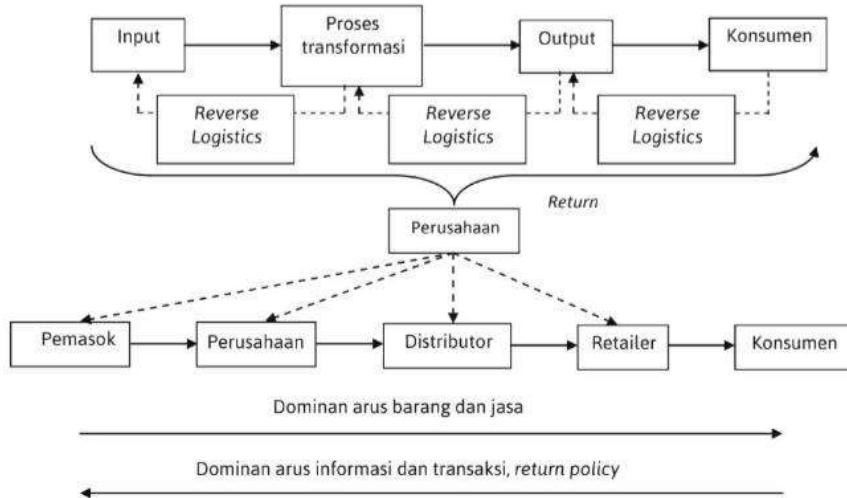
Dalam era globalisasi dan kompleksitas bisnis saat ini, *supply chain* atau rantai pasok menjadi elemen krusial dalam menjaga kelancaran operasional perusahaan. Rantai pasok mencakup serangkaian proses yang melibatkan pergerakan produk atau layanan dari tahap produksi hingga konsumen akhir. Istilah *Supply Chain* diperkenalkan oleh *Michael Porter* dalam bukunya yang berjudul "*Competitive Advantage: Creating and Sustaining Superior Performance*" pada tahun 1985. Porter menggambarkan rantai pasok sebagai serangkaian kegiatan yang melibatkan perusahaan, mulai dari pemasok hingga konsumen, untuk menciptakan dan memberikan nilai kepada pelanggan. Dengan demikian, pemahaman mendalam terhadap rantai pasok menjadi landasan penting untuk mencapai dan mempertahankan keunggulan kompetitif. Menurut Christopher (2005), Rantai pasok adalah serangkaian kumpulan dari perusahaan yang saling tergantung secara berurutan dan bekerja sama dalam pengendalian, pengelolaan dan perbaikan arus barang dan informasi dari sisi penyalur sampai ke sisi pengguna akhir. Dalam konteks ini, ketergantungan antar perusahaan mencakup kolaborasi erat dengan pemasok, produsen, distributor, dan pengecer. Koordinasi yang efektif dalam pengendalian, pengelolaan, dan perbaikan arus barang dan informasi membantu mengoptimalkan efisiensi operasional serta meminimalkan risiko seperti keterlambatan pengiriman dan persediaan berlebih. Rantai pasok yang terintegrasi dengan baik memberikan keunggulan kompetitif melalui responsibilitas yang cepat terhadap perubahan pasar dan kebutuhan pelanggan.

Sedangkan, Menurut Vorst, (2004) dalam Adenekan et al.,(2024) menjelaskan definisi Supply chain yaitu : "Supply chain is a network of physical and related activities involved in the movement of materials and information within or across company boundaries." Pada pengertian tersebut menjelaskan bahwa rantai pasok adalah jaringan fisik dan aktivitas terkait yang memfasilitasi aliran bahan dan informasi di dalam atau

melintasi batas perusahaan. Ini mencakup transportasi, pergudangan, produksi, dan pertukaran informasi kritis untuk meningkatkan efisiensi operasional dan responsibilitas terhadap kebutuhan pelanggan. Artinya, rantai pasok melibatkan serangkaian aktivitas dan komponen fisik yang berperan dalam pergerakan bahan dan informasi. Komponen-komponen tersebut mencakup transportasi untuk memindahkan barang, pergudangan untuk menyimpan dan mengelola stok, serta proses produksi sebagai bagian dari manufaktur. Poin yang tak kalah penting adalah pertukaran informasi, yang mencakup semua data penting terkait persediaan, produksi, dan distribusi. Mekanisme berbagi informasi ini dapat menghasilkan tingkat layanan yang lebih tinggi secara keseluruhan dan biaya keseluruhan yang lebih rendah untuk rantai pasokan (Kaipia et al., 2002 dalam Mohamadi et al., 2024). Keseluruhan sistem ini dirancang untuk meningkatkan efisiensi operasional dan responsibilitas terhadap kebutuhan pelanggan. Pendekatan ini mencerminkan integrasi yang holistik dalam manajemen rantai pasok untuk mencapai kinerja optimal.

Supply Chain sebuah produk berfungsi sebagai alur kompleks yang membawa produk dari tahap awal produksi hingga akhirnya tiba di tangan konsumen. Jantung dari suksesnya suatu bisnis sering kali terletak pada efisiensi dan ketepatan dalam rantai pasoknya. Rantai pasok suatu produk melibatkan serangkaian pihak yang bekerja bersama secara sinergis untuk memastikan produk sampai ke tangan konsumen dengan kesempurnaan. Pertama-tama, supplier bertindak sebagai penyedia bahan baku yang sangat krusial, menyuplai pabrik dengan elemen-elemen yang diperlukan untuk memulai proses produksi. Pabrik berperan sebagai sentral produksi yang tidak hanya memproses bahan baku, melainkan juga merakit dan membentuknya menjadi produk jadi. Distributor menjadi tulang punggung dalam mengelola transportasi dan distribusi produk ke titik-titik penjualan, membentuk jembatan antara produsen dan pengecer. Pada tingkat pengecer, produk akhir dijajakan dan diarahkan kepada konsumen akhir. Tak kalah penting, perusahaan logistik menyusun strategi pengiriman yang efisien, memastikan produk melewati setiap fase dengan lancar. Keseluruhan proses ini mencerminkan kerja sama erat antara berbagai pihak yang saling terkait, menciptakan suatu jaringan pasok

yang kokoh dan dapat diandalkan. Struktur dari rangkaian proses yang terdapat dalam supply chain tersebut saling terkait. Seperti yang tersaji pada gambar II.1 yang mengilustrasikan keterkaitan antara berbagai pihak hingga mencapai konsumen. Merujuk pada Levi et al. (2008) dalam Levi Ben et al., 2019 rantai pasok terdiri dari beberapa elemen, antara lain supplier, pusat manufaktur, gudang, pusat distribusi, sistem transportasi, retail outlet, dan konsumen. supplier, dan mengembangkan supplier partnership atau strategic alliance. Berikut ini skema yang menggambarkan mengenai konsep rantai pasok.



Gambar II.1 Konsep Rantai Pasok (Supply Chain)

Sumber: R.V. Martono, 2019

Pada skema tersebut, rantai pasok dimulai dengan peran krusial pemasok, yang menyediakan bahan baku atau produk kepada perusahaan tahap ini disebut Input. Tahap selanjutnya adalah proses transformasi, di mana bahan baku diubah menjadi produk jadi melalui berbagai aktivitas produksi. Hasil dari proses ini, yang disebut output, kemudian siap didistribusikan ke pasar atau konsumen akhir. Setelah konsumen menggunakan produk, tahap reverse logistics memainkan peran penting dalam mengelola pengembalian produk, baik untuk tujuan retur maupun daur ulang. Perusahaan berada di pusat rantai pasok, mengelola semua aspek produksi, distribusi, dan koordinasi untuk memastikan responsivitas dan efisiensi. Akhirnya, konsumen menjadi fokus utama dalam rantai pasok, sebagai penerima akhir produk atau layanan. Siklus ini menciptakan suatu sistem yang kompleks namun terintegrasi,

memastikan aliran yang efektif dari pemasok hingga konsumen, dengan potensi untuk kembali ke dalam rantai melalui proses reverse logistics jika diperlukan.

Dalam konteks skema tersebut, eksplorasi keterkaitan antar komponen konsep rantai pasok menjadi kunci untuk mengidentifikasi potensi kesalahan. Manajemen rantai pasok yang optimal diperlukan untuk mengatasi temuan tersebut dan memastikan kelancaran operasional perusahaan. Kompleksitas rantai pasokan modern, ditambah dengan keharusan terus-menerus untuk meningkatkan pendapatan, mengurangi biaya, dan meningkatkan kepuasan pelanggan, dengan latar belakang ekonomi global dan persaingan yang ketat di antara perusahaan, menggaris bawahi perlunya praktik manajemen yang lebih baik di mana strategi yang ketat dapat memainkan peran penting (Sarimveis et al., 2008, Hugos, 2018). Masalah kompleks ini melibatkan pengambilan keputusan mengenai pemilihan pemasok, fasilitas produksi, moda transportasi, tingkat persediaan, dan saluran distribusi, dan lain-lain (Zhang, 2006, Chandra dan Tumanyan, 2007 dalam Rezapour 2017H). Berbagai karya komprehensif dan analisis mendalam menyatakan bahwa peningkatan dan pengoptimalan manajemen secara dinamis secara signifikan meningkatkan daya saing dan profitabilitas sistem rantai pasokan (Schwartz dan Rivera, 2010).

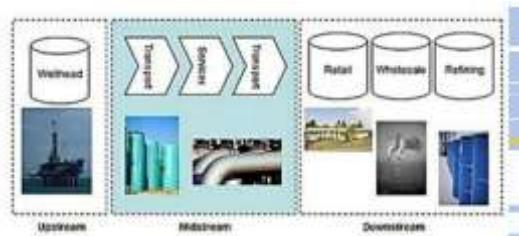
Diperkenalkan pada tahun 1980-1990-an, Manajemen Rantai Pasok (Supply Chain Management/SCM), seperti yang didefinisikan oleh Ivanov dkk. (2012) dan Bozarth dkk. (2008), mengacu pada manajemen aliran jasa dan barang. Supply chain Management juga mencakup serangkaian strategi yang digunakan untuk mengintegrasikan secara efisien semua pelaku dalam rantai pasok, termasuk pemasok, produsen, distributor, dan pengecer (Aghezzaf et al., 2011, Chopra dan Meindl, 2003). Pada tahun 1990-an isu manajemen rantai pasok telah menjadi agenda para manajemen senior sebagai kebijakan strategis perusahaan (Muhammad Asir, 2021). Manajemen rantai pasok, sebagaimana diungkapkan oleh Samal (2019) adalah upaya mengintegrasikan pengelolaan pasokan dan permintaan di dalam dan antar perusahaan, dan bertanggung jawab memadukan fungsi-fungsi dan proses-proses di dalam dan antar perusahaan dalam suatu model perusahaan yang unggul. Dengan fokus pada keseluruhan aliran nilai, manajemen rantai pasok berusaha menciptakan suatu lingkungan di mana koordinasi yang efektif dan efisiensi operasional dapat meningkatkan daya saing dan

responsibilitas terhadap pasar. Fu (2012) mendefinisikan manajemen rantai pasok sebagai pendekatan integratif dalam menangani masalah perencanaan dan pengawasan aliran material dari pemasok sampai ke pengguna. Dalam konteks ini, pendekatan tersebut mencakup pengelolaan persediaan, logistik, produksi, dan distribusi secara holistik. Tujuannya bukan hanya untuk mengoptimalkan aliran material, tetapi juga untuk meningkatkan kolaborasi antar mitra bisnis, meningkatkan efisiensi operasional, dan merespons perubahan pasar dengan lebih cepat.

II.1.1 Komponen Utama Supply Chain

Dalam Supply Chain Management terdapat tiga komponen utama yang mendukung berjalannya suatu proses bisnis yaitu Upstream Supply Chain Management Merupakan keseluruhan kegiatan perusahaan manufaktur dengan pendistribusianya atau hubungan antara manufaktur, hubungan distributor dapat diperluas menjadi kepada beberapa tingkatan, semua jalur dari asal bahan baku/material. Kegiatan utama dalam upstream supply chain adalah pengadaan produk.

Supply Chain Management Merupakan keseluruhan proses pengiriman produk ke gudang yang kemudian akan digunakan untuk proses bisnis masukan bahan baku dari distributor ke dalam hasil keluaran perusahaan tersebut. Kegiatan utama manajemen produksi, pabrikasi, pengendalian persediaan. Downstream supply chain segment Merupakan keseluruhan kegiatan yang melibatkan pengiriman kepada konsumen akhir. Kegiatan utama : Distribusi, gudang, transportasi dan layanan penjualan. Berikut contoh sekema yang gambar dari ketiga komponen tersebut.



Gambar II.2 Komponen Utama Supply Chain

Sumber: Hertina et al., 2023

Dari gambar tersebut dapat disimpulkan, rantai pasokan terdiri dari tiga komponen utama, yaitu hulu, tengah, dan hilir. Pada bagian hulu, pemasok memainkan peran kunci dengan menyediakan bahan baku atau komponen yang diperlukan untuk produksi. Proses ini melibatkan penerimaan, pemeriksaan, dan pengelolaan persediaan dari pemasok, sambil menjaga koordinasi untuk memastikan ketersediaan dan kualitas bahan. Di bagian tengah, produsen bertanggung jawab untuk mentransformasikan bahan baku menjadi produk jadi, sedangkan distributor dan logistik mengelola distribusi produk antara produsen dan pengecer. Proses ini melibatkan produksi, manajemen kualitas, distribusi, dan pengelolaan persediaan. Pada akhirnya, di bagian hilir, pengecer menjual produk langsung kepada pelanggan akhir, yang merupakan konsumen atau organisasi yang membeli dan menggunakan produk. Proses hilir melibatkan penjualan, interaksi dengan pelanggan, layanan purna jual, serta pemantauan tren pasar dan umpan balik pelanggan untuk perbaikan dan inovasi produk. Keseluruhan rantai pasokan ini menciptakan ekosistem yang berfungsi secara sinergis untuk memenuhi kebutuhan pasar dan pelanggan dengan efisien.

II.1.2 Tujuan Supplay Chain

Tujuan dari manajemen rantai pasok adalah untuk meminimalkan total biaya keseluruhan namun tetap memaksimalkan dampak positif secara sosial dan lingkungan. Biaya-biaya tersebut meliputi biaya langsung seperti biaya pengadaan, transportasi, dan biaya penyimpanan persediaan, serta biaya tidak langsung seperti biaya emisi karbon, pengelolaan limbah, dan tanggung jawab sosial (Costantino et al., 2012, Angerhofer dan Angelides, 2000). Manajemen rantai pasok bertujuan untuk dapat membantu perusahaan dalam merumuskan strategi perencanaan yang tepat. Tujuan tersebut menjadi landasan untuk mengidentifikasi langkah-langkah yang perlu diambil dalam mengelola rantai pasok secara efektif dan efisiensi. Menurut Dede hertina dkk, (2023) dalam bukunya yang berjudul *Manajemen Rantai Pasok*, hal 4 menjelaskan beberapa tujuan dari manajemen rantai pasok sebagai berikut:

Mencapai keunggulan kompetitif. Manajemen rantai pasok bertujuan untuk memberikan keunggulan kompetitif kepada Perusahaan melalui integrasi yang efektif, kolaborasi yang erat, dan pengelolaan yang optimal dalam arus barang dan informasi. Hal ini

dapat membantu Perusahaan membedakan dirinya dari pesaing dan mencapai posisi yang lebih kuat di pasar. Meningkatkan efisiensi. Dengan mengoptimalkan aliran barang, mengurangi waktu siklus, menghindari stok berlebihan dan mengelola biaya dengan baik, perusahaan dapat mencapai penggunaan sumber daya yang lebih efisien dan mengurangi pemborosan. Memenuhi kebutuhan pelanggan. Melalui koordinasi yang baik antara berbagai tahapan dalam rantai pasok, Perusahaan dapat memberikan produk atau jasa tepat waktu, dengan kualitas yang diharapkan, dan dengan biaya yang kompetitif. Hal ini dapat meningkatkan kepuasan pelanggan dan membangun hubungan jangka Panjang yang kuat. Mengoptimalkan keuntungan.

Manajemen rantai pasok berusaha untuk mengoptimalkan keuntungan Perusahaan dengan cara mengelola biaya, meningkatkan efisiensi, dan menciptakan nilai tambah dalam rantai pasok. Dengan mengoptimalkan proses produksi-produk jasa secara keseluruhan, perusahaan dapat meningkatkan profitabilitasnya. Mengelola risiko. Tujuan penting dalam manajemen rantai pasok adalah mengelola risiko yang terkait dengan rantai pasok. Risiko seperti ketidakstabilan pasokan, fluktuasi harga, gangguan operasional dan perubahan kebijakan dapat berdampak negatif pada kinerja Perusahaan. Oleh karena itu manajemen rantai pasok bertujuan untuk mengidentifikasi risiko, mengambil langkah-langkah mitigasi, dan menjaga keberlanjutan operasional dalam menghadapi ketidakpastian.

II.1.3 Manfaat Supplay Chain

Salah satu manfaat manajemen rantai pasokan adalah untuk meminimalkan total biaya keseluruhan namun tetap memaksimalkan dampak positif secara sosial dan lingkungan. Biaya-biaya tersebut meliputi biaya langsung seperti biaya pengadaan, transportasi, dan biaya penyimpanan persediaan, serta biaya tidak langsung seperti biaya emisi karbon, pengelolaan limbah, dan tanggung jawab sosial (Costantino et al., 2012, Angerhofer dan Angelides, 2000). Manfaat lainnya, adalah mengefesienkan operasional dengan memastikan aliran barang dan informasi berjalan secara efisien, mengurangi pemborosan dan biaya operasional. Manajemen rantai pasok yang baik juga dapat meningkatkan Kualitas Layanan dengan mengelola persediaan dengan baik dan mengkoordinasikan proses operasional, serta dapat memberikan layanan pelanggan yang lebih baik dan tepat waktu.

II.2 Manajemen Persediaan

Persediaan menjadi pondasi utama dalam membangun ketangguhan operasional suatu perusahaan. Keefisienan persediaan tidak hanya mempengaruhi ketersediaan produk, tetapi juga berdampak pada kinerja keseluruhan rantai pasokan. Persediaan, sebagaimana yang dijelaskan oleh Handoko (2015), persediaan (inventory) adalah suatu istilah umum yang menunjukan segala sesuatu atau sumber daya-sumber daya organisasi yang disimpan dalam antisipasinya terhadap pemenuhan permintaan. Hal ini tidak hanya sebagai tanggapan terhadap permintaan pasar, tetapi juga sebagai strategi antisipatif untuk mengatasi fluktuasi permintaan dan meminimalkan risiko kekurangan stok. Konsep dasar persediaan melibatkan pengelolaan barang atau sumber daya yang disimpan oleh suatu entitas dengan tujuan memenuhi permintaan atau kebutuhan di masa depan. Hal ini merupakan proses perencanaan, pengadaan, penyimpanan, dan pengendalian stok untuk menjaga keseimbangan optimal antara ketersediaan dan biaya. Konsep ini juga melibatkan aspek manajemen risiko, di mana organisasi berusaha menghindari kelebihan atau kekurangan persediaan yang dapat berdampak pada kinerja operasional dan keuangan. Keseluruhan, konsep dasar persediaan mencakup strategi untuk mengoptimalkan pengelolaan stok agar sesuai dengan tuntutan pasar dan keberlanjutan operasional perusahaan.

II.2.1 Inventori Control

Inventory Control merupakan sistem yang meningtegrasikan frekuensi, waktu dan ukuran pesanan menjadi satu kesatuan. Masing-masing faktor tersebut menjadi desain utama dari inventory control. Faktor frekuensi menjelaskan bagaimana sebuah perusahaan melakukan review secara terus menerus atau pada waktu tertentu saja terhadap stok inventori. Waktu merupakan salah satu faktor masalah dalam inventori, yaitu bagaimana perusahaan dapat menyediakan persediaan tepat pada saat produk habis. Faktor terakhir, ukuran pesanan tergantung pada beberapa faktor lainnya dalam pemesanan yaitu ordering cost, holding cost, shortage cost dan lainnya (Shenoy & Rosas, 2018).

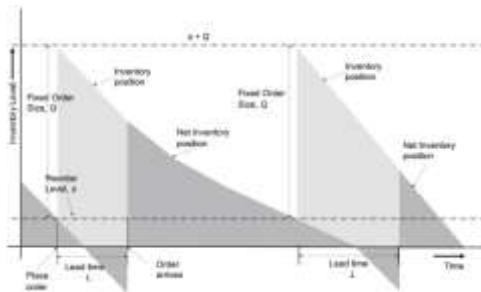
Biaya penyimpanan akan berakibat pada total biaya penyimpanan yang mencakup biaya pergudangan yang diantaranya adalah biaya sewa, perlengkapan, gaji karyawan. Pengelolaan inventory merupakan bentuk kelanjutan dari proses perecanaan, pengaturan, dan

pengawasan inventory yang bertujuan meminimalkan biaya investasi di dalam inventory namun dengan tetap menyeimbangkan antara permintaan dan penawaran (Raphella, Nathan, & G, 2014).

Berikut adalah hal-hal yang harus dimiliki Inventory yang efektif (Kumari & Jagadeeswaran, 2018): Memelihara ketersediaan permintaan stock , Mengantisipasi perubahan harga, Mengendalikan investasi dan membuatnya tetap berada pada tingkat yang optimum, Memfasilitasi pembukuan biaya secara jelas, Menghindari stock yang berlebihan ataupun kekurangan

II.2.1.1 Continous Review Fixed Order Quantity (s,Q)

Dalam perusahaan yang menggunakan sistem tinjauan berkelanjutan, posisi persediaan suatu barang dipantau secara terus menerus dan diketahui setiap saat. Posisi persediaan suatu barang didefinisikan sebagai jumlah barang yang dimiliki saat ini dalam persediaan ditambah jumlah barang yang dipesan. Saat permintaan muncul, barang akan ditarik dari persediaan. Secara bersamaan, posisi persediaan diperbarui. Proses ini terus berlanjut hingga tingkat persediaan mencapai tingkat yang telah ditentukan, s , yang disebut sebagai titik pemesanan ulang. Pada titik ini, pesanan pengisian ulang baru dengan ukuran Q ditempatkan, yang diisi setelah waktu L , yang disebut sebagai waktu tunggu. Penerimaan pesanan meningkatkan posisi persediaan. Proses sistem titik-pesanan Sistem ini juga dikenal sebagai sistem (s, Q) dan sistem duPerusahaan yang menerapkan sistem tinjauan berkelanjutan harus memutuskan hal-hal berikut ini sehubungan dengan pengelolaan persediaan mereka: Ukuranpesanan tetap yang optimal, Q . Tingkatpemesanan ulang, s .

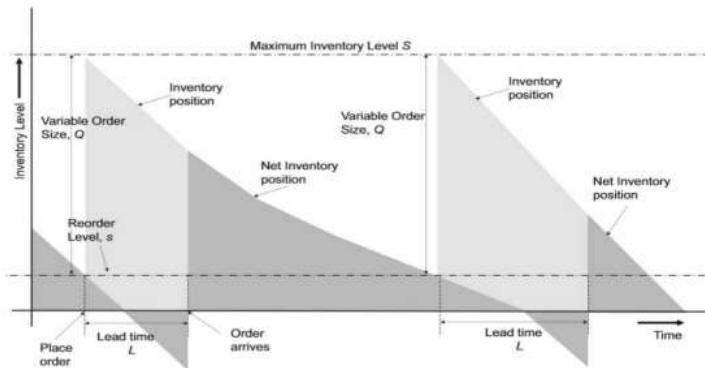


Gambar II.3 Grafik Continous Review, Fixed Order Quantity (s, Q)

Sumber: Shenoy & Rosas, 2018

II.2.1.2 Continous Review Pixed Order Up-To-Level (s,S)

Sistem pengendalian persediaan berbasis tinjauan berkelanjutan ini mirip dengan sistem titik pemesanan dan tingkat pemesanan dengan satu perbedaan - jumlah pesanan dalam sistem ini bervariasi. Sistem ini juga disebut sebagai sistem (s, S) atau sistem Min-Max. Metode Min-Max ini dilakukan dengan mengendalikan jumlah minimum dan maksimum persediaan dengan mengatur rencana pemesanan persediaan (plan order) agar tidak terjadi kekurangan (stockout) atau kelebihan persediaan (overstock) (Vergianti, 2018). Dalam bentuk aslinya, Min-Max dianggap metode yang cukup statis pengendalian persediaannya di mana nilai-nilai min-max jarang berubah, mungkin beberapa kali per tahun. Dari perspektif pemesanan Min / Max, ketika pemesanan ulang adalah untuk dilakukan, baik nilai-nilai min dan max harus diperbarui dengan nilai reorder point yang dihasilkan dari perhitungan perkiraan kuantil (Rizky, Sudarso, & Sadriatwati, 2016). Ketika tingkat persediaan mencapai tingkat yang telah ditentukan, s , yang disebut sebagai titik pemesanan ulang, pesanan pengisianulang dengan ukuran Q ditempatkan sehingga menaikkan tingkat persediaan ke tingkat maksimum, S . Perlu diketahui bahwa ukuran pesanan Q dalam sistem ini adalah variabel. Variabel keputusan dalam sistem ini Tingkat pemesanan ulang, s . Tingkat persediaan maksimum , S .

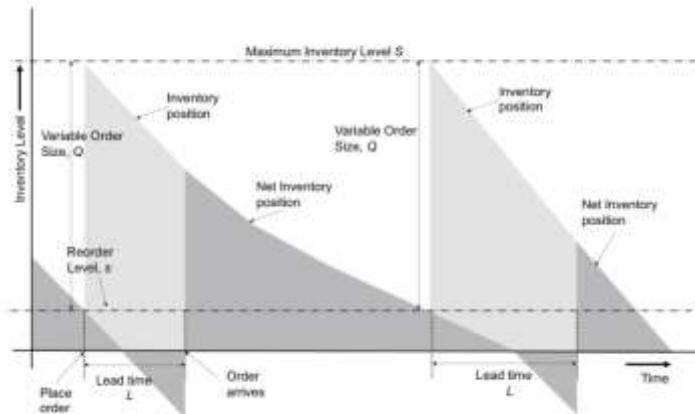


Gambar II.4 Grafik Continous Review, Order-Up-to-Level (s, S)

Sumber: Shenoy & Rosas, 2018

II.2.1.3 Periodic Review Order Up-To-Level (T,S)

Dalam sistem kontrol persediaan berbasis tinjauan berkala ini, yang juga disebut sistem (T, S) , tingkat persediaan barang ditinjau pada titik-titik waktu tertentu yang telah ditentukan. Jika tinjauan status pertama terjadi pada waktu T , tinjauan kedua adalah dilakukan pada waktu $2T$, dan seterusnya. Berikut ini adalah variabel-variabel keputusan dalam organisasi yang menggunakan sistem ini: Waktu tetap yang optimal, T , diantara pesanan pengisian ulang. Tingkat persediaan maksimum, S . Ukuran pesanan variabel, Q . Ukuran pesanan dihitung pada saat peninjauan. Bergantung pada tingkat persediaan (jumlah persediaan di tangan dan dalam pesanan) pada saat tinjauan, ukuran pesanan dihitung yang akan membawa tingkat persediaan kembali ke S .



Gambar II.5 Periodic Review, Order-Up-to-Level (T, S)

Sumber: Shenoy & Rosas, 2018

II.2.2 Jenis-jenis Persediaan

Dalam buku yang berjudul Rantai Pasok Jagung oleh Abdul Azis, dkk (2021) memaparkan jenis-jenis persediaan menurut Djoko dan Harsasi diantara bahan baku, barang setengah jadi, barang komoditas dan barang proyek dimana, Bahan baku (raw materials) Mata rantai pertama ada di pabrik pembuat bahan baku, dan mata rantai terakhir ada di pabrik pembuat produk akhir (bukan di konsumen akhir). Bahan baku ini di pabrik pembuat produk akhir digabung dengan bahan penolong, dan dengan teknologi tertentu diolah menjadi bahan setengah jadi dan bahan jadi.

Barang setengah jadi (work in process product) Permulaan mata rantai ada di pabrik pembuat bahan jadi. Seperti telah dijelaskan sebelumnya, bahan setengah jadi adalah hasil dari proses bahan baku. Bahan setengah jadi dapat langsung diproses di pabrik yang sama menjadi bahan jadi, tetapi dapat juga dijual kepada konsumen sebagai komoditas. Jadi, akhir dari mata rantai akan sangat tergantung dari hal di atas, bisa pendek dan bisa panjang. Akhir mata rantai ada di konsumen akhir pengguna atau pembeli hasil produksi tersebut.

Barang komoditas (commodity) Persediaan jenis ini adalah barang yang dibeli oleh perusahaan tertentu sudah dalam bentuk barang jadi dan diperdagangkan, dalam arti dijual kembali kepada konsumen. Di perusahaan tersebut, barang ini dapat diproses lagi, misalnya diganti bungkusnya atau diperkecil kemasannya, tetapi dapat juga dijual lagi langsung dalam bentuk asli seperti saat dibeli. Mata rantai persediaan jenis ini bermula dari pabrik pembuat komoditas tersebut dan berakhir pada konsumen akhir pengguna barang tersebut. Barang komoditas kadang-kadang juga disebut resales commodities karena memang barang tersebut dibeli untuk dijual lagi dengan keuntungan tertentu.

Barang proyek Persediaan jenis ini adalah material dan suku cadang yang digunakan untuk membangun proyek tertentu, misalnya membuat pabrik baru. Mata rantai panjangnya hampir sama dengan MRO materials, jadi bermula dari pabrik pembuat barang-barang tersebut dan berakhir di perusahaan pembuat barang jadi yang dimaksud. Masing-masing dari persediaan tersebut memiliki kompleksitas dan tantangan manajemen tersendiri. Pengelolaan persediaan yang cermat memiliki peran sentral dalam menjaga kestabilan operasional dan mencegah kesenjangan antara permintaan dan ketersediaan barang atau bahan. Sebagaimana diutarakan oleh Dasgupta et al., (2019), investasi dalam persediaan diakui sebagai elemen kunci yang ikut serta dalam mengalami fluktuasi siklus bisnis. Meskipun demikian, mengelola persediaan, baik bagi perusahaan dagang maupun manufaktur, sering kali menjadi tantangan yang kompleks, sebagaimana yang diungkapkan oleh Kolias et al. (2011). Oleh karena itu, efisiensi dalam manajemen persediaan menjadi aspek vital dalam pengambilan keputusan bisnis, terutama dalam menghadapi meningkatnya risiko eksternal, seperti yang telah ditekankan oleh Schiff dan Lieber (1974) serta Zietsman dan Van Vuuren (2022). Dalam konteks ini, strategi manajemen persediaan yang bijaksana

menjadi suatu keharusan, membantu perusahaan tidak hanya menghadapi tantangan fluktuasi pasar, tetapi juga meningkatkan ketahanan terhadap risiko ekonomi yang mungkin timbul. Manajemen persediaan yang efektif telah terbukti secara signifikan mengurangi biaya (Shirley & Winston, 2004) dan meningkatkan posisi pasar perusahaan (Atnafu dan Balda, 2018, Maji et al., 2022) dan daya saing (Chan et al., 2017), yang pada akhirnya berkontribusi pada peningkatan kinerja bisnis (Nyabwanga dan Ojera, 2012, Prinotakis dan Argyropoulos, 2018).

II.2.3 Biaya Dalam Persediaan

Terdapat beberapa komponen biaya yang dapat digunakan dalam sistem persediaan yaitu biaya pembelian (*purchase cost*), biaya pemesanan (*order cost*) dan biaya simpan (*holding cost*) (Hansen , Mowen, & Guan, 2001 dalam Abdirad 2014). Biaya pemesanan (*order cost* atau *setup cost*) Biaya pemesanan adalah biaya yang dikeluarkan sehubungan dengan pemesanan bahan atau barang, sejak dari penempatan pemesanan sampai pada tersedianya barang digudang, yang mencakup biaya administrasi, biaya pemilihan *vendor*, biaya pengangkutan, biaya penerimaan dan inspeksi barang. Biaya pemesanan tidak tergantung pada jumlah barang yang di pesan, namun tergantung dari berapa kali dilakukannya pemesanan. Jika perusahaan memproduksi persediaan sendiri, dalam artian tidak melakukan pemesanan kepada pemasok, maka biaya pemesanan disebut dengan *set-up cost*, yaitu biaya yang dikeluarkan untuk persiapan mesin atau proses manufaktur

Biaya simpan (*holding cost*) Biaya simpan (*holding cost*) adalah biaya yang dikeluarkan berkaitan dengan diadakannya persediaan barang. Biaya simpan terdiri dari biaya modal, biaya gudang, biaya penyusutan, biaya kadaluarsa, biaya asuransi, dan biaya administrasi dan pemindahan, biaya kehilangan atau penyusutan barang selama penyimpanan. Biaya simpan dapat dinyatakan dalam dua bentuk yaitu sebagai persentase dari rata-rata nilai persediaan per tahun dan dalam bentuk rupiah per tahun per unit barang.

Biaya kekurangan persediaan (*shortage cost*) Biaya kekurangan persediaan (*shortage cost*) adalah biaya yang dikeluarkan akibat tidak tersedianya barang pada waktu yang diperlukan. Biaya kekurangan persediaan ini pada dasarnya bukan biaya nyata (riil), melainkan berupa biaya kehilangan kesempatan. Termasuk dalam biaya kekurangan persediaan adalah semua

biaya kesempatan yang timbul akibat terhentinya proses produksi akibat tidak adanya bahan yang diproses, biaya administrasi tambahan, biaya akibat tertundanya penerimaan keuntungan, hingga biaya kehilangan (Handoko,1999 dalam Yordan, 2020)

II.2.4 Pengelolaan Persediaan

Dalam konteks global, ketidakpastian geopolitik, perubahan regulasi, dan gangguan dalam rantai pasokan dapat menciptakan persoalan baru yang perlu dikelola secara hati-hati. Terdapat beberapa tantangan umum yang kerap muncul dalam pengelolaan persediaan. Biaya simpan yang tinggi Melalui tata kelola manajemen yang efektif, perusahaan dapat mengurangi dampak biaya penyimpanan persediaan yang signifikan. Tujuan dari manajemen persediaan adalah untuk meningkatkan layanan pelanggan, meningkatkan variasi produk, dan mengurangi biaya (Giannoccaro et al., 2003 dalam Sharma et al., 2022). Banyak Perusahaan semakin tertarik dengan tujuan yang melekat pada rantai pasokan, yaitu meminimalkan total biaya (Simchi-Levi et al., 2003 dalam Barbosa et al.,2018). Kondisi biaya penyimpanan tinggi dipicu oleh kompleksitas sistem lapisan persediaan, di mana pengelolaan persediaan pada setiap tingkat memerlukan pemahaman mendalam terhadap permintaan yang fluktuatif, waktu tunggu yang tidak terkoordinasi, dan keterlambatan informasi. Integrasi yang tidak optimal dalam rantai pasokan multi eselon dapat meningkatkan biaya simpan secara signifikan, memaksa perusahaan untuk mencari solusi yang lebih canggih dan terintegrasi dalam pengelolaan persediaan guna mengatasi tantangan kompleks ini.

Kehilangan Pendapatan akibat Kekurangan Stok penting yang diamati dan dipertimbangkan oleh Brightstar dalam proses perencanaan rantai pasokannya adalah substitusi kehabisan stok. Dalam dinamika persediaan, konsekuensi yang timbul saat pelanggan menemui kehabisan stok pilihan pertama dapat menjadi sangat signifikan. Seperti yang ditegaskan oleh Anupindi et al., (1998), sebagian pelanggan cenderung beralih ke pilihan kedua sebagai alternatif. Pada kondisi seperti ini, penting bagi sebuah perusahaan untuk memiliki manajemen persediaan yang efektif sebagai garda terdepan. Tanpa manajemen yang baik, risiko kehilangan pelanggan atau kekecewaan pelanggan menjadi lebih tinggi, menciptakan tantangan yang dapat mempengaruhi citra perusahaan dan daya

saingnya di pasar. Oleh karena itu, menerapkan strategi manajemen persediaan yang cermat bukan hanya tentang memastikan ketersediaan stok, tetapi juga memitigasi potensi dampak negatif terhadap kepuasan pelanggan. Dengan begitu, perusahaan dapat tidak hanya memenuhi permintaan pelanggan dengan lebih baik tetapi juga membangun reputasi yang kuat dalam menjaga keandalan dan kualitas layanan.

Kadaluwarsa Produk Persoalan manajemen persediaan yang lain adalah kadaluwarsa produk. Jika perusahaan tidak memiliki manajemen yang baik maka mereka tidak dapat mengatasi tantangan, seperti mengelola produk dengan siklus hidup yang lebih pendek, ekspektasi pelanggan, dan risiko lainnya (Emrouznejad et al., 2023). Keterlambatan dalam pemesanan atau produksi hal ini merupakan tantangan nyata yang sering dihadapi perusahaan. Pemesanan produk dalam jumlah yang tepat untuk menjamin ketersediaan produk dan tingkat kesegaran yang tinggi serta meminimalkan pemborosan merupakan salah satu masalah yang paling menantang bagi para peritel (Wyman, 2014).

Produk cacat/rusak Studi penelitian sebelumnya menunjukkan bahwa mengurangi tingkat kerusakan dapat meningkatkan tingkat layanan pelanggan dan daya saing bisnis (Ferguson dan Koenigsberg, 2007 dalam Cao Y et al., 2022zha). Namun, tidak semua produk menunjukkan tingkat kualitas yang optimal, terkadang adanya cacat atau kerusakan dapat mengganggu efektivitas manajemen perusahaan, merugikan reputasi perusahaan, dan mengurangi kepuasan pelanggan. Nilai atau kualitas produk yang mudah rusak akan memburuk seiring waktu setelah diproduksi dan sepenuhnya rusak (tidak bernilai) ketika masa pakainya habis (Amorim et al., 2012). Hal ini akan merugikan finansial perusahaan dan menjadi tantangan dalam pengelolaan manajemen persediaan. Manajemen yang tidak memadai juga berpotensi menyebabkan kelebihan penimbunan, artinya aset tertahan dalam persediaan sehingga membatasi arus kas dan perkembangan organisasi. Sebaliknya, menurut Moons et al., (2019), understoking menyebabkan kekurangan dan mengganggu stabilitas layanan yang diberikan perusahaan. Persoalan-persoalan tersebut adalah kenyataan signifikan yang kerap dihadapi oleh perusahaan dagang maupun manufaktur, terutama ketika melibatkan struktur multi eselon dalam operasional perusahaan.

Setelah perusahaan memahami tantangan dalam manajemen persediaan, penerapan strategi yang efektif dapat menghasilkan sejumlah manfaat yang signifikan. Menurut Nanik kustiningsih dan Ali Farhan (2022) Manfaat Manajemen Persediaan adalah menjamin kebebasan atau kelancaran kegiatan operasional internal dan eksternal sehingga permintaan pelanggan dapat terpenuhi tanpa tergantung pemasok. Manajemen persediaan merupakan aspek krusial dalam mengelola rantai pasok suatu perusahaan. Dengan merencanakan, mengendalikan, dan mengawasi persediaan, perusahaan dapat menjaga ketersediaan barang yang optimal untuk memenuhi permintaan pelanggan. Namun, semakin kompleksnya sistem distribusi dan tingkat permintaan yang bervariasi mendorong pengembangan pendekatan yang lebih canggih. Salah satu konsep yang muncul sebagai solusi adalah manajemen persediaan multi eselon. Dalam pendekatan ini, persediaan tidak hanya dipandang sebagai entitas linier, tetapi sebagai sistem yang melibatkan beberapa tingkat atau lapisan. Hal ini memungkinkan perusahaan untuk merespons perubahan permintaan secara lebih adaptif, mengoptimalkan biaya persediaan, dan meminimalkan risiko tidak tersedianya barang. Dengan mengintegrasikan konsep manajemen persediaan multi eselon, perusahaan dapat meningkatkan efisiensi operasional dan meraih keunggulan kompetitif di pasar yang terus berubah.

