

BAB I PENDAHULUAN

Bab pendahuluan berisi pembahasan mengenai latar belakang permasalahan, rumusan masalah, tujuan penelitian, dan manfaat yang diharapkan dari penelitian yang dilakukan, serta sistematika dalam penulisan skripsi.

1.1 Latar Belakang

Distribusi merupakan proses pemindahan barang hasil produksi dari tingkat produsen ke konsumen (Pujawan & Mahendrawati, 2010). Distribusi menjadi aspek penting bagi perusahaan karena menentukan keberhasilan pemasaran barang hasil produksi. Berdasarkan hasil survey Badan Pusat Statistik dan Kementerian Perindustrian biaya distribusi mencapai 30 persen dari total biaya produksi (Kemenperin, 2014). Biaya distribusi tersebut akan semakin meningkat apabila jangkauan distribusi mencakup wilayah yang luas. Meningkatnya biaya distribusi tersebut tentu akan berdampak pada peningkatan harga jual dari suatu barang hasil produksi. Dengan mahalnya harga jual barang hasil produksi dapat mengurangi daya beli konsumen dan berdampak pada kerugian perusahaan. Distribusi dua tahap dibutuhkan untuk mengoptimalkan biaya distribusi barang dari produsen ke konsumen.

Pada distribusi dua tahap akan dibentuk sebuah distributor sebagai perantara antara produsen dan konsumen. Distributor dibangun sesuai dengan wilayah pemasaran yang dekat dengan lokasi agen atau konsumen. Barang hasil produksi dari produsen akan dikirimkan ke beberapa distributor dan selanjutnya distributor akan menyalurkan barang hasil produksi kepada beberapa agen atau konsumen yang berdekatan (Ningrum, 2011), sehingga memudahkan produsen untuk memenuhi permintaan konsumen dan dapat meminimalkan biaya distribusi.

Penelitian sebelumnya menyelesaikan permasalahan distribusi dua tahap menggunakan metode *heuristic* berdasarkan mekanisme evolusi biologis. Metode tersebut digunakan untuk menghasilkan solusi terbaik dari ruang solusi yang luas. Salah satu algoritma *heuristic* yang digunakan adalah algoritma genetika. Algoritma genetika digunakan dalam menyelesaikan permasalahan distribusi rantai pasok dua tingkat yang dipengaruhi oleh biaya tetap dan menggunakan banyak

model transportasi (Chun & Yi, 2009), pencarian biaya minimum dalam distribusi suatu komoditi dengan transportasi dua tahap membandingkan algoritma genetika dengan algoritma *brute force* (Ningrum, 2011), algoritma genetik dua populasi diterapkan pada kasus distribusi dua tahap perusahaan susu fermentasi dengan menggunakan metode *weight mapping crossover* dan teknik mutasi *swap mutation* (Prabowo, 2011).

Selain menggunakan algoritma genetika, algoritma *evolution strategies* juga dapat digunakan untuk menyelesaikan permasalahan optimasi. Dimana ciri utama dari algoritma *evolution strategies* adalah representasi solusi yang digunakan berupa vektor bilangan pecahan (Mahmudy, 2013). Dalam penerapannya algoritma *evolution strategies* terbukti dapat menyelesaikan *Vehicle Routing Problem With Time Windows* pada distribusi minuman soda XYZ dengan teknik mutasi *exchange mutation* (Harun, Mahmudy, & Yudistira, 2014), pencarian rute optimum dengan studi kasus layanan pesan antar *Pizza Hut Delivery* (PHD) yang memperoleh nilai fitness terbesar pada generasi 50 dan 100 (Endarwati, Mahmudy, & Ratnawati, 2014).

Oleh karena itu, pada penelitian ini menggunakan algoritma *evolution strategies* untuk menyelesaikan permasalahan optimasi distribusi barang dua tahap melalui penelitian yang berjudul “Penerapan Algoritma *Evolution Strategies* untuk Optimasi Distribusi Barang Dua Tahap”. Penelitian ini bertujuan memperoleh solusi yang mendekati optimal pada permasalahan distribusi dua tahap, yaitu distribusi dari produsen ke beberapa agen dan dari agen ke beberapa sub agen.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang diuraikan pada bagian latar belakang, maka dirumuskan permasalahan sebagai berikut:

1. Bagaimana menerapkan algoritma *evolution strategies* untuk optimasi distribusi barang dua tahap.
2. Bagaimana menentukan parameter algoritma *evolution strategies* yang tepat untuk permasalahan distribusi barang dua tahap.

3. Bagaimana pengaruh parameter algoritma *evolution strategies* terhadap *fitness* dari solusi yang dihasilkan algoritma *evolution strategies* untuk optimasi distribusi barang dua tahap.

1.3 Batasan Masalah

Agar penelitian ini dapat terfokus, perlu dilakukan pembatasan masalah, antara lain:

1. Distribusi barang dua tahap yang diselesaikan adalah menentukan jalur distribusi optimal dengan meminimalkan biaya distribusi dari produsen ke beberapa agen dan dari agen ke beberapa sub agen.
2. Parameter penentuan optimasi adalah kapasitas produsen, kebutuhan agen dan sub agen, serta biaya distribusi tanpa memperhitungkan jadwal produksi, jumlah kendaraan, dan pengaruh lainnya.
3. Distribusi yang diselesaikan adalah distribusi seimbang, dimana jumlah permintaan tidak lebih besar atau sama dengan persediaan produsen.
4. Input data ke sistem terdiri dari parameter ES dan parameter kasus. Parameter ES adalah ukuran populasi (μ), *offspring* (λ), parameter a , range nilai sigma, jumlah generasi, dan perbandingan mutasi segmen. Sedangkan parameter kasus merupakan data *dummy* yang dibuat peneliti sendiri, yaitu jumlah produsen, agen, dan sub agen, kapasitas produsen, kebutuhan agen dan sub agen, serta biaya distribusi dari produsen ke agen, maupun agen ke sub agen.
5. Metode mutasi yang digunakan adalah *exchange mutation* dan seleksi menggunakan *elitism selection*.
6. Output yang dihasilkan oleh sistem adalah jalur distribusi optimal, total biaya distribusi, dan nilai *fitness* dari solusi optimal.

1.4 Tujuan

Tujuan yang ingin dicapai dari penelitian ini adalah:

1. Menerapkan algoritma *evolution strategies* untuk optimasi distribusi barang dua tahap.
2. Mengetahui parameter algoritma *evolution strategies* yang tepat untuk menyelesaikan permasalahan distribusi barang dua tahap.

3. Mengetahui pengaruh parameter algoritma *evolution strategies* terhadap *fitness* dari solusi yang dihasilkan algoritma *evolution strategies* untuk optimasi distribusi barang dua tahap.

1.5 Manfaat

Manfaat dari hasil penelitian ini adalah dihasilkan suatu sistem yang menerapkan algoritma *evolution strategies* yang dapat membantu perusahaan untuk menentukan jalur distribusi barang dua tahap yang optimal.

1.6 Sistematika Penulisan

Sistematika penulisan memberikan gambaran dan uraian dari penyusunan skripsi secara garis besar yang meliputi beberapa bab, antara lain:

BAB I PENDAHULUAN

Bab ini menguraikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat dari penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini menguraikan penelitian yang telah dilakukan sebelumnya dan teori-teori yang berhubungan dengan algoritma evolusi, algoritma *Evolution Strategies*, dan distribusi dua tahap.

BAB III METODOLOGI PENELITIAN

Bab ini menguraikan tentang langkah kerja yang dilakukan dalam penulisan skripsi yang terdiri dari studi pustaka, perancangan perangkat lunak, implementasi, pengujian, serta pengambilan kesimpulan.

BAB IV PERANCANGAN

Bab ini menguraikan tentang perancangan perangkat lunak penerapan algoritma *evolution strategies* untuk optimasi distribusi barang dua tahap.

BAB V IMPLEMENTASI

Bab ini menguraikan tentang implementasi dari perancangan yang dijabarkan pada bab sebelumnya.

BAB VI PENGUJIAN DAN ANALISIS

Bab ini memuat hasil pengujian dan analisis hasil pengujian penerapan

algoritma *Evolution Strategies* untuk optimasi distribusi barang dua tahap.

BAB VII PENUTUP

Bab ini menguraikan kesimpulan dari pembahasan dan saran yang diharapkan bermanfaat untuk pengembangan penelitian ini selanjutnya.



BAB II

TINJAUAN PUSTAKA

Bab tinjauan pustaka membahas mengenai penelitian terkait, distribusi dua tahap, algoritma evolusi, dan algoritma *evolution strategies*.