II.2.5 Model Persediaan

Dalam lingkungan rantai pasokan, apakah persaingan antar perusahaan atau mengejar kepentingan seluruh aliansi perusahaan, secara langsung berubah menjadi manajemen dan optimalisasi rantai pasokan, dan manajemen persediaan merupakan mata rantai penting dari manajemen rantai pasokan (Zhao 2003; Zhu et al. 2008; Liang 2005 dalam De Oliveira et al., 2018) biaya persediaan adalah salah satu bagian terpenting dari pemeran rantai pasokan (30% atau lebih) (Yu 2004; Liao 2006).

II.2.5.1 Satu Eselon

Model eselon tunggal atau satu eselon mengikuti konsep satu gudang, kelompok gudang, atau toko yang tidak saling mengirim barang. Mereka menempati satu tingkat dalam hierarki. (Robert A. Darvis, Demand-Driven Inventory Optimization and Replenishment, 2013 hal 42) Mode kontrol persediaan satu eselon adalah strategi manajemen persediaan

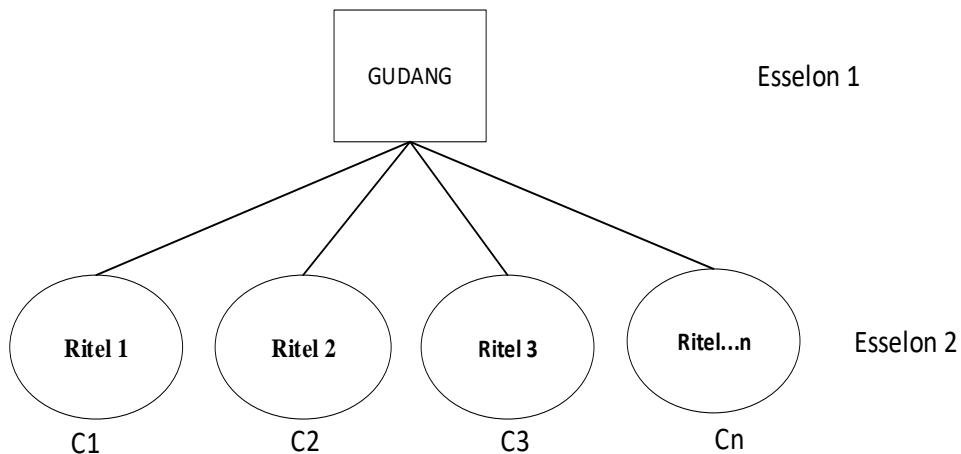
yang diterapkan pada perusahaan tertentu dalam rantai pasokan. Biasanya, strategi ini diterapkan pada rantai pasokan yang lebih pendek dan memiliki sedikit pengaruh eksternal terhadap perusahaan. Mode kontrol persediaan satu eselon dalam rantai pasokan perusahaan dapat dibagi menjadi dua jenis: mode penyimpanan pasti dan mode persediaan acak. Mode penyimpanan pasti memiliki parameter yang tidak berubah di setiap siklus pesanan atau dapat dihitung rata-ratanya. (Ershi Qi etc, International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012) ProceedingsCore Areas of Industrial Engineering, 2013, hal 1320) Mode kontrol persediaan satu eselon dalam rantai pasokan perusahaan dapat dibagi menjadi dua jenis: mode penyimpanan pasti dan mode persediaan acak.

Mode penyimpanan pasti memiliki parameter yang tidak berubah di setiap siklus pesanan atau dapat dihitung rata-ratanya. Satu lagi adalah jenis mode persediaan acak, yang beberapa parameternya merupakan variabel yang tidak pasti (Bowersox dan Closs 1996; Dong 2002; Cheung dan Lee 2002). Tipe penyimpanan yang pasti memiliki lima jenis yaitu stok seketika dan tidak ada kekurangan stok, stok seketika dan penyisihan kekurangan stok (Chun 2007), stok tidak seketika dan penyisihan kekurangan stok, harga pembelian diputuskan berdasarkan diskon besar-besaran, dan mode penyimpanan multispecies; Tipe penyimpanan acak termasuk mode penyimpanan satu fase dimana permintaan merupakan variabel acak diskrit, dan mode penyimpanan dimana permintaan merupakan variabel acak kontinu (Liu 2011b; Xian 2006). Yang paling umum adalah mode pemesanan di mana permintaan adalah fungsi yang bervariasi terhadap waktu dan tidak diperbolehkan kekurangan stok, dan itu adalah mode persediaan pesanan penjualan dari ekonomi kemerosotan tanpa kekurangan stok berdasarkan variasi linier harga sesuai dengan perubahan permintaan dan situasi di mana persediaan memiliki pengaruh terhadap penjualan (Zhu 2004; Dai 2007; Li 2009).

II.2.5.2 Dua Eselon

Model persediaan dua eselon (two-echelon inventory model) melibatkan dua tingkat dalam rantai pasokan, biasanya terdiri dari gudang pusat (depot) dan beberapa ritel. Model ini dirancang untuk mengoptimalkan pengelolaan persediaan pada kedua tingkatan tersebut

dengan tujuan meminimalkan biaya total yang mencakup biaya pemesanan, penyimpanan, dan kekurangan stok.



Gambar II.6 Sistem Dua Eselon

Berikut adalah komponen utama dalam model persediaan dua eselon Gudang Pusat yaitu untuk Menyimpan persediaan dalam jumlah besar, Memasok barang ke gudang local, Memesan barang dari pemasok eksternal, Biaya terkait: biaya pemesanan dari pemasok, biaya penyimpanan digudang pusat.

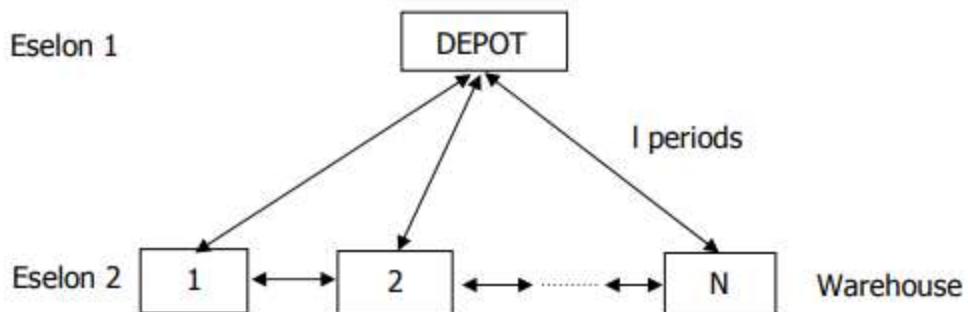
Ritel Menyimpan persediaan dalam jumlah lebih kecil dibandingkan gudang pusat, Memenuhi permintaan pelanggan akhir, Memesan barang dari gudang pusat, Biaya terkait: biaya pemesanan dari gudang pusat, biaya penyimpanan di gudang lokal, biaya kekurangan stok jika terjadi kehabisan barang. Model ini mempertimbangkan beberapa faktor penting, seperti: Tingkat permintaan Permintaan di tingkat ritel yang mempengaruhi jumlah stok yang harus disediakan. Lead time Waktu yang dibutuhkan untuk pemesanan dan pengiriman barang dari pemasok ke gudang pusat dan dari gudang pusat ke gudang lokal. Tujuan utama dari model persediaan dua eselon adalah mencapai keseimbangan antara biaya penyimpanan dan biaya pemesanan, serta meminimalkan risiko kehabisan stok dengan strategi pemesanan yang efisien pada kedua tingkat persediaan tersebut.

II.2.5.3 Multi Eselon

Konsep sistem pengendalian persediaan pada multieselon diperkenalkan dalam penelitian Clark dan Scarf yang pada prinsipnya menyatakan bahwa persediaan pada suatu

eselon adalah semua persediaan yang ada pada unit fasilitas yang bersangkutan dan semua persediaan yang ada pada semua fasilitas yang mengikutinya pada semua fasilitas yang mengikutinya pada eselon berikutnya. Setiap fasilitas di masing-masing eselon mempunyai karakteristik yang berbeda-beda dipengaruhi oleh tipe permintaan yang terjadi di fasilitas tersebut. Karena permintaannya yang berbeda-beda maka kemungkinan terjadinya kekurangan atau kelebihan persediaan sangat besar. Tidak menutup kemungkinan dalam satu eselon terdapat fasilitas yang kekurangan persediaan tetapi difasilitas lainnya mengalami kelebihan persediaan. Interaksi antar fasilitas yang kekurangan dan kelebihan persediaan tersebut dapat terjadi dengan mengadakan pengiriman persediaan dari yang berlebih kepada fasilitas yang kekurangan, tanpa meminta pengiriman dari fasilitas di eselon di atasnya.

Interaksi tersebut dapat berupa permintaan pada fasilitas yang lain bila di fasilitas yang bersangkutan terjadi kekurangan atau sebaliknya dapat mengirimkan produk ke fasilitas yang kekurangan persediaan. Masing-masing fasilitas sendiri mempunyai permintaan yang dapat sama atau bervariasi dalam setiap periode waktu. Permasalahan persediaan multi eselon seperti itu digambarkan sebagai berikut:



Gambar II.7 System Persediaan Multieselon

Sumber Gambar (Fifi Herni Mustofa, 2013)

Pada eselon 1 terdapat satu depot, fungsi depot adalah memasok produk untuk eselon 2. Pada eselon 2 terdapat n buah warehouse, warehouse merupakan unit fasilitas pada sub sistem distribusi yang berfungsi melayani konsumen dan mengelola pasokan produk yang berasal

dari depot. Pada Gambar 1, lokasi pada eselon 1 disebut dengan depot dan lokasi pada eselon 2 disebut dengan fasilitas 1,2, ..., n. Pengiriman dari supplier luar diterima hanya pada depot dan permintaan eksternal serta pengembalian terjadi hanya pada fasilitas di eselon 2. Diasumsikan bahwa permintaan bersih (permintaan total dikurangi pengembalian) adalah independent dari lokasi ke lokasi dan dari periode waktu yang satu ke periode waktu yang lain. Setiap fasilitas di eselon 2 mempunyai karakteristik yang dapat sama atau berbeda dipengaruhi oleh tipe permintaan yang terjadi di fasilitas tersebut. Pada suatu fasilitas yang terjadi kekurangan persediaan dan di fasilitas yang lainnya terjadi kelebihan persediaan dan dapat terjadi interaksi. Interaksi dapat terjadi dengan mengadakan pengiriman persediaan dari yang berlebih kepada fasilitas yang kekurangan, tanpa meminta pengiriman dari eselon di atasnya. Costumer datang ke toko ritel untuk membeli barang yang mereka butuhkan. Sebagai contoh stok barang di toko ritel akan habis seiring dengan meningkatnya permintaan. Pada titik tertentu, pemilik toko ritel melakukan pemesanan pengisian ulang ke gudang, yang ketika diterima, akan meningkatkan jumlah barang yang ada di toko ritel. Demikian pula, ketika tingkat persediaan di gudang menipis, manajer gudang perlu melakukan pemesanan kepada produsen untuk memasok stok baru.

Setiap fasilitas dapat berinteraksi dengan fasilitas di eselon atasnya atau dalam eselon yang sama dalam mendistribusikan produknya. Interaksi tersebut dapat berupa permintaan pada fasilitas yang lain bila di fasilitas yang bersangkutan terjadi kekurangan atau sebaliknya dapat mengirimkan produk ke fasilitas yang kekurangan persediaan. Masing-masing fasilitas sendiri mempunyai permintaan yang dapat sama atau bervariasi dalam setiap periode waktunya. Fasilitas dengan eselon yang di bawah akan meminta distribusi produk pada fasilitas di eselon atasnya.

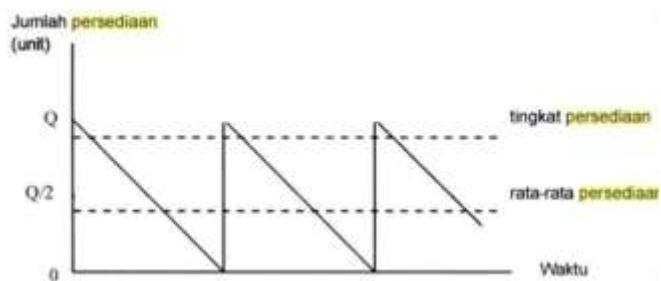
II.3 Economic Order Quantity (EOQ)

EOQ (Economic Order Quantity) merupakan sebuah metode atau model dalam manajemen persediaan. Metode ini digunakan untuk menentukan jumlah optimal barang yang harus dipesan dalam satu kali pesanan agar biaya total persediaan dapat diminimalkan. EOQ sangat penting dalam pengelolaan persediaan karena membantu perusahaan untuk mengoptimalkan tingkat persediaan mereka dengan mempertimbangkan biaya pemesanan

dan biaya penyimpanan. Sebagai respons terhadap tantangan kompleks dalam menjaga keseimbangan antara biaya simpan dan ketidaktersediaan barang, perusahaan sering mengadopsi model perhitungan persediaan yang terstruktur. Model-model ini dirancang untuk memberikan panduan dalam menentukan jumlah optimal persediaan yang harus dijaga, dengan mempertimbangkan faktor-faktor seperti biaya simpan, biaya pemesanan, dan tingkat permintaan. Salah satu model yang terkenal dan banyak digunakan dalam hal ini adalah Economic Order Quantity (EOQ). EOQ memberikan pendekatan matematis yang membantu perusahaan dalam menemukan keseimbangan yang optimal antara biaya simpan dan biaya pemesanan, dengan tujuan meminimalkan total biaya persediaan. Dalam konteks ini, pembahasan selanjutnya akan mengeksplorasi konsep EOQ lebih lanjut untuk memahami bagaimana model ini dapat menjadi alat efektif dalam mengoptimalkan manajemen persediaan perusahaan.

Seperti yang diketahui, model persediaan pertama kali diperkenalkan oleh Harris (1913). Dia mengembangkan model persediaan economic order quantity (EOQ) yang terkenal yang menganggap bahwa permintaan konstan dan diketahui. Setelah itu, banyak penelitian yang telah dilakukan di bidang persediaan. Banyak peneliti yang mengembangkan model persediaan Harris (1913) dengan mempertimbangkan asumsi-asumsi yang lebih meyakinkan dengan tujuan untuk memodelkan lebih dekat situasi kehidupan nyata. Menurut Cárdenas-Barrón et al., (2014), Harris dianggap dianggap sebagai peletak dasar teori persediaan terutama dalam metode perhitungan EOQ. Menurut Handoko (1994) dalam Ramadhani dan Ardiansyah, (2022).

Grafik persediaan dalam model ini berbentuk gigi gergaji, Karena permintaan dianggap konstan, persediaan berkurang dalam jumlah yang sama (linear) dari waktu ke waktu.



Gambar II.8 Grafik Persediaan dalam Model EOQ

Sumber: Eddy Herjanto, 2008

Dalam grafik persediaan EOQ, sumbu x umumnya mewakili jumlah pesanan atau tingkat produksi, sedangkan sumbu y menggambarkan biaya total persediaan. Pada grafik tersebut, terdapat dua kurva yang saling berhubungan. Yaitu, Kurva Biaya Pemesanan (*Ordering Cost Curve*). Biasanya merupakan kurva turun yang mencerminkan biaya pemesanan yang semakin rendah seiring dengan peningkatan jumlah pesanan. Ini mencerminkan *trade-off* antara biaya pemesanan yang lebih rendah untuk pesanan besar dan biaya penyimpanan yang lebih tinggi, dan Kurva Biaya Penyimpanan (*Holding Cost Curve*). Umumnya merupakan kurva naik yang mencerminkan biaya penyimpanan yang meningkat seiring dengan peningkatan jumlah persediaan. Hal ini menggambarkan *trade-off* antara biaya penyimpanan yang lebih rendah untuk persediaan kecil dan biaya pemesanan yang lebih tinggi.

Pada grafik tersebut terdapat titik Kesetimbangan (*Balance Point*). Titik yang menggambarkan dua kurva tersebut bertemu dan biaya total persediaan mencapai nilai minimum. Inilah jumlah pesanan optimal atau EOQ, di mana biaya pemesanan dan biaya penyimpanan saling seimbang. Nanik Kustiningih & Ali Farhan dalam bukunya yang berjudul Manajemen Keuanga: Dasar-Dasar Pengelolaan Keuangan, memaparkan mengenai EOQ. Dimana, dua keputusan utama yang berkaitan dengan pengendalian persediaan tersebut adalah berapa banyak sumber daya yang harus dipesan (dibeli atau diproduksi) dan kapan waktunya untuk melakukan pemesanan (pembelian atau produksi) untuk mengurangi biaya-biaya persediaan tersebut.

Terdapat dua biaya yang harus dipertimbangkan pada saat melakukan keputusan terhadap "jumlah yang harus dipesan", yaitu biaya penyimpanan (*carrying cost/holding cost*) dan biaya pemesanan (*ordering cost/acquisition cost*). Jika jumlah kuantitas yang dipesan meningkat maka biaya penyimpanan akan meningkat sedangkan biaya pemesanan akan menurun. Oleh karena itu, diperlukan suatu perhitungan yang berfungsi untuk menyeimbangkan kedua biaya tersebut. Salah satu metode yang paling sering digunakan dalam menentukan jumlah kuantitas pesanan pada Manajemen Persediaan adalah Metode

Economic Order Quantiy (EOQ) atau dalam bahasa Indonesia disebut juga dengan Jumlah Pemesanan Ekonomis.

II.3.1 Persediaan Dalam EOQ

Persediaan (inventory) merupakan stok barang yang disimpan oleh suatu perusahaan untuk memenuhi permintaan pelanggan. Umumnya setiap jenis perusahaan memiliki berbagai bentuk persediaan. Berdasarkan jenis dan posisi barang dalam urutan penggerjaan produk, persediaan dapat dibagi menjadi lima macam (Taylor,2005 dalam Aditya,2015) Persediaan bahan baku, yaitu persediaan dari barang-barang berwujud yang digunakan dalam proses produksi. Persediaan komponen produk atau parts yang dibeli, yaitu persediaan barangbarang yang terdiri dari parts yang diterima dari perusahaan lain, yang dapat secara langsung di-assembling dengan parts lain tanpa melalui proses produksi sebelumnya

Persediaan bahan-bahan pembantu atau barang-barang perlengkapan, yaitu persediaan barang-barang atau bahan baku yang diperlukan dalam proses produksi untuk membantu kelancaran produksi. Persediaan barang setengah jadi, yaitu persediaan barang-barang yang keluar dari tiap-tiap bagian dalam suatu pabrik atau barang yang masih perlu diproses kembali untuk kemudian menjadi barang jadi. Persediaan barang jadi, yaitu persediaan barang-barang yang telah selesai diproses atau diolah dalam pabrik dan siap untuk dijual kepada para konsumen. Sedangkan menurut Ristono (2013) dalam Yul, (2019) persediaan dapat diartikan sebagai barang-barang yang disimpan untuk digunakan atau dijual pada masa atau periode yang akan datang. Persediaan terdiri dari persediaan bahan baku, persediaan bahan setengah jadi dan persediaan barang jadi. Persediaan bahan baku dan bahan setengah jadi disimpan sebelum digunakan atau dimasukkan ke dalam proses produksi, sedangkan persediaan barang jadi atau barang dagangan disimpan sebelum dijual atau dipasarkan. Dengan demikian setiap perusahaan yang melakukan kegiatan usaha umumnya memiliki persediaan. Menurut Nasution dan Prasetyawan (2008) dalam Hidayat dkk, (2020) jika dilihat dari jenisnya, ada 4 macam persediaan secara umum yaitu: Bahan baku (raw materials) adalah barang-barang yang dibeli dari pemasok (supplier) dan akan digunakan atau diolah menjadi produk jadi yang akan dihasilkan oleh perusahaan.

Bahan setengah jadi (work in process) adalah bahan baku yang sudah diolah atau dirakit menjadi komponen namun masih membutuhkan langkah-langkah lanjutan agar menjadi produk jadi. Barang jadi (finished goods) adalah barang jadi yang telah selesai diproses, siap untuk disimpan di gudang barang jadi, dijual, atau didistribusikan ke lokasi-lokasi pemasaran. Bahan-bahan pembantu (supplies) adalah barang-barang yang dibutuhkan untuk menunjang produksi, namun tidak akan menjadi bagian pada produk akhir yang dihasilkan perusahaan .

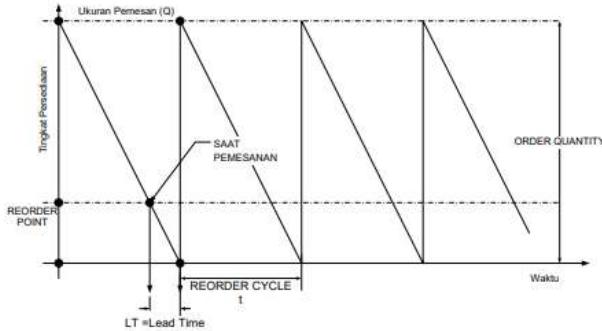
II.3.1.1 Model Persediaan EOQ

Secara umum bahwa model persediaan terbagi menjadi dua bagian yaitu Model Deterministik (Taha,2003 dalam Garcia et al.,2015), Model Deterministik yaitu model yang menganggap bahwa semua parameter telah diketahui dengan pasti. Model ini dibagi lagi menjadi dua yaitu deterministic static dan deterministic dynamic. Contoh model yang dipakai adalah model Economic Order Quantity (EOQ) dan pemesanan barang multi-item dengan Metode Lagrange Multiplier.

Model Stukistik (Probabilistik), yaitu model yang menganggap bahwa semua parameter mempunyai nilai-nilai yang tidak pasti dan satu atau lebih parameter tersebut merupakan variabel-variabel acak. Contoh dari model ini antara lain adalah model pengendalian persediaan Sistem P dan Sistem Q. Model ini dibagi lagi menjadi dua yaitu probabilistic static dan probabilistic dynamic.

II.3.1.2 Model Persediaan EOQ Deterministik

Model EOQ merupakan model persediaan yang sederhana yang bertujuan untuk menentukan ukuran pemesanan yang ekonomis dan dapat meminimumkan biaya total persediaan. Model ini dapat diterapkan apabila terdapat asumsi-asumsi berikut (Render dan Heizer,2001 dalam Haming,2022) yaitu kebutuhan permintaan adalah tetap dan diketahui, lead time (waktu tunggu) adalah tetap, harga beli per unit tetap, biaya simpan dan biaya setiap kali pesan tetap, diskon kuantitas tidak diperkenankan, tidak terjadi kekurangan persediaan atau back order, Model ini dapat digambarkan sebagai berikut:



Gambar II.9 Grafik Siklus Persediaan

Sumber: Dahdah (2009)

Dalam kaitannya dengan model persediaan tersebut, biaya-biaya yang relevan dengan model ini adalah biaya pemesanan dan biaya penyimpanan. Jika D adalah jumlah permintaan, dalam kasus ini per minggu, y adalah kuantitas pesanan, dan K adalah biaya setiap kali pesan, maka biaya pemesanan per minggu dirumuskan (Taha, 2003)

$$\text{Biaya pemesanan per periode} = K \frac{D}{y}$$

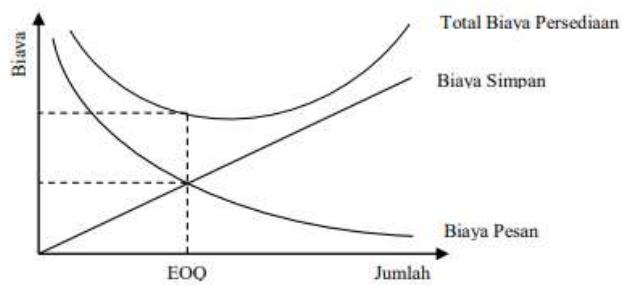
Biaya simpan mingguan dihitung dengan mencari rata-rata biaya penyimpanan tiap bulan yang dikonversi menjadi mingguan. Rata-rata persediaan dihitung sebanyak setengah kali kuantitas pesanan dikali biaya simpan per unit dan nilai ini akan berkurang terus-menerus hingga mencapai nol, sehingga biaya simpan dapat dirumuskan (Setiawan dan Hayati ,2012):

$$\text{Biaya Penyimpanan} = h \frac{y}{2}$$

Berdasarkan persamaan (2.1) dan persamaan (2.2) maka biaya yang muncul dalam persediaan adalah hasil penjumlahan biaya pemesanan dan biaya penyimpanan per periode waktu, dalam kasus ini adalah per minggu, dan dapat dirumuskan sebagai (Taha, 2003):

$$\text{Biaya persediaan per minggu} (TC) = K \frac{D}{y} + h \frac{y}{2}$$

Hubungan dari ketiga persamaan tersebut dapat dilihat dalam Gambar 2

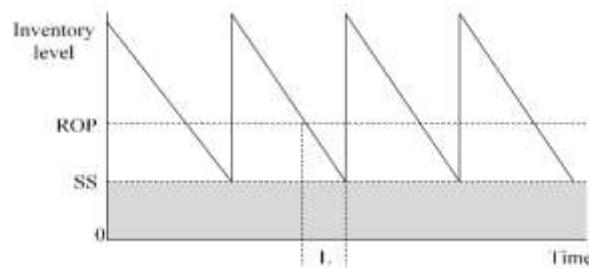


Gambar II.10 Kurva Biaya Persediaan

Sumber: Setiawan dan Hayati (2012)

II.3.1.3 Model Persediaan EOQ Probabilistik

Model persedian Stokastik adalah salah satu model persediaan yang paling mendasar dari semua model persediaan, selain itu model ini adalah model yang paling banyak digunakan dalam industri, dan sebagai dasar pengembangan model persediaan yang lainnya. Probabilitas distribusi biasanya diestimasikan berdasarkan data masa lalu. Model ini digunakan ketika ketidak pastian diperlakukan sebagai keacakan dan ditangani oleh teori probabilitas. Diasumsikan bahwa permintaan diwakili distribusi normal dengan memperkirakan rata-rata permintaan harian dan standart deviasinya (Thanthatemee dan Phruksapharnat, 2012)



Gambar II.11 Grafik Siklus Persediaan Stokastik

Sumber: Taha 2003

II.4 Reorder Point (Titik Pemesanan Kembali)

Selain menentukan EOQ, pengendalian persediaan juga menentukan kapan dilakukan pesanan atau pembelian kembali bahan. Pembelian atau pemesanan bahan jangan menunggu

sampai persediaan habis, karena kalau itu terjadi maka akan mengganggu kontinuitas produksi. Penentuan kapan melakukan pemesanan ini disebut dengan Reorder Point (ROP), yaitu saat dimana perusahaan atau manajer produksi harus melakukan kembali pembelian bahan. Hal ini diperlukan karena karena tidak selamanya pesanan bahan baku dapat segera dikirim oleh pihak pemasok atau levararisir, sehingga diperlukan waktu beberapa lama. (Ristono, 2009). Bila kebiasaan pesanan bahan baku datang dengan memakan waktu 7 hari misalnya, maka perusahaan harus memiliki persediaan yang cukup untuk memenuhi kebutuhan selama 7 hari tersebut. Waktu 7 hari tersebut dikenal dengan Lead-time. Dari EOQ dapat diketahui frekwensi pemesanan dan waktu siklus pemesanan. dari Waktu siklus tersebut dapat diketahui kapan persedian akan habis oleh karena itu ROP di lakukan sebelum waktu siklus itu berakhir, sehingga ketika waktu siklus tersebut berakhir maka barang yang dipesan akan datang dan mengisi persediaan kembali. Pemesanan kembali ditentukan berdasarkan kebutuhan selama tenggang waktu pemesanan. Jika posisi persediaan cukup untuk memenuhi permintaan selama tenggang waktu pemesanan, maka pemesanan kembali dilakukan sebanyak y (Ristono, 2009).

II.5 Metaheuristik

Dalam bahasa, konsep metaheuristik berasal dari penggabungan kata "meta," yang berarti "di atas" atau "melebihi," dan "heuristik," yang merujuk pada suatu metode atau teknik penyelesaian masalah yang bersifat temu-balik atau berbasis pengalaman. Dalam ilmu komputer dan optimasi matematika, metaheuristik dapat didefinisikan sebagai suatu prosedur tingkat tinggi atau heuristik yang dirancang untuk menemukan, menghasilkan, menyempurnakan, atau memilih heuristik (algoritme pencarian parsial) yang mampu memberikan solusi yang cukup baik untuk masalah optimasi atau masalah pembelajaran mesin. Secara harfiah, Metaheuristik adalah algoritme pengoptimalan yang dapat memecahkan berbagai masalah, termasuk masalah yang sulit atau tidak mungkin dipecahkan dengan metode tradisional (Tzanetos dan Dounias, 2021, He dkk., 2022, Sharma dkk., 2021). Metaheuristik mencakup berbagai pendekatan adaptif yang memungkinkan penyesuaian terhadap perubahan kondisi atau kompleksitas masalah secara dinamis. Kemampuan ini membedakan metaheuristik dari metode tradisional, karena dapat menangani ruang pencarian

yang besar atau memiliki sifat yang tidak terstruktur dengan efisien. Oleh karena itu, dalam berbagai konteks aplikasi, metaheuristik memberikan solusi yang lebih fleksibel dan serbaguna untuk permasalahan optimasi yang kompleks. Sebagaimana yang disebutkan oleh Chawla dan Duhan, (2018) dan (Halim et al., 2021), Algoritme ini didasarkan pada penjelajahan ruang pencarian masalah secara cerdas dan efisien untuk menemukan solusi berkualitas tinggi. Penelusuran yang cerdas ini melibatkan penggunaan heuristik yang adaptif dan strategi pencarian yang mengoptimalkan eksplorasi dan eksloitasi dalam ruang pencarian.

Menurut Fathonali-Fard et al., (2023) Metaheuristik adalah metode optimasi yang memanfaatkan algoritma stokastik, iteratif, dan acak. Tidak seperti pendekatan optimasi konvensional, metaheuristik tidak menjamin solusi yang optimal. Sebaliknya, mereka berusaha untuk mengidentifikasi solusi berkualitas tinggi yang mendekati solusi optimal dalam jangka waktu yang masuk akal (Ezugwu et al., 2021). Namun, metaheuristik memiliki kemampuan untuk beradaptasi dengan kondisi yang berubah-ubah dan mengeksplorasi spektrum solusi yang luas, sehingga cocok untuk memodelkan sistem cerdas rantai pasokan melingkar (Liu et al., 2021). Selain itu, metaheuristik dapat membantu dalam mengidentifikasi pertukaran terbaik antara tujuan yang saling bertentangan. Misalnya, menyeimbangkan biaya, dampak lingkungan, dan kesetaraan sosial sangat penting dalam merancang sistem sirkular yang berkelanjutan, dan metaheuristik dapat membantu mencapai tujuan ini (Pop et al., 2015). Dalam beberapa dekade terakhir, metaheuristik telah mendapatkan perhatian yang semakin meningkat sebagai metode yang kuat dan efisien untuk mengatasi tantangan komputasi yang kompleks dan masalah dunia nyata dalam manajemen rantai pasokan (Lin et al., 2018). Peningkatan perhatian terhadap metaheuristik mencerminkan keberhasilannya dalam memberikan solusi yang adaptif dan optimal untuk berbagai masalah dalam manajemen rantai pasokan. Metaheuristik memungkinkan penyesuaian dinamis terhadap ketidakpastian permintaan, fluktuasi stok, dan perubahan kondisi operasional.

Salah satu karakteristik utama dari metaheuristik adalah kemampuannya untuk menangani fungsi-fungsi yang kompleks dan nonlinier (Mazaheri dan Khodadadi, 2020, De

León-Aldaco dkk., 2015), yang membuatnya sangat berguna untuk masalah optimasi yang melibatkan banyak variabel atau kendala (Cuong-Le dkk., 2021, Sang-To dkk., 2023). Keuntungan lain dari metaheuristik adalah fleksibilitasnya, karena dapat diadaptasi ke domain masalah yang berbeda dan disesuaikan dengan tujuan tertentu (Nama et al., 2023, Nama et al., 2017). Dalam manajemen rantai pasokan yang semakin kompleks dan dinamis, metaheuristik membantu perusahaan untuk mengoptimalkan keefisienan operasional, mengurangi biaya penyimpanan, dan meningkatkan ketepatan waktu pengiriman. Dengan demikian, peningkatan perhatian terhadap metaheuristik mencerminkan kontribusinya yang signifikan dalam meningkatkan kinerja rantai pasokan secara keseluruhan, menjadikannya pilihan yang semakin penting dalam menghadapi tantangan dalam lingkungan bisnis yang terus berubah.

II.5.1 Jenis-jenis Metaheuristik

Dalam literatur yang ada, terdapat beberapa jenis metaheuristik yang berbeda, masing-masing dengan kelebihan dan kekurangannya (Abualigah et al., 2022). Penelitian yang dilakukan oleh Abualigah ini mencerminkan keragaman dalam jenis-jenis metaheuristik yang tersedia. Setiap jenis metaheuristik memiliki keunikan dan aplikabilitasnya sendiri tergantung pada sifat masalah yang dihadapi. Dengan mengakomodasi keberagaman ini, penelitian dan implementasi metaheuristik memberikan fleksibilitas dalam pendekatan terhadap berbagai masalah optimasi, memungkinkan pemilihan metode yang paling sesuai dengan karakteristik spesifik dari setiap tugas yang dihadapi. Beberapa contoh pendekatan metaheuristik secara umum adalah sebagai berikut.

Genetic Algorithm (GA). Algoritma ini terinspirasi dari seleksi alam dan menggunakan prinsip-prinsip genetika dan evolusi alamiah untuk mengoptimalkan suatu populasi solusi dalam beberapa generasi. Prosesnya mirip dengan evolusi biologis, dengan individu dalam populasi yang mewakili solusi potensial untuk suatu masalah. Algoritma genetika mencakup beberapa langkah kunci yaitu; inisialisasi Populasi. Dimana sebuah populasi awal solusi dihasilkan secara acak; Evaluasi Fitness. Setiap solusi dalam populasi dievaluasi berdasarkan sejauh mana solusi tersebut memenuhi kriteria tujuan (fitness function); Selanjutnya adalah Seleksi. Solusi-solusi yang memiliki kinerja lebih baik lebih

mungkin dipilih untuk bertahan hidup dan berkembang ke generasi berikutnya. Pemilihan dapat dilakukan dengan berbagai metode, seperti seleksi turnamen atau roulette wheel; lalu crossover (Crossover). Pada tahap ini pasangan solusi dipilih, dan informasi genetik antar mereka dipertukarkan untuk menghasilkan solusi baru. Ini mensimulasikan konsep reproduksi dan persilangan dalam evolusi biologis; langkah selanjutnya, Mutasi. Beberapa solusi dipilih secara acak, dan sebagian kecil dari informasi genetiknya diubah untuk memperkenalkan variasi baru ke dalam populasi; terakhir Evaluasi dan Iterasi. Langkah-langkah 2-5 diulang untuk setiap generasi. Populasi berevolusi seiring waktu, dan solusi yang lebih baik memiliki probabilitas lebih tinggi untuk bertahan dan memberikan kontribusi pada generasi berikutnya. Proses ini diulang beberapa kali, dan solusi yang dihasilkan pada akhirnya diharapkan mendekati atau mencapai solusi optimal untuk masalah yang diberikan. Algoritma genetika efektif untuk eksplorasi ruang pencarian yang besar dan kompleks, serta untuk menemukan solusi yang memenuhi kriteria optimasi pada berbagai jenis masalah.

Simulated Annealing. Algoritme ini didasarkan pada proses anal yang digunakan dalam metalurgi, di mana logam dipanaskan dan kemudian didinginkan secara perlahan untuk mengurangi cacat. Tujuannya adalah mencari solusi optimal dalam ruang pencarian yang mungkin memiliki banyak minimum lokal. Proses dimulai dengan inisialisasi solusi awal, yang kemudian diubah secara iteratif. Langkah-langkah pencarian solusi optimal dengan Simulated Annealing dimulai dengan inisialisasi suhu awal. Suhu ini menentukan seberapa besar kemungkinan solusi yang buruk akan diterima pada awal pencarian. Selama iterasi, solusi baru dihasilkan dengan melakukan perubahan acak pada solusi saat ini. Jika solusi baru lebih baik, diterima tanpa ragu. Namun, jika solusi baru lebih buruk, masih bisa diterima dengan probabilitas tertentu, yang berkurang seiring berjalananya waktu. Proses simulasi ini menciptakan efek "penurunan suhu" secara bertahap, di mana penerimaan solusi buruk menjadi semakin sulit. Hal ini membantu mencegah terjebak di minimum lokal dan memungkinkan pencarian solusi yang lebih baik secara global. Algoritma berlanjut hingga suhu mencapai nilai ambang tertentu atau kriteria berhenti lainnya. Hasilnya adalah solusi yang dapat dianggap sebagai optimal dalam konteks ruang pencarian yang kompleks.

Simulated Annealing terbukti efektif dalam menangani masalah optimisasi nonkonvensional yang sulit.

Ant Colony Optimization (ACO). Algoritma ini terinspirasi dari perilaku semut, yang menggunakan feromon untuk berkomunikasi dan menemukan jalur terpendek antara sarang mereka dan sumber makanan (Bertsimas dan Tsitsiklis, 1993 dalam Salama,2022). Tujuannya adalah menemukan solusi optimal dalam ruang pencarian. Proses ini melibatkan koloni semut yang bergerak melalui solusi potensial, melepaskan senyawa feromon, dan mengikuti jalur dengan tingkat feromon tertinggi. Langkah-langkah pencarian solusi optimal dengan ACO dimulai dengan inisialisasi solusi dan penyebaran semut ke seluruh solusi. Semut melepaskan feromon selama perjalanan mereka. Semakin baik solusi yang ditemukan, semakin tinggi tingkat feromonnya. Selanjutnya, solusi dengan feromon tinggi lebih cenderung dipilih oleh semut berikutnya. Optimasi koloni semut menggunakan pendekatan yang sama untuk menemukan jalur terbaik antara titik awal dan akhir dalam sebuah graf (Dorigo et al., 2006 dalam Shami et al.,2022). Proses ini berlanjut seiring waktu, dengan feromon menguap secara alami. Tetapi, solusi yang optimal akan terus diperkuat karena semut lebih sering memilih jalur tersebut. Algoritma berlanjut hingga kriteria berhenti terpenuhi, seperti mencapai jumlah iterasi yang ditentukan atau solusi optimal ditemukan. Hasilnya adalah solusi optimal yang dicapai oleh koloni semut, menciptakan pendekatan adaptif untuk menangani masalah optimisasi yang kompleks. ACO telah terbukti efektif dalam menyelesaikan berbagai masalah, terutama yang melibatkan kombinatorial optimization.

Tabu Search (TS). Tabu Search (TS) adalah metode optimisasi yang fokus pada pencarian solusi optimal dengan menghindari kejebakan di minimum lokal. Metode ini dimodelkan berdasarkan konsep pencarian heuristik yang dinamakan "tabu list." TS memulai dengan solusi awal dan melakukan perubahan iteratif dengan tujuan meningkatkan kualitas solusi. Algoritma ini menggunakan pendekatan berbasis memori untuk mencegah algoritma mengunjungi kembali solusi sebelumnya yang telah dieksplorasi (Glover, 1989, Chakraborty et al., 2021). Langkah-langkah pencarian solusi optimal dengan Tabu Search dimulai dengan inisialisasi solusi dan tabu list. Tabu list berfungsi untuk mencatat langkah-langkah yang

telah diambil untuk mencegah revisiting solusi sebelumnya. Selama iterasi, solusi baru dievaluasi dan langkah-langkah yang diambil dimasukkan ke dalam tabu list untuk beberapa iterasi ke depan. Proses iteratif ini membantu TS menghindari solusi yang telah dieksplorasi sebelumnya, mendorong pencarian menuju ruang pencarian yang lebih luas. Tabu Search menggunakan kriteria aspirasi untuk memutuskan apakah solusi yang secara normal dihindari oleh aturan tabu dapat diambil jika memiliki peningkatan yang signifikan. Algoritma terus melakukan perubahan dan evaluasi solusi hingga kriteria berhenti terpenuhi, seperti mencapai jumlah iterasi maksimum atau mencapai tingkat akurasi yang diinginkan. Hasilnya adalah solusi optimal yang mencerminkan hasil pencarian yang efisien dan efektif dalam mengatasi kendala minimum lokal pada masalah optimisasi kompleks. Tabu Search telah sukses diterapkan dalam berbagai konteks optimisasi, termasuk masalah kombinatorial dan kontinu.

Particle Swarm Optimization (PSO). Menurut Kennedy dan Ebhart dalam (Samir Khatir et al., 2023) Algoritma ini didasarkan pada pergerakan partikel dalam ruang pencarian, yang bergerak menuju solusi terbaik yang ditemukan sejauh ini sambil menjelajahi wilayah baru di ruang pencarian. PSO merupakan metode optimisasi yang mirip dengan perilaku kelompok burung atau ikan. Partikel-partikel bergerak dalam ruang pencarian untuk mencari solusi optimal. Langkah-langkahnya melibatkan inisialisasi, evaluasi solusi, dan pembaruan iteratif posisi partikel berdasarkan pengaruh pribadi dan tetangga. Proses ini berlanjut hingga kriteria berhenti terpenuhi, menghasilkan posisi partikel terbaik sebagai solusi optimal. PSO efektif dalam menangani masalah kompleks dengan ruang pencarian yang besar atau tidak terstruktur. Meskipun algoritme-algoritme ini berbeda dalam hal prinsip dan mekanisme yang mendasarinya, semuanya bertujuan untuk menemukan solusi berkualitas tinggi untuk masalah-masalah pengoptimalan (Agushaka dkk., 2022a, Oyelade dkk., 2022, Ezugwu dkk., 2022). Dengan menggunakan teknik eksplorasi dan optimasi cerdas, metaheuristik dapat membantu memecahkan masalah kompleks di berbagai domain (Agushaka et al., 2022). \

II.6 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) pertama kali diperkenalkan oleh Kennedy dan Eberhart pada tahun 1995 dan telah terbukti sangat efektif untuk masalah-masalah statis.

(Yang tai et al., 2008). Particle Swarm Optimization (PSO), adalah sebuah algoritma metode swarm intelligence yang meniru pergerakan kawanan burung yang dikembangkan oleh Kennedy dan Eberhart (1995). Menurut Clerks dkk, (2002), PSO bekerja berdasarkan perilaku sosial dari kawanan dan menemukan solusi global dengan mengacu pada posisi terbaik partikel lain di setiap waktu. Selain itu, PSO dapat meniru kerja sama antar individu dalam kelompok yang sama dengan menggunakan swarm intelligence dan bertukar pengalaman dari generasi ke generasi (Kuo dan Han, 2011). Pendekatan ini memfasilitasi pencarian solusi yang adaptif dan kemampuan algoritma untuk terus berkembang dari generasi ke generasi.

Koefisien percepatan yang digunakan untuk kecepatan dan pembaruan posisi bergeser ke arah waktu yang bervariasi untuk secara efektif mengontrol pencarian global (Ratnaweera et al., 2004). Artinya, dalam Particle Swarm Optimization (PSO), koefisien percepatan digunakan untuk mengatur kecepatan dan pembaruan posisi partikel. Penggunaan koefisien percepatan ini dirancang untuk mengarahkan perubahan posisi partikel ke seluruh ruang pencarian dengan cara yang bervariasi sepanjang waktu. Tujuannya adalah untuk secara efektif mengendalikan proses pencarian secara global, membantu partikel menjelajahi ruang pencarian dengan cara yang dapat menemukan solusi optimal tanpa terjebak dalam minimum local. Alogaritma PSO berkembang cukup pesat, baik dari sisi aplikasi maupun dari sisi pengembangan metode yang digunakan, hal itu berlangsung sejak penama kali diperkenalkannya alogaritma tersebut (Jong jek siang 2009 dalam Ginantara et al.,2021), Maka dengan begitu, mereka mengategorikan alogaritma sebagai bagian dari kehidupan rekayasa/ buatan Artificial Life. Alogaritma Ini terhubung dengan komputasi evolusioner, algoritma genetik, dan pemrograman. Dalam Particle Swarm Optimization (PSO), kawanan diasumsikan mempunyai ukuran tertentu dengan setiap partikel posisi awalnya terletak disuatu lokasi yang acak dalam ruang multidimensi (Hsieh, L.F., Huang, C.J. & Huang, C.I. 2007 dalam Yener et al.,2019). Diasumsikan setiap partikel memiliki dua karakteristik yaitu posisi dan kecepatan. Setiap partikel bergerak dalam ruang atau Space tertentu dan mengingat posisi terbaik yang pernah dilalui atau ditemukan sumber makanan atau nilai fungsi objektif. Setiap partikel menyampaikan informasi atau posisi terbaiknya kepada partikel yang lain dan

menyesuaikan posisi dan kecepatan masing-masing berdasarkan yang diterima mengenai posisi yang bagus Asriningtias, s. R. dkk,2015).

Particle Swarm Optimization (PSO) adalah salah satu dari teknik komputasi evolusioner, yang mana populasi pada PSO didasarkan pada penelusur algoritma dan diawali dengan suatu populasi yang random yang disebut dengan particle. Berbeda dengan teknik komputasi evolusioner lainnya, setiap particle di dalam PSO juga berhubungan dengan suatu velocity. Partikel-partikel tersebut bergerak melalui penelusuran ruang dengan velocity yang dinamis yang disesuaikan menurut perilaku historisnya. Oleh karena itu, partikel-partikel mempunyai kecenderungan untuk 6 bergerak ke area penelusuran yang lebih baik setelah melewati proses penelusuran. Particle Swarm Optimization (PSO) mempunyai kesamaan dengan genetic algorithm yang mana dimulai dengan suatu populasi yang random dalam bentuk matriks. Namun PSO tidak memiliki operator evolusi yaitu crossover dan mutasi seperti yang ada pada genetic algorithm. Baris pada matriks disebut particle atau dalam genetic algorithm sebagai kromosom yang terdiri dari nilai suatu variable. Setiap particle berpindah dari posisinya semula ke posisi yang lebih baik dengan suatu velocity. Pada algoritma PSO vektor velocity di update untuk masing-masing partikel kemudian menjumlahkan vektor velocity tersebut ke posisi particle. Update velocity dipengaruhi oleh kedua solusi yaitu global best yang berhubungan dengan biaya yang paling rendah yang pernah diperoleh dari suatu particle dan solusi local best yang berhubungan dengan biaya yang paling rendah pada populasi awal (Asriningtias, s. R. dkk, 2015).

Jika solusi local best mempunyai suatu biaya yang kurang dari biaya solusi global yang ada, maka solusi local best menggantikan solusi global best. Kesederhanaan algoritma dan performansinya yang baik, menjadikan PSO telah menarik banyak perhatian di kalangan para peneliti dan telah diaplikasikan dalam berbagai persoalan optimisasi. PSO telah populer menjadi optimisasi global dengan sebagian besar permasalahan dapat diselesaikan dengan baik di mana variabel- variabelnya adalah bilangan riil. Menurut (Wati, Dwi Ana Ratna, 2011) terdapat beberapa istilah umum yang biasa digunakan dalam Particle Swarm Optimization dapat didefinisikan sebagai berikut: Swarm: populasi dari suatu algoritma. Particle: anggota (individu) pada suatu swarm. Setiap particle merepresentasikan suatu solusi

yang potensial pada permasalahan yang diselesaikan. Posisi dari suatu particle adalah ditentukan oleh representasi solusi saat itu. Pbest (Personal best): posisi Pbest suatu particle yang menunjukkan posisi particle yang dipersiapkan untuk mendapatkan suatu solusi yang terbaik. Gbest (Global best): posisi terbaik particle pada swarm atau posisi terbaik diantara Pbest yang ada. Velocity (v): vektor yang menggerakkan proses optimisasi yang menentukan arah di mana suatu particle diperlukan untuk berpindah (move) untuk memperbaiki posisinya semula atau kecepatan yang menggerakkan proses optimasi yang menentukan arah dimana particle diperlukan untuk berpindah dan memperbaiki posisinya semula. Inertia weight (0): inertia weight di simbolkan w, parameter ini digunakan untuk mengontrol dampak dari adanya velocity yang diberikan oleh suatu particle. Learning Rates (c1 dan c2): suatu konstanta untuk menilai kemampuan particle (c1) dan kemampuan sosial swarm (c2) yang menunjukkan bobot dari particle terhadap memorinya.

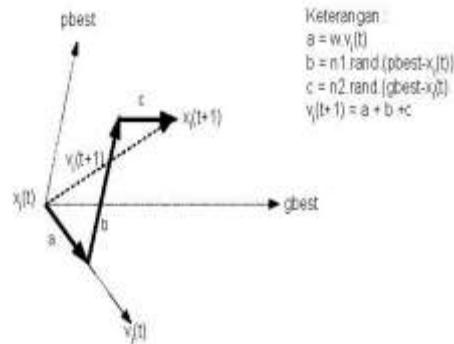
Posisi dari tiap partikel dapat dianggap sebagai calon solusi (candidate solution) bagi suatu masalah optimisasi. Tiap-tiap partikel diberi suatu fungsi fitness merancang sesuai dengan menunjuk masalah yang yang bersesuaian. Ketika masing- masing partikel bergerak ke suatu posisi baru didalam ruang pencarian, itu akan mengingat sebagai personal best (Chen, R.M. & Shih, H.F. 2013). Sebagai tambahan terhadap ingatan informasi sendiri, masing-masing partikel akan juga menukar informasi dengan partikel yang lain dan mengingat global best (Gbest). Kemudian masing-masing partikel akan meninjau kembali arah dan percepatannya sesuai dengan Pbest dan Gbest untuk bergerak ke arah yang optimal dan menemukan solusi yang optimal. Dengan keuntungan dari aplikasi yang mudah dan sederhana, lebih sedikit parameter yang diperlukan, dan hasil yang baik, PSO telah diadopsi didalam banyak bidang, seperti TSP, flowshop, VRP, task-resource assignment, penjadwalan khusus dan lain lain. Sebab itu PSO telah pula diterapkan dalam membentuk penjadwalan yang optimal untuk university courses. Seperti halnya dengan algoritma evolusioner yang lain, algoritma PSO adalah sebuah populasi yang didasarkan penelusuran inisialisasi partikel secara random dan adanya interaksi diantara partikel dalam populasi. Di dalam PSO setiap partikel bergerak melalui ruang solusi dan mempunyai kemampuan untuk mengingat posisi terbaik sebelumnya dan dapat bertahan dari generasi ke generasi. Algoritma PSO

dikembangkan dengan berdasarkan pada model berikut (Engelbrecht, A.P. 2006 dalam Ezugu et al., 2022; Kusumawati dkk, 2015): Ketika seekor burung mendekati target atau makanan (atau bisa minimum atau maximum suatu fungsi tujuan) secara cepat mengirim informasi kepada burung- burung yang lain dalam kawanan tertentu. Burung yang lain akan mengikuti arah menuju ke makanan tetapi tidak secara langsung. Ada komponen yang tergantung pada pikiran setiap burung, yaitu memorinya tentang apa yang sudah dilewati pada waktu sebelumnya. Menurut Bai (2010) keuntungan dari Algoritma PSO adalah (Raharjo, j. S. D.,2013): Particle Swarm Optimization berdasar pada kecerdasan (intelligence). Ini dapat diterapkan ke dalam kedua penggunaan dalam bidang teknik dan riset ilmiah. PSO tidak punya overlap dan kalkulasi mutasi. Pencarian dapat dilakukan oleh kecepatan dari partikel. Selama pengembangan beberapa generasi, kebanyakan hanya partikel yang optimis yang dapat mengirim informasi kepartikel yang lain, dan kecepatan dari pencarian adalah sangat cepat. Perhitungan didalam Algoritma PSO sangat sederhana, menggunakan kemampuan optimisasi yang lebih besar dan dapat diselesaikan dengan mudah. PSO memakai kode/jumlah yang riil, dan itu diputuskan langsung dengan solusi, dan jumlah dimensi tetap sama dengan solusi yang ada.

Menurut Raharjo, j. S. D. (2013) menyatakan bahwa ada berapa kerugian dari Algoritma PSO adalah: Metode mudah mendapatkan optimal parsial (sebagian), yang mana menyebabkan semakin sedikit ketepatannya untuk peraturan tentang arah dan kecepatan. Metode tidak bisa berkembang dari permasalahan sistem yang tidak terkoordinir, seperti solusi dalam bidang energi dan peraturan yang tidak menentu didalam bidang energi. Model Algoritma PSO ini akan disimulasikan dalam ruang dengan dimensi tertentu dengan sejumlah iterasi sehingga di setiap iterasi, posisi partikel akan semakin mengarah ke target yang dituju (minimasi atau maksimasi fungsi). Ini dilakukan hingga maksimum iterasi dicapai atau bisa juga digunakan kriteria penghentian yang lain. Hal ini disebabkan, PSO merupakan algoritma optimasi yang mudah dipahami, cukup sederhana, dan memiliki unjuk kerja yang sudah terbukti handal. Algoritma PSO dapat digunakan pada berbagai masalah optimasi baik kontinyu maupun diskrit, linier maupun nonlinier. PSO memodelkan aktivitas pencarian solusi terbaik dalam suatu ruang solusi sebagai aktivitas terbangnya kelompok

partikel dalam suatu ruang solusi tersebut. Dengan demikian, awal penelusuran pada algoritma PSO dilakukan dengan populasi yang random (acak) yang disebut dengan partikel dan jika suatu partikel atau seekor burung menemukan jalan yang tepat atau pendek menuju sumber makanan, maka sisa kelompok yang lain juga akan segera mengikuti jalan tersebut meskipun lokasi mereka jauh di kelompok tersebut. Posisi partikel dalam ruang solusi tersebut merupakan kandidat solusi yang berisi variabel-variabel optimasi. Setiap posisi tersebut akan dikaitkan dengan sebuah nilai yang disebut nilai objektif atau nilai fitness yang dihitung berdasarkan fungsi objektif dari masalah optimasi yang akan diselesaikan.

(Wati & Rochman,2013) PSO memodelkan aktivitas pencarian solusi terbaik dalam suatu ruang solusi sebagai aktivitas terbangnya kelompok partikel dalam suatu ruang solusi tersebut. Posisi partikel dalam ruang solusi tersebut merupakan kandidat solusi yang berisi variabel-variabel optimasi. Setiap posisi tersebut akan dikaitkan dengan sebuah nilai yang disebut nilai objektif atau nilai fitness yang dihitung berdasarkan fungsi objektif dari masalah optimasi yang akan diselesaikan. Proses di dalam algoritma PSO dapat dijelaskan berdasarkan pada gambar sebagai berikut:



Gambar II.12 Algoritma PSO

Sumber: wati & Rochman,2013

Pada gambar diatas menjelaskan terdapat p partikel yang tersebar se cara acak pada ruang solusi yang ada. Posisi partikel i saat waktu t, yaitu $x_i(t)$ akan diperbaiki menurut persamaan posisi sebagai berikut: Persamaan ke-2

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

dengan $v_i(t+1)$ adalah kecepatan partikel yang dihitung dengan persamaan berikut:
Persamaan ke-3

$$v_i(t+1) = wv_i(t) + n_1r_1[pbest_i(t) - x_i(t)] + n_2r_2[gbest_i(t) - x_i(t)]$$

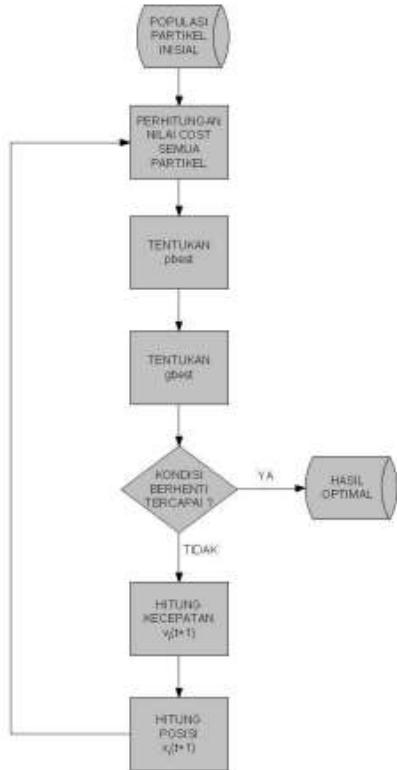
Titik $pbest_i(t)$ adalah solusi lokal terbaik yang dicapai oleh partikel i sampai saat t , dan merepresentasikan kontribusi kognitif terhadap vektor $v_i(t+1)$. Titik $gbest(t)$ adalah solusi global terbaik yang telah dicapai diantara partikel-partikel sampai saat t dan merepresentasikan kontribusi sosial terhadap vektor kecepatan. Bilangan acak r_1 dan r_2 terdistribusi seragam pada interval $[0,1]$. Faktor skala kognitif (n_1) dan faktor skala sosial (n_2) pada algoritma PSO umumnya bernilai dua. Variabel w adalah bobot incrsia. Inersia yang besar memfasilitasi eksplorasi global (pencarian dalam daerah yang luas) sedangkan inersia yang kecil menghasilkan eksplorasi lokal (pencarian dalam daerah yang sempit). Oleh karena itu, nilai w merupakan faktor kritis yang menentukan perilaku konvergen algoritma PSO. Untuk itu, direkomendasikan untuk memilih nilai w yang besar pada awalnya agar menghasilkan eksplorasi global pada ruang solusi, selanjutnya nilai w diturunkan secara bertahap untuk mendapatkan solusi yang lebih baik.

Pers. 2 dan 3 dijelaskan oleh Gambar 1 sebagai penjumlahan vektor. Faktor pembelajaran kognitif, yang dirumuskan sebagai $n_1r_1[pbest_i(t) - x_i(t)]$ pada Pers. 2 merupakan memori partikel jangka pendek. Faktor pembelajaran kognitif ini menunjukkan inklinasi partikel untuk mengulang perilaku sebelumnya yang terbukti sukses pada partikel tersebut. Hal ini juga menunjukkan adanya pengaruh memori partikel tersebut. Faktor pembelajaran sosial, yang diberikan oleh $n_2r_2[gbest_i(t) - x_i(t)]$, merupakan "peer pressure" bagi sebuah partikel. Faktor pembelajaran sosial tersebut menunjukkan inklinasi partikel untuk meniru atau mengemulasi perilaku partikel lain yang telah sukses. Hal ini menunjukkan adanya pengaruh tetangga partikel. Setiap iterasi, sekawanan partikel dievaluasi nilai fitness-nya dan berdasarkan nilai tersebut, kecepatan dan posisi partikel diperbaharui. Proses dalam algoritma PSO dapat disajikan dengan Diagram alir algoritma

PSO secara detail dijelaskan sebagai berikut. Untuk masalah optimasi dengan fungsi objektif f dan variabel x_1, \dots, x_n sebagai persamaan berikut: Persamaan ke-4

$$\min_{(x_1, \dots, x_n)} f_{(x_1, \dots, x_n)}$$

Dengan ukuran populasi n_{pop} dan ukuran partikel n langkah-langkah optimasi dengan PSO adalah Inisialisasi populasi partikel: dilakukan dengan membangkitkan populasi partikel (par) secara acak pada ruang solusi $[0,1]$, sebanyak n_{pop} yang masing- masing berisi variabel sebanyak n . Jadi populasi awal berupa matriks dengan ukuran $n_{pop} \times n$. Selanjutnya juga dilakukan pembangkitan nilai kecepatan partikel vpar berupa bilangan random pada domain $[0,1]$. vpar berukuran sama dengan par yaitu $n_{pop} \times n$. Perhitungan nilai fitness semua partikel diawali dengan melakukan proses dekoding partikel ke dalam domain variabel x_1, \dots, x_n yaitu $[x_{lower}, x_{upper}]$. Setelah itu nilai variabel hasil dekoding dievaluasi oleh fungsi $f(x_1, \dots, x_n)$. Tentukan pbest (terbaik lokal); hasil evaluasi fungsi $f(x_1, \dots, x_n)$ diperoleh nilai cost sebanyak n_{pop} yaitu $f_1, \dots, f_{n_{pop}}$. Selanjutnya pbest ditentukan sebagai berikut: jika nilai cost dari sebuah partikel lebih baik (lebih kecil) dari nilai cost sebelumnya maka pbest akan diset ke nilai cost yang baru, atau dapat dijelaskan sebagai berikut: Untuk iterasi ke k ; Untuk $i=1$ sampai n_{pop} Jika $pbest_i(k-1) > f_i(k)$ maka $pbest_i(k) = f_i(k)$ Jika $pbest_i(k-1) < f_i(k)$ maka $pbest_i(k) = pbest_i(k-1)$ Tentukan gbest (terbaik global); jika nilai minimum cost dari seluruh partikel lebih baik (lebih kecil) dari nilai minimum cost dari seluruh partikel pada iterasi sebelumnya maka nilai gbest akan diset sebagai nilai minimum cost yang baru tersebut.



Gambar II.13 Diagram alir algoritma PSO

Sumber: wati & Rochman,2013

Atau untuk iterasi ke k , tentukan; minimum $f_l(k), \dots, f_{n_{pop}}(k) = f_{\min}(k)$. Jika $f_{\min}(k) \neq g_{\min}(k-1)$ maka $g_{\min}(k) = f_{\min}(k)$. Jika tidak demikian maka $g_{\min}(k) = g_{\min}(k-1)$. Lalu lakukan pengecekan ketika kondisi berhenti, jika kondisi berhenti tercapai maka keluar dari proses perulangan (iterasi), jika belum tercapai lanjutkan ke iterasi berikutnya yaitu perhitungan kecepatan untuk masingmasing partikel sesuai pers. 3 dan perhitungan posisi yang baru untuk masing-masing partikel sesuai pers. 2

II.6.1 Prosedur Algoritma Particle Swarm Optimization

Prosedur standar untuk menerapkan algoritma PSO adalah sebagai Berikut: (puspitasari G,) Inisialisasi populasi dari particle-particle dengan position value dan velocity secara random dalam suatu ruang dimensi penelusuran. Evaluasi fungsi objektif optimisasi

yang diinginkan pada setiap particle. Membandingkan nilai objektif pada tiap particle dengan personal best yang ada. Jika nilai yang ada lebih baik dibandingkan dengan nilai personal best, maka nilai tersebut dipakai sebagai personal best yang baru dan _ sama dengan lokasi particle yang ada_dalam ruang dimensional Identifikasi particle dalam lingkungan dengan hasil terbaik sejauh ini. Update velocity dan posisi particle Kembali ke langkah 2 sampai kriteria terpenuhi, berhenti pada nilai objektif yang cukup baik atau sampai pada jumlah maksimum iterasi. (Teugeh dkk., 2009 dalam Mubarok et al.,2023)

II.6.2 Inisialisasi Populasi

PSO dimulai dengan membangkitkan populasi secara random. Populasi berisi kandidat-kandidat solusi yang digunakan untuk menyelesaikan sebuah permasalahan yang akan terus diperbaiki pada tiap iterasinya. Pada PSO, populasi yang diambil umumnya tidak terlalu besar, antara 20 sampai 50. Untuk permasalahan permutation flowshop sendiri, menurut Tasgetiren (2004) ukuran populasi yang diambil sebanyak $2 \times$ jumlah job dengan batas atas untuk pengambilan populasi secara acak X_{max} merupakan bilangan bulat positif.

II.6.3 Personal Best p_i^t

Personal best p_i^t merupakan vektor yang menggambarkan posisi terbaik untuk *particel* i berdasarkan nilai *fitness* terbaik hingga iterasi t . Fungsi objektif yang digunakan untuk meminimalkan makespan adalah $f(\pi_i^t x_i^t)$ diman π_i^t menunjukkan permutasi job pada particle x_i^t , sedangkan personal best pada particle i ditunjukkan seperti $f(\pi_i^t x_i^t) \leq f(\pi_i^{t-1} x_i^{t-1})$. Secara sederhana, fungsi fitness pada personal best dapat ditulis $f_i^p = f(\pi_i^t p_i^t)$. Personal best dapat ditulis $p_i^t = [p_{i1}^t, p_{i2}^t, \dots, p_{in}^t]$ untuk tiap particle i , dimana posisi terbaik (p_{ij}^t) direpresentasikan sebagai position value untuk personal best ke- i terhadap job j pada iterasi t . Pada setiap iterasinya, personal best akan diperbaiki apabila $f_i^t < f_i^p$, untuk $i = 1, 2, \dots, p$ (Tasgetiren dkk., 2004).

II.6.4 Global Best (G^t)

Global best (G^t) merupakan particel terbaik, yang berisi kumpulan posisi terbaik, dari seluruh personal best pada iterasi t . Berdasarkan pengertian tersebut, global best dapat diperoleh dengan $f(\pi^t G^t) \leq f(\pi_i^t p_i^t)$ untuk $i = 1, 2, \dots$, sederhana, fungsi fitness pada global

best dapat ditulis $f^g = f(\pi^t G^t)$. Global best sendiri dapat ditulis $G_i^t = [g_1^t, g_2^t, \dots, g_n^t]$ dimana g_j^t merupakan position value untuk particle yang terpilih sebagai global best terhadap job j pada iterasi t. pada setiap iterasinya, gelobal best akan diperbaiki apabila $f_i^t < f^g$, untuk $i=1,2,\dots,p$. (Tasgeterin dkk, 20004).

II.6.5 Update Velocity v_{ij}^t

Particle dapat bergerak menelusuri ruang solusi dengan velocity. Pada dasarnya velocity digunakan untuk menentukan arah dimana suatu particle diperlukan untuk berpindah dan memperbaiki posisi sebelumnya sehingga particle dapat menuju ke ruang solusi yang lebih baik. Velocity akan diperbaiki pada tiap iterasi dengan memperhatikan beberapa hal, antara lain velocity sebelumnya, pengaruh personal best dan global best pada iterasi sebelumnya. Velocity diperbaiki dengan menggunakan persamaan berikut:

$$v_{ij}^t = w^{t-1} v_{ij}^{t-1} + c_1 r_1 (p_{ij}^{t-1} - x_{ij}^{t-1}) + c_2 r_2 (g_{ij}^{t-1} - x_{ij}^{t-1})$$

Selanjutnya akan dibentuk populasi baru dengan memperbaiki position value pada tiap particel dengan persamaan berikut: $x_{ij}^t = x_{ij}^{t-1} + v_{ij}^t$, dimana $I=1,2,\dots,p$; $j=1,2,\dots,n$

Ketereangan:

v_{ij}^t : velocity yang terletak pada particel i, job j untuk iterasi 1-1

w^{t-1} : inertia weight pada iterasi 1-1

x_{ij}^t : position value yang terletak pada particel i, job j untuk iterasi t

x_{ij}^{t-1} : position value yang terletak pada particel i, job j untuk iterasi t-1

p_{ij}^{t-1} : personal best yang terletak pada particel i, job j untuk iterasi 1-

g_{ij}^{t-1} : global best pada job j untuk iterasi (-1

$c_1 c_2$: cognitivedan social parameter

$r_1 r_2$: random uniforam [0,1]

II.6.6 Interia Weight

Inertia weight merupakan salah satu parameter yang ada pada PSO yang berfungsi sebagai pengontrol pengaruh dari velocity sebelumnya untuk velocity yang sekarang. Pada dasarnya, inertia weight diperkenalkan untuk keseimbangan antara kemampuan penelusuran global dan local. Inertia weight akan diperbaiki dengan menggunakan persamaan:

$$w^t = w^{t-1} \times a$$

Keterangan:

w^t : inertia weight pada iterasi

w^{t-1} : inertia weight pada iterasi 1-1

a : faktor pengurangan (decrement factor)

Nilai pada parameter ini biasanya berkisar antara 0.4 sampai dengan 0.9 dan nilai a yang biasa digunakan 0.975 (Uysal dan Bulkan, 2008).

II.6.7 Cognitive Dan Social Parameter C1 C2

Cognitive parameter merupakan parameter yang digunakan untuk mengontrol pengaruh dari personal best terhadap position value pada iterasi sebelumnya. Social parameter merupakan parameter yang digunakan untuk mengontrol pengaruh dari global best terhadap position value pada iterasi sebelumnya. Cognitive dan social parameter berisi konstanta-konstanta yang umumnya bernilai 1.5-2.0 dan 2.0 -2.5 untuk masing-masing parameter (Hasan, 2004).

II.7 Penelitian Terdahulu

Bagian ini menjelaskan beberapa jurnal atau artikel penelitian sebelumnya yang dapat dijadikan rujukan dalam topik penelitian ini. Pemilihan penelitian terdahulu disesuaikan dengan permasalahan yang ada, diharapkan dapat memberikan pemahaman dan referensi untuk menyelesaikan penelitian ini. Berikut adalah pembahasan mengenai artikel penelitian terdahulu yang mencakup Pada penelitian (Tao Zhang dkk. 2015) yang berjudul Pencocokan Persediaan multi-level dan perencanaan Pesanan dalam Lingkungan Produksi Hibrida Make-To-Order/Make-To-Stock untuk Pabrik Baja melalui *Particle Swarm Optimization*,

menjelaskan penyelesaian masalah persediaan yang dihadapi oleh industri Baja. Tantangan itu bagaimana menyediakan pengiriman tepat waktu, pengiriman akurat, dan produk berkualitas tinggi. Selain itu, industri ini harus memiliki *utilisasi* peralatan yang tinggi dan biaya produksi yang rendah untuk memenuhi kebutuhan pasar. Strategi manajemen yang dilakukan untuk menyelesaikan masalah perencanaan pemesanan pada MTO/MTS *hybrid* dilakukan dengan menggunakan metode pendekatan Metaheuristik yaitu *Particle Swarm Optimization* yang ditingkatkan dan dikombinasikan dengan pencarian lokal yang inovatif untuk memecahkan masalah tersebut. Hasil penelitian menunjukkan bahwa model memiliki validitas yang tinggi dan metode PSO yang ditingkatkan memberikan kemanjuran yang baik.

Pada penelitian (Chung-Yuan Dye, 2012) yang berjudul Model Persediaan yang Memburuk dalam Jangka Waktu Terbatas dengan Penetapan Harga Dua Fase dan Permintaan serta Biaya yang Bervariasi Terhadap Waktu dalam Pembiayaan Kredit Perdagangan Menggunakan *Particle Swarm Optimization*, menjelaskan tentang masalah persediaan. Selama fase pengenalan dan pertumbuhan siklus hidup produk, perusahaan ini menghadapi permintaan yang besar dengan sedikit persaingan. Jadi pola permintaan yang bervariasi terhadap waktu perlu dipertimbangkan untuk mencerminkan penjualan di berbagai fase siklus hidup produk. Dalam beberapa tahun terakhir, sebagian besar penelitian mengenai model kebijakan pengisian persediaan yang melibatkan permintaan bervariasi terhadap waktu, berasumsi bahwa biaya pembelian satuan dan harga jual satuan adalah tetap. Asumsi biaya pembelian unit dan harga jual unit tetap pada model EOQ klasik tidak mencerminkan situasi di mana tingkat inflasi tinggi atau situasi di mana terjadi kenaikan harga atau penurunan diperkirakan terjadi. Sama halnya pada penelitian ini yang mengalami masalah terkait persediaan, termasuk asumsi permintaan konstan, pembayaran tepat waktu, dan ketidaksesuaian dengan kondisi pasar riil. Model kurang memperhitungkan fluktuasi biaya, kenaikan harga, dan strategi harga dinamis yang penting untuk optimalkan keuntungan. Penelitian ini bertujuan untuk menemukan nilai optimal dari harga jual, jumlah pengisian ulang, dan skema pengisian ulang yang memaksimalkan total keuntungan dalam jangka waktu perencanaan yang terbatas. Dengan memperkenalkan metodologi baru yang mempertimbangkan dinamika pasar yang lebih kompleks, termasuk integrasi kebijakan

kredit perdagangan dua tingkat maka digunakan Pendekatan *Particle Swarm Optimization*. Hasilnya, penelitian yang disajikan memasukkan banyak model sebelumnya sebagai kasus khusus. Dari hasil percobaan menunjukkan bahwa PSO dengan faktor penyempitan merupakan algoritma stabil konvergen untuk model ini karena perbedaan maksimal antar solusi berada pada kisaran 0,0389%. Juga telah ditunjukkan bahwa PSO dengan faktor penyempitan memiliki kinerja lebih baik dibandingkan yang lain dalam aspek efisiensi dan ketahanan komputasi.

Penelitian selanjutnya adalah penelitian oleh (Ali Akbar dkk, 2019) dengan judul Model EOQ Dua Gudang dengan Biaya Persediaan Bernilai Interval dan Pembayaran di Muka untuk Barang yang Rusak dalam *Particle Swarm Optimization*. Penelitian ini menguraikan dua masalah utama tentang tantangan manajemen persediaan. Pertama, kesulitan menentukan jumlah persediaan optimal dan biaya penyimpanan tinggi akibat penggunaan dua gudang, terutama gudang sewaan. Kedua implementasi skema pembayaran di muka menciptakan tantangan dalam menangani ketidakpastian situasi pemasaran yang kompetitif dalam model persediaan dua gudang. Dari analisis masalah ini, berbagai varian teknik *Particle Swarm Optimization* (yaitu PSO-CO, WQPSO dan GQPSO) telah dikembangkan untuk memecahkan masalah model persediaan yang diusulkan dengan menggunakan aritmatika interval dan hubungan urutan interval. Terakhir, untuk mengilustrasikan dan juga memvalidasi model yang diusulkan, contoh numerik telah diselesaikan dan solusi terbaik yang ditemukan (baik solusi optimal atau solusi mendekati optimal) yang diperoleh dari varian PSO yang berbeda telah dibandingkan. Kemudian, analisis sensitivitas dilakukan untuk mempelajari pengaruh perubahan berbagai parameter model terhadap kebijakan optimal.

Penelitian (Asoke Kumar dkk, 2016) berjudul Investigasi Masalah Persediaan Dua Gudang dalam Lingkungan Interval di Bawah Inflasi Melalui *Particle Swarm Optimization*. Dalam tulisan ini, masalah persediaan dua gudang telah diselidiki dalam kondisi inflasi dengan efek kerusakan yang berbeda pada dua gudang terpisah (gudang sewaan, RW, dan gudang milik, OW). Tujuan dari penelitian ini adalah untuk menentukan ukuran lot dari siklus sistem persediaan dua gudang dengan meminimalkan biaya rata-rata sistem. Dengan

mempertimbangkan kebijakan persediaan yang berbeda, model yang sesuai telah dirumuskan untuk tren linier dalam parameter permintaan dan nilai interval biaya. Penelitian ini menjelaskan model dua gudang menggunakan SFI (*Set-Valued Fuzzy Interval*) dan IFS (*Interval-Valued Fuzzy Set*) dapat berkaitan dengan cara menangani ketidakpastian dalam lingkungan persediaan. SFI dan IFS memungkinkan representasi interval dan *fuzzy* untuk parameter yang tidak pasti, yang dapat mencerminkan variasi dalam permintaan, waktu pengiriman, atau faktor lainnya. Setelah model dua gudang dijelaskan dengan menggunakan SFI dan IFS, penelitian kemudian membahas penerapan teknik optimasi kawanan partikel (PSO) untuk menyelesaikan model tersebut. PSO digunakan untuk mencari solusi optimal atau mendekati optimal dalam lingkungan yang tidak pasti tersebut. Akhirnya, kesimpulan yang bermanfaat telah dicapai mengenai pemilihan kebijakan persediaan yang tepat untuk sistem dua gudang.

Penelitian selanjutnya oleh (Taofik Safrodin dkk, 2023) yang berjudul Optimasi Algoritma K- *Nearest Neighbor* Berbasis *Particle Swarm Optimization* Untuk Meningkatkan Kebutuhan Barang. Masalah yang dihadapi pada perusahaan ini adalah difektivitas penyelenggaraan persediaan di gudang kurang optimal, memungkinkan mengalami *overstock* atau kekurangan barang ; Kesulitan menentukan nilai k terbaik pada algoritma K-NN, yang sensitif terhadap hasil klasifikasi ; harga dan volume permintaan karet fluktuatif dan tidak stasioner. Untuk menyelesaikan masalah tersebut digunakan pendekatan K-NN untuk Klasifikasi, dengan menentukan nilai k optimal dengan algoritma genetika, menerapkan K-NN pada *dataset* IKG desa. Selain itu, penelitian ini menggunakan K-NN dan PSO untuk menentukan harga prediksi karet. Prosedurnya, K-NN digunakan untuk prediksi harga karet, dan optimalisasi hasil dengan PSO. Hasilnya K-NN dan PSO menunjukkan peningkatan performa dengan RMSE 0,082, lebih baik daripada K-NN saja.

Penelitian lainnya oleh (Jiu Ping et al, 2012) yang berjudul Algoritme *Particle Swarm Optimization* Berbasis Pemrograman Dinamis untuk Masalah Manajemen Persediaan dalam Kondisi Ketidakpastian. Dalam proyek konstruksi skala besar, seperti yang diatur oleh ketentuan kontrak FIDIC, setiap item material penting harus dibeli melalui undangan penawaran, dengan hanya satu penawar yang berhasil untuk setiap item. Penelitian ini

mengangkat kompleksitas masalah manajemen persediaan untuk proyek konstruksi skala besar (IMP-LCP), di mana terdapat hubungan kompleks antar item persediaan. Contohnya, proporsi pembelian barang utama dan bantu dalam proyek beton bendungan selalu saling terkait, bergantung pada desain proporsi campuran beton untuk konstruksi. Manajer konstruksi perlu menentukan proporsi pembelian untuk setiap item berdasarkan hubungan tersebut. Untuk mengatasi ketidakpastian tersebut, pendekatan *hybrid crisp* digunakan untuk mengubah parameter acak *fuzzy* menjadi variabel *fuzzy* yang selanjutnya didefuzzifikasi menggunakan operator nilai ekspektasi dengan indeks optimis-pesimis. Sifat berulang dari model penulis memotivasi mereka untuk mengembangkan algoritma PSO berbasis DP. Lebih khusus lagi, pendekatan mereka memperlakukan variabel keadaan sebagai parameter tersembunyi. Hal ini pada gilirannya menghilangkan banyak pemeriksaan kelayakan yang berlebihan selama inisialisasi dan pembaruan partikel pada setiap iterasi. Hasil dan analisis sensitivitas disajikan untuk menyoroti kinerja metode optimasi penulis, yang sangat efektif dibandingkan dengan algoritma PSO standar.

Penelitian selanjutnya oleh (Dragan Simic et al, 2019) berjudul *Particle Swarm Optimization* dan Pencarian Adaptif Murni dalam Manajemen Persediaan Barang Jadi. Penelitian ini menyajikan efisiensi baru berdasarkan kerja sama antara kedua teknik untuk mengoptimalkan manajemen persediaan. Salah satu teknik yang diterapkan adalah kecerdasan gerombolan yang terinspirasi secara biologis secara umum, khususnya *Particle Swarm Optimization* (PSO) dan yang kedua adalah teknik optimisasi global yang disebut *Pure Adaptive Search* (PAS) untuk memodelkan pengendalian persediaan dalam sistem produksi. Tujuan dari penelitian ini adalah untuk membuat model untuk meminimalkan nilai persediaan, yang diimplementasikan dalam dua tahap. Tahap pertama adalah meminimalkan biaya produksi berdasarkan permintaan produk dan harga barang yang direalisasikan dengan teknik PSO. Fase kedua mengimplementasikan hasil eksperimen yang dikumpulkan pada fase pertama proses optimasi global berdasarkan pengendalian persediaan pemodelan PAS untuk meminimalkan nilai persediaan dalam sistem produksi. Berdasarkan pembahasan dalam penelitian ini, dapat disimpulkan bahwa penggunaan teknik optimasi saja tidak cukup untuk mengoptimalkan proses persediaan-produksi. Alasan utamanya adalah jenis masalah

persediaan-produksi yang memerlukan koordinasi berbagai elemen dan faktor menjadi satu kesatuan yang harmonis yang dapat digunakan dalam situasi ketika faktor-faktor tersebut bertindak berlawanan arah.