2.1 Penelitian Terkait

Penelitian sebelumnya menerapkan algoritma genetika untuk menyelesaikan permasalahan distribusi dua tahap dengan biaya tetap dan menggunakan banyak model transportasi. Distribusi yang dimaksud adalah distribusi barang dari pabrik ke distributor dan dari distributor ke konsumen. Pengujian pada penelitian ini dilakukan pada kasus distribusi dua tahap dari 2 pabrik, 2 distributor, 4 konsumen, dengan 2 model transportasi. Operator *crossover* yang digunakan adalah *Partially Mapped Crossover* (PMX) dengan *two-point crossover*. Algoritma genetika yang diterapkan untuk menyelesaikan permasalahan distribusi dua tahap dengan biaya tetap dan menggunakan banyak model transportasi menghasilkan solusi optimal setelah 50 generasi (Chun & Yi, 2009).

Penelitian mengenai pencarian biaya minimum dalam distribusi suatu komoditi dengan transportasi dua tahap menggunakan algoritma genetik menggunakan representasi kromosom array satu dimensi (*priority based*). Teknik *crossover* dan mutasi yang digunakan adalah *one-cut point crossover* dan *swap mutation*, serta seleksi *roulette wheel*. Dengan studi kasus distribusi dua tahap dari 3 pabrik, 4 distributor, dan 5 konsumen, penelitian ini menunjukkan bahwa penggunaan algoritma genetika lebih efektif dibandingkan dengan algoritma *brute force*. Hal ini dapat dilihat dari perbandingan iterasi dan waktu komputasi yang dibutuhkan algoritma genetik lebih sedikit dibandingkan dengan algoritma *brute force* dalam memperoleh nilai *cost* paling minimum (Ningrum, 2011).

Algoritma *evolution strategies* juga menerapkan prinsip evolusi biologis dan dapat digunakan sebagai teknik optimasi untuk menyelesaikan *Vehicle Routing Problem With Time Windows* (VRPTW). Penelitian ini menggunakan studi kasus distribusi minuman soda XYZ dengan populasi awal (μ) sebanyak 80, teknik mutasi *exchange mutation* untuk menghasilkan solusi optimal setelah generasi 2000. Pada penelitian ini menyatakan bahwa algoritma *evolution strategies* dapat menyelesaik-

kan *Vehicle Routing Problem With Time Windows* (VRPTW) dalam waktu kurang dari satu menit (Harun, Mahmudy, & Yudistira, 2014).

2.2 Distribusi

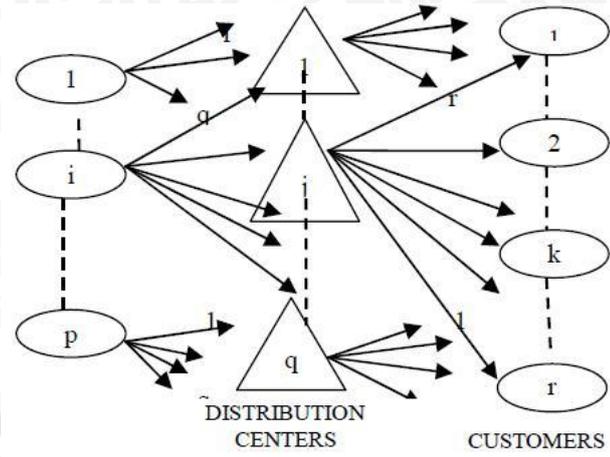
2.2.1 Pengertian Distribusi

Distribusi merupakan proses untuk menyalurkan barang hasil produksi dari produsen menuju konsumen melalui rantai distribusi (Pujawan & Mahendrawati, 2010). Distribusi merupakan aspek penting bagi perusahaan. Distribusi yang optimal memberikan peluang bagi perusahaan untuk memenuhi kebutuhan konsumen dan memperoleh keuntungan yang lebih besar.

2.2.2 Distribusi Dua Tahap

Biaya distribusi barang hasil produksi dari pabrik ke konsumen akan bertambah besar apabila wilayah pemasaran barang hasil produksi semakin meluas. Hal ini berpengaruh pada peningkatan harga jual barang hasil produksi. Peningkatan harga jual juga akan berdampak pada penurunan daya beli konsumen sehingga dapat menyebabkan kerugian bagi perusahaan. Untuk itu perusahaan menerapkan distribusi dua tahap sebagai solusi untuk meminimalkan biaya distribusi.