Penelitian yang dilakukan oleh (Ming Feng et al, 2009) berjudul Menerapkan *Particle Swaem Optimization* Linier ke Model Persediaan Multi Eselon Serial. Penelitian ini membahas aspek manajemen rantai pasok, menanggapi kebutuhan pelanggan secara akurat dan secara efektif mengurangi total biaya merupakan kegiatan penting bagi perusahaan untuk mencapai keunggulan kompetitif mereka dan memiliki peluang untuk memperoleh sejumlah besar keuntungan dalam lingkungan manajemen rantai pasokan global yang sangat kompetitif saat ini. Oleh karena itu, penelitian ini memberikan model serial multi-eselon terintegrasi *just in time* (JIT) berdasarkan pertimbangan waktu tunggu pengiriman yang tidak pasti dan kualitas yang tidak dapat diandalkan (model SMEIJI). Maka pada penelitian ini menerapkan pendekatan *Particle Swarm Optimization* (PSO) sebagai metode untuk menghasilkan solusi yang lebih baik dalam memecahkan masalah bilangan bulat non linier campuran. Pembahasan utama artikel ini adalah untuk membuat model persediaan *just-intime* terintegrasi *multi echelon* serial dengan waktu tunggu pengiriman yang tidak pasti dan pertimbangan kualitas yang tidak dapat diandalkan. Hal terpenting, di dalamnya fokus membandingkan *Liner decreasing weight particle swarm optimization* (LDW-PSO) dengan *non-liner decreasing weight particle swarm optimization* (NLDW-PSO). Berdasarkan pengaturan parameter pemasangan yang dilakukan, hasil akhir menunjukkan bahwa (LDW-PSO) akan dapat dilakukan secara efisien dan memiliki solusi utama dalam menyelesaikan masalah persediaan multi-eselon.

Penelitian yang dilakukan oleh (Chi-Yang Tsai et al, 2007) berjudul Pendekatan *Particle Swarm Optimization* Beberapa Tujuan untuk Klasifikasi Persediaan. Masalah utama penelitian ini adalah kompleksitas manajemen persediaan dalam perusahaan, terutama terkait dengan pengelompokan dan klasifikasi persediaan. Skema klasifikasi yang umum digunakan seperti ABC mungkin terlalu sederhana untuk perusahaan dengan persediaan yang terdiversifikasi. Selain itu, pendekatan tunggal atau kriteria tunggal dalam skema klasifikasi mungkin tidak memadai untuk memenuhi berbagai tujuan perusahaan, seperti pengurangan

biaya, korelasi permintaan barang, dan perputaran persediaan. Oleh karena itu, perlu adanya skema klasifikasi yang lebih fleksibel dan dapat diadaptasi dengan mempertimbangkan berbagai faktor serta tujuan perusahaan yang beragam. Implementasi teknik optimasi gerombolan partikel (PSO) dalam penelitian ini bertujuan untuk mengatasi beberapa keterbatasan skema klasifikasi yang ada. Hasil penelitian tersebut mencakup penerapan pendekatan *Particle Swarm Optimization* (PSO) pada masalah klasifikasi persediaan. Melalui eksperimen dan studi numerik, penelitian ini berhasil menentukan kombinasi terbaik dari nilai parameter algoritma. Kinerja algoritma PSO dibandingkan dengan metode klasifikasi lain yang diketahui, dan hasilnya disajikan untuk mengukur efektivitasnya.

Penelitian oleh (Huthaifa et al, 2018) yang berjudul Pendekatan *Particle Swarm Optimization* Multi Objektif untuk Sistem kontrol produksi Persediaan. Penelitian ini mencari Solusi optimal yang dipilih berdasarkan kriteria bersaing untuk mengurangi variabilitas pesanan (*bullwhip effect*) dan meningkatkan respons persediaan. Tujuan penelitian ini untuk mengoptimalkan kinerja dinamis sistem pengendalian produksi-persediaan dalam hal minimalisasi rasio varians antara tingkat pesanan dan konsumsi, dan meminimalkan integral kesalahan absolut antara tingkat persediaan aktual dan target dengan memasukkan *Pareto Optimality* ke dalam *Particle Swarm Optimization* (PSO). Desain/metode/pendekatan –Sistem kendali produksi-persediaan dimodelkan dan dioptimalkan melalui teori dan simulasi kendali. Dinamika sistem pengendalian produksi-persediaan dimodelkan melalui persamaan diferensial waktu berkelanjutan dan transformasi Laplace. Perancangan simulasi dilakukan dengan menggunakan model *state-space* dari sistem. Hasil *Multi Objective Particle Swarm Optimizaton* (MOPSO) dibandingkan dengan hasil publikasi yang diperoleh dari optimisasi *weighted genetic algorithm* (WGA) . Temuan –Hasil yang diperoleh dari proses optimasi MOPSO memastikan bahwa kinerjanya secara sistematis lebih baik daripada WGA dalam hal mengurangi variabilitas pesanan (*bullwhip effect*) dan meningkatkan respons persediaan (tingkat layanan pelanggan) dalam kondisi operasional yang sama. Penelitian ini menunjukkan penerapan unik PSO dalam mengoptimalkan kinerja dinamis sistem pengendalian produksi-persediaan.

Penelitian yang dilakukan oleh Huayang Deng et al,2021, dengan judul Model Optimasi Gabungan Strategi Persediaan dan Persediaan s,S, Menggunakan Algoritma Berbasis PSO yang Disempurnakan. Penelitian ini bertujuan untuk mengoptimalkan kebijakan persediaan dalam jaringan pasokan multiechelon dan multiperiode dengan menggunakan algoritma Particle Swarm Optimization (PSO) yang ditingkatkan. Fokus utama adalah memperbaiki kebijakan s, S dengan mempertimbangkan waktu tunggu acak dan variasi persediaan maksimum untuk meminimalkan biaya total, termasuk biaya transportasi, biaya persediaan, dan kerugian akibat keterlambatan. Model ini memanfaatkan skema optimasi PSO dengan bobot inersia adaptif dan fungsi penalti untuk mengatasi kompleksitas nonlinier dan multiperiode. Hasil penelitian menunjukkan bahwa metode ini secara signifikan mengurangi total biaya dibandingkan dengan pendekatan tradisional, menawarkan skema pasokan yang lebih optimal. Penelitian ini memberikan kontribusi penting dalam pengelolaan pasokan suku cadang dengan menyediakan model yang lebih efisien dan referensi untuk penelitian lebih lanjut di bidang ini.

Penelitian saat ini yang dilakukan oleh Surya Daliman dengan judul Pemodelan Sistem Persediaan (s,S) yang Mempertimbangkan Keterbatasan Kapasitas Angkutan Menggunakan Particle Swarm Optimization (PSO) pada Pengiriman Berjenjang Dua Eselon. Penelitian ini bertujuan untuk mengoptimalkan sistem persediaan berbasis kebijakan (s, S) dengan memperhitungkan keterbatasan kapasitas angkutan dalam sistem pengiriman berjenjang dua eselon. Metode yang digunakan mencakup kebijakan (s, S) untuk menentukan waktu dan jumlah pengisian ulang persediaan, serta menggunakan salah satu metoda Metaheuristik yaitu algoritma Particle Swarm Optimization (PSO) untuk mencari solusi optimal dalam pengelolaan persediaan dan kapasitas angkutan. Penelitian berfokus dengan kemampuannya untuk memasukkan faktor keterbatasan kapasitas angkutan, yang memberikan perspektif lebih realistik dalam manajemen persediaan. Selain itu, kompleksitas penelitian ini meningkat karena melibatkan pengoptimalan dalam konteks dua eselon dan keterbatasan kapasitas angkutan. Penelitian ini berbeda dari studi sebelumnya dengan fokus pada keterbatasan kapasitas angkutan dan penggunaan PSO dalam model dua eselon, yang belum sepenuhnya dieksplorasi dalam literatur terkait.

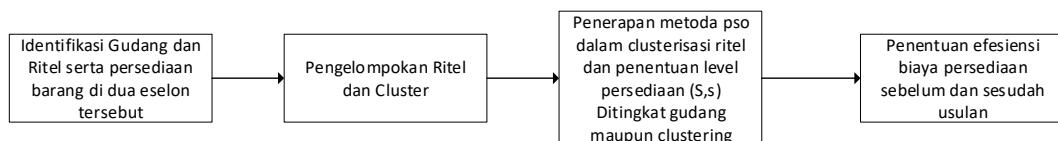
Nama Peneliti	Metoda												
	Sistem Persediaan				Metaheuristik					Pola Deskripsi			
	EOQ	Inventori	s,Q	s,S	T,S	GA	SA	ACO	TS	PSO	Satu Eselon	Dua Eselon	Multi Eselon
Tao Zhang et al., 2015		✓								✓		✓	
Chung-Yuan dye, 2012	✓	✓								✓		✓	
Ali Akbar dkk., 2019	✓	✓								✓		✓	
Asoke Kumar et al., 2016		✓								✓		✓	
Taofik Safrodin dkk, 2023		✓								✓			
Jiu Ping et al., 2012		✓								✓			
Dragan Simic et al., 2019		✓								✓			
Ming Fing et al., 2009		✓								✓			✓
Chi-Yang Tsai et al., 2019		✓		✓						✓			
Huthaifa et al., 2018		✓								✓			
Huayang Deng et al., 2021	✓	✓		✓						✓			✓
Surya Daliman, 2023	✓	✓		✓						✓		✓	

Tabel II. 1 Penelitian Terdahulu

BAB III Usulan Pemecahan Masalah

III.1 Metodologi Penelitian

PT. XYZ di Kota Bandung mengoperasikan sistem distribusi dua eselon. Eselon pertama yang menjadi pusatnya adalah gudang. Dan eselon kedua adalah ritel. Terdapat 8 ritel dalam perusahaan ini yang tersebar luas di kota Bandung. Dalam konteks rantai pasok, pengelolaan persediaan menghadapi beberapa tantangan utama. PT. XYZ mengalami tantangan signifikan dalam manajemen persediaan dan logistik pengiriman barang. Permasalahan utamanya meliputi penentuan jumlah pesanan optimal untuk menghindari kelebihan atau kekurangan stok, sambil mempertimbangkan keterbatasan kapasitas angkutan dalam sistem pengiriman. Selain itu, perusahaan perlu mencari solusi untuk meminimalkan total biaya persediaan. Oleh karena itu, diperlukan pendekatan yang tepat untuk mengidentifikasi dan menerapkan solusi yang efektif dalam menyelesaikan masalah tersebut. Solusi yang diterapkan melibatkan metode persediaan (s, S), yang merupakan model manajemen persediaan berbasis titik pemesanan. Model persediaan (s,S) dan algoritma PSO (Particle Swarm Optimization) diterapkan dalam penelitian ini untuk dapat menekan biaya persediaan. Model (s,S) digunakan untuk menghitung kuantitas pesanan barang yang optimal dengan memperhatikan titik pemesanan kembali. Sementara PSO digunakan sebagai pendekatan optimisasi dalam menemukan solusi (s,S) pada sistem persediaan perusahaan. Penelitian ini dilakukan pada sistem persediaan dua eselon. Berikut Flowchart yang menggambarkan metodologi penelitian ini :



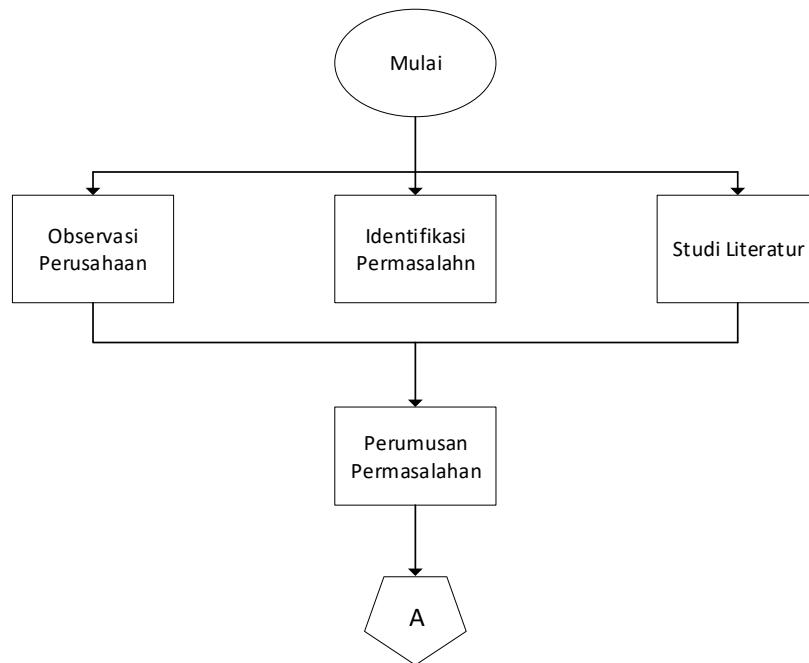
Gambar III. 1 Metodologi Penelitian

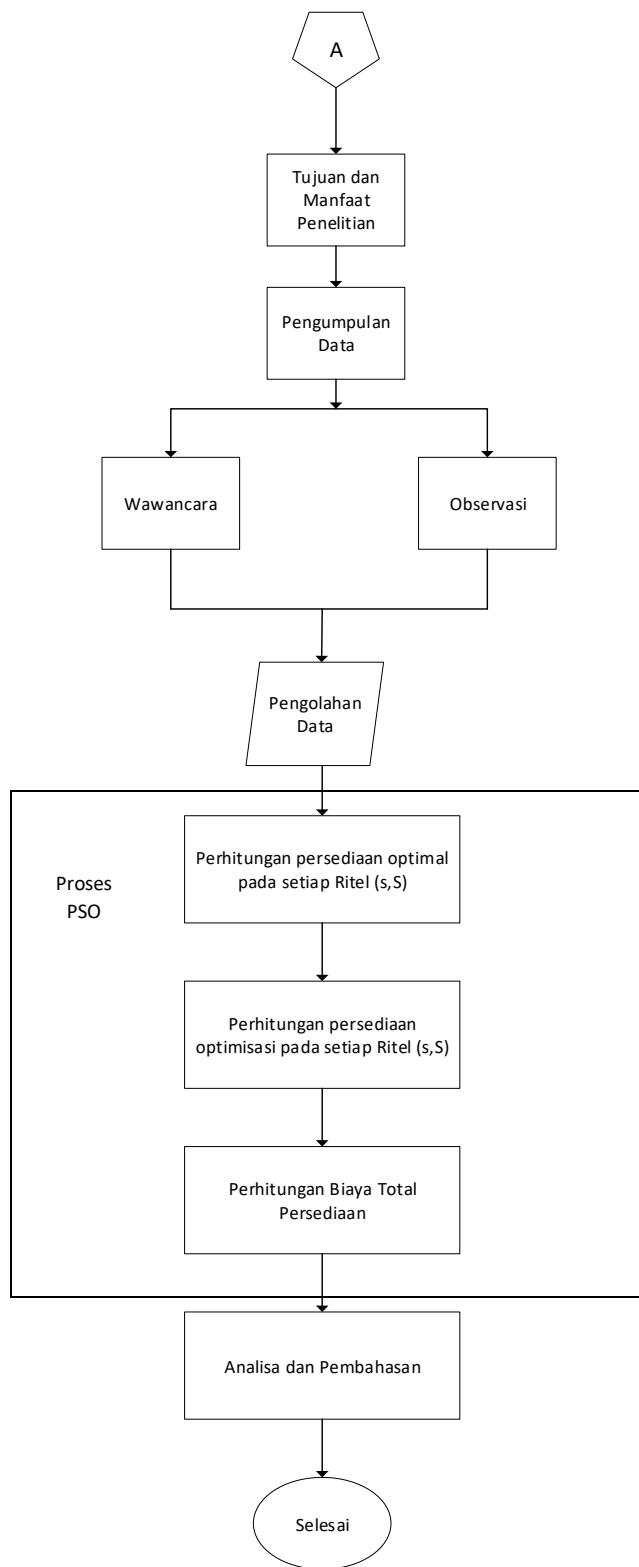
Dengan demikian, penelitian ini bertujuan untuk memberikan solusi terhadap masalah ini melalui pemodelan persediaan dua eselon dengan menggunakan pendekatan metaheuristik, berfokus pada integrasi model persediaan s,S dan PSO. Berdasarkan kasus yang dihadapi maka didapatkan model pemecahan masalah yang menjelaskan tiap

tahapan untuk dapat menyelesaikan penelitian ini terhadap kasus tersebut, ialah sebagai berikut:

III.2 Model Pemecahan Masalah

Pada langkah-langkah pemecahan masalah disajikan flowchart yang menggambarkan langkah-langkah pemecahan masalah secara sistematis. Tahapan ini, dimulai dengan observasi perusahaan, identifikasi permasalahan, dan studi literatur, yang merupakan langkah awal dalam memahami konteks dan kebutuhan penelitian. Selanjutnya, langkah-langkah tersebut berkembang menjadi beberapa tahap yang lebih rinci, mulai dari penentuan tujuan dan manfaat penelitian, pengumpulan data melalui wawancara dan observasi, hingga pengolahan data. Proses selanjutnya melibatkan perhitungan persediaan optimal untuk setiap ritel dengan metode Continuous Review, Order-Up-to-Level (s, S) System, serta perhitungan biaya total persediaan. serta optimisasi persediaan menggunakan Particle Swarm Optimization (PSO). Terakhir, analisa dan pembahasan dilakukan untuk menarik kesimpulan dari hasil perhitungan dan memberikan rekomendasi yang sesuai. Gambar flowchart berikut mengilustrasikan secara keseluruhan langkah-langkah ini.





Gambar III. 2 Flowchart Langkah-langkah Pemecahan Masalah

III.3 Studi Literatur

Dalam studi literatur menjelaskan berbagai macam teori yang berkaitan dengan tahapan pemecahan masalah pada kasus tersebut diantaranya, Model persediaan s,S dan algoritma PSO (Particle Swarm Optimization) diterapkan dalam penelitian ini untuk dapat menekan biaya persediaan. Model s,S digunakan untuk menghitung kuantitas pesanan barang yang optimal dengan memperhatikan titik pemesanan kembali. Sementara PSO digunakan sebagai pendekatan metaheuristik untuk menemukan solusi alternative/global dalam sistem persediaan perusahaan. Penelitian ini dilakukan pada sistem persediaan dua eselon. Dalam pembahasan ini, teori diambil dari buku, jurnal, serta karya ilmiah lain yang membahas penelitian yang serupa dengan masalah yang dihadapi pada penelitian ini. Studi literatur bertujuan guna memahami konteks penelitian, menghindari kesenjangan penelitian, dan meningkatkan kredibilitas penelitian. Hal ini dilakukan sebagai bahan pendukung dalam proses pengolahan data sehingga mendapatkan kejelasan atau keakuratan sebagai pandangan dalam penelitian ini.

III.4 Identifikasi Masalah

Adapun identifikasi masalah yang terdapat di PT.XYZ , berdasarkan hasil pengamatan pada PT.XYZ yang sedang berjalan didapatkan permasalahan yang berdampak besarnya biaya persediaan, dan kuantitas pemesanan barang yang tidak optimal maka dengan itu pada penelitian ini dilakukan secara mendalam untuk dapat menekan biaya persediaan dan menghitung kuantitas pesanan barang yang optimal dengan memperhatikan titik pemesanan kembali. Sementara PSO digunakan sebagai pendekatan metaheuristik untuk menemukan solusi alternative/global dalam sistem persediaan perusahaan. Penelitian ini dilakukan pada sistem persediaan dua eselon, dengan tujuan mempercepat pengembalian modal serta memperhatikan ketersediaan barang. penelitian focus mencakup analisis dampak keterlambatan dalam keluarnya barang terhadap stabilitas keuangan perusahaan, upaya mencapai keseimbangan optimal antara percepatan distribusi dan pemeliharaan ketersediaan yang memadai, evaluasi kebijakan pengendalian persediaan yang ada, serta langkah-langkah konkret dalam mengoptimalkan penerapan pendekatan metaheuristik seperti EOQ dan PSO. Dengan

memahami secara mendalam tantangan ini, pada penelitian ini diharapkan dapat memberikan kontribusi solutif untuk meningkatkan efisiensi manajemen persediaan dua eselon di PT. XYZ.

III.5 Tujuan Pemecahan Masalah

Tujuan pemecahan masalah ini bertujuan untuk mengetahui mengetahui tingkat persediaan di dalam tahapan Gudang pusat dan ritel, banyaknya jumlah persediaan di tingkat ritel, dan untuk mengetahui kapan waktu yang tepat untuk melakukan permintaan ulang (Reorder Point) di setiap ritel.

III.6 Pengumpulan Data

Pada penelitian ini dibutuhkan beberapa data diantanya data perusahaan dan observasi. Data ini akan diolah sesuai dengan tujuan penelitian. Data-data yang diperlukan sebagai berikut: Data PT.XYZ Hijab Fashion Data yang diperlukan di PT.XYZ Hijab Fashion terdiri dari visi, misi, motto, budaya, dan esensi perusahaan. Demand Gudang pusat dan ritel PT.XYZ Hijab Fashion selama 24 bulan data riwayat pemintaan produk ritel-ritel ke perusahaan. Data lead time Gudang pusat dan setiap ritel Kumpulan data waktu tunggu produk Biaya pesan Gudang pusat Biaya yang timbul setiap kali gudang pusat melakukan pemesanan atau pengadaan barang dari pemasok atau unit lain dalam rantai pasokan. Biaya pesan ritel Biaya pesan pembelian bahan baku yang dilakukan ritel ke Gudang pusat Biaya simpan persediaan Gudang Biaya yang timbul karena menyimpan barang atau persediaan di dalam gudang selama periode tertentu. Harga jual produk Kumpulan data mengenai harga jual setiap produk dari Gudang pusat ke ritel.

III.7 Pengolahan Data

Pengolahan data merupakan sebuah proses untuk mengubah, memproses sebuah informasi sehingga dapat membantu dalam pengambilan keputusan. Dalam pengolahan data penelitian ini, melakukan perhitungan model Economic Order Quantity (EOQ) yang di optimasi dengan metode Particle Swarm OptimIZATION Optimization (PSO) Menggunakan Sofware MATLAB.

III.7.1 Optimisasi Model Persediaan s,S Menggunakan Particle Swarm Optimization (PSO)

III.7.1.1 Model Persediaan s,S

Untuk mengetahui total biaya persediaan pada system Dua Eselon terlebih dahulu harus diketahui tingkat persediaan pada masing-masing tingkat dan juga reorder point pada tingkat ritel. Ada beberapa notasi yang harus dipahami sebelum melakukan perhitungan yaitu sebagai berikut:

C_{ow} : biaya pemesanan per pesanan untuk gudang.

C_{or} : biaya pemesanan per pesanan untuk pengecer.

C_w : harga satuan barang di Gudang.

C_r : harga satuan barang di Ritel.

C_w^c : biaya penambahan Nilai di Gudang

C_r^c : biaya penambahan Nilai di Ritel

i : nilai tercatat per tahun

s_w : titik pemesanan ulang di Gudang

s_r : titik pemesanan ulang di Ritel

Q_w : jumlah pesanan untuk Gudang.

Q_r : jumlah pesanan untuk Ritel.

D_w : permintaan waktu tunggu di Gudang

D_r : permintaan waktu tunggu di Ritel

L : waktu tunggu

HK : Hari Kerja

I_w^c : tingkat persediaan Eselon Gudang Pusat

I_r^c : tingkat persediaan Eselon Ritel

III.7.1.2 Model Persediaan s,S Tingkat Ritel

PT.XYZ Memiliki 8 ritel untuk Produk Fashion Hijab. Setiap ritel memiliki perilaku yang berbeda-beda seperti dari demand dan lead time maka dari itu, perlu dilakukan perhitungan untuk masing-masing ritelnya agar diketahui kebutuhannya jumlah pesan optimal (Q_r) dan Reorder Point (s_r) yang berpengaruh kepada Total Inventory Cost (TIC) untuk perusahaan. Berikut merupakan model perhitungan yang digunakan oleh setiap ritelnya:

1. Menghitung Biaya Nilai Tambah di Tingkat Ritel (C_r^c)

Biaya nilai tambah ini dihitung untuk mengetahui biaya yang dikeluarkan akibat adanya biaya depresiasi dan biaya tenaga kerja berikut merupakan modelnya:

$$C_r^c = C_r - C_w \quad (\text{III.1})$$

2. Menghitung nilai n di Tingkat Ritel

Nilai n merupakan bilangan integer yang digunakan untuk perhitungan Total Inventory Cost (TIC) yang minimal. Untuk mencari nilai n tersebut bisa menggunakan persamaan seperti berikut.

$$n = \sqrt{\frac{C_{ow}C_r^c}{C_{or}C_w^c}} \quad (\text{III.2})$$

frekuensi pemesanan gudang 1 kali

Jika hasil dari perhitungan nilai n menggunakan persamaan diatas bukan bilangan integer atau bilangan bulat, maka perlu dilakukan perhitungan untuk nilai $f(n_1)$ dan $f(n_2)$ dengan menggunakan nilai $n_1 = 1$ dan $n_2 = 2$ berikut merupakan model persamaannya

$$f(n_1) = \left(C_{or} + \frac{C_{ow}}{n_1} \right) (n_1 C_w^c + C_r^c) \quad (\text{III.3})$$

$$f(n_2) = \left(C_{or} + \frac{C_{ow}}{n_2} \right) (n_2 C_w^c + C_r^c)$$

Dengan pembatas dimana

$n = n_1$, jika $f(n_1) \leq f(n_2)$

$n = n_2$, jika $f(n_2) > f(n_1)$

3. Menghitung jumlah Persamaan Optimal (Q_r) di Tingkat Ritel

Jumlah Persamaan Optimal (Q_r) dihitung untuk mengetahui jumlah unit yang harus dipesan oleh setiap ritel yang berdampak untuk meminimalkan Total Inventory Cost (TIC) Pada perusahaan. Berikut merupakan model Persamaannya

$$Q_r = \sqrt{\frac{2(C_{or} + \frac{C_{ow}}{n})D}{i(nC_w^c + C_r^c)}} \quad (\text{III.4})$$

Menghitung Reorder Point di Tingkat Ritel (s_r)

Reorder Point merupakan tingkat persediaan minimum dimana pesanan baru harus dilakukan untuk menghindari kehabisan stok. Berikut merupakan model persamaan untuk menghitung Reorder Point (s_r) di setiap ritel.

$$s_r = L \times \frac{Q_r}{\text{Hari Kerja}} \quad (\text{III.5})$$

4. Menghitung Tingkat Persediaan di Tingkat Ritel (I_r^c)

Untuk mengetahui tingkat persediaan di Gudang, maka perlu diketahui tingkat persediaan di ritel juga. Dimana, persediaan ditingkat ritel

$$I_r^c = \frac{1}{2} Q_r \quad (\text{III.6})$$

Jadi, Persediaan eselon untuk tingkat ritel adalah sebagai berikut

$$I_r^c = I_r \quad (\text{III.7})$$

III.7.1.3 Perhitungan Model Persediaan s,S Tingkat Gudang Pusat

Setelah melakukan perhitungan untuk di setiap ritel, maka perlu melakukan perhitungan persediaan juga di tingkat Gudang pusat. Berikut merupakan model persamaan yang dapat digunakan untuk menghitung jumlah persamaan optimal (Q_w)

dan Reorder Point (s_w) di tingkat Gudang pusat untuk mengetahui total inventory Cost (TIC) keseluruhan.

1. Menghitung Biaya Nilai Tambah di Tingkat Gudang (C_w^c)

Biaya nilai tambah ini dihitung untuk mengetahui biaya yang dikeluarkan akibat adanya biaya depresiasi dan biaya tenaga kerja. Berikut merupakan model persamaannya.

$$C_w^c = C_w$$

2. Menghitung Jumlah Pesan Produk (Q_w) di Tingkat Gudang

Untuk perhitungan jumlah pesan produk (Q_w) di Tingkat Gudang diperlukan data dari setiap ritel. Karena setiap ritel memiliki perilaku masing-masing sehingga setelah didapatkan hasil dari setiap ritel kemudian dijumlah dan didapatkan jumlah pesan untuk tingkat gudang. Berikut merupakan model persamaannya. Kemudian, tahap kedua dilakukan perhitungan biaya penambahan nilai tingkat pada ritel. Menggunakan persamaan berikut:

$$Q_w = \sum_{i=1}^n (Q_r) \quad (\text{III.9})$$

3. Menghitung Reorder Point di Tingkat Gudang (s)

Berikut merupakan model persamaan untuk menghitung Reorder Point(s)

$$s_w = L \times \frac{Q_r}{\text{Hari Kerja}} \quad (\text{III.10})$$

4. Menghitung Tingkat Persediaan di Gudang (I_w^c) Untuk mengetahui tingkat persediaan eselon di gudang, maka perlu diketahui tingkat persediaan di setiap ritel juga. Sebelumnya perlu dihitung terlebih dahulu rata-rata persediaannya, yaitu menggunakan persamaan berikut:

$$I_w^c = \frac{1}{2} Q_w \quad (\text{III.11})$$

Sehingga, untuk perhitungan tingkat persediaan eselonnya yaitu sebagai berikut.

$$I_w^c = I_w + \sum_{i=1}^n (I_r) \quad (\text{III.12})$$

III.7.1.4 Model Perhitungan Tingkat Inventory Cost (TIC)

Perhitungan Total Inventory Cost (TIC) dilakukan perhitungan untuk biaya pesan dan simpan di tingkat ritel. Kemudian, setelah diketahui biaya dari setiap ritelnya ditambahkan dengan biaya pesan dan biaya simpan di tingkat gudang pusat. Berikut merupakan model persamaan untuk perhitungan Total Inventory Cost (TIC)

1. Perhitungan Biaya Pesan di Tingkat Gudang Pusat

Perhitungan biaya pesan ini diambil dari biaya pesan di tingkat gudang pusat, demand dan jumlah pemesanan di tingkat gudang pusat. Sehingga, untuk mendapatkan biaya pesan adalah sebagai berikut.

$$\text{Biaya Pesan} = \frac{c_{ow} D_w}{Q_w} \quad (\text{III.13})$$

2. Perhitungan Biaya Simpan di Tingkat Gudang Pusat

Perhitungan biaya simpan diambil dari tingkat persediaan yang ada, biaya simpan dan biaya nilai tambah yang terjadi di tingkat gudang pusat. Sehingga, didapatkan persamaan berikut.

$$\text{Biaya Simpan} = I_w^c i C_w^c \quad (\text{III.14})$$

3. Perhitungan Biaya Pesan di Tingkat Ritel

Perhitungan biaya pesan ini diambil dari biaya pesan, demand dan jumlah pemesanan di tingkat ritel. Sehingga, untuk mendapatkan biaya pesan adalah sebagai berikut

$$\text{Biaya Pesan} = \frac{c_{ow} D_r}{Q_r} \quad (\text{III.15})$$

4. Perhitungan Biaya Simpan di Tingkat Gudang Pusat

Perhitungan biaya simpan diambil dari tingkat persediaan yang ada, biaya simpan dan biaya nilai tambah yang terjadi di tingkat ritel. Sehingga, didapatkan persamaan berikut

$$Biaya Simpan = I_r^c i C_r^c$$

Dari setiap perhitungan biaya pesan dan biaya simpan dijumlahkan untuk mengetahui Total Inventory Cost (TIC) di masing-masing eselon. Jadi, untuk mengetahui Total Inventory Cost (TIC) terlebih dahulu menghitung biaya pesan dan biaya simpan di masing-masing ritel. Setelah diketahui maka dijumlahkan untuk mengetahui Total Inventory Cost (TIC) di tingkat ritel, kemudian hasil dari masing-masing ritel dijumlahkan dan dihitung TIC di dua eselon dengan menambahkan hasil Total Inventory Cost (TIC) di tingkat gudang, berikut merupakan persamaan modelnya.

$$TIC = \frac{C_{ow} D_w}{Q_w} + I_w^c i C_w^c + \sum_{i=1}^n \left(\frac{C_{ow} D_w}{Q_w} + I_r^c i C_r^c \right) \quad (\text{III.16})$$

III.8 Analisa dan Pembahasan

Hasil dari pengumpulan data dan pengolahan data yang sudah peneliti lakukan, kemudian dilakukan Analisa dan diuraikan berdasarkan dari hasil pemrosesan. Dalam bagian analisa dan pembahasan ini, dijelaskan hasil dari pengolahan data persediaan pada setiap ritel. Data persediaan dianalisis untuk mengetahui stok persediaan pada setiap tingkat Ritel, mengidentifikasi kekurangan dan kelebihan dalam persediaan setiap ritel bahkan biaya yang optimal

III.9 Kesimpulan dan Saran

Hasil dari pengolahan data dan Analisa yang sudah dilakukan, kemudian menyusun kesimpulan dan memberikan saran serta masukan berdasarkan dari hasil penelitian yang sudah dilakukan.

BAB IV Pengumpulan dan Pengolahan Data

IV.1 Pengumpulan Data

Pengumpulan data dilakukan untuk mengumpulkan data-data beserta informasi yang berhubungan dengan topik penelitian. Pengumpulan data ini dilakukan dengan observasi secara langsung di lapangan dengan melakukan wawancara kepada pihak-pihak yang terkait. Proses observasi dilakukan dengan mengamati dan mempelajari secara langsung mengenai hal -hal yang berkaitan dengan permasalahan yang diteliti. Untuk proses pengumpulan data dengan melakukan wawancara dilakukan dengan beberapa pihak terkait, sehingga informasi yang didapatkan sesuai dengan permasalahan yang akan diteliti.

Adapun data-data yang didapatkan dari hasil observasi dan wawancara, yang dibutuhkan dalam penelitian ini adalah data persedian, data permintaan tahun 2022 dan 2023, data Lead Time, biaya pesan gudang pusat dan ritel, biaya penyimpanan produk, dan data harga produk. Data-data tersebut selanjutnya akan diolah untuk mendapatkan saran berdasarkan hasil dari penelitian yang dilakukan.

IV.2 Gambaran Umum Perusahaan

IV.2.1 Sejarah PT.XYZ Hijab Fashion

PT XYZ Fashion didirikan pada tahun 2015 di Bandung oleh sekelompok pengusaha lokal yang berkomitmen untuk menghadirkan produk fashion berkualitas, trendi, dan terjangkau. Terinspirasi dari kebutuhan masyarakat akan busana yang mencerminkan gaya hidup modern namun tetap nyaman dan relevan, para pendiri menyadari bahwa fashion adalah bagian penting dari identitas dan ekspresi diri. Dengan modal awal yang terbatas namun tekad yang kuat, Berawal dari sebuah workshop kecil, PT XYZ kini telah berkembang menjadi salah satu perusahaan fashion terpercaya di Bandung, dengan produk-produk yang telah menjangkau berbagai daerah di Indonesia. PT XYZ dikenal karena komitmennya terhadap desain inovatif, kualitas tinggi, dan keberlanjutan industri fashion lokal. Sejak awal berdiri, PT XYZ mengusung misi:

“Menciptakan Fashion Berkualitas untuk Gaya Hidup yang Lebih Percaya Diri.” Filosofi ini tidak hanya tercermin dalam desain dan bahan produk yang dihasilkan, tetapi juga dalam budaya perusahaan yang berfokus pada kepercayaan, inovasi, dan kepedulian terhadap tren serta keberlanjutan. PT XYZ percaya bahwa fashion yang baik tidak hanya mempercantik penampilan, tapi juga membawa rasa bangga dan percaya diri bagi pemakainya.

Pada tahun 20020, PT XYZ mulai berinvestasi dalam teknologi produksi terbaru dan sistem manajemen desain yang terstandarisasi. Dengan mesin-mesin jahit industri modern dan software desain digital dari Jepang, perusahaan berhasil meningkatkan efisiensi sekaligus mempertahankan detail dan kualitas premium pada setiap produk. PT XYZ kemudian membentuk tim Quality Control untuk memastikan bahwa setiap koleksi yang diluncurkan memenuhi standar pasar domestik dan siap bersaing di kancah internasional. Dengan semakin tingginya permintaan akan produk fashion lokal yang stylish dan berkualitas, pada tahun 2022 PT XYZ mulai memperluas jangkauan distribusi ke berbagai kota besar di Indonesia. Langkah ini disertai dengan peluncuran beberapa lini baru, seperti: fashion muslim modern. Inovasi produk dan ekspansi pasar ini menjadikan PT XYZ sebagai salah satu pionir fashion lokal yang mampu menjawab kebutuhan konsumen modern tanpa meninggalkan akar nilai budaya dan kualitas produksi.

IV.2.2 Profil Perusahaan

PT XYZ Fashion adalah perusahaan yang bergerak di bidang produksi dan distribusi produk fashion berkualitas tinggi. Didirikan pada tahun 2000 di Bandung, PT XYZ hadir untuk menjawab kebutuhan masyarakat Indonesia akan pakaian yang tidak hanya trendi dan nyaman, tetapi juga terjangkau dan berkualitas.

a. Visi

Menjadi perusahaan fashion lokal terbaik dan terpercaya di Indonesia, dengan produk yang berkualitas tinggi, trendi, dan berdampak positif bagi gaya hidup masyarakat serta keberlanjutan lingkungan.

b. Misi

1. Menyediakan produk fashion berkualitas, nyaman, dan stylish yang dapat diakses oleh semua kalangan masyarakat, dengan mengacu pada standar kualitas nasional dan tren global.
2. Menggunakan teknologi produksi dan desain terkini untuk menghasilkan produk dengan presisi tinggi dan mendukung proses produksi yang efisien serta ramah lingkungan.
3. Menjadi pelopor dalam industri fashion lokal dengan terus menghadirkan inovasi desain, kolaborasi kreatif, dan produk-produk yang mengikuti perkembangan gaya hidup dan kebutuhan konsumen.
4. Membangun hubungan yang kuat dan berkelanjutan dengan konsumen, mitra bisnis, dan karyawan melalui transparansi, kepercayaan, dan pelayanan yang berkualitas.

IV.2.3 Data Permintaan

Data permintaan adalah informasi yang berisi catatan jumlah barang atau produk yang diminta (dibutuhkan atau dibeli) oleh konsumen dalam suatu periode waktu tertentu.

IV.2.3.1 Data Permintaan (Demand) Tingkat Gudang Pusat

Data permintaan (Demand) tingkat Gudang pusat diambil berdasarkan data dari gudang pusat ke Supplier selama 12 bulan terakhir (2023 dan 2024) . Berikut ini adalah merupakan data permintaan produk ditingkat gudang pusat.

BULAN	JUMLAH PRODUK/PACK	
	2023	2024
Jan	653	1499
Feb	458	869
Mar	645	634
Apr	317	565
May	852	368

Jun	446	589
Jul	521	543
Aug	577	609
Sep	352	551
Oct	365	563
Nov	578	480
Dec	537	933
TOTAL	6301	8203
Rata-rata Demand	7252	

Tabel IV. 1 Data Permintaan (Demand) Tingkat Gudang Pusat

Sumber: Data Perusahaan PT.XYZ Tahun 2023 dan 2024

IV.2.3.2 Data Permintaan (Demand) Tingkat Ritel

Data permintaan (Demand) ritel diambil berdasarkan data permintaan dari 8 ritel yang tersebar di Kota Bandung kepada gudang pusat selama 24 bulan terakhir (2023 dan 2024) . Berikut ini adalah merupakan data permintaan produk ditingkat gudang pusat.

Ritel	demand		Rata-rata Demand
	2023	2024	
1	849	923	886
2	634	1128	881
3	942	972	957
4	532	968	750
5	556	954	755
6	719	1250	984
7	946	926	936
8	879	1052	965

Tabel IV. 2 Data Permintaan (Demand) Tingkat Ritel

Sumber: Data Perusahaan PT.XYZ Tahun 2023 dan 2024

IV.2.4 Data Lead Time

Lead team merupakan waktu tenggang yang diperlukan untuk melakukan pengadaan produk. Pada PT.XYZ pemesanan dilakukan jika persediaan produk digudang sudah habis atau saat persediaan mencapai titik pemesanan, sama halnya dengan ritel melakukan pemesanan pada Gudang pusat ketika persediaan produk mencapai titik pemesanan.

IV.2.4.1 Data Lead Time Tingkat Gudang Pusat

Data lead team tingkat Gudang pusat merupakan data yang diambil dari waktu setiap pesan produk ke supplier hingga produk ke Gudang pusat. Dilihat dari data 24 bulan (2023 dan 2024) keadaan lead team tidak konstan, maka data tersebut diambil rata rata keseluruhannya sehingga lead team nya yaitu 3 hari.

IV.2.4.2 Data Lead Time Tingkat Ritel

Data Lead Time ditingkat ritel juga merupakan data yang diambil dari waktu setiap pesan produk ke gudang pusat untuk masing-masing ritel dilihat dari 24 bulan terakhir (2023 dan 2024). Setiap ritel memiliki lead time yang berbeda - beda sesuai dengan kebutuhan dari setiap ritelnya. Berikut merupakan rincian data lead time PT.XYZ di setiap ritel.

Ritel	Rata-rata Lead Team (hari)
1	3
2	3
3	3
4	3
5	3
6	3
7	3
8	3

Tabel IV. 3 Lead Time Tingkat Ritel

IV.2.5 Biaya Pemesanan

Biaya pemesanan (ordering cost) adalah seluruh biaya yang timbul setiap kali perusahaan melakukan pemesanan barang kepada pemasok atau melakukan proses pengadaan persediaan. Biaya ini muncul setiap kali terjadi pemesanan.

IV.2.5.1 Biaya Pemesanan Tingkat Gudang Pusat

Biaya pesan gudang pusat diambil dari biaya yang terkait dengan proses pengiriman dan penanganan pesanan dari supplier ke gudang pusat. Dari hasil wawancara dan berdasarkan data perusahaan biaya untuk setiap kali proses pemesanan beragama karena terjadinya dinamika di setiap kali pemesanannya, maka untuk biaya pemesanan diasumsikan dari biaya transportasi dan pengiriman, biaya pemeliharaan dan perbaikan kendaraan, biaya perlengkapan kantor, biaya bahan bakar dan biaya penyusutan kendaraan. Sehingga, biaya pemesanan untuk gudang pusat sebesar Rp150.000 per pesanan. Berikut merupakan rincian dari biaya pesan gudang pusat:

1. Biaya Transportasi dan Pengiriman

Biaya transportasi dan pengiriman tingkat gudang pusat perpesanan mengeluarkan biaya sebesar Rp150.000

2. Biaya Pemeliharaan dan Perbaikan Kendaraan

Biaya pemeliharaan dan perbaikan kendaraan yang dikeluarkan perpesanan yaitu sebesar Rp25.000

3. Biaya Bahan Bakar

Biaya bahan bakar yang digunakan oleh perusahaan untuk akomodasi yaitu sebesar Rp12.000 perpesanan.

4. Biaya Penyusutan Kendaraan

Kendaraan akan mengalami penurunan nilai jual akibat pemakaian, perusahaan memiliki biaya sendiri untuk keadaan ini. Biaya Penyusutan Kendaraan yang dikeluarkan oleh perusahaan per-harinya yaitu Rp18.000.

5. Biaya Pesanan Produk

Biaya ini merupakan biaya diperlukan untuk membeli produk dari supplier. Biaya pesan produk ini rata-rata pertahunnya mencapai Rp 95.000

No	Cost Category	Biaya/Pesan
1	biaya transportasi dan pengiriman	Rp 150.000
2	biaya pemeliharaan dan perbaikan kendaraan	Rp 25.000
3	biaya bahan bakar	Rp 12.000
4	biaya penyusutan mobil	Rp 18.000
5	biaya pesan produk	Rp 95.000
TOTAL		Rp 300.000

Tabel IV. 4 Biaya Pemesanan Tingkat Gudang Pusat

IV.2.5.2 Biaya Pemesanan Tingkat Ritel

Biaya pesan ritel diambil dari biaya yang terkait dengan proses pengiriman dan penanganan pesanan dari gudang pusat ke ritel. Biaya ini terdiri dari biaya transportasi dan pengiriman, biaya pemeliharaan dan perbaikan kendaraan, biaya bahan bakar, biaya penyusutan kendaraan, dan biaya pesan produk. Sehingga, biaya pemesanan di setiap ritel adalah sebesar Rp.65.000.

IV.2.6 Biaya Penyimpanan Produk

Biaya penyimpanan produk diambil dari biaya yang timbul karena adanya produk yang disimpan. Biaya penyimpanan produk dapat dinyatakan dalam dua bentuk, yaitu sebagai persentase dari nilai rata-rata persediaan per tahun dan dalam bentuk rupiah per tahun per unit barang. Pada penelitian ini digunakan bentuk persentase dari biaya penyimpanan. Bentuk persentase biaya penyimpanan ini diambil dan diasumsikan dari biaya modal perusahaan dari bank sehingga biaya penyimpanan ini mengacu pada tingkat suku bunga yang dikeluarkan oleh Bank Rakyat Indonesia (BRI) yaitu sebesar 5.00% di periode tahun 2025. dikeluarkan perusahaan sehingga diperoleh biaya penyimpanan sebesar 20% per tahun. Untuk biaya penyimpanan di setiap ritel diasumsikan sama.

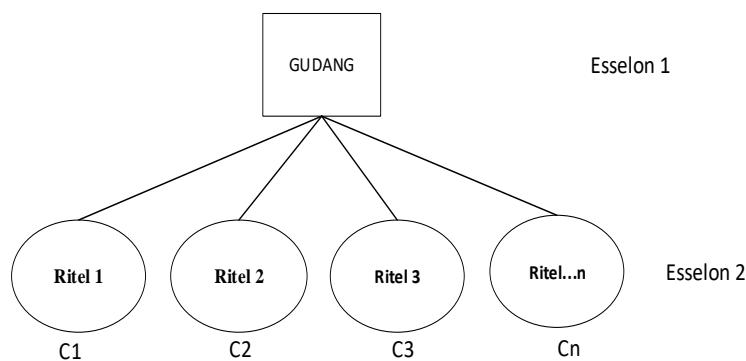
IV.2.7 Data Harga Produk

Data harga produk pada penelitian menggunakan data harga pembelian per produk Stelan Kurta ASH-VN ke supplier dan penjualan produk ke ritel dimana produk dijual satu pes nya 12 PCS. Sehingga, harga pembelian produk ke supplier per pcs- nya yaitu sebesar Rp75.000, sedangkan harga penjualan produk ke ritel per pcs- nya yaitu sebesar Rp95.000

IV.3 Pengolahan Data

IV.3.1 Optimisasi Model Persediaan s,S Menggunakan Particle Swarm Optimization

Permasalahan jaringan distribusi dua eselon di PT.XYZ yang terdiri dari gudang pusat dan ritel. Produk dikirim dari supplier ke gudang pusat kemudian dari gudang pusat produk akan dikirim ke ritel sebagai eselon kedua. Karena permintaan produk dari ritel terjadi secara fluktuasi dengan waktu permintaan yang berbeda setiap periodenya. Sehingga, menjadikan tingkat persediaan yang ada di eselon pertama dan eselon kedua memiliki perbedaan yang cukup jauh, dengan itu tujuan dari model ini adalah untuk meminimalkan jumlah pesan produk, reorder point dan Total Inventory Cost (TIC) di setiap eselon supaya tingkat persediaan yang ada di kedua eselon tidak terlalu jauh



Gambar IV. 1 Jaringan Pasokan Dua Eselon

Pengolahan data pada penelitian ini dilakukan dengan menghitung jumlah persediaan yang ada pada gudang pusat dan ritel. Dengan menggunakan model persediaan s,S pada dua eselon, dimana model persediaan ini dihitung menggunakan proses metaheuristik Particle Swarm Optimization (PSO) di dalam software Matlab sehingga akan

mendapatkan hasil yang paling optimal dari jumlah persediaan dan TIC (Total Inventory Cost).

IV.3.2 Model Persediaan s,S

Langkah awal yang dilakukan adalah melakukan pemodelan persediaan s,S pada dua eselon di software MATLAB. Sebelum dilakukannya optimisasi menggunakan Particle Swarm Optimization (PSO) maka, perlu melakukan pendefinisian fungsi objektif yang akan digunakan dalam proses optimisasi. Permasalahan yang terjadi di penelitian ini terjadi di dua tingkatan atau eselon, eselon pertama yaitu gudang pusat dan eselon kedua yaitu ritel. Maka perlu melakukan pendefinisian secara rinci di setiap eselonnya. Pertama yaitu melakukan pemodelan Function biaya persediaan untuk gudang pusat terlebih dahulu, dimana model perhitungan dan input-an variabel persediaan s,S didefinisikan terlebih dahulu secara rinci sebelum masuk kedalam proses Optimasi. Berikut merupakan deklarasi yang digunakan di software Matlab sebagai alat pendukung penelitian untuk melakukan optimisasi.

```

function
[qw,qrr1,qrr2,qrr3,qrr4,qrr5,qrr6,qrr7,qrr8,sw,sr1,sr2,sr3,sr4,sr5,sr6,sr7,
sr8,biaya] = fungsi_biayapersediaan(partikel_posisi)

%qw;
%qrr1;
%qrr2;
%qrr3;
%qrr4;
%qrr5;
%qrr6;
%qrr7;
%qrr8;
%faktor_1;
%faktor_2;
%faktor_3;
%faktor_4;

```

```
%qw  = partikel_posisi(1);
%qrr1 = partikel_posisi(2);
%qrr2 = partikel_posisi(3);
%qrr3 = partikel_posisi(4);
%qrr4 = partikel_posisi(5);
%qrr5 = partikel_posisi(6);
%qrr6 = partikel_posisi(7);
%qrr7 = partikel_posisi(8);
%qrr8 = partikel_posisi(9);
%faktor_1 = partikel_posisi(10);
%faktor_2 = partikel_posisi(11);
%faktor_3 = partikel_posisi(12);
%faktor_4 = partikel_posisi(13);
```

Gambar IV. 2 Deklarasi Variabel Model s,S

Hasil Pengolahan Data Menggunakan Matlab

Pendeklarasikan fungsi biaya persediaan ini sebagai input fungsi objektif di Particle Swarm Optimization (PSO). Deklarasi variabel yang menggunakan tanda % digunakan untuk *modulo* atau *sisa pembagian*. Dan deklarasi partikel posisif(...) mendefinisikan bahwa variabel tersebut menjadi variabel yang akan di optimisasi oleh PSO. Adapun notasi yang digunakan di dalam Matlab sama dengan notasi persamaan yang sudah dirancang di bab III, akan tetapi ada penyesuaian dalam penulisannya karena keterbatasan sistem. Berikut merupakan notasi yang digunakan di dalam software Matlab

cw :Biaya Satuan Produk Gudang
 cr :Biaya Satuan Produk Ritel
 ccw :Biaya Nilai Tambah Gudang
 ccr :Biaya Nilai Tambah Ritel
 cow :Biaya Pesan Gudang
 cor_n :Biaya Pesan Ritel ke-1 sampai Ritel ke-n
 i :Tingkat Biaya Simpan

dw	:Demand Gudang
dr_n	:Demand Ritel ke-1 sampai Ritel ke-n
qw	:Jumlah Pesan Optimal Gudang
qrr_n	:Jumlah Pesan Optimal Ritel ke-1 sampai Ritel ke-n
iw	:Persediaan Gudang
irr_1	:Persediaan Ritel ke-1 sampai Ritel ke-n
icw	:Tingkat Persediaan Gudang
icr_1	:Tingkat Persediaan Ritel ke-1 sampai Ritel ke-n
L_w	:Lead Time Gudang
L_r	:Lead Time Ritel
s_w	:Reorder Point Gudang
s_r	:Reorder Point Ritel

Setelah dilakukan pendefinisian variabel yang akan digunakan didalam fungsi objektif. Kemudian, melakukan pendefinisian model persediaan s, S yang akan digunakan sebagai fungsi objektif tersebut. Terdapat beberapa perhitungan yang terlibat dalam model persediaan ini. Perhitungan Biaya Nilai Tambah di tingkat gudang pusat dan ritel menjadi awal perhitungan untuk menjadi input di perhitungan selanjutnya. Dengan menggunakan persamaan III.I dan III.5 Berikut merupakan deklarasi perhitungan biaya nilai tambah yang terjadi di tingkat gudang pusat.

```
%Model Persediaan s,S Tingkat Gudang Pusat dan Ritel
%Menghitung biaya nilai tambah di gudang (ccw)
cw = 65000;
cr = 95000;
ccw = cw;
```

Gambar IV. 3 Perhitungan Biaya Nilai Tambah

Hasil Pengolahan Data Menggunakan Matlab

Kemudian, melakukan pendefinisian untuk melakukan perhitungan nilai n disetiap ritelnya dan juga terdapat input nilai i atau tingkat biaya simpan, dimana nilai i ini bersifat konstan. Nilai n perlu diketahui karena akan menjadi fungsi untuk perhitungan Total Inventory Cost (TIC), Persamaan nilai n tersebut menggunakan persamaan III.6 Berikut merupakan deklarasinya.

```
%Menghitung biaya nilai tambah di setiap ritel (ccr)
ccr = cr - cw;
cow = partikel_posisi(1);
cor = partikel_posisi(2);
i = 0.20;
dw = partikel_posisi(3);
dr1 = partikel_posisi(4);
dr2 = partikel_posisi(5);
dr3 = partikel_posisi(6);
dr4 = partikel_posisi(7);
dr5 = partikel_posisi(8);
dr6 = partikel_posisi(9);
dr7 = partikel_posisi(10);
dr8 = partikel_posisi(11);

%Menghitung Nilai n
%Menghitung Nilai n Ritel 1
nr1 = sqrt((cow.*ccr)./(cor.*ccw));

n11 = ((cor+(cow./1)).*(1.*ccw+ccr));
n12 = ((cor+(cow./2)).*(2.*ccw+ccr));

if n11 <= n12
    nr1 = n11;
    nr1 = 1;
else
    nr1 = n12;
    nr1 = 2;
end
```

```
%Menghitung Nilai n Ritel 2
nr2 = sqrt((cow.*ccr)./(cor.*ccw));

n21 = ((cor+(cow./1)).*(1.*ccw+ccr));
n22 = ((cor+(cow./2)).*(2.*ccw+ccr));

if n21 <= n22
    nr2 = n21;
    nr2 = 1;
else
    nr2 = n22;
    nr2 = 2;
end

%Menghitung Nilai n Ritel 3
nr3 = sqrt((cow.*ccr)./(cor.*ccw));

n31 = ((cor +(cow./1)).*(1.*ccw+ccr));
n32 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n31 <= n32
    nr3 = n31;
    nr3 = 1;
else
    nr3 = n32;
    nr3 = 2;
end

%Menghitung Nilai n Ritel 4
nr4 = sqrt((cow.*ccr)./(cor.*ccw));

n41 = ((cor +(cow./1)).*(1.*ccw+ccr));
n42 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n41 <= n42
```

```
nr4 = n41;
nr4 = 1;
else
    nr4 = n42;
    nr4 = 2;
end

%Menghitung Nilai n Ritel 5
nr5 = sqrt((cow.*ccr)./(cor.*ccw));

n51 = ((cor +(cow./1)).*(1.*ccw+ccr));
n52 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n51 <= n52
    nr5 = n51;
    nr5 = 1;
else
    nr5 = n52;
    nr5 = 2;
end

%Menghitung Nilai n Ritel 6
nr6 = sqrt((cow.*ccr)./(cor.*ccw));

n61 = ((cor +(cow./1)).*(1.*ccw+ccr));
n62 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n61 <= n62
    nr6 = n61;
    nr6 = 1;
else
    nr6 = n62;
    nr6 = 2;
end
```

```

%Menghitung Nilai n Ritel 7
nr7 = sqrt((cow.*ccr)./(cor.*ccw));

n71 = ((cor +(cow./1)).*(1.*ccw+ccr));
n72 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n71 <= n72
    nr7 = n71;
    nr7 = 1;
else
    nr7 = n72;
    nr7 = 2;
end

%Menghitung Nilai n Ritel 8
nr8 = sqrt((cow.*ccr)./(cor.*ccw));

n81 = ((cor +(cow./1)).*(1.*ccw+ccr));
n82 = ((cor +(cow./2)).*(2.*ccw+ccr));

if n81 <= n82
    nr8 = n81;
    nr8 = 1;
else
    nr8 = n82;
    nr8 = 2;
end

```

Gambar IV. 4 Deklarasi Nilai n Setiap Ritel

Hasil Pengolahan Data Menggunakan Matlab

Terdapat beberapa variabel yang didefinisikan sebagai partikel_posisi (1 ... n) dimana variabel ini yang akan di random dengan batas bawah dan batas atas yang sudah ditentukan. Dan juga terdapat fungsi if else yang digunakan dalam deklarasi ini untuk mendefinisikan pembatas nilai n yang akan digunakan untuk fungsi TIC.

$$n = n_{11-n}, \text{ jika } f(n_{11-n}) \leq f(n)_{21-n}$$

$$n = n_{21-n}, \text{ jika } f(n_{11-n}) \leq f(n)_{21-n}$$

Setelah mendeklarasikan perhitungan nilai n, kemudian mendeklarasikan perhitungan jumlah pemesanan untuk ritel (qr) dan gudang (qw). Deklarasi ini menggunakan persamaan III.4 dan III.9

```
%Menghitung Jumlah Pemesanan Optimal Untuk Ritel
%Menghitung Jumlah Pemesanan Optimal Ritel 1
qrr1 = sqrt((2.*(cor+(cow./nr1)).*dr1)./(i.*(nr1.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 2
qrr2 = sqrt((2.*(cor+(cow./nr2)).*dr2)./(i.*(nr2.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 3
qrr3 = sqrt((2.*(cor+(cow./nr3)).*dr3)./(i.*(nr3.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 4
qrr4 = sqrt((2.*(cor+(cow./nr4)).*dr4)./(i.*(nr4.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 1
qrr5 = sqrt((2.*(cor+(cow./nr5)).*dr5)./(i.*(nr5.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 6
qrr6 = sqrt((2.*(cor+(cow./nr6)).*dr6)./(i.*(nr6.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 7
qrr7 = sqrt((2.*(cor+(cow./nr7)).*dr7)./(i.*(nr7.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal Ritel 8
qrr8 = sqrt((2.*(cor+(cow./nr8)).*dr8)./(i.*(nr8.*ccw+ccr)));
%Menghitung Jumlah Pemesanan Optimal di Gudang Pusat
qw
=((nr1.*qrr1)+(nr2.*qrr2)+(nr3.*qrr3)+(nr4.*qrr4)+(nr5.*qrr5)+(nr6.*qrr6)+(nr7.*qrr7)+(nr8.*qrr8));
```

Gambar IV. 5 Deklarasi Perhitungan Jumlah Pesan Optimal (Qw dan Qr)

Hasil Pengolahan Data Menggunakan Matlab

Kemudian, mendeklarasikan perhitungan untuk Reorder Point dan tingkat persediaan di tingkat Gudang pusat dan ritel dengan menggunakan persamaan III.5 dan III.10

```
%Menghitung Reorder Point Tingkat Gudang (sg)
dw1 = partikel_posisi(12);
L = 3;
HK = 24;

sw = L.* (qw./HK);

%Menghitung Reorder Point Tingkat Ritel (sr)
%d11 = partikel_posisi(13);
%d12 = partikel_posisi(14);
%d13 = partikel_posisi(15);
%d14 = partikel_posisi(16);
%d15 = partikel_posisi(17);
%d16 = partikel_posisi(18);
%d17 = partikel_posisi(19);
%d18 = partikel_posisi(20);

sr1 = L.* (qrr1./HK);
sr2 = L.* (qrr2./HK);
sr3 = L.* (qrr3./HK);
sr4 = L.* (qrr4./HK);
sr5 = L.* (qrr5./HK);
sr6 = L.* (qrr6./HK);
sr7 = L.* (qrr7./HK);
sr8 = L.* (qrr8./HK);

%Menghitung Persediaan di gudang (icw)
iw = 8205;
irr1 = 923;
irr2 = 1128;
irr3 = 972;
irr4 = 968;
```

```

irr5 = 954;
irr6 = 1250;
irr7 = 926;
irr8 = 1052;

icw = iw + (irr1 + irr2 + irr3 + irr4 + irr5 + irr6 + irr7 + irr8);

%Menghitung Persediaan di ritel (icr)
icr1 = irr1;
icr2 = irr2;
icr3 = irr3;
icr4 = irr4;
icr5 = irr5;
icr6 = irr6;
icr7 = irr7;
icr8 = irr8;

biaya = hitung_tic(icw,icr1,icr2,icr3,icr4,icr5,icr6,icr7,icr8,qw,qrr1,qrr2,qrr3,qrr4,qrr5,qrr6,qrr7,qrr8,cow,cor,ccw,ccr,dw,dr1,dr2,dr3,dr4,dr5,dr6,dr7,dr8,i);

```

Gambar IV. 6 Deklarasi Perhitungan Reorder Point dan Tingkat Persediaan Ritel

Hasil Pengolahan Data Menggunakan Matlab

Di deklarasi terakhir ada biaya-hitung_tic(...) deklarasi ini berfungsi untuk memanggil Function hitung_tic ke Function biayapersediaan yang dimana Function biayapersediaan ini akan yang akan dideklarasikan menjadi fungsi objektif di program optimisasi PSO. Setelah mendeklarasikan semua fungsi perhitungan biaya kemudian melakukan deklarasi untuk perhitungan untuk Total Inventory Cost (TIC). Perhitungan tersebut menggunakan persamaan III.11 Berikut deklarasi di Matlabnya.

```

function tic = hitung_tic
(icw,icr1,icr2,icr3,icr4,icr5,icr6,icr7,icr8,qw,qrr1,qrr2,qrr3,qrr4,qrr5,qrr6,qrr7,qrr8,cow,cor,ccw,ccr,dw,dr1,dr2,dr3,dr4,dr5,dr6,dr7,dr8,i);

faktor_1 = ((cow.*dw)./qw);

```

```

faktor_2 = icw.*i.*ccw;
faktor_3
= ((cor.*dr1)./qrr1)+((cor.*dr2)./qrr2)+((cor.*dr3)./qrr3)+((cor.*dr4)./qrr4)
+((cor.*dr5)./qrr5)+((cor.*dr6)./qrr6)+((cor.*dr7)./qrr7)+((cor.*dr8)./qrr8
);
faktor_4
= (icr1.*i.*ccr)+(icr2.*i.*ccr)+(icr3.*i.*ccr)+(icr4.*i.*ccr)+(icr5.*i.*ccr)+
(icr6.*i.*ccr)+(icr7.*i.*ccr)+(icr8.*i.*ccr);

tic = faktor_1 + faktor_2 + faktor_3 + faktor_4;

```

Gambar IV. 7 Deklarasi Perhitungan Total Inventory Cost

Hasil Pengolahan Data Menggunakan Matlab

Function tic ini digunakan untuk melakukan perhitungan TIC, sama dengan dengan Function sebelumnya perlu dilakukan pendeklarasian variabel yang menjadi input untuk perhitungan TIC ini. Setelah itu, karena perhitungan TIC kompleks untuk mempermudah sistem Matlab membacanya model perhitungan dibagi menjadi 4 faktor penyelesaian yang akhirnya akan ditambahkan setiap faktornya untuk mengetahui hasil akhir TIC. Perhitungan pada faktor_3 dan faktor_4 merupakan perhitungan untuk tingkat ritel, dimana hasil dari setiap ritel dijumlahkan terlebih dahulu kemudian baru dilakukan perhitungan dengan tingkat gudang pusat. Berikut merupakan model matematis yang digunakan dalam menyelesaikan permasalahan TIC terdapat di persamaan III.17.

$$TIC = \frac{C_{ow}D}{Q_w} + I_w^c i C_w^c + \sum_{i=1}^n \left(\frac{C_{ow}D}{Q_r} + I_r^c i C_r^c \right)$$

IV.3.3 Model Particle Swarm Optimization

Selanjutnya Setelah melakukan pendefinisian fungsi objektif, selanjutnya dilakukan pengoptimisasian menggunakan Particle Swarm Optimization (PSO) yang nantikan akan menghasilkan Total Biaya Persediaan yang paling minimum. Berikut merupakan langkah-langkah pengoptimisasian dengan bantuan software Matlab.

1. Inisialisasi parameter awal

Langkah pertama yaitu inisialisasi parameter awal, inisialisasi merupakan proses pengaturan nilai awal untuk variabel yang akan digunakan dalam program. dimana pada proses ini, parameter permasalahan dan parameter PSO di-input. Sebagi pembatas optimisasi. Berikut merupakan parameter awal yang digunakan

```

clc;
clear;
close all;

%% Data

%% Problem Definition

%CostFunction = @(x) Sphere(x);           % Cost Function

nVar = 19;           % Number of Decision Variables

VarSize = [1 nVar];   % Size of Decision Variables Matrix

VarMin = [150000 65000 6000 845 936 774 845 830 722 643 767 826 697 785
720 840 645 655 730];   % Lower Bound of Variables
cow,cor,dw,dr1,dr2,dr3,dr4,dr5,dr6,dr7,dr8,irr1,irr2,irr3,irr4,irr5,irr6,ir
r7,irr8

VarMax = [300000 95000 8205 923 1128 972 968 954 1250 926 1052 849 634 942
532 556 719 946 879];   % Upper Bound of Variables
cow,cor,dw,dr1,dr2,dr3,dr4,dr5,dr6,dr7,dr8,irr1,irr2,irr3,irr4,irr5,irr6,ir
r7,irr8

%% PSO Parameters

MaxIt = 1000;           % Maximum Number of Iterations

nPop=50;           % Population Size (Swarm Size)

```

```
% PSO Parameters
w=1; % Inertia Weight
wdamp=1; % Inertia Weight Damping Ratio
c1=1.3; % Personal Learning Coefficient
c2=1.8; % Global Learning Coefficient
```

Gambar IV. 8 Deklarasi Inisialisasi Parameter Awal

Hasil Pengolahan Data Menggunakan Matlab

Didalam deklarasi Matlab diatas, parameter yang digunakan ada parameter untuk Problem Definition dimana variabel yang akan di random untuk mendapatkan solusi terbaik ada 20 variabel yang didefinisikan sebagai nVar 20; Lalu ada Varsize = [1 nVar]; adalah untuk mendefinisikan ukuran matriks hasil dari variabel, Varmin [...]; dan Varmax [...]; adalah batas bawah dan batas atas dari variabel yang akan dirandomisasi atau nVar.

Kemudian, untuk parameter PSO nya terdapat Maxit atau Iterasi Maksimal yaitu menunjukkan banyaknya iterasi yang akan dilakukan, didalam permasalahan ini Maxlt yang digunakan adalah 1000 karena dapat memberikan hasil yang paling optimal. Parameter nPop adalah untuk mendefinisikan ukuran populasi atau ukuran swarm yang akan diuji, nPop yang digunakan adalah sebanyak 50.

Parameter lain seperti w atau berat inersia, ci dan c2 atau elemen akselerasi. Nilainya didapatkan berdasarkan hasil penelitian (Kakarndee & Manomat, 2020) dan jurnalnya yang berjudul "A Comparatice study on Particle Swarm Optimization and Bat Algorithm for Three Echelon Inventory Model" penelitian ini melakukan perbandingan antara metode Particle Swarm Optimization (PSO) dan Bat Algothrm mengenai permasalahan persediaan di tiga eselon. Adapun parameter yang digunakan dalam penelitian tersebut diantara lain ada ci dan c2 yang digunakan masing-masing 1.3 dan 1.8 untuk nilai berat inersianya (w) sama dengan 1. Dilakukan tiga kali percobaan dengan kondisi iterasi 300, 500 dan 1000 hasil yang ditunjukkan bahwa iterasi 1000 memberikan hasil yang paling optimal untuk permasalahan tersebut. Dengan pengujian parameter yang digunakan dalam penelitian ini berhasil mendapatkan solusi biaya total

yang lebih rendah daripada metode pembandingnya. Sehingga, parameter parameter tersebut dijadikan ketetapan yang digunakan dalam penelitian ini

2. Inisialisasi random position dan velocity

Langkah selanjutnya adalah melakukan inisialisasi secara random untuk position dan velocity. Inisialisasi ini dilakukan untuk memanggil variabel yang akan dijalankan. Dan juga memanggil variabel position dan velocity secara random.

```
%% Initialization

empty_particle.Position=[];
empty_particle.Cost=[];
empty_particle.qw=[];
empty_particle.qrr1=[];
empty_particle.qrr2=[];
empty_particle.qrr3=[];
empty_particle.qrr4=[];
empty_particle.qrr5=[];
empty_particle.qrr6=[];
empty_particle.qrr7=[];
empty_particle.qrr8=[];
empty_particle.sw=[];
empty_particle.sr1=[];
empty_particle.sr2=[];
empty_particle.sr3=[];
empty_particle.sr4=[];
empty_particle.sr5=[];
empty_particle.sr6=[];
empty_particle.sr7=[];
empty_particle.sr8=[];
empty_particle.Velocity=[];
empty_particle.Best.Position=[];
empty_particle.Best.qw=[];
empty_particle.Best.qrr1=[];
empty_particle.Best.qrr2=
```

```

empty_particle.Best.qrr3=[];
empty_particle.Best.qrr4=[];
empty_particle.Best.qrr5=[];
empty_particle.Best.qrr6=[];
empty_particle.Best.qrr7=[];
empty_particle.Best.qrr8=[];
empty_particle.Best.sw=[];
empty_particle.Best.sr1=[];
empty_particle.Best.sr2=[];
empty_particle.Best.sr3=[];
empty_particle.Best.sr4=[];
empty_particle.Best.sr5=[];
empty_particle.Best.sr6=[];
empty_particle.Best.sr7=[];
empty_particle.Best.sr8=[];
empty_particle.Best.Cost=[];

particle = repmat(empty_particle,nPop,1);

GlobalBest.Cost=inf;

```

Gambar IV. 9 Deklarasi Inisialisasi Partikel

Hasil Pengolahan Data Menggunakan Matlab

Deklarasi empty particle position []: ini akan menunjukkan output variabel yang diinginkan dengan posisi dan hasil yang paling optimal. Deklarasi particle- repmat (empty particle, nPop,1) menjelaskan bahwa partikel yang akan menjadi output berdasarkan yang sudah didefinisikan di empty particle dan hasil posisi akan membentuk sebuah matrik yang berdasarkan banyaknya nPop. GlobalBest.Cost inf menjelaskan bahwa posisi terbaik masih bersifat tak terbatas, global best merupakan posisi terbaik dari sekumpulan partikel dalam populasi.

```

for i=1:nPop
    % Initialize Position

    particle(i).Position=unifrnd(VarMin,VarMax,VarSize);

```

```
% Initialize Velocity
particle(i).Velocity=zeros(VarSize);
```

Gambar IV. 10 Deklarasi Randomize Position dan Valecity

Hasil Pengolahan Data Menggunakan Matlab

Deklarasi for $i=1: nPop$ mendefinisikan bahwa inisialisasi secara random dilakukan dimulai dari swarm pertama sampai dengan $nPop$ atau populasi yang sudah ditentukan sebelumnya. Inisialisasi ini dilakukan untuk position dan velocity yang dilakukan secara random terlebih dahulu dengan batasan yang berdasarkan dengan *Varmin*, *Varmax* dan *VarSize*. Inisialisasi ini dilakukan untuk memberikan kecepatan terbang dan posisi yang akan dituju oleh setiap burung untuk mendapatkan posisi yang terbaik. Kedua kondisi tersebut diinisialisasi menggunakan persamaan berikut.

$x = \text{unifrnd}(\text{VarMin}, \text{VarMax}, \text{VarSize})$

Persamaan ini menunjukkan bahwa posisi yang akan dirandom, berdasarkan batasan variabel minimum, variabel maksimum dan ukuran variabelnya. Sedangkan, untuk inisialisasi velocity menggunakan persamaan berikut

$v = \text{zeros}(\text{VarSize})$

Persamaan ini menunjukkan bahwa velocity atau kecepatan terbang swarm dilakukan berdasarkan dengan *VarSize* yang sudah ditentukan di awal.

3. Hitung pBest dan gBest

Langkah selanjutnya yaitu menghitung pBest dan gBest. Pbest ini merupakan hasil solusi atau posisi terbaik untuk individu sedangkan gBest ini merupakan solusi atau posisi individu terbaik pada populasi. Dimana gBest ini yang nantinya akan menjadi solusi optimal.

```
% Evaluation
%particle(i).Cost= CostFunction(particle(i).Position);
[particle(i).qw    particle(i).qrr1    particle(i).qrr2    particle(i).qrr3
particle(i).qrr4    particle(i).qrr5    particle(i).qrr6    particle(i).qrr7
particle(i).qrr8    particle(i).sw     particle(i).sr1     particle(i).sr2
```

```

particle(i).sr3    particle(i).sr4    particle(i).sr5    particle(i).sr6
particle(i).sr7    particle(i).sr8    particle(i).Cost]    =
fungsi_biyapersediaan(particle(i).Position);

% Update Personal Best
particle(i).Best.Position = particle(i).Position;
particle(i).Best.qw    = particle(i).qw;
particle(i).Best.qrr1 = particle(i).qrr1;
particle(i).Best.qrr2 = particle(i).qrr2;
particle(i).Best.qrr3 = particle(i).qrr3;
particle(i).Best.qrr4 = particle(i).qrr4;
particle(i).Best.qrr5 = particle(i).qrr5;
particle(i).Best.qrr6 = particle(i).qrr6;
particle(i).Best.qrr7 = particle(i).qrr7;
particle(i).Best.qrr8 = particle(i).qrr8;
particle(i).Best.sw    = particle(i).sw;
particle(i).Best(sr1  = particle(i).sr1;
particle(i).Best(sr2  = particle(i).sr2;
particle(i).Best(sr3  = particle(i).sr3;
particle(i).Best(sr4  = particle(i).sr4;
particle(i).Best(sr5  = particle(i).sr5;
particle(i).Best(sr6  = particle(i).sr6;
particle(i).Best(sr7  = particle(i).sr7;
particle(i).Best(sr8  = particle(i).sr8;
particle(i).Best.Cost = particle(i).Cost;

% Update Global Best
if particle(i).Best.Cost < GlobalBest.Cost
    GlobalBest=particle(i).Best;
end

end

BestCost = zeros(MaxIt,1);
Best_qw = zeros(MaxIt,1);

```

```

Best_qrr1 = zeros(MaxIt,1);
Best_qrr2 = zeros(MaxIt,1);
Best_qrr3 = zeros(MaxIt,1);
Best_qrr4 = zeros(MaxIt,1);
Best_qrr5 = zeros(MaxIt,1);
Best_qrr6 = zeros(MaxIt,1);
Best_qrr7 = zeros(MaxIt,1);
Best_qrr8 = zeros(MaxIt,1);

Best_sw = zeros(MaxIt,1);
Best_sr1 = zeros(MaxIt,1);
Best_sr2 = zeros(MaxIt,1);
Best_sr3 = zeros(MaxIt,1);
Best_sr4 = zeros(MaxIt,1);
Best_sr5 = zeros(MaxIt,1);
Best_sr6 = zeros(MaxIt,1);
Best_sr7 = zeros(MaxIt,1);
Best_sr8 = zeros(MaxIt,1);

```

Gambar IV. 11 Deklarasi Hitung pBest dan gBest

Hasil Pengolahan Data Menggunakan Matlab

Pada proses ini terlebih melakukan evaluasi fungsi objektif yang akan digunakan, dimana fungsi objektif ini sudah di deklarasikan diawal. Diproses ini fungsi biaya persediaan akan mulai dihitung dan dioptimisasi. Kemudian, setelah melakukan evaluasi terhadap fungsi objektif dilakukan melakukan update terhadap pbest dan gbest di setiap partikelnya sampai hasil cost lebih kecil dari GlobalBest.Cost. Hasil ini masih berdasarkan dengan posisi dan kecepatan yang ditentukan secara random.

4. Update velocity and position pada setiap partikel

Langkah selanjutnya, yaitu melakukan update position dan velocity untuk setiap partikelnya untuk mencari nilai gBest yang konvergen

```

%% PSO Main Loop
for it=1:MaxIt

    for i=1:nPop

        % Update Velocity
        particle(i).Velocity = w.*particle(i).Velocity ...
        +c1*rand(VarSize).*(particle(i).Best.Position-
        particle(i).Position) ...
        +c2*rand(VarSize).*(GlobalBest.Position-particle(i).Position);

        % Apply Velocity Limits
        particle(i).Velocity = max(particle(i).Velocity,VelMin);
        particle(i).Velocity = min(particle(i).Velocity,VelMax);

        % Update Position
        particle(i).Position = particle(i).Position + particle(i).Velocity;

        % Velocity Mirror Effect
        IsOutside=(particle(i).Position<VarMin
        particle(i).Position>VarMax);
        particle(i).Velocity(IsOutside)=-particle(i).Velocity(IsOutside);

        % Apply Position Limits
        particle(i).Position = max(particle(i).Position,VarMin);
        particle(i).Position = min(particle(i).Position,VarMax);
    end
end

```

Gambar IV. 12 Deklarasi Update Valocity dan Position

Hasil Pengolahan Data Menggunakan Matlab

Deklarasi for i=1:MaxIt mendefinisikan bahwa pencarian jumlah pesan optimal, reorder point dan Total Inventory Cost (TIC) akan dilakukan dari iterasi pertama sampai Maksimum iterasi yaitu 1000. Dan for i=1:nPop mendefinisikan bahwa iterasi akan dimulai dari swarm ke-1 sampai swarm ke-10. Kemudian, dilakukan update velocity dan position yang berdasarkan dengan velocity dan position sebelumnya sehingga

partikel akan terbang menuju ke posisi yang baru. Untuk melakukan update velocity terdapat persamaan yang dapat digunakan yaitu sebagai berikut.

$$v_i^t = wv_i^{t-1} + \phi C_1(P_i - X_i^{t-1}) + \phi C_2(g - X_i^{t-1})$$

Didalam Matlab rumus tersebut dideklarasikan sebagai berikut

Particle (i). Velocity = w x particle (i). Velocity + cl x rand (Varsize) x (particle (i) Best Position Particle (i) Position) + c2 x rand (Varsize) x (GlobalBest Position particle (i) Position)

Dan terdapat pembatas update velocity yaitu sebagai berikut

Particle (i). Velocity = max (particle (i).velocity, velmin)

Particle (i). Velocity = min (particle (i).velocity, velmax)

Kemudian, untuk update position menggunakan persamaan sebagai berikut.

$$X_i^t = X_i^{t-1} + v_i^k$$

Didalam Matlab rumus tersebut dideklarasikan sebagai berikut

Particle (i). Position = particle (i).position + particle (i).velocity

Dengan pembatas sebagai berikut.

Particle (i). Position = max (particle (i).velocity, varmin)

Particle (i). Position = min (particle (i). velocity, varmax)

5. Evaluasi fungsi objektif dan fungsi fitness pada partikel

Langkah selanjutnya, karena ada proses update velocity dan position maka perlu dilakukan evaluasi terhadap perhitungan kembali berdasarkan dengan fungsi objektif yang sudah di deklarasikan sebelumnya.

```
% Evaluation
    %particle(i).Cost = CostFunction(particle(i).Position);
    %particle(i).qw particle(i).qrr1 particle(i).qrr2 particle(i).qrr3
    %particle(i).qrr4 particle(i).qrr5 particle(i).qrr6 particle(i).qrr7
```

```

particle(i).qrr8      particle(i).sw      particle(i).sr1      particle(i).sr2
particle(i).sr3      particle(i).sr4      particle(i).sr5      particle(i).sr6
particle(i).sr7      particle(i).sr8      particle(i).Cost]      =
fungsi_biyapersediaan(particle(i).Position);

```

Gambar IV. 13 Deklarasi Evaluasi Fungsi Objektif dan Fitness

Hasil Pengolahan Data Menggunakan Matlab

Selain mengevaluasi fungsi objektif di langkah ini juga menyesuaikan nilai fitness dimana fungsi fitness ini digunakan untuk mengukur tingkat kesesuaian (finess) antara alaternatif solusi yang dicari. Hasil evaluasi ini akan menjadi solusi terbaik.