Pada distribusi dua tahap perusahaan akan membangun sebuah distributor yang nantinya menjadi fasilitas perantara antara pabrik dan konsumen. Distributor dibangun sesuai dengan wilayah pemasaran yang dekat dengan lokasi agen atau konsumen (Ningrum, 2011). Mekanisme distribusi dua tahap ini adalah dengan memindahkan barang hasil produksi dari pabrik dikirimkan ke beberapa distributor, selanjutnya distributor akan menyalurkan barang sesuai permintaan kepada beberapa agen atau konsumen yang lokasinya berdekatan dengan gudang distributor (Chun & Yi, 2009). Tujuan dari distribusi dua tahap adalah mengoptimalkan biaya transportasi pada proses distribusi. Model distribusi dua tahap ditunjukkan pada Gambar 2.1.



Gambar 2.1 Model Distribusi Dua Tahap

Sumber: (Chun & Yi, 2009)

Berdasarkan Gambar 2.1 biaya optimal distribusi dua tahap dapat dimodelkan secara matematis dengan fungsi yang dinyatakan oleh Gen dalam Prabowo (2011) pada Persamaan 2.1.

$$Z = \sum_{i=1}^i \sum_{j=1}^j t_{ij} x_{ij} + \sum_{j=1}^j \sum_{k=1}^k c_{jk} y_{jk} \tag{2.1}$$

dimana,

- i : Jumlah pabrik (i= 1,2,...i).
- k : Jumlah konsumen (k= 1,2,k).
- j : Jumlah distributor (j= 1,2,...j).
- a_i : Jumlah kapasitas produksi pabrik i.
- b_j : Jumlah permintaan distributor j.
- d_k : Jumlah permintaan dari konsumen k.
- x_{ij} : Jumlah barang yang dikirimkan dari pabrik menuju distributor.
- y_{jk} : Jumlah barang yang dikirimkan dari distributor menuju konsumen.
- t_{ij} : Biaya perjalanan dari pabrik menuju distributor.
- c_{jk} : Biaya perjalanan dari distributor menuju konsumen.

Fungsi tersebut memiliki beberapa batasan (*constraint*) dimana distribusi barang tidak melebihi atau sama dengan kapasitas produksi seperti yang ditunjukkan pada Persamaan 2.2 dan 2.3.

$$\sum_{j=1}^j x_{ij} \leq a_i, \quad \forall i \tag{2.2}$$



$$\sum_{k=1}^k y_{jk} \leq b_j, \quad \forall j \quad 2.3$$

Barang yang dikirimkan dari distributor menuju konsumen tidak boleh kurang dari permintaan konsumen, dinyatakan pada Persamaan 2.4

$$\sum_{j=1}^j y_{jk} \geq d_k, \quad \forall k \quad 2.4$$

Persediaan yang dimiliki pabrik diasumsikan tidak nol atau sama dengan permintaan dari konsumen, dinyatakan pada Persamaan 2.5 dan 2.6.

$$\sum_{i=1}^i \sum_{j=1}^j x_{ij} = \sum_{j=1}^j \sum_{k=1}^k y_{jk} \quad 2.5$$

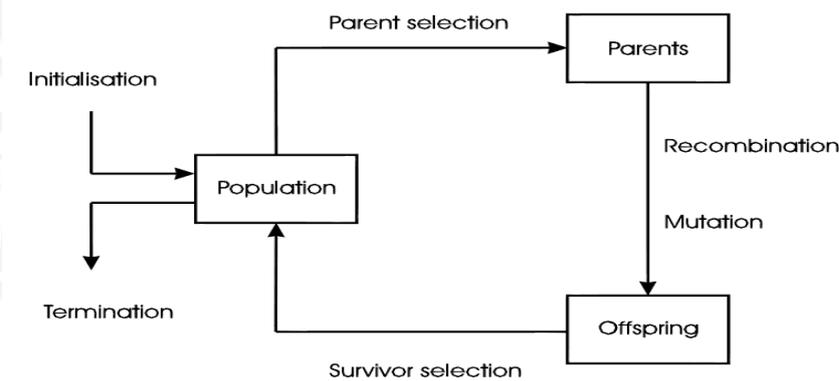
$$x_{ij}, y_{jk} \geq 0, \quad \forall i, j, k \quad 2.6$$

2.3 Algoritma Evolusi

Algoritma Evolusi (*Evolutionary Algorithm, EAs*) merupakan metode *heuristic* yang didasarkan atas prinsip teori evolusi yang dikemukakan oleh Charles Darwin (Lange, 2007). Sejumlah individu dalam populasi berperan sebagai induk (*parent*) melakukan reproduksi untuk menghasilkan keturunan (*offspring*). Individu-individu tersebut berevolusi dan menghadapi seleksi alam. Hanya individu-individu terbaik yang mampu beradaptasi dan bertahan hidup hingga generasi berikutnya.