6. Update Pbest dan Gbest

Langkah selanjutnya melakukan update Pbest dan Gbest setiap partikel. Dimana Pbest dan Gbest ini didapatkan dari partikel yang memiliki nilai fitness terbaik.

```

% Update Personal Best
if particle(i).Cost < particle(i).Best.Cost

    particle(i).Best.Position=particle(i).Position;
    particle(i).Best.qw = particle(i).qw;
    particle(i).Best.qrr1 = particle(i).qrr1;
    particle(i).Best.qrr2 = particle(i).qrr2;
    particle(i).Best.qrr3 = particle(i).qrr3;
    particle(i).Best.qrr4 = particle(i).qrr4;
    particle(i).Best.qrr5 = particle(i).qrr5;
    particle(i).Best.qrr6 = particle(i).qrr6;
    particle(i).Best.qrr7 = particle(i).qrr7;
    particle(i).Best.qrr8 = particle(i).qrr8;
    particle(i).Best.sw = particle(i).sw;
    particle(i).Best.sr1 = particle(i).sr1;
    particle(i).Best.sr2 = particle(i).sr2;
    particle(i).Best.sr3 = particle(i).sr3;
    particle(i).Best.sr4 = particle(i).sr4;
    particle(i).Best.sr5 = particle(i).sr5;

```

```

particle(i).Best.sr6 = particle(i).sr6;
particle(i).Best.sr7 = particle(i).sr7;
particle(i).Best.sr8 = particle(i).sr8;
particle(i).Best.Cost = particle(i).Cost;

% Update Global Best
if particle(i).Best.Cost<GlobalBest.Cost

    GlobalBest=particle(i).Best;

end

end

Best_qw(it) = GlobalBest.qw;
disp(['Iteration ' num2str(it) ': Best qw = ' num2str(Best_qw(it))]);
Best_qrr1(it) = GlobalBest.qrr1;
disp(['Iteration ' num2str(it) ': Best qrr1 = ' num2str(Best_qrr1(it))]);
Best_qrr2(it) = GlobalBest.qrr2;
disp(['Iteration ' num2str(it) ': Best qrr2 = ' num2str(Best_qrr2(it))]);
Best_qrr3(it) = GlobalBest.qrr3;
disp(['Iteration ' num2str(it) ': Best qrr3 = ' num2str(Best_qrr3(it))]);
Best_qrr4(it) = GlobalBest.qrr4;
disp(['Iteration ' num2str(it) ': Best qrr4 = ' num2str(Best_qrr4(it))]);
Best_qrr5(it) = GlobalBest.qrr5;
disp(['Iteration ' num2str(it) ': Best qrr5 = ' num2str(Best_qrr5(it))]);
Best_qrr6(it) = GlobalBest.qrr6;
disp(['Iteration ' num2str(it) ': Best qrr6 = ' num2str(Best_qrr6(it))]);
Best_qrr7(it) = GlobalBest.qrr7;
disp(['Iteration ' num2str(it) ': Best qrr7 = ' num2str(Best_qrr7(it))]);
Best_qrr8(it) = GlobalBest.qrr8;
disp(['Iteration ' num2str(it) ': Best qrr8 = ' num2str(Best_qrr8(it))]);

```

```

Best_sw(it) = GlobalBest.sw;
disp(['Iteration ' num2str(it) ': Best sw = ' num2str(Best_sw(it))]);
Best_sr1(it) = GlobalBest(sr1;
disp(['Iteration ' num2str(it) ': Best sr1 = ' num2str(Best_sr1(it))]);
Best_sr2(it) = GlobalBest(sr2;
disp(['Iteration ' num2str(it) ': Best sr2 = ' num2str(Best_sr2(it))]);
Best_sr3(it) = GlobalBest(sr3;
disp(['Iteration ' num2str(it) ': Best sr3 = ' num2str(Best_sr3(it))]);
Best_sr4(it) = GlobalBest(sr4;
disp(['Iteration ' num2str(it) ': Best sr4 = ' num2str(Best_sr4(it))]);
Best_sr5(it) = GlobalBest(sr5;
disp(['Iteration ' num2str(it) ': Best sr5 = ' num2str(Best_sr5(it))]);
Best_sr6(it) = GlobalBest(sr6;
disp(['Iteration ' num2str(it) ': Best sr6 = ' num2str(Best_sr6(it))]);
Best_sr7(it) = GlobalBest(sr7;
disp(['Iteration ' num2str(it) ': Best sr7 = ' num2str(Best_sr7(it))]);
Best_sr8(it) = GlobalBest(sr8;
disp(['Iteration ' num2str(it) ': Best sr8 = ' num2str(Best_sr8(it))]);

BestCost(it) = GlobalBest.Cost;
disp(['Iteration ' num2str(it) ': Best Cost = ' num2str(BestCost(it))]);

w=w*wdamp;

end

```

Gambar IV. 14 Deklarasi Update pBest dan gBest

Hasil Pengolahan Data Menggunakan Matlab

Untuk melakukan update Pbest dapat menggunakan persamaan sebagai berikut

$$P_i(t+1) = \begin{cases} P_i(t) & \text{jika } f(X_i(t+I)) \geq f(P_i(t)) \\ X_i(t+I) & \text{jika } f(X_i(t+I)) \leq f(P_i(t)) \end{cases}$$

Didalam Matlab persamaan tersebut dideklarasikan dengan *if* particle (i).Cost < particle (i).Best Cost. Sedangkan untuk melakukan update Gbest dapat menggunakan persamaan sebagai berikut.

Didalam

$$F(G(t)) = \min \{f(X_1(t)), \dots, f(X_n(t+m))\}$$

Matlab persamaan tersebut dideklarasikan if particle (1).Best.Cost < GlobalBest. Cost,. Hasil solusi baru Pbest dan Gbest setelah dilakukan update velocity dan position ini akan dibandingkan dengan hasil sebelumnya untuk mendapatkan nilai Pbest dan Gbest terbaik dari q_w, q_{r1-n}, s_w, s_r dan Total Inventory Cost (TIC).

7. Hasil Optimisasi

Kemudian, setelah semua dideklarasikan maka program dilakukan running untuk memunculkan Best Solution dari setiap partikel.

```
%% Results

figure;
%plot(BestCost,'LineWidth',2);
semilogy(BestCost,'LineWidth',2);
xlabel('Iteration');
ylabel('Best Cost');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qw,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qw');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr1,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr1');
grid on;
```

```
figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr2,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr2');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr3,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr3');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr4,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr4');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr5,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr5');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr6,'LineWidth',2);

xlabel('Iteration');
ylabel('Best qrr6');
grid on;
```

```
figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr7,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr7');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_qrr8,'LineWidth',2);
xlabel('Iteration');
ylabel('Best qrr8');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sw,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sw');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr1,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr1');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr2,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr2');
grid on;
```

```
figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr3,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr3');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr4,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr4');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr5,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr5');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr6,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr6');
grid on;

figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr7,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr7');
grid on;
```

```

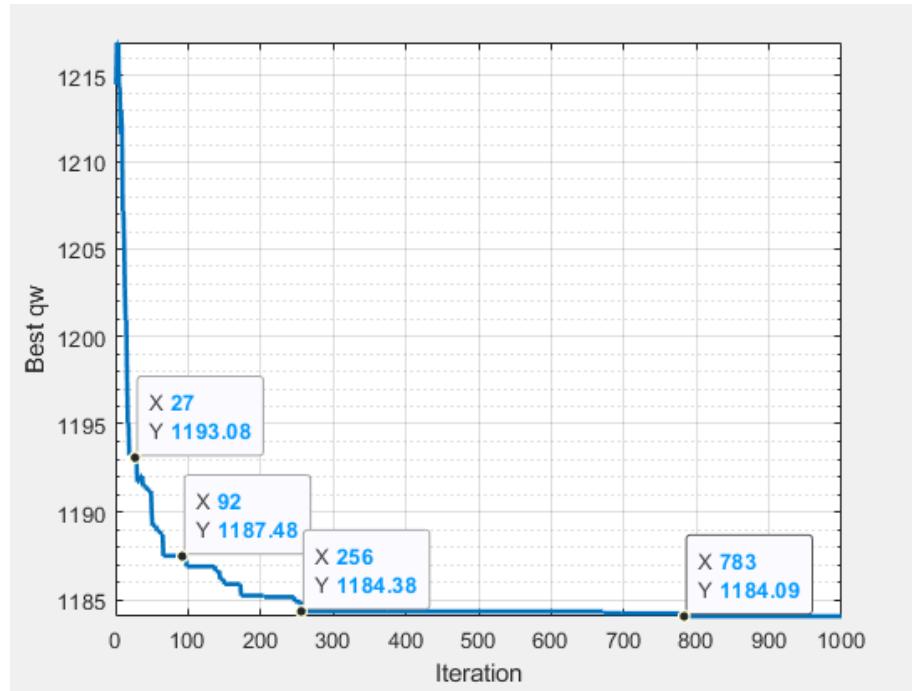
figure;
%plot(BestCost,'LineWidth',2);
semilogy(Best_sr8,'LineWidth',2);
xlabel('Iteration');
ylabel('Best sr8');
grid on;

```

Gambar IV. 15 Deklarasi Reorder Point

Hasil Pengolahan Data Menggunakan Matlab

Dalam langkah ini, Bestsol GlobalBest mendefinisikan bahwa solusi terbaik untuk jumlah pemesanan optimal di gudang (q_w), di ritel (q_r), Reorder Point untuk gudang dan ritel (s) dan biaya total persediaan (TIC) yang dihasilkan oleh PSO. Results yang akan ditampilkan yaitu berupa grafik dimana didalam matlab didefinisikan sebagai figure.

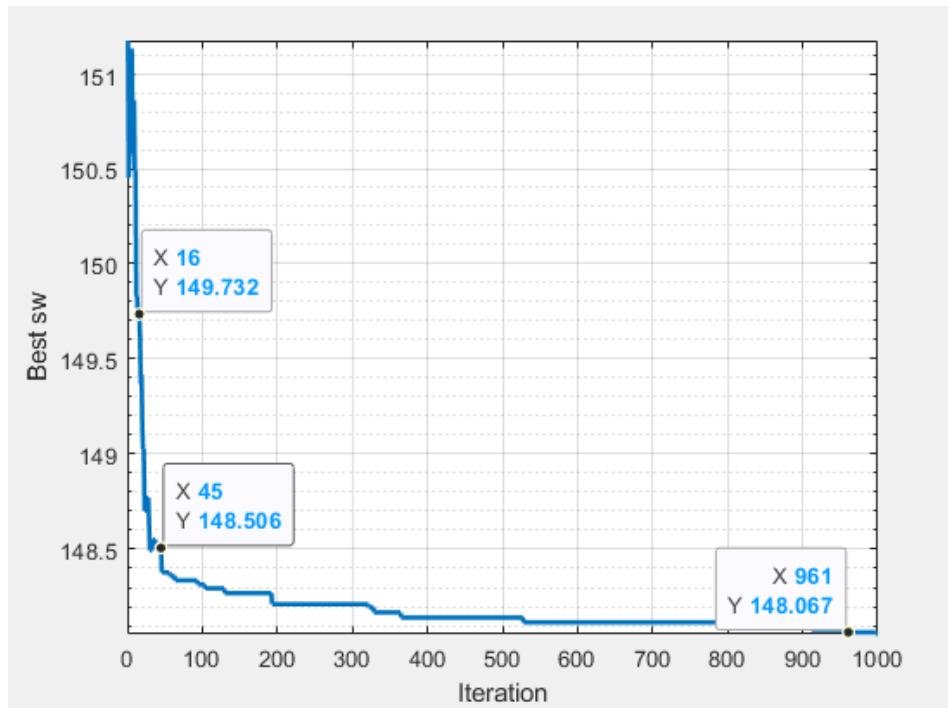


Gambar IV. 16 Grafik Iteration Jumlah Optimal Tingkat Gudang Pusat

Hasil Pengolahan Data Menggunakan Matlab

Grafik di atas menunjukkan posisi dari Jumlah Optimal Tingkat Gudang Pusat

q_w dimana Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi ke posisi yang berbeda-beda dan mencari posisi untuk memberikan solusi terbaik disetiap iterasinya. Untuk mendapatkan solusi optimal Pada awal iterasi (sekitar iterasi ke-0 hingga ke 92), nilai q_w mengalami perubahan yang cukup signifikan. yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. yang menemukan satu titik solusi terbaik yaitu posisi paling minimum, Pada grafik di atas menggambarkan jika line bergerak secara fluktatif dimana di setiap iterasinya partikel-partikel akan kembali untuk mencari posisi yang terbaik, namun karena TIC belum menunjukkan posisi yang optimal maka partikel-partikel akan terus mencari di setiap iterasinya posisi yang benar menunjukkan posisi yang paling optimal untuk variable q_w . Sampai mendapatkan posisinya yang paling optimalnya di iterasi ke-783 dengan nilai 1184,09.



Gambar IV. 17 Grafik Iteration Reorder Point Tingkat Gudang Pusat

Hasil Pengolahan Data Menggunakan Matlab

Grafik di atas menunjukkan posisi dari Reorder Point Tingkat Gudang Pusat s_w dimana Proses ini menandakan bahwa partikel-partikel PSO masih melakukan

eksplorasi ruang solusi ke posisi yang berbeda-beda dan mencari posisi untuk memberikan solusi terbaik disetiap iterasinya. Untuk mendapatkan solusi optimal Pada awal iterasi (sekitar iterasi ke-0 hingga ke 16), nilai s_w mengalami perubahan yang cukup signifikan. yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. yang menemukan satu titik solusi terbaik yaitu posisi paling minimum, Pada grafik di atas menggambarkan jika line bergerak secara fluktatif dimana di setiap iterasinya partikel-partikel akan kembali untuk mencari posisi yang terbaik, namun karena TIC belum menunjukkan posisi yang optimal maka partikel-partikel akan terus mencari di setiap iterasinya posisi yang benar menunjukkan posisi yang paling optimal untuk variable S_w . Sampai mendapatkan posisinya yang paling optimalnya di iterasi ke-961 dengan nilai 148.06.

Gudang	Jumlah Pemesanan Optimal (Qw)	Reorder Point (Sw)
Pusat	1181	145

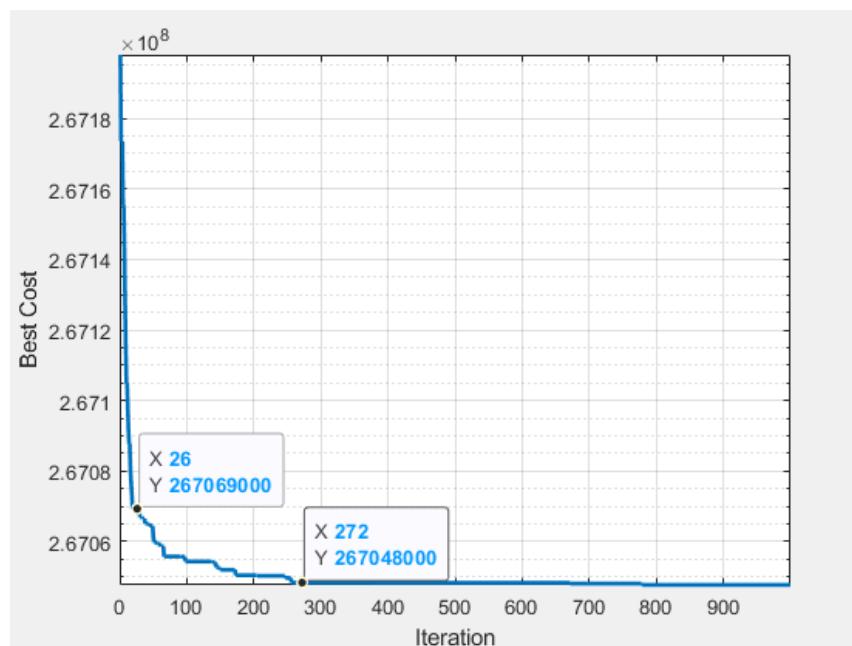
Gambar IV. 18 Hasil Optimisasi PSO di Tingkat Gudang Pusat

Sumber: Hasil Pengolahan Data Menggunakan Matlab

Ritel	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)
1	139	17
2	156	19
3	137	17
4	145	18
5	138	17
6	163	20
7	146	18
8	157	19

Gambar IV. 19 Hasil Optimisasi PSO di Tingkat Ritel

Dari tabel tersebut dapat dilihat bahwa Hasil perhitungan menunjukkan bahwa setiap ritel memiliki nilai jumlah pesanan optimal (Q_r) dan reorder point (S_r) yang berbeda-beda. Nilai Q_r menunjukkan jumlah unit barang yang sebaiknya dipesan setiap kali pemesanan dilakukan agar total biaya persediaan (biaya pesan dan biaya simpan) berada pada tingkat minimum. Dengan demikian, pemesanan dalam jumlah tersebut akan memberikan efisiensi dalam pengelolaan persediaan. Sementara itu, nilai reorder point (S_r) menunjukkan batas atau titik pemesanan kembali, yaitu ketika persediaan telah turun sampai pada jumlah tertentu, perusahaan harus segera melakukan pemesanan ulang agar tidak terjadi kekosongan stok (stockout) selama waktu tunggu. Dari tabel tersebut dapat dilihat bahwa jumlah pesanan optimal yang bervariasi yang menunjukkan adanya perbedaan kebutuhan permintaan pada setiap ritel. Ritel dengan permintaan yang lebih tinggi memiliki Q_r yang lebih besar, hasil ini menunjukkan bahwa semakin besar permintaan (demand) suatu ritel, maka semakin besar pula nilai Q_r dan S_r -nya. Dengan menggunakan hasil optimasi ini, setiap ritel dapat mengatur jadwal pemesanan dengan lebih efisien, meminimalkan biaya simpan, serta menghindari kekurangan stok.



Gambar IV. 20 Grafik Iteration Jumlah Optimal Total Inventory Cost

Sumber: Hasil Pengolahan Data Menggunakan Matlab

Gambar di atas menunjukkan hasil proses optimasi dalam bentuk grafik Jumlah Optimal Total Inventory Cost terhadap Iterasi merepresentasikan jumlah iterasi (0–1000), yang menggambarkan total biaya minimum yang ditemukan selama proses optimasi. Pada awal iterasi, terlihat penurunan nilai Best Cost yang cukup tajam, menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik dalam waktu singkat. Nilai Best Cost mengalami penurunan signifikan hingga iterasi ke-26 dengan nilai sebesar 267.069.000, kemudian kembali menurun hingga iterasi ke-272 dengan nilai 267.048.000. Setelah iterasi tersebut, grafik terlihat stabil tanpa perubahan signifikan hingga akhir proses menandakan bahwa algoritma telah berkonvergensi pada solusi terbaik. Hal ini menandakan bahwa hasil optimasi sudah mencapai titik optimal (baik lokal maupun global) sesuai parameter yang digunakan dalam algoritma. Secara keseluruhan, proses optimasi menunjukkan konvergensi yang cepat dan stabil, dengan nilai akhir Best Cost sebesar 267.048.000.

BAB V Analisa dan Pembahasan

V.1 Analisa

Analisa ini dilakukan berdasarkan dengan hasil pengolahan data yang telah dilakukan pada bab IV yang bertujuan untuk mendapatkan penyelesaian masalah pada permasalahan yang sudah dirumuskan sebelumnya, untuk meningkatkan kinerja persediaan yang terjadi di gudang pusat dan ritel di PT.XYZ. Pengolahan data dalam penelitian ini terlebih dahulu melakukan uji kenormalan pada data demand untuk mengetahui demand tersebut berdistribusi normal atau tidak untuk mengetahui apakah data tersebut dapat diolah dengan perhitungan probabilistik. Kemudian, setelah melakukan uji kenormalan dlanjutkan ke pengolahan data untuk mengetahui jumlah lot optimal di tingkat dua eselon, reorder point dan Total Inventory Cost minimum dengan menggunakan model persediaan s,Q yang diolah dengan metode Metaheuristik Particel Swarm Optimization (PSO).

V.1.1 Analisa Perbandingan Hasil Pengolahan Data

Analisa ini diambil dari hasil pengolahan data optimisasi menggunakan metode Particel Swarm Optimization (PSO) dengan model persediaan s,S di tingkat dua eselon untuk mendapatkan hasil optimal pada jumlah pemesanan di Gudang pusat dan ritel. Setelah dilakukan pengolahan data hasilnya menunjukkan bahwa optimisasi model persediaan s,S dengan metode Particle Swarm Optimization (PSO) memberikan hasil yang lebih optimal dibandingkan dengan pengolahan model persediaan s,S secara manual. Perbedaan hasil antara metode manual dan PSO disebabkan oleh kemampuan algoritma PSO dalam melakukan pencarian solusi global dengan mempertimbangkan berbagai kemungkinan kombinasi nilai variabel keputusan. Sementara metode manual cenderung menghasilkan solusi lokal karena hanya berdasarkan pada rumus dan asumsi tertentu tanpa eksplorasi ruang solusi yang lebih luas.

PT.XYZ memiliki 8 ritel dan juga terdapat gudang pusat berikut merupakan perbandingan jumlah pemesanan optimal tingkat gudang pusat (Q_w), jumlah pemesanan optimal tingkat ritel (Q_r), reorder point tingkat gudang pusat (S_w), dan

reorder point tingkat ritel (Sr) jika di hitung secara manual dan setelah dilakukan optimisasi menggunakan software matlab yang terjadi di gudang pusat ataupun di ritel langkah-langkah perhitungannya ada di lampiran

Berikut adalah Tabel hasil perbandingan antara perhitungan manual dan hasil optimisasi menggunakan metode Particle Swarm Optimization (PSO) menunjukkan adanya perbedaan nilai pada jumlah pemesanan optimal (Qr) dan reorder point (Sr) di setiap ritel. Perbandingan hasil tersebut dapat dilihat pada Tabel 4.4 berikut.

Gudang Pusat	Hasil Optimisasi Matlab PSO		Hasil Perhitungan Manual	
	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)
	1185	145	1223	124

Tabel V. 1 Perbandingan Hasil Perhitungan Manual dan optimisasi Gudang pusat

Ritel	Hasil Optimisasi Matlab PSO		Hasil Perhitungan Manual	
	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)
1	139	17	141	14
2	156	19	156	17
3	137	17	145	15
4	145	18	145	15
5	138	17	144	15
6	163	20	164	19
7	146	18	141	14
8	157	19	151	16

Tabel V. 2 Perbandingan Hasil Perhitungan Manual dan optimisasi Ritel

Hasil perhitungan menghasilkan nilai statistik uji sebesar $t = 0,442$ dengan $p\text{-value} = 0,672$. Nilai $p\text{-value}$ tersebut lebih besar dari tingkat signifikansi 0,05, sehingga dapat

disimpulkan bahwa: Tidak terdapat perbedaan yang signifikan antara Q_r sebelum dan sesudah optimasi

Berdasarkan tabel V.1 perbandingan yang terjadi di gudangpusat dan disetiap ritel bisa dilihat Dari Tabel di atas hasil perbandingan optimisasi antara perhitungan manual dan hasil optimisasi menggunakan metode Particle Swarm Optimization (PSO) menunjukkan nilai optimisasi pada gudang pusat (Q_w dan S_w) dengan hasil optimisasi pada setiap ritel (Q_r dan S_r). kedua tingkat persediaan tersebut saling memengaruhi dalam menjaga ketersediaan stok dan menekan biaya total persediaan (Total Inventory Cost/TIC) secara keseluruhan. Hasil optimisasi gabungan antara gudang pusat dan ritel juga menunjukkan bahwa total biaya persediaan (Total Inventory Cost/TIC) mengalami penurunan signifikan dibandingkan dengan hasil perhitungan manual.

Penurunan TIC terjadi karena PSO mampu mengatur frekuensi pemesanan yang efisien, menyesuaikan jumlah stok optimal di setiap tingkat, serta menjaga tingkat pelayanan agar tidak terjadi kekurangan persediaan. Dengan kata lain, metode PSO tidak hanya mengoptimalkan kebijakan pemesanan secara individual (ritel per ritel), tetapi juga secara terintegrasi antara gudang pusat. Berdasarkan hasil tersebut, dapat disimpulkan bahwa penerapan metode Particle Swarm Optimization (PSO) mampu menciptakan keseimbangan optimal antara gudang pusat dan ritel. Nilai Q_w dan Q_r yang dihasilkan saling menyesuaikan untuk menekan biaya total sistem, sedangkan S_w dan S_r yang meningkat menunjukkan adanya penguatan sistem terhadap ketidakpastian permintaan.

V.2 Pembahasan

Pembahasan ini merupakan penjelasan Berdasarkan hasil pengolahan data yang telah dilakukan pada Bab IV, tujuan dari pembahasan ini adalah untuk memberikan gambaran jelas mengenai penelitian dan hasil yang sudah di dapatkan di PT.XYZ,berdasarkan permasalahan yang terjadi di PT.XYZ mengenai biaya persediaan dan penyimpanan baik di Gudang pusat dan ritel dimana ada perbedaan yang diperoleh jumlah pemesanan optimal (Q_r) dan reorder point (S_r) untuk delapan ritel dengan dua metode, yaitu perhitungan manual dan hasil optimisasi menggunakan

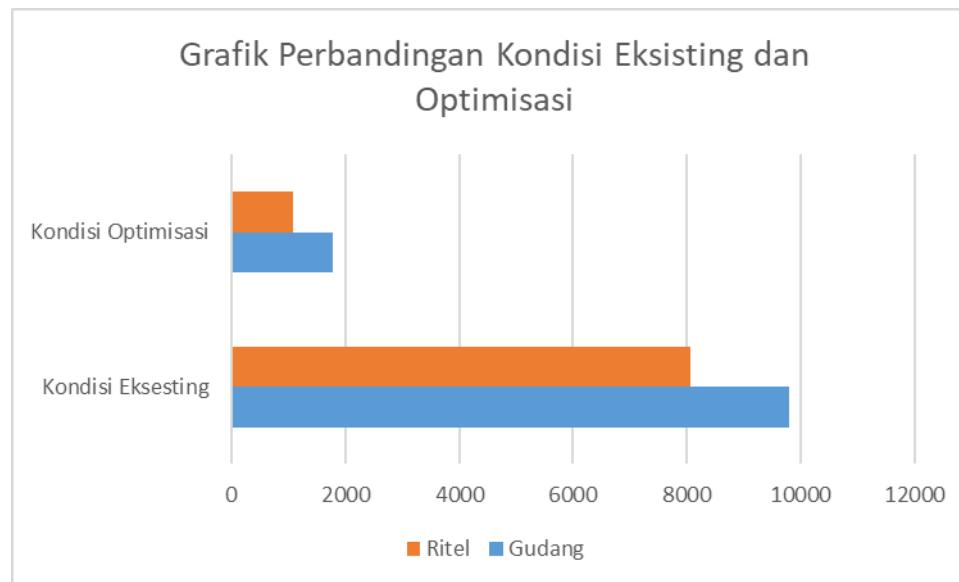
Particle Swarm Optimization (PSO) pada MATLAB. Secara umum, hasil menunjukkan bahwa nilai Q_r dan S_r pada metode PSO memiliki perbedaan kecil dibandingkan hasil perhitungan manual. Nilai Q_r hasil PSO cenderung sedikit lebih tinggi pada beberapa ritel, sedangkan nilai S_r mengalami peningkatan dari hasil manual.

Hal ini menunjukkan bahwa metode PSO mampu menyesuaikan nilai parameter agar mendekati kondisi optimal, di mana keseimbangan antara biaya pemesanan dan biaya simpan tercapai secara lebih efisien. Dengan demikian, PSO dapat memberikan solusi yang lebih tepat untuk menentukan jumlah pemesanan optimal dan titik pemesanan kembali (reorder point) pada setiap ritel. jumlah pemesanan optimal (Q_r) menunjukkan seberapa banyak produk yang harus dipesan oleh masing-masing ritel setiap kali melakukan pemesanan agar biaya total persediaan minimal. Berdasarkan hasil perhitungan, Q_r manual berkisar antara 1187 unit, sedangkan hasil optimisasi PSO berkisar antara 1181 unit. Perbedaan ini relatif kecil, namun cukup signifikan dalam konteks efisiensi biaya. biaya total bisa signifikan karena mempengaruhi frekuensi pemesanan dan biaya simpan secara keseluruhan. Perubahan nilai Q_r pada hasil PSO menunjukkan bahwa metode ini mampu menyeimbangkan hubungan antara biaya pesan dan biaya simpan dengan lebih optimal dibandingkan metode manual konvensional. Titik pemesanan kembali (S_r) menunjukkan jumlah persediaan minimum yang harus dimiliki sebelum dilakukan pemesanan ulang.

Hasil perhitungan menunjukkan bahwa nilai S_r manual berada pada kisaran 125 unit, sedangkan hasil optimisasi PSO meningkat menjadi 145 unit. Peningkatan nilai S_r ini menunjukkan bahwa metode PSO mempertimbangkan variasi permintaan dan waktu tunggu (lead time) yang lebih realistik, sehingga perusahaan atau gudang dapat menghindari risiko stock out (kehabisan stok) saat permintaan meningkat. Dengan adanya penyesuaian nilai S_r ini, sistem persediaan menjadi lebih responsif terhadap fluktuasi permintaan, tanpa harus menambah jumlah pemesanan secara berlebihan. Dari hasil perhitungan yang telah dilakukan, nilai total TIC pada hasil optimisasi PSO menunjukkan penurunan dibandingkan dengan hasil manual. Hal ini membuktikan

bahwa metode PSO mampu menghasilkan kombinasi jumlah pemesanan (Q_r) dan reorder point (S_r) yang meminimalkan biaya total persediaan tahunan. Dengan kata lain, penerapan metode PSO menghasilkan kebijakan persediaan yang lebih efisien, karena mampu menurunkan total biaya yang harus dikeluarkan tanpa mengurangi tingkat pelayanan terhadap ritel. Dari hasil penelitian dan pengolahan data yang sudah dilakukan terdapat usulan perbaikan yaitu dengan memaksimalkan jumlah pemesanan baik dari gudang pusat ke supplier, maupun dari ritel ke gudang pusat.

Dari Hasil penelitian dan pengolahan data yang sudah dilakukan terdapat usulan perbaikan yaitu dengan memaksimalkan jumlah pemesanan baik dari gudang pusat ke supplier, maupun dari ritel ke gudang pusat. Dapat dilihat dari grafik dibawah ini, setelah adanya optimasi tingkat persediaan antara gudang pusat dan ritel berkurang.



Gambar V. 1 Grafik perbandingan Kondisi Eksisting dan Optimasi

Grafik tersebut menampilkan tingkat persediaan ketika kondisi sudah di optimisasi dan kondisi eksisting. Perbedaan persediaan yang terjadi di kondisi optimisasi tidak terlalu berbeda antara gudang dan ritel. Dan juga jumlah persediaannya lebih optimal karena sesuai dengan kebutuhan dari setiap ritel dan gudang hal ini berpengaruh dapat mengurangi biaya yang berlebih atas penyimpanan produk yang terlalu lama ataupun kerusakan produknya.

Karena optimisasi yang dilakukan dengan menggunakan metode Metaheuristik Particle Swarm Optimization dapat memberikan penurunan pada Total Inventory Cost (TIC) yang dikeluarkan perusahaan. Perusahaan dapat menghemat cukup besar dengan menerapkan perhitungan ini. Berikut merupakan tabel hasil perbandingan mengenai kondisi Total Inventory Cost (TIC) setelah dilakukan optimisasi.

No	Kondisi	Total Biaya (Rp)	Selisih terhadap Eksisting (Rp)	Penghematan (%)
1	Eksisting	649.202.000	382.154.000	58,87%
2	Total Inventory Cost (TIC)	267.048.000		

Tabel V. 3 Hasil Perbandingan Total Inventory Cost (TIC)

Berdasarkan hasil analisis, diketahui bahwa total biaya pada kondisi eksisting sebesar Rp 649.202.000. Setelah dilakukan proses optimisasi menggunakan model particle swarm optimization, total biaya menurun menjadi Rp 267.048.000, sehingga diperoleh penghematan sebesar Rp 382.154.000 atau sekitar 58,87% dibandingkan dengan kondisi awal.

Dalam penelitian lainnya yang dilakukan oleh (Ozturk & Senel, 2019) yang berjudul "An Analysis of Multi-item Inventory Model Using Particle Swarm Optimization Under Discrete Delivery Orders and Limited Storage Space" dimana penelitian ini bertujuan untuk menentukan jumlah pesanan optimal dan jumlah pengiriman optimal pada multi-item untuk meminimalkan total biaya dengan membandingkan dua metode optimisasi yaitu Particle Swarm Optimization (PSO) dan Genetic Algorithm (GA). Hasilnya menunjukkan bahwa dengan kondisi jumlah iterasi dan populasi yang sama Particle Swarm Optimization (PSO) dapat memberikan hasil yang lebih baik dengan

waktu yang lebih singkat dari pada Genetic Algorithm (GA). Berdasarkan pertimbangan dari hasil penelitian terdahulu, metode Particle Swarm Optimization (PSO) digunakan untuk menyelesaikan permasalahan persediaan yang ada di PT.XYZ. Hasilnya menunjukkan bahwa PSO bisa memberikan solusi yang optimal

BAB VI Kesimpulan dan Saran

VI.1 Kesimpulan

Adapun kesimpulan berdasarkan hasil analisis dan pembahasan sebelumnya telah dipaparkan mengenai pengolahan data untuk dapat menyelesaikan permasalahan yang terangkum dalam perumusan masalah yang telah di paparkan dengan begitu setelah dilakukan pengolahan data yang sudah dilakukan dengan tujuan untuk mendapatkan solusi jumlah pesan optimal, reorder point dan Total Inventory Cost (TIC) yang dapat mempengaruhi terhadap tingkat persediaan di dalam dua eselon yaitu gudang pusat dan ritel. Dengan menggunakan metode optimisasi Particle Swarm Optimization (PSO) maka didapatkan hasil optimisasi yang memberikan dampak signifikan terhadap penurunan total biaya persediaan. Dan juga didapatkan penurunan kondisi persediaan di Gudang pusat maka didapatkan hasil sebagai berikut :

1. Berdasarkan hasil optimisasi menggunakan PSO didapatkan penurunan kondisi persediaan di gudang pusat sehingga, kondisi persediaan tidak jauh berbeda antara pasokan ke gudang pusat dan penjualan ke ritel. Didapatkan hasil optimal tingkat persediaan untuk gudang pusat adalah sebesar 1181 unit.
2. Terjadi Penghematan Total Inventory Cost (TIC) setelah dilakukan penelitian ini, bahwa penerapan metode optimisasi memberikan dampak signifikan terhadap penurunan total biaya persediaan. Kondisi eksisting menunjukkan total biaya sebesar Rp 649.202.000, sedangkan setelah dilakukan proses optimisasi biaya tersebut menurun menjadi Rp 267.048.000, sehingga terjadi penghematan sebesar 58,87%.

Secara keseluruhan, hasil penelitian ini membuktikan bahwa metode optimisasi particle swarm optimization (PSO), mampu memberikan solusi yang efektif dalam menekan total biaya persediaan. Metode ini dapat dijadikan sebagai alternatif pengambilan keputusan yang lebih efisien dalam menentukan jumlah pemesanan optimal (Order Quantity) dan titik pemesanan ulang (Reorder Point). Dengan

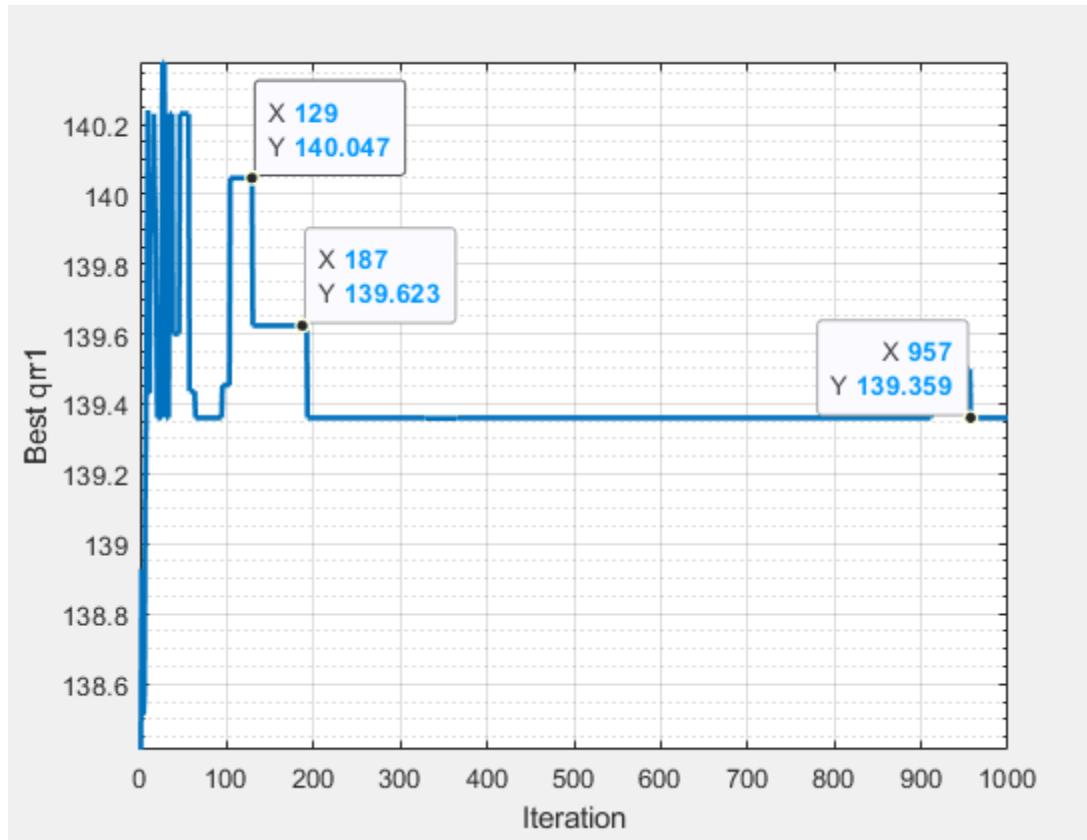
demikian, penerapan model optimisasi dalam sistem pengendalian persediaan terbukti dapat meningkatkan efisiensi operasional dan mengurangi pemborosan biaya.

VI.2 Saran

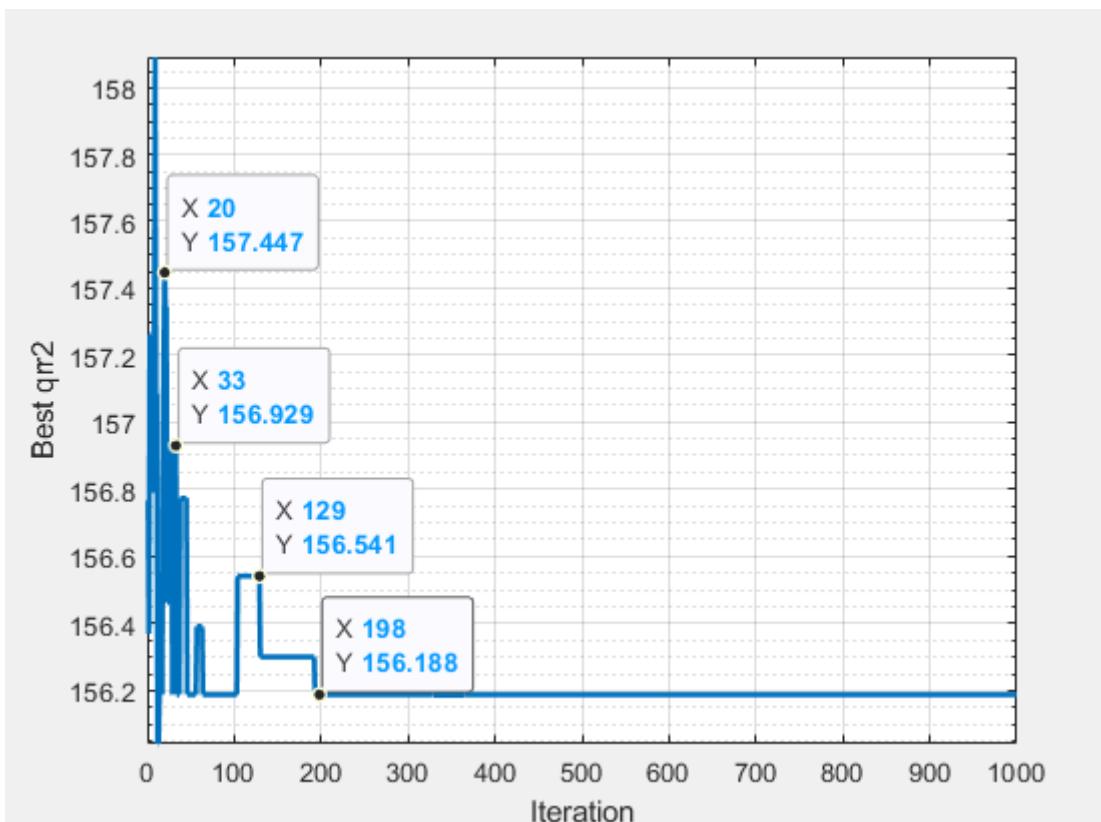
Adapun saran berdasarkan kesimpulan yang sudah dipaparkan sebelumnya, maka penulis selaku yang melakukan penelitian ini Berdasarkan pengolahan data, analisa dan pembahasan yang telah dilakukan. Berikut merupakan saran usulan bagi perusahaan PT.XYZ

1. Bagi pihak perusahaan, disarankan untuk mempertimbangkan penerapan metode optimisasi seperti Particle Swarm Optimization (PSO) dalam sistem pengendalian persediaan. Penggunaan metode ini dapat membantu perusahaan dalam menentukan kebijakan pemesanan yang lebih optimal, sehingga mampu menurunkan total biaya dan meningkatkan efektivitas pengelolaan stok barang.
2. Perusahaan perlu melakukan evaluasi secara berkala terhadap parameter-parameter penting seperti biaya simpan, biaya pesan, dan tingkat permintaan. Hal ini diperlukan agar model optimisasi yang digunakan selalu sesuai dengan kondisi aktual dan dapat memberikan hasil yang optimal secara berkelanjutan.
3. Dengan adanya penelitian lanjutan dan penerapan metode optimisasi yang berkelanjutan, diharapkan sistem pengendalian persediaan dapat berjalan lebih efisien, adaptif terhadap perubahan permintaan, serta mendukung peningkatan produktivitas dan profitabilitas perusahaan secara keseluruhan.

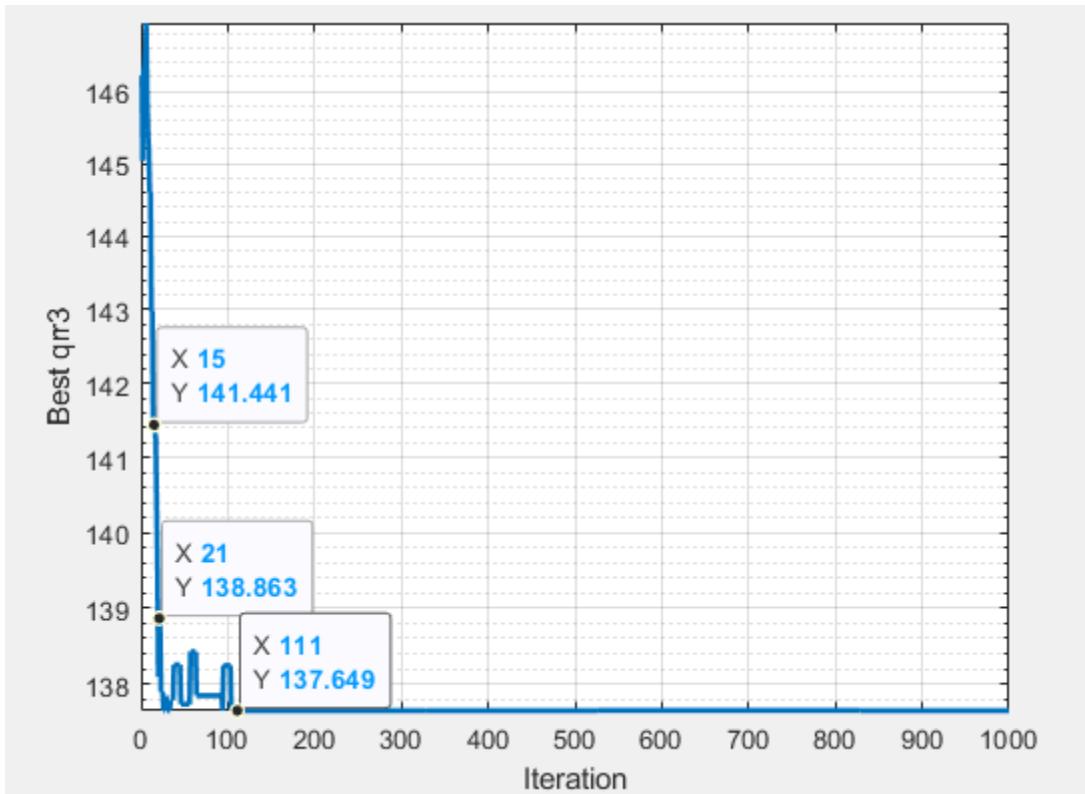
Lampiran 1 Gambar Grafik Hasil Optimasi PSO Jumlah Optimal Disetiap Ritel



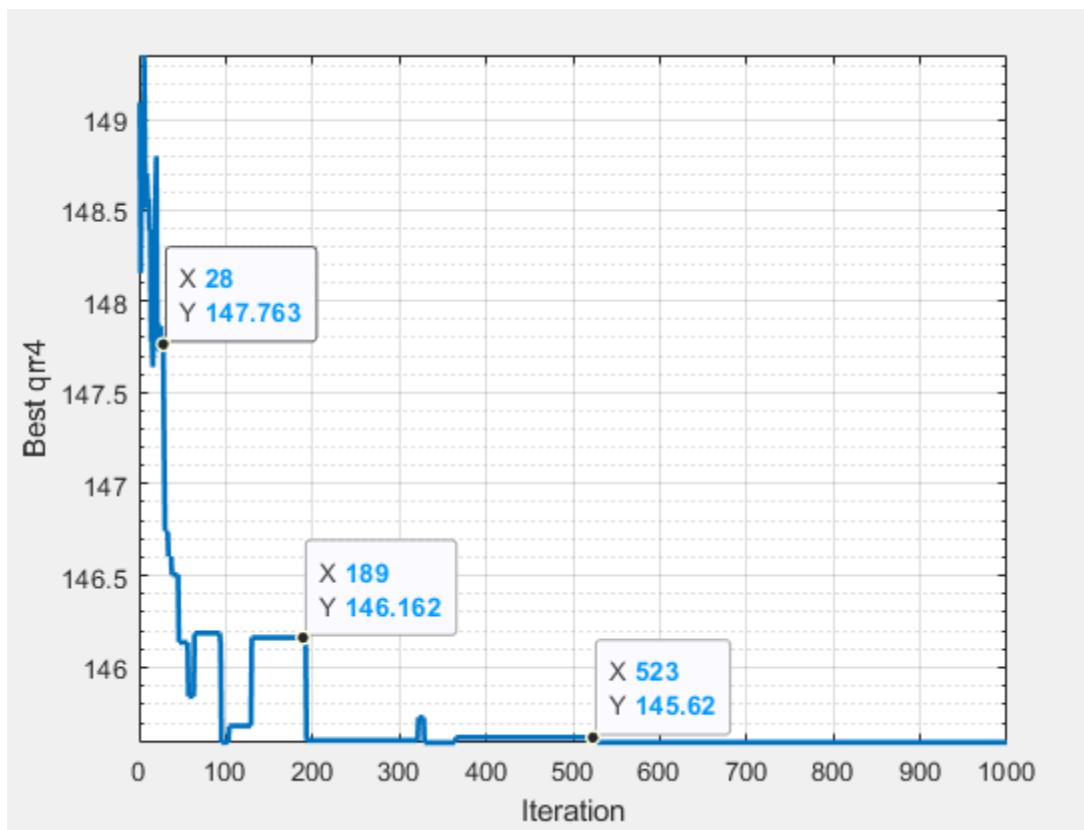
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr1} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-200), nilai q_{rr1} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 140,0 kemudian menurun hingga mencapai 139,3, yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-200, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-957, nilai terbaik akhir q_{rr1} tercatat sebesar 139,35.



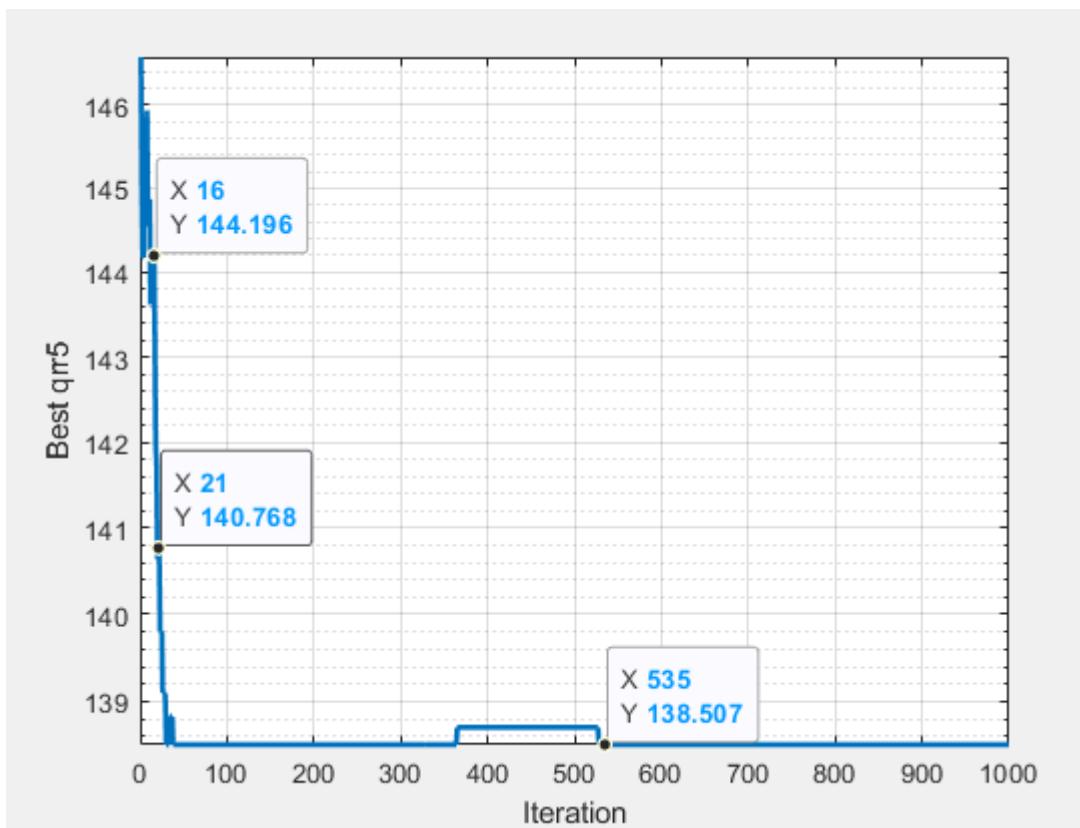
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr2} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-20), nilai q_{rr2} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 157,44 kemudian menurun hingga mencapai 156,92, yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-20, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-198, nilai terbaik akhir q_{rr2} tercatat sebesar 156,18.



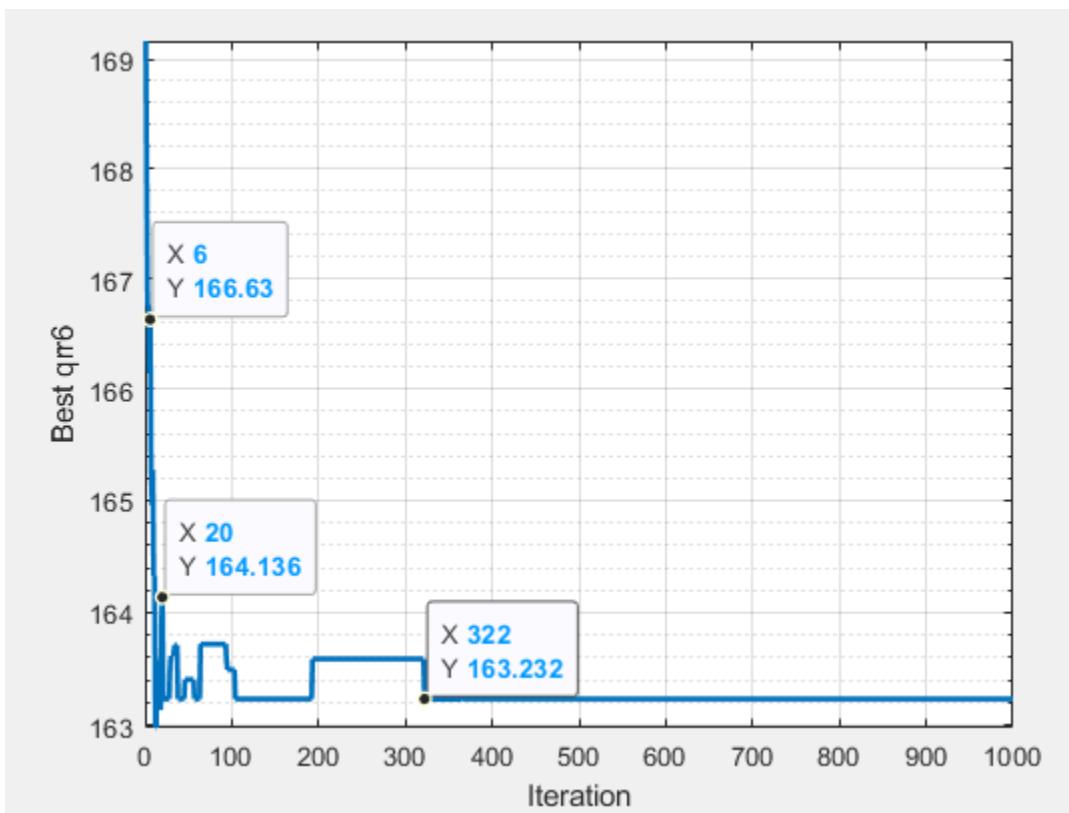
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr3} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-15), nilai q_{rr3} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 141,44 kemudian menurun hingga mencapai 138,86 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-21, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-111, nilai terbaik akhir q_{rr3} tercatat sebesar 137,64.



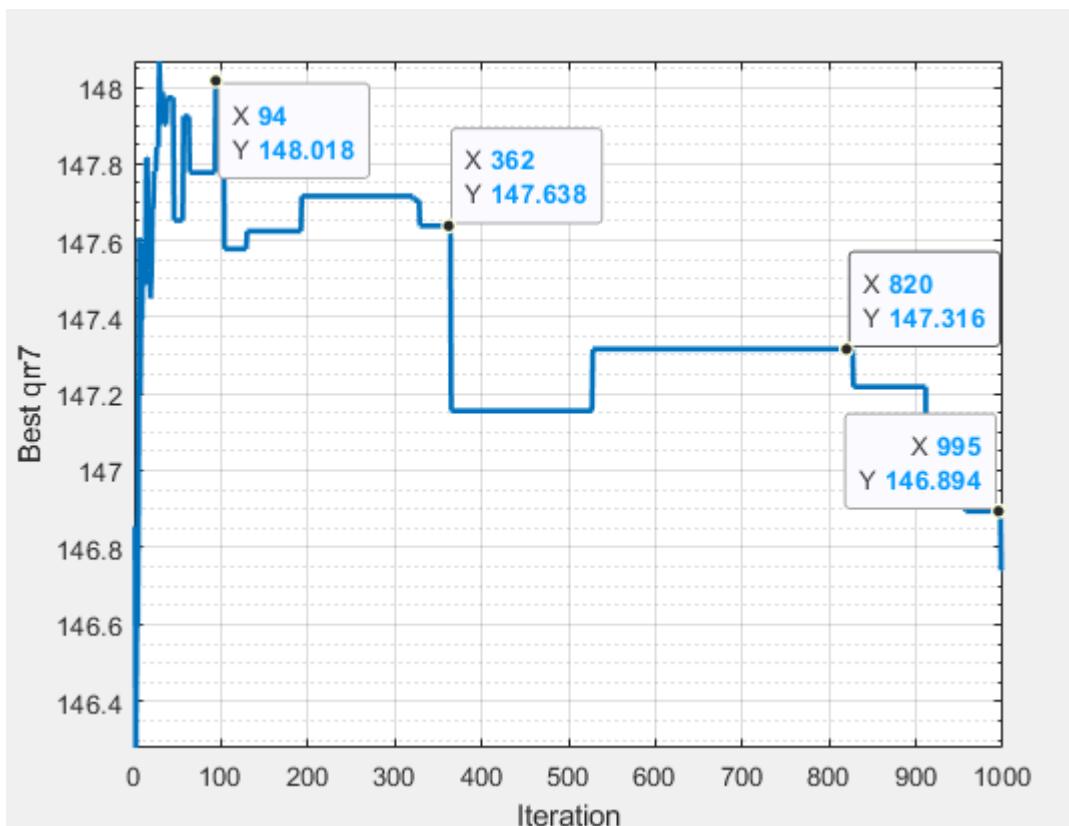
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr4} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-28), nilai q_{rr4} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 147,76 kemudian menurun hingga mencapai 146,16 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-189, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-523, nilai terbaik akhir q_{rr4} tercatat sebesar 145,62.



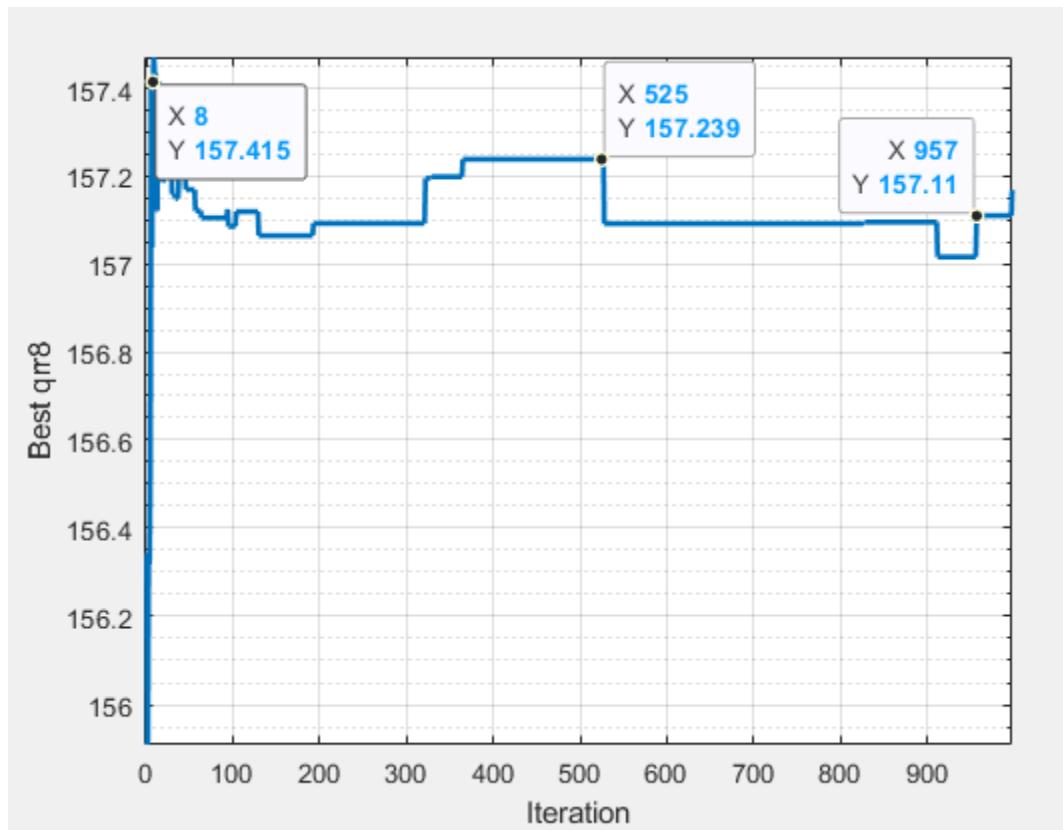
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr5} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-16), nilai q_{rr5} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 144,19 kemudian menurun hingga mencapai 140,76 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-21, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-535, nilai terbaik akhir q_{rr5} tercatat sebesar 138,50.



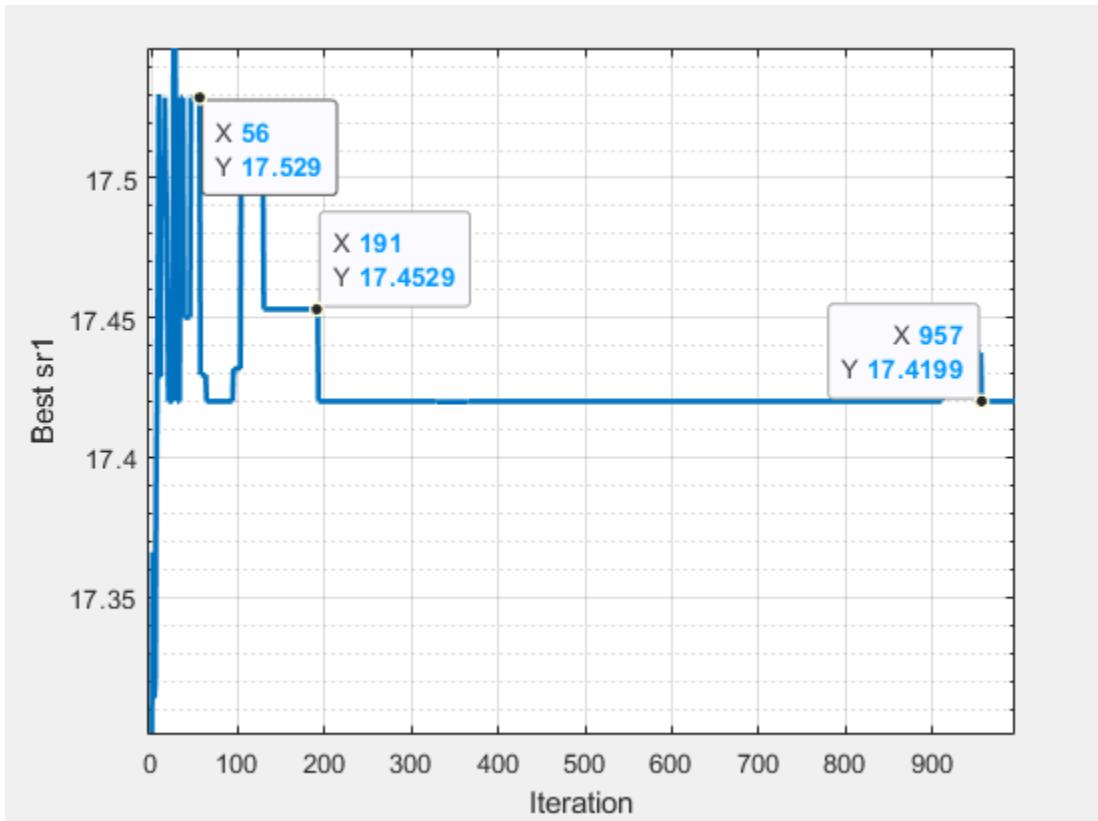
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr6} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-6), nilai q_{rr6} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 166,63 kemudian menurun hingga mencapai 164,13 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-20, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-322, nilai terbaik akhir q_{rr4} tercatat sebesar 163,23.



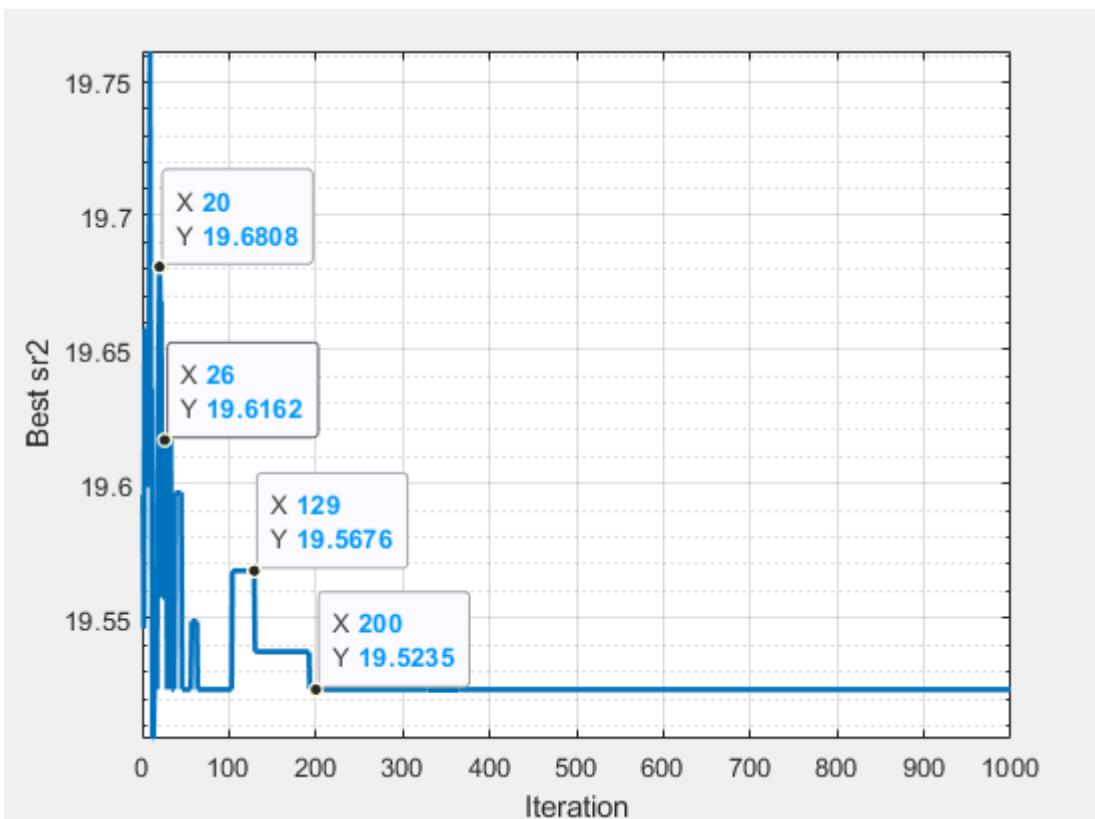
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr7} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-94), nilai q_{rr7} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 148 kemudian menurun hingga mencapai 147,31 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-820, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-995, nilai terbaik akhir q_{rr7} tercatat sebesar 146,89.



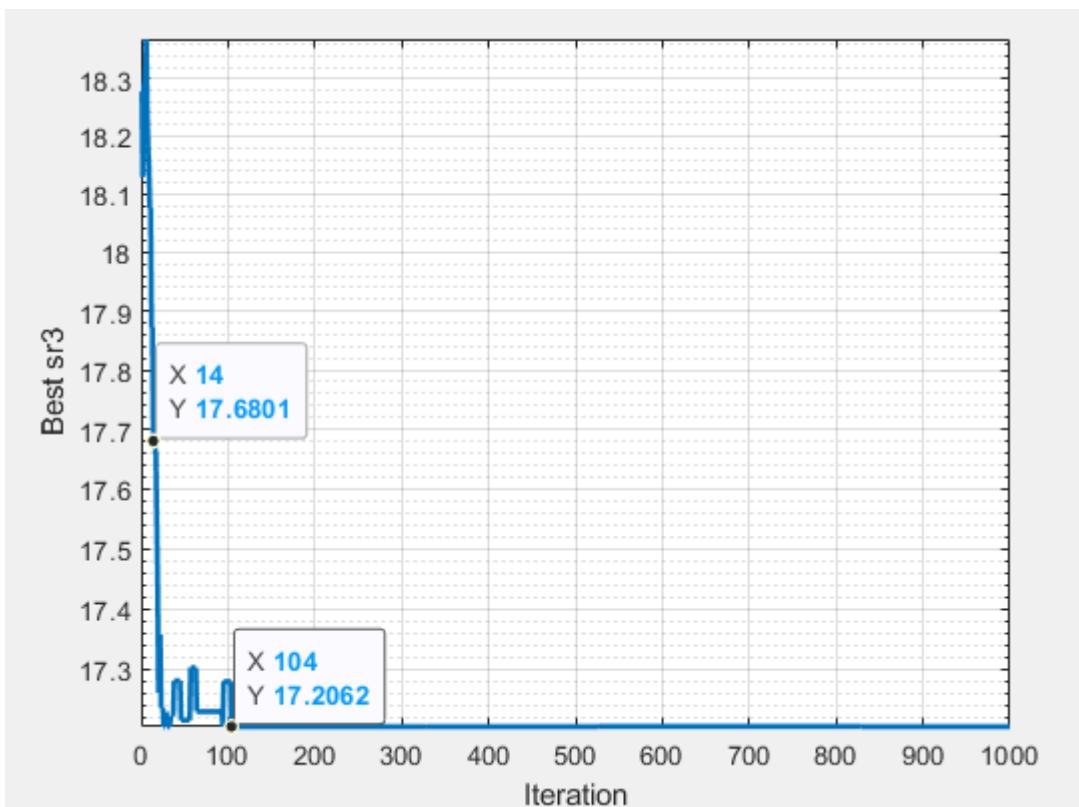
Gambar ini menunjukkan grafik hasil optimasi variabel q_{rr3} dengan metode Particle Swarm Optimization (PSO). Grafik tersebut menggambarkan jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung.). Pada awal iterasi (sekitar iterasi ke-0 hingga ke-8), nilai q_{rr8} mengalami perubahan yang cukup signifikan. Nilai awal sekitar 157,41 kemudian menurun hingga mencapai 157,41 yang menunjukkan bahwa algoritma berhasil menemukan solusi yang lebih baik pada tahap awal. Proses ini menandakan bahwa partikel-partikel PSO masih melakukan eksplorasi ruang solusi untuk mencari posisi optimal. Setelah iterasi ke-525, grafik mulai menunjukkan perubahan yang relatif kecil dan cenderung stabil hingga iterasi terakhir. Hal ini menandakan bahwa proses optimasi telah memasuki tahap konvergensi, di mana nilai terbaik sudah mendekati hasil optimal dan tidak mengalami peningkatan berarti. Pada iterasi ke-957, nilai terbaik akhir q_{rr8} tercatat sebesar 157,11.

Lampiran 2 Gambar Grafik Hasil Optimasi PSO Reorder Point Disetiap Ritel

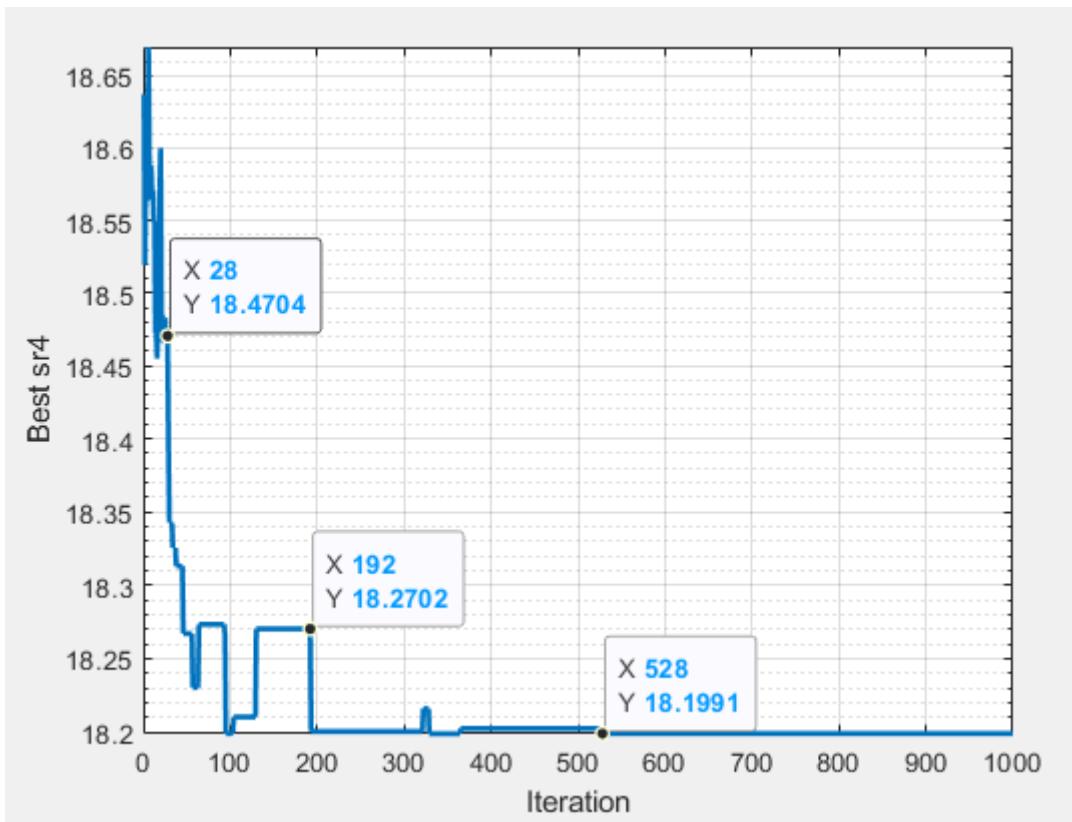
Gambar ini menunjukkan grafik hasil optimasi variabel S_{r1} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r1} dengan cara memperbaiki solusi. S_{r1} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-957 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 17,41.



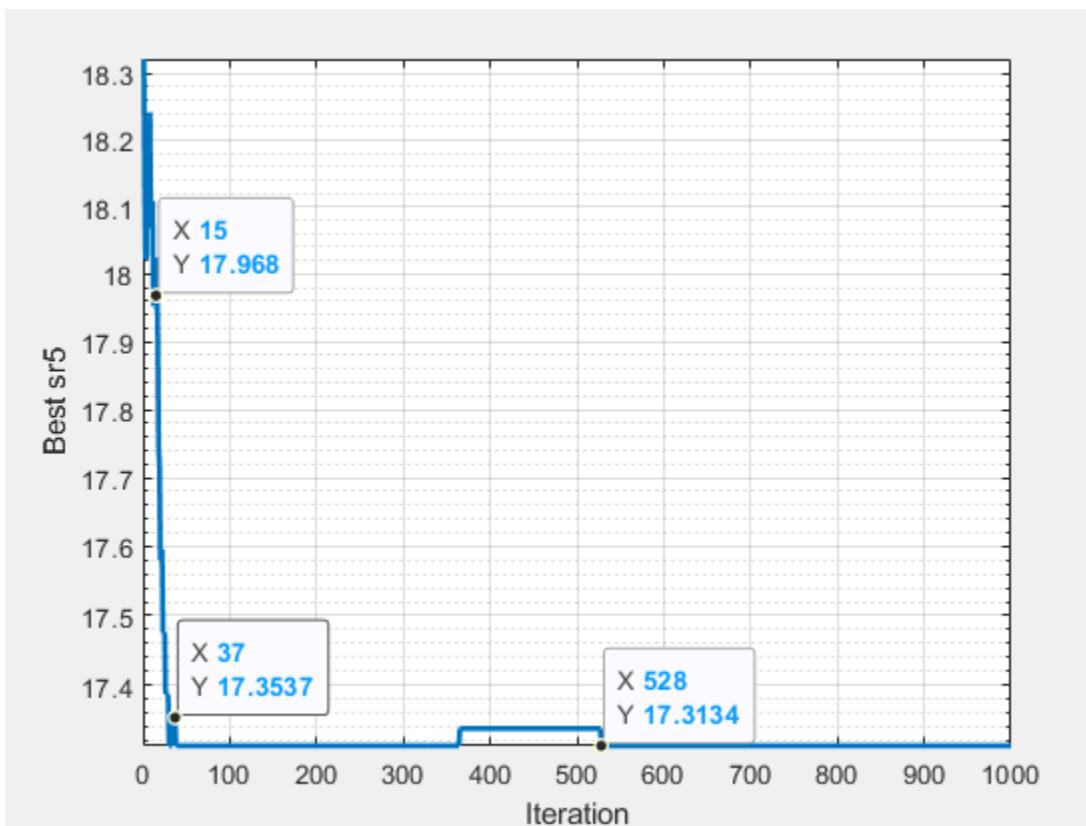
Gambar ini menunjukkan grafik hasil optimasi variabel S_{r2} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r2} dengan cara memperbaiki solusi. S_{r2} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-200 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 19,52.



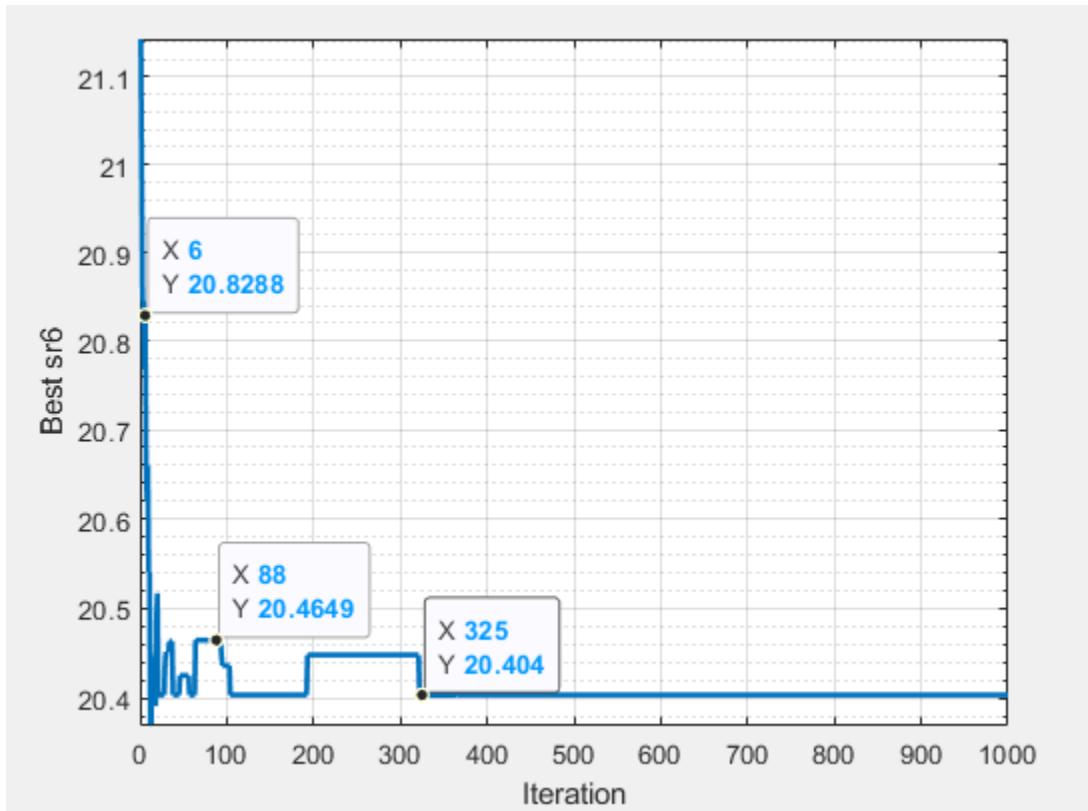
Gambar ini menunjukkan grafik hasil optimasi variabel S_{r3} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r3} dengan cara memperbaiki solusi. S_{r3} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-104 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 17,20.



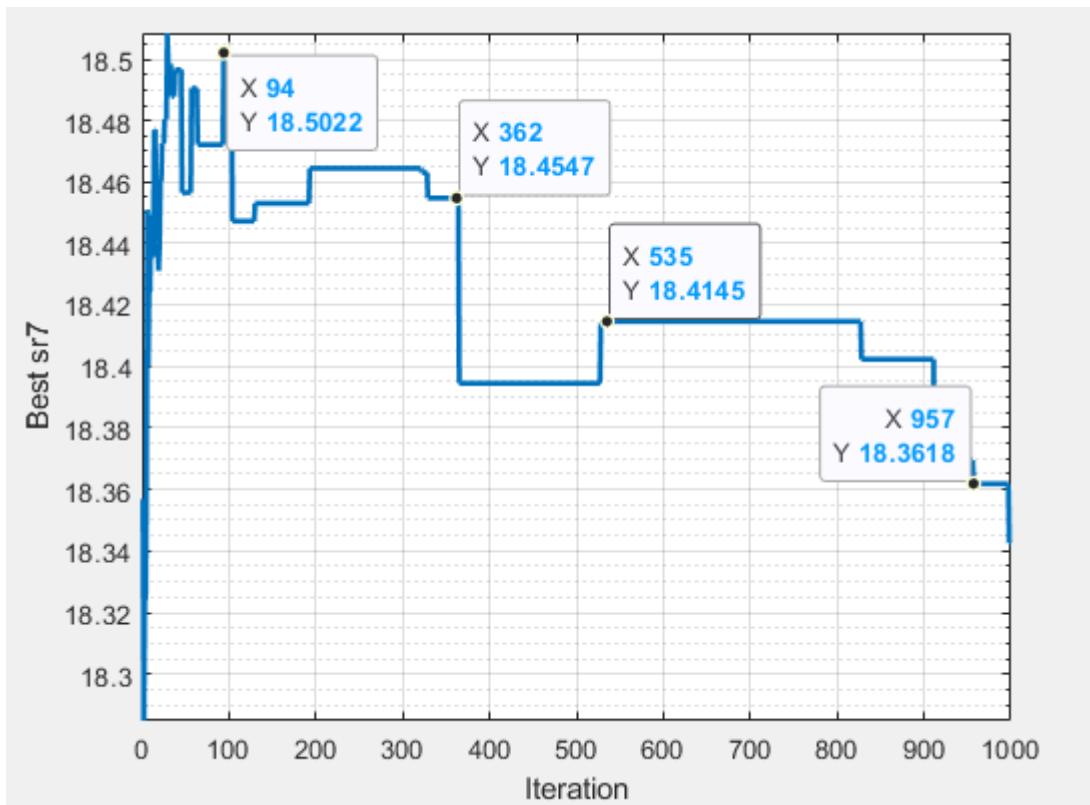
Gambar ini menunjukkan grafik hasil optimasi variabel S_{r4} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r4} dengan cara memperbaiki solusi. S_{r4} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-528 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 18,19.



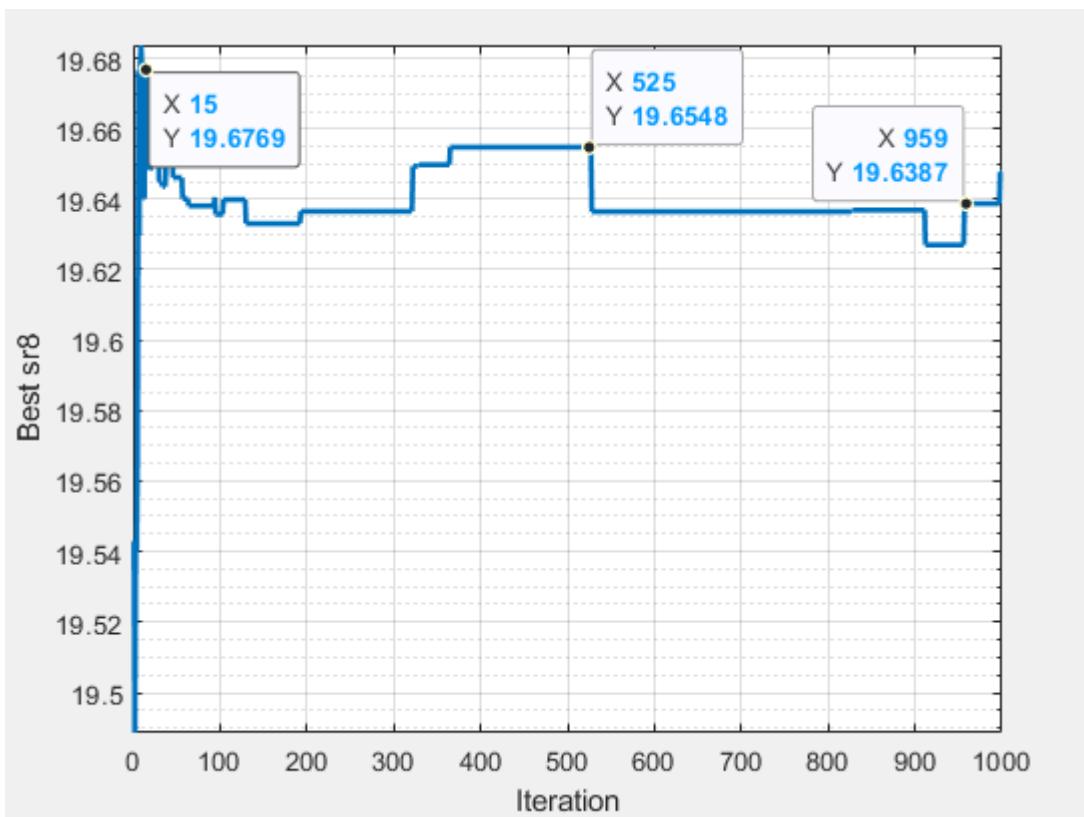
Gambar ini menunjukkan grafik hasil optimasi variabel S_{r4} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r4} dengan cara memperbaiki solusi. S_{r4} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-528 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 17,31.



Gambar ini menunjukkan grafik hasil optimasi variabel S_{r6} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r6} dengan cara memperbaiki solusi. S_{r6} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. Iterasi ke-325 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 20,40.



Gambar ini menunjukkan grafik hasil optimasi variabel S_{r7} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal S_{r7} dengan cara memperbaiki solusi. S_{r7} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-957 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 18,36.



Gambar ini menunjukkan grafik hasil optimasi variabel S_{r8} menggunakan metode Particle Swarm Optimization (PSO). Grafik ini menampilkan hubungan antara jumlah iterasi dengan nilai terbaik (best value) yang diperoleh selama proses optimasi berlangsung. Tujuan dari proses ini adalah untuk menemukan nilai optimal s_{r1} dengan cara memperbaiki solusi. S_{r8} mulai stabil dan tidak menunjukkan perubahan yang signifikan. Hal ini menandakan bahwa proses optimasi telah memasuki fase eksplorasi, ketika partikel-partikel mulai berkumpul di sekitar solusi terbaik yang telah ditemukan. Stabilitas ini menunjukkan bahwa algoritma telah mendekati titik konvergensi atau kondisi optimal. iterasi ke-959 merupakan hasil terbaik (best value) yang dicapai selama proses optimasi dengan nilai 19,63.

Lampiran 3 Perhitungan Manual Jumlah pemesanan optimal gudang

- Jumlah pemesanan optimal gudang adalah jumlah barang yang dipesan oleh gudang pusat kepada pemasok dalam satu kali pemesanan agar total biaya persediaan di tingkat gudang berada pada kondisi minimum.

Rumus untuk menentukan jumlah pemesanan optimal pada Gudang diantaranya :

$$Q_w = \sqrt{\frac{2(C_{or} + \frac{C_{ow}}{n})D}{i(nC_w^c + C_r^c)}}$$

Keterangan :

Q_w : jumlah pemesanan optimal

D : permintaan tahunan

C_{or} : biaya pemesanan di tingkat ritel

C_{ow} : biaya pemesanan di tingkat gudang

n : jumlah ritel yang dilayani Gudang

i : biaya simpan tahunan

C_r^c : biaya simpan per unit di ritel

C_w^c : biaya simpan per unit di gudang

Berikut Perhitungan jumlah optimal di tingkat gudang diantaranya :

$$Q_w = \sqrt{\frac{2(65.000 + \frac{300.000}{8})8205}{0,20(8 \times 95.000 + 75.000)}}$$

$$Q_w = \sqrt{\frac{1682025000}{34.000}}$$

$$Q_w = \sqrt{49472} = 222,42 = 223 \text{ unit}$$

Jadi, pemesanan optimal pada Gudang pusat adalah 223 unit

- Biaya pemesanan gudang

Biaya pemesanan gudang (Warehouse Ordering Cost) adalah biaya yang dikeluarkan oleh pihak gudang pusat setiap kali melakukan pemesanan barang kepada pemasok.

Rumus Untuk menentukan Biaya pemesanan di gudang diantaranya :

$$cow = \left(\frac{D}{Q_w} \times C_w \right) + \left(\frac{Q_w}{2} \times C_w^c \right)$$

Keterangan :

D : Jumlah permintaan tahunan di Ritel

Q_w : Jumlah pemesanan optimal di Ritel

C_w : Biaya pemesanan di Ritel

C_w^c : Biaya simpan di Ritel

Hitungan Untuk menentukan Biaya pesan pada Gudang diantaranya :

$$cow = \left(\frac{8205}{223} \times 90.000 \right) + \left(\frac{223}{2} \times 95.000 \right) = Rp. 12.620.432$$

Jadi, ongkos pesan pada gudang adalah *Rp. 12.620.432*

- Reorder Point (ROP)

Reorder Point (ROP) gudang adalah titik atau batas jumlah persediaan di gudang yang menjadi tanda kapan gudang harus melakukan pemesanan kembali kepada pemasok agar tidak terjadi kekurangan stok (stockout) selama waktu tunggu (lead time)

Rumus untuk menentukan Reorder point di gudang diantaranya :

$$s_w = L \times \frac{Q_w}{Hari Kerja}$$

Keterangan :

s_w : Jumlah persediaan gudang

L : Lead Time

Q_w : Jumlah Pemesanan gudang

Hitungan Untuk Menentukan Reorder Point pada Gudang pusat diantaranya :

$$s_w = 3 \times \frac{8205}{200} = 124 \text{ unit}$$

Jadi, reorder point pada Gudang adalah 124 unit

- Rata-rata persediaan gudang

Rata-rata persediaan adalah jumlah rata-rata barang yang tersedia selama periode tertentu yang digunakan untuk memenuhi kebutuhan disetiap ritel.

Rumus untuk menentukan rata-rata persediaan pada gudang diantaranya :

$$I_r^c = \frac{1}{2} Q_r$$

Keterangan :

I_r^w : rata-rata persediaan di gudang

Q_w : jumlah pemesanan optimal di gudang

Hitungan Untuk menentukan rata-rata persediaan pada Gudang diantaranya :

$$I_r^c = \frac{1}{2} 8205 = 4102,5$$

Jadi, Rata-rata persedian Gudang = 4103 unit

- Jumlah pemesanan optimal ritel

Jumlah pemesanan optimal ritel adalah jumlah barang yang sebaiknya dipesan oleh ritel kepada gudang dalam satu kali pemesanan agar total biaya persediaan di tingkat ritel berada pada kondisi minimum. Dengan kata lain, nilai Q_r menunjukkan kuantitas pemesanan yang paling efisien bagi ritel untuk menyeimbangkan antara biaya pemesanan (ordering cost) dan biaya penyimpanan (holding cost).

Rumus untuk menentukan jumlah pemesanan optimal pada Gudang diantaranya :

$$Q_w = \sqrt{\frac{2 \left(C_{or} + \frac{C_{ow}}{n} \right) D}{i(nC_w^c + C_r^c)}}$$

Keterangan :

Q_w : jumlah pemesanan optimal

D : permintaan tahunan

C_{or} : biaya pemesanan di tingkat ritel

C_{ow} : biaya pemesanan di tingkat gudang

n : jumlah ritel yang dilayani Gudang

i : biaya simpan tahunan

C_r^c : biaya simpan per unit di ritel

C_w^c : biaya simpan per unit di gudang

- Berikut pengolahan data untuk menentukan jumlah pemesanan optimal

Ritel 1

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 923}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{673.790.000}{34.000}}$$

$$Q_r = \sqrt{19818} = 140,77 = 141 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 1 adalah 141 unit

Ritel 2

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 1128}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{823.440.000}{34.000}}$$

$$Q_r = \sqrt{24219} = 155,62 = 156 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 2 adalah 156 unit

Ritel 3

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 972}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{709.560.000}{34.000}}$$

$$Q_r = \sqrt{20870} = 144,46 = 145 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 3 adalah 145 unit

Ritel 4

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 968}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{706.640.000}{34.000}}$$

$$Q_r = \sqrt{20783} = 144,16 = 145 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 4 adalah 145 unit

Ritel 5

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 954}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{696.420.000}{34.000}}$$

$$Q_r = \sqrt{20483} = 143,11 = 144 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 5 adalah 144 unit

Ritel 6

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 1250}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{912.500.000}{34.000}}$$

$$Q_r = \sqrt{26838} = 163,82 = 164 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 6 adalah 164 unit

Ritel 7

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 926}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{675.980.000}{34.000}}$$

$$Q_r = \sqrt{19881} = 141 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 7 adalah 141 unit

$$Q_r = \sqrt{\frac{2 \left(65.000 + \frac{300.000}{1} \right) 1052}{0,20(1 \times 95.000 + 75.000)}}$$

$$Q_r = \sqrt{\frac{767.960.000}{34.000}}$$

$$Q_r = \sqrt{22587} = 151 \text{ unit}$$

Jadi, pemesanan optimal pada ritel 8 adalah 151 unit

- Biaya pemesanan ritel

Biaya pemesanan ritel adalah biaya yang dikeluarkan oleh pihak ritel setiap kali melakukan pemesanan barang kepada gudang pusat atau pemasok.

Rumus Untuk menentukan Biaya pemesanan di gudang diantaranya :

$$cor_1 = \left(\frac{D}{Q_r} x C_r \right) + \left(\frac{Q_r}{2} x C_r^c \right)$$

Keterangan :

D : Jumlah permintaan tahunan

Q_r : Jumlah pemesanan optimal di ritel

C_r : Biaya pemesanan di ritel

C_r^c : Biaya simpan di ritel

- Berikut pengolahan data untuk menentukan Biaya Pesan

Ritel 1

$$cor_1 = \left(\frac{923}{141} x 65.000 \right) + \left(\frac{141}{2} x 75.000 \right) = Rp. 5.712.996$$

Jadi, ongkos pesan pada ritel 1 adalah Rp. 5.712.996

Ritel 2

$$cor_2 = \left(\frac{1128}{156} x 65.000 \right) + \left(\frac{156}{2} x 75.000 \right) = Rp. 6.320.000$$

Jadi, ongkos pesan pada ritel 2 adalah Rp. 6.320.000

Ritel 3

$$cor_3 = \left(\frac{972}{145} x 65.000 \right) + \left(\frac{145}{2} x 75.000 \right) = Rp. 5.873.224$$

Jadi, ongkos pesan pada ritel 2 adalah Rp. 5.873.224

Ritel 4

$$cor_4 = \left(\frac{968}{145} x 65.000 \right) + \left(\frac{145}{2} x 75.000 \right) = Rp. 5.871.431$$

Jadi, ongkos pesan pada ritel 4 adalah Rp. 5.871.431

Ritel 5

$$cor_5 = \left(\frac{954}{144} x 65.000 \right) + \left(\frac{144}{2} x 75.000 \right) = Rp. 5.830.625$$

Jadi, ongkos pesan pada ritel 5 adalah *Rp. 5.830.625*

Ritel 6

$$cor_6 = \left(\frac{1250}{164} x 65.000 \right) + \left(\frac{164}{2} x 75.000 \right) = Rp. 6.645.426$$

Jadi, ongkos pesan pada ritel 6 adalah *Rp. 6.645.426*

Ritel 7

$$cor_7 = \left(\frac{926}{141} x 65.000 \right) + \left(\frac{141}{2} x 75.000 \right) = Rp. 5.714.379$$

Jadi, ongkos pesan pada ritel 7 adalah *Rp. 5.714.379*

Ritel 8

$$cor_8 = \left(\frac{1052}{151} x 65.000 \right) + \left(\frac{151}{2} x 75.000 \right) = Rp. 6.115.347$$

Jadi, ongkos pesan pada ritel 8 adalah *Rp. 6.115.347*

- Reorder point pada setiap ritel ritel

Reorder Point ritel adalah titik atau batas jumlah persediaan di tingkat ritel yang menandakan kapan ritel harus melakukan pemesanan ulang kepada gudang agar tidak terjadi kehabisan stok (stockout) selama menunggu kedatangan barang yang baru

Rumus untuk menentukan Reorder point pada setiap ritel diantaranya :

$$s_r = L \times \frac{Q_r}{Hari\ Kerja}$$

Keterangan :

s_r : Jumlah persediaan di ritel

L : Lead Time

Q_r : Jumlah Pemesanan di ritel

Ritel 1

$$s_r = 3 \times \frac{923}{200} = 13,84 \text{ unit}$$

Jadi, reorder point pada ritel 1 adalah 14 unit

Ritel 2

$$s_r = 3 \times \frac{1128}{200} = 16,92 \text{ unit}$$

Jadi, Reorder point pada ritel ke 2 adalah 17 unit

Ritel 3

$$s_r = 3 \times \frac{972}{200} = 14,58 \text{ unit}$$

Jadi, Reorder point pada ritel ke 3 adalah 15 unit

Ritel 4

$$s_r = 3 \times \frac{968}{200} = 14,52 \text{ unit}$$

Reorder point pada ritel ke 4 adalah 15 unit

Ritel 5

$$s_r = 3 \times \frac{954}{200} = 14,31 \text{ unit}$$

Jadi, Reorder point pada ritel ke 5 adalah 8 unit

Ritel 6

$$s_r = 3 \times \frac{1250}{200} = 18,75 \text{ unit}$$

Jadi, Reorder point pada ritel ke 6 adalah 19 unit

Ritel 7

$$s_r = 3 \times \frac{926}{200} = 13,89 \text{ unit}$$

Jadi, Reorder point pada ritel ke 7 adalah 14 unit

Ritel 8

$$s_r = 3 \times \frac{1052}{200} = 15,78 \text{ unit}$$

Jadi, Reorder point pada ritel ke 8 adalah 16 unit

- Rata-rata persediaan di setiap ritel

Rata-rata persediaan ritel adalah jumlah rata-rata barang yang tersedia di ritel untuk memenuhi permintaan pelanggan.

Rumus untuk menentukan rata-rata persediaan pada ritel diantaranya :

$$I_r^c = \frac{1}{2} Q_r$$

Keterangan :

I_r^c : rata-rata persediaan di gudang

Q_r : jumlah pemesanan optimal di gudang

Ritel 1

$$I_r^c = \frac{1}{2} 923 = 462 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 1 adalah = 462 unit

Ritel 2

$$I_r^c = \frac{1}{2} 1128 = 564 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 2 adalah = 564 unit

Ritel 3

$$I_r^c = \frac{1}{2} 972 = 486 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 3 adalah = 486 unit

Ritel 4

$$I_r^c = \frac{1}{2} 968 = 474 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 4 adalah = 474 unit

Ritel 5

$$I_r^c = \frac{1}{2} 954 = 477 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 5 adalah = 477 unit

Ritel 6

$$I_r^c = \frac{1}{2} 1250 = 625 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 6 adalah = 625 unit

Ritel 7

$$I_r^c = \frac{1}{2} 926 = 463 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 7 adalah = 463 unit

Ritel 8

$$I_r^c = \frac{1}{2} 1052 = 526 \text{ unit}$$

Jadi, Rata-rata persedian pada Ritel 8 adalah = 526 unit

Total Jumlah Total Inventory Cost TIC diantaranya :

$$TIC = \frac{C_{ow}D}{Q_w} + I_w^c i C_w^c + \sum_{i=1}^n \left(\frac{C_{or}D}{Q_r} + I_r^c i C_r^c \right)$$

Keterangan :

Q_w : jumlah pemesanan optimal

D : permintaan tahunan

C_{or} : biaya pemesanan di tingkat ritel

C_{ow} : biaya pemesanan di tingkat gudang

n : jumlah ritel yang dilayani Gudang

i : biaya simpan tahunan

I_w^c : rata-rata gudang

I_r^c : rata rata ritel

C_r^c : biaya simpan per unit di ritel

C_w^c : biaya simpan per unit di gudang

Perhitungan Total Inventory Cost (TIC) diantaranya sebagai beriku :

$$TIC = \frac{300.000 X 8205}{223} + 4102,5 X 0,20 X 95.000 = Rp. 88.995.117$$

$$\sum_{i=1}^n \left(\frac{520.000 X 8173}{149} + 4087 X 0,20 X 75.00 \right) = Rp. 89.828.222$$

Jadi Total Inventory Cost (TIC) = Rp. 178.823.339 + Rp. 48.083.428

$$= Rp.226.906.767$$

Tabel hasil perbandingan antara perhitungan manual dan hasil optimisasi menggunakan metode Particle Swarm Optimization (PSO) menunjukkan adanya perbedaan nilai pada jumlah pemesanan optimal (Qr) dan reorder point (Sr) di setiap ritel. Perbandingan hasil tersebut dapat dilihat pada Tabel berikut ini

Ritel	Hasil Perhitungan Manual		Hasil Optimisasi Matlab PSO	
	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)	Jumlah Pemesanan Optimal (Qr)	Reorder Point (Sr)
1	141	14	139	17
2	156	17	156	19
3	145	15	137	17
4	145	15	145	18
5	144	15	138	17
6	164	19	163	20
7	141	14	146	18
8	151	16	157	19

DAFTAR PUSTAKA

- Chopra, S. and Meindl, P. (2007) "Supply chain management: strategy, planning, and Operation." Upper Saddle River: Pearson Prentice Hall: 3.
- Tan K.C, Lyman S.B, Wisner J.D. (2002), "Supply chain management: A strategic perspective", International Journal of Operations and Production Management, 22, 6, 614-631
- Neeraj Gupta (2012), "A Write Up on Valye Chain Analysis" International Jurnal of Project Management, 1:7
- Christopher, M. (2005) Logistics and Supply Chain Management. Pearson Education, London.
- Van der Vorst JGAJ. 2004. Supoly Chain Management: theory and practices. The Emerging World Of Chains & Networks, Elsevier. Hoofdstuk 2.1.
- Kaipia, Riikka, Jan Holmström, and Kari Tanskanen. "VMI: What are you losing if you let your customer place orders?" Production Planning & Control 13.1 (2002): 17-25.
- Kaipia, R., Holmström, J., & Tanskanen, K. (2002). VMI: What are you losing if you let your customer place orders?. Production Planning & Control, 13(1), 17-25.
- KAIPIA, Riikka; HOLMSTRÖM, Jan; TANSKANEN, Kari. VMI: What are you losing if you let your customer place orders?. Production Planning & Control, 2002, 13.1: 17-25.
- Simchi-Levi, D. – Kaminsky, P. – Simchi-Levi, E. (2008) Designing and Managing the Supply Chain, Concepts, Strategies & Case Studies, 3rd ed. McGraw-Hill: New York.

- Nya, D. N., & Abouaïssa, H. (2023). A robust inventory management in dynamic supply chains using an adaptive model-free control. *Computers & Chemical Engineering*, 179, 108434.
- Zhang, Ding. "A network economic model for supply chain versus supply chain competition." *Omega* 34.3 (2006): 283-295.
- Zhang, D. (2006). A network economic model for supply chain versus supply chain competition. *Omega*, 34(3), 283-295.
- ZHANG, Ding. A network economic model for supply chain versus supply chain competition. *Omega*, 2006, 34.3: 283-295.
- Mbhele, T. P., & Phiri, M. A. (2016). CHALLENGING BULLWHIP EFFECT DYNAMICS WITH ELECTRONICALLY ENABLED-SUPPLY CHAIN MANAGEMENT SYSTEMS. *Risk Governance & Control: Financial Markets & Institutions*, 6(4).
- Asir, M. (2021). Rantai pasok kakao: Karakteristik & peran stakeholder. Penerbit NEM.
- Fu, J., & Fu, Y. (2012). Case-based reasoning and multi-agents for cost collaborative management in supply chain. *Procedia Engineering*, 29, 1088-1098.
- Bambang Leo Handoko, Rudy Aryanto, Idris Gautama So, The Impact of Enterprise Resources System and Supply Chain Practices on Competitive Advantage and Firm Performance: Case of Indonesian Companies, *Procedia Computer Science*, Volume 72, 2015, Pages 122-128.
- Madzivhandila, N. L. (2022). An Investigation of Factors Affecting Plasterboard

Warehouse

Inventory Management: The Case of a South African Company. University of Johannesburg (South Africa).

Hertina, D., Afati, L., Munizu, M., Riyadi, S., Thamrin, J. R., & Irawan, D. A. (2023).

MANAJEMEN RANTAI PASOK: Efektifitas MRP dalam mencapai kesuksesan bisnis. PT.

Sonpedia Publishing Indonesia.

Biswas, S. K., Karmaker, C. L., Islam, A., Hossain, N., & Ahmed, S. (2017).

Analysis of different

inventory control techniques: a case study in a retail shop. Journal of Supply Chain Management

Systems, 6(3), 35.

Costantino, N., Dotoli, M., Falagario, M., Fanti, M. P., & Mangini, A. M. (2012).

A model for

supply management of agile manufacturing supply chains. International Journal of Production

Economics, 135(1), 451-457.

Cornellia, R., & Kurniawan, D. W. (2022). Inventory Control of Raw Materials Sajadah Product

Type royal foam density 22 at UMKM Danika Online Shop using the minmax method,

deterministic model, and C4. 5 Algorithm

Dalam buku yang berjudul Rantai Pasok Jagung oleh Abdul Azis, dkk (2021) memaparkan jenisjenis persediaan menurut Djoko dan Harsasi

DR Ir Abdul Azis Jakfar, M. T., Hery Purwanto, S. P., & Norita Vibriyanto, S. E. (2022). Rantai

Pasok Jagung. Media Nusa Creative (MNC Publishing).

He, X., Xu, X., & Shen, Y. (2023). How climate change affects enterprise

inventory management—From the perspective of regional traffic. *Journal of Business Research*, 162, 113864.

Kolias, G. D., Dimelis, S. P., & Filios, V. P. (2011). An empirical analysis of inventory turnover behaviour in the Greek retail sector: 2000-2005. *International Journal of Production Economics*, 133(1), 143–153.

Hansen, D. R., Mowen, M.M., and Guan, L. (2009). *Cost Management Accounting and Control*, SouthWestern Cengage Learning, Sixth Edition

Handoko, T. H. (1999). *Dasar Dasar Manajemen Produksi dan Operasi* (7th ed.). Yogyakarta: BPFE.

Giannoccaro, Ilaria & Pontrandolfo, Pierpaolo & Scozzi, Barbara. (2003). A fuzzy echelon approach for inventory management in supply chains. *European Journal of Operational Research*. 149. 185-196. 10.1016/S0377-2217(02)00441-1.

Simchi-levi, David & Kaminsky, Philip & Simchi-Levi, Edith. (2003). *Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies*.

Li, Z., & Fu, Q. G. (2017). Robust inventory management with stock-out substitution. *International Journal of Production Economics*, 193, 813-826.

Emrouznejad, Ali & Amin, Gholam R. & Ghiyasi, Mojtaba & Michali, Maria. (2023). A review of inverse data envelopment analysis: origins, development, and future directions. *IMA Journal of*

- Management Mathematics. 34. 10.1093/imaman/dpad006.
- Ferguson, M. E., & Koenigsberg, O. (2007). How should a firm manage deteriorating inventory? Production and Operations Management, 16 (3), 306–312.
- Amorim, P., Meyr, H., Almeder, C., & Almada-Lobo, B. (2013). Managing perishability in production-distribution planning: a discussion and review. Flexible Services and Manufacturing Journal, 25, 389-413.
- Moons, K., Waeyenbergh, G., & Pintelon, L. (2019). Measuring the logistics performance of internal hospital supply chains—a literature study. Omega, 82, 205-217.
- Kustiningsih, N., & Farhan, A. (2022). Manajemen Keuangan: Dasar-Dasar Pengelolaan Keuangan. Cv Globalcare.
- Zhu, Q., Sarkis, J., Lai, K. H., & Geng, Y. (2008). The role of organizational size in the adoption of green supply chain management practices in China. Corporate social responsibility and environmental management, 15(6), 322-337.
- Lee, S. Y. (2008). Drivers for the participation of small and medium-sized suppliers in green supply chain initiatives. Supply chain management: an international journal, 13(3), 185-198.
- Chu, P., Yang, K. L., Liang, S. K., & Niu, T. (2004). Note on inventory model with a mixture of back orders and lost sales. European Journal of Operational Research, 159 (2), 470-475
- Davis, R. A. (2015). Demand-driven inventory optimization and replenishment: Creating a more

- efficient supply chain. John Wiley & Sons.
- Han, P., Huo, Y., & Qi, E. (2021). Effect of seeding on the benefits of the manufacturer and retailer. *Computers & Industrial Engineering*, 153, 107074.
- Bowersox, D. J., Closs, D. J., Cooper, M. B., & Bowersox, J. C. (2020). Supply chain logistics management. McGraw-Hill.
- Davis, R. A. (2015). Demand-driven inventory optimization and replenishment: Creating a more efficient supply chain. John Wiley & Sons.
- Qi, E., Shen, J., & Dou, R. (Eds.). (2013). International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012) Proceedings: Core Areas of Industrial Engineering. Springer Science & Business Media.
- Cárdenas-Barrón, L. E., Shaikh, A. A., Tiwari, S., & Treviño-Garza, G. (2020). An EOQ inventory model with nonlinear stock dependent holding cost, nonlinear stock dependent demand and trade credit. *Computers & Industrial Engineering*, 139, 105557.
- Erlenkotter, Donald. (2014). Ford Whitman Harris's economical lot size model. *International Journal of Production Economics*. 155. 12–15. 10.1016/j.ijpe.2013.12.008.
- Handoko, T. Hani. 1994. Dasar-dasar Manajemen Produksi dan Operasi. Edisi 1. BPFF UGM. Yogyakarta
- Herdjanto, Eddy. 2008. Manajemen Operasi Edisi 3. Jakarta : PT. Raja Grasindo Persada.
- Taylor III, B.W. (2005). Sains Manajemen. Edisi 8. Jakarta: Salemba Empat.
- Ristono, Agus. 2009. Manajemen Persediaan. Penerbit Graha Ilmu. Yogyakarta.

- Nasution, A. H., & Prasetyawan, Y., 2008, Perencanaan dan Pengendalian Produksi, Graha Ilmu, Yogyakarta.
- Taha, H.A. (2007). Operations Research : An Introduction Eight Edition. Pearson Prentice Hall.
- Render dan Heizer. (2001). Prinsip-prinsip Manajemen Operasi. Edisi 8. Jakarta: Salemba Empat.
- Tanthamee,T dan Phruksapharnat,B. (2012). Fuzzy Inventory Control System for Uncertain Demand and Supply: IMECS,March 14-16,2012.
- Kamal,L. dan Sculfort,J.-L. (2007). Fuzzy Modeling of Inventory control system in uncertain envirovment, International Sympo. Sep 2007,53-57. Logistic and Industrial Informatic.
- He, F., Kalantari, K. R., Ebrahimnejad, A., Branch, Q., Azad, I., & Motameni, H. (2022). An Effective Fault-Tolerance Technique in Web Services: An Approach based on Hybrid.
- Chawla, M., & Duhan, M. (2018). Levy flights in metaheuristics optimization algorithms—a review. *Applied Artificial Intelligence*, 32(9-10), 802-821.
- Fathollahi-Fard, A. M., Wong, K. Y., & Aljuaid, M. (2023). An efficient adaptive large neighborhood search algorithm based on heuristics and reformulations for the generalized quadratic assignment problem. *Engineering Applications of Artificial Intelligence*, 126, 106802.
- Ezugwu, A. E., Shukla, A. K., Nath, R., Akinyelu, A. A., Agushaka, J. O., Chiroma, H., & Muhuri, P. K. (2021). Metaheuristics: a comprehensive overview and classification along

with bibliometric analysis. *Artificial Intelligence Review*, 54, 4237-4316.

Liu, P., Hendalianpour, A., Razmi, J., & Sangari, M. S. (2021). A solution algorithm for integrated production-inventory-routing of perishable goods with transshipment and uncertain demand. *Complex & Intelligent Systems*, 7, 1349-1365.

Pop, P. C., Pintea, C. M., Sitar, C. P., & Hajdu-Măcelaru, M. (2015). An efficient Reverse Distribution System for solving sustainable supply chain network design problem. *Journal of Applied Logic*, 13(2), 105-113.

Lin, Y., Jia, H., Yang, Y., Tian, G., Tao, F., & Ling, L. (2018). An improved artificial bee colony for facility location allocation problem of end-of-life vehicles recovery network. *Journal of cleaner production*, 205, 134-144.

Mazaheri, V., & Khodadadi, H. (2020). Heart arrhythmia diagnosis based on the combination of morphological, frequency and nonlinear features of ECG signals and metaheuristic feature selection algorithm. *Expert Systems with Applications*, 161, 113697.

Cuong-Le, T., Minh, H. L., Khatir, S., Wahab, M. A., Tran, M. T., & Mirjalili, S. (2021). A novel version of Cuckoo search algorithm for solving optimization problems. *Expert Systems with Applications*, 186, 115669.

Nama, S., Saha, A. K., Chakraborty, S., Gandomi, A. H., & Abualigah, L. (2023). Boosting particle swarm optimization by backtracking search algorithm for optimization problems.

Swarm and
Evolutionary Computation, 79, 101304.

Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M., & Gandomi, A. H. (2021).
The arithmetic
optimization algorithm. Computer methods in applied mechanics and engineering,
376, 113609.

Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-Qaness, M. A., &
Gandomi, A. H.
(2021). Aquila optimizer: a novel meta-heuristic optimization algorithm.
Computers & Industrial
Engineering, 157, 107250.

Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z. W., & Gandomi, A. H.
(2022). Reptile Search
Algorithm (RSA): A nature-inspired meta-heuristic optimizer. Expert Systems
with Applications,
191, 116158.

Bertsimas, D., & Tsitsiklis, J. (1993). Simulated annealing. Statistical science,
8(1), 10-15.

Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. IEEE
computational
intelligence magazine, 1(4), 28-39.

Glover, F., & Laguna, M. (1998). Tabu search (pp. 2093-2229). Springer US.

Chakraborty, S., Saha, A. K., Nama, S., & Debnath, S. (2021). COVID-19 X-ray
image
segmentation by modified whale optimization algorithm with population
reduction. Computers in
Biology and Medicine, 139, 104984.

Abualigah, L., Hanandeh, E. S., Zitar, R. A., Thanh, C. L., Khatir, S., & Gandomi,
A. H. (2023).
Revolutionizing sustainable supply chain management: A review of

metaheuristics. *Engineering Applications of Artificial Intelligence*, 126, 106839.

Agushaka, J. O., Ezugwu, A. E., & Abualigah, L. (2022). Dwarf mongoose optimization algorithm. *Computer methods in applied mechanics and engineering*, 391, 114570

Oyelade, O. N., Ezugwu, A. E. S., Mohamed, T. I., & Abualigah, L. (2022). Ebola optimization search algorithm: A new nature-inspired metaheuristic optimization algorithm. *IEEE Access*, 10, 16150-16177.

Ezugwu, A. E., Agushaka, J. O., Abualigah, L., Mirjalili, S., & Gandomi, A. H. (2022). Prairie dog optimization algorithm. *Neural Computing and Applications*, 34(22), 20017-20065.

Agushaka, J. O., Ezugwu, A. E., & Abualigah, L. (2023). Gazelle optimization algorithm: a novel nature-inspired metaheuristic optimizer. *Neural Computing and Applications*, 35(5), 4099-4131.

Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). ieee.

Xu, G., Cui, Q., Shi, X., Ge, H., Zhan, Z. H., Lee, H. P., ... & Wu, C. (2019). Particle swarm optimization based on dimensional learning strategy. *Swarm and Evolutionary Computation*, 45, 33-51.

Clerc, M., & Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE transactions on Evolutionary*

- Computation, 6(1), 58-73.
- Kuo, R. J., & Han, Y. S. (2011). A hybrid of genetic algorithm and particle swarm optimization for solving bi-level linear programming problem—A case study on supply chain model. *Applied Mathematical Modelling*, 35(8), 3905-3917.
- Ratnaweera, Asanga & Halgamuge, Saman & Watson, Harry. (2004). Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients. *Evolutionary Computation, IEEE Transactions on*. 8. 240 - 255. 10.1109/TEVC.2004.826071.
- Jong, Jek siang. (2009). *Jaringan Syaraf Tiruan & Pemrograman Menggunakan Matlab*. Penerbit Andi. Yogyakarta
- Hsieh, L.F., Huang, C.J. & Huang, C.L. 2007. Applying Particle Swarm Optimization To Schedule Order Picking Routes In A Distribution Center. *Asian Journal of Management and Humanity Sciences*. Vol. 1, No. 4. pp. 558-576.
- Asriningtias, s. R., dachlan, h. S., & yudaningtyas, e. (2015). Optimasi Training Neural Network Menggunakan Hybrid Adaptive Mutation PSO-BP. *Jurnal eeccis*, 9(1), pp-79.
- Chen, R.M. & Shih, H.F. 2013. Solving University Course Timetabling Problems Using Constriction Particle Swarm Optimization with Local Search. *Article Algorithms* 2013, 6, 227-244; doi:10.3390/a6020227. ISSN 1999-4893
- Engelbrecht, A.P. 2006. *Fundamentals of Computational Swarm Intelligence*. Wiley.

Kusumawati, d., winarno, w. W., & arief, m. R. (2015). Prediksi kelulusan mahasiswa menggunakan metode neural network dan particle swarm optimization. Semnasteknimedia online, 3(1), 3-8.

Bai, Qinghai. 2010. Analysis Of Particle Swarm Optimization Algorithm. CCSE, Computer and Information Science. www.ccsenet.org/cis College of Computer Science and Technology. Inner Mongolia University for Nationalities. Tongliao 028043: China

Raharjo, j. S. D. (2013). Model artificial neural network berbasis particle swarm optimization untuk prediksi laju inflasi. Jurnal sistem komputer, 3(1), 10-21.

Wati, D.A.R dan Rochman, Y.A. 2013. Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Particle Swarm Optimization (PSO). Fakultas Teknologi Industri, Jurusan Teknik Elektro, Universitas Islam Indonesia.

Teugeh, M., Soeprijanto, dan Purnomo, H.M., 2009, Modified Improved Particle Swarm Optimization for Optimal Generator Scheduling, Seminar Nasional Aplikasi Teknologi Informasi, Yogyakarta, ISSN: 1907- 5022.

Zhang, T., Zheng, Q. P., Fang, Y., & Zhang, Y. (2015). Multi-level inventory matching and order planning under the hybrid Make-To-Order/Make-To-Stock production environment for steel plants via Particle Swarm Optimization. Computers & Industrial Engineering, 87, 238-249.

Dye, C. Y. (2012). A finite horizon deteriorating inventory model with two-phase

pricing and timevarying demand and cost under trade credit financing using particle swarm optimization. *Swarm and Evolutionary Computation*, 5, 37-53.

Shaikh, A. A., Das, S. C., Bhunia, A. K., Panda, G. C., & Al-Amin Khan, M. (2019). A twowarehouse EOQ model with interval-valued inventory cost and advance payment for deteriorating item under particle swarm optimization. *Soft Computing*, 23(24), 13531-13546.

Bhunia, A. K., & Shaikh, A. A. (2016). Investigation of two-warehouse inventory problems in interval environment under inflation via particle swarm optimization. *Mathematical and Computer Modelling of Dynamical Systems*, 22(2), 160-179.

Safrudin, T., Pranoto, G. T., & Hadikristanto, W. (2023). Optimasi Algoritma K-Nearest Neighbor Berbasis Particle Swarm Optimization Untuk Meningkatkan Kebutuhan Barang. *Bulletin of Information Technology (BIT)*, 4(3), 281-286.

Xu, J., Zeng, Z., Han, B., & Lei, X. (2013). A dynamic programming-based particle swarm optimization algorithm for an inventory management problem under uncertainty. *Engineering Optimization*, 45(7), 851-880.

Simić, D., Svirčević, V., Ilin, V., Simić, S. D., & Simić, S. (2019). Particle swarm optimization and pure adaptive search in finish goods' inventory management. *Cybernetics and Systems*, 50(1), 58-77.

Yang, M. F., & Lin, Y. (2010). Applying the linear particle swarm optimization to a serial multiechelon inventory model. *Expert Systems with Applications*, 37(3), 2599-2608.

- Tsai, C. Y., & Yeh, S. W. (2008). A multiple objective particle swarm optimization approach for inventory classification. *International journal of production Economics*, 114(2), 656-666.
- Al-Khazraji, H., Cole, C., & Guo, W. (2018). Multi-objective particle swarm optimisation approach for production-inventory control systems. *Journal of Modelling in Management*, 13(4), 1037-1056.
- Jaberipour, M., Khorram, E., & Karimi, B. (2011). Particle swarm algorithm for solving systems of nonlinear equations. *Computers & Mathematics with Applications*, 62(2), 566-576.
- Clark, Andrew & Scarf, Herbert. (2004). Optimal Policies for a Multi-Echelon Inventory Problem. *Management Science*. 50. 1782-1790. 10.1287/mnsc.1040.0265.
- Dasgupta, S. P. (2021). The economics of biodiversity the Dasgupta review abridged version.