Fungsi *fitness* dalam algoritma evolusi digunakan untuk mengukur seberapa baik suatu individu (Mahmudy, 2013). Individu terbaik yang bertahan hingga generasi terakhir dapat dijadikan sebagai solusi yang mendekati optimal. Individu terbaik dengan nilai *fitness* yang baik cenderung menghasilkan solusi yang terbaik, tetapi tidak menutup kemungkinan solusi dengan nilai *fitness* yang kurang baik dapat menghasilkan solusi baru yang lebih baik (Harun, Mahmudy, & Yudistira, 2014).

Beberapa tipe dalam algoritma evolusi yang telah dikembangkan memiliki proses yang hampir sama, yaitu inisialisasi populasi awal, reproduksi yang terdiri dari mutasi dan rekombinasi menghasilkan *offspring*, evaluasi dan seleksi untuk menghasilkan himpunan individu baru yang bertahan hingga generasi berikutnya. Proses dalam algoritma evolusi ditunjukkan pada Gambar 2.2.



Gambar 2.2 Proses Algoritma Evolusi

Sumber: (Eiben & Smith, 2004)

Terdapat beberapa tipe algoritma evolusi. Perbedaan dari masing-masing tipe algoritma evolusi adalah representasi individunya (Mahmudy, 2013). Algoritma genetika (*Genetic Algorithm, GAs*) menggunakan representasi biner, vektor bilangan integer, dan *real*, *Evolution Strategies (ES)* menggunakan representasi vektor bilangan pecahan, *Genetic Programming* menggunakan representasi *trees based*, *Evolutionary Programming* menggunakan representasi *Finite State Machines (FSM)* (Mahmudy, 2013).

2.4 Algoritma *Evolution Strategies*

Algoritma *Evolution Strategies (ES)* pertama kali dikembangkan oleh Schwefel and Rechenberg dari University of Berlin sekitar tahun 1960-an (Lange, 2007). Algoritma *evolution strategies* dapat digunakan untuk menyelesaikan permasalahan optimasi karena menghasilkan solusi yang mendekati optimal dari suatu permasalahan yang kompleks (Beyer & Schwefel, 2002).

Ciri utama dari algoritma *evolution strategies* yang membedakan dari tipe algoritma evolusi lainnya adalah penggunaan vektor bilangan pecahan (*real vector*) sebagai representasi suatu solusi (Mahmudy, 2013). Proses reproduksi dalam algoritma *evolution strategies* lebih menekankan pada proses mutasi yang diterapkan pada *offspring* hasil proses rekombinasi. Rekombinasi pada *ES* seperti operator *crossover* dalam *GAs* tetapi penerapannya bisa menggunakan lebih dari satu *parent* (Mahmudy, 2013). Mekanisme *self-adaptation* yang terdapat pada algoritma *evolution strategies* digunakan untuk mengontrol perubahan nilai

strategy parameters yang menyatakan level mutasi setiap individu pada setiap generasi (Hansen, Arnold, & Auger, 2013).

2.4.1 Struktur Dasar Evolution Strategies

Notasi μ (*miu*) digunakan untuk menyatakan ukuran populasi, λ (*lamda*) menyatakan banyaknya *offspring*, dan r menyatakan operator rekombinasi (Mahmudy, 2013) pada literatur lain ditulis sebagai ρ (*rho*) (Hansen, Arnold, & Auger, 2013). Algoritma *evolution strategies* memiliki beberapa siklus penyelesaian yang berbeda. Perbedaan dari siklus penyelesaian dalam algoritma *evolution strategies* adalah pada ada tidaknya proses rekombinasi dan individu yang terlibat dalam seleksi. Terdapat empat macam siklus dalam algoritma *evolution strategies*, yaitu (Mahmudy, 2013):

1. (μ, λ)

Algoritma *evolution strategies* dengan siklus (μ, λ) tidak menggunakan rekombinasi dalam proses reproduksinya. Reproduksi hanya melibatkan proses mutasi. Metode seleksi yang digunakan adalah *elitism selection* dengan melibatkan individu dalam *offspring* (λ).

2. $(\mu + \lambda)$

Reproduksi siklus $(\mu + \lambda)$ hanya menggunakan mutasi. Seleksi yang digunakan adalah *elitism selection* dengan melibatkan *parent* (μ) dan *offspring* (λ).

3. $(\mu/r, \lambda)$

Reproduksi siklus ES $(\mu/r, \lambda)$ melibatkan proses rekombinasi dan mutasi. Metode seleksi yang digunakan adalah *elitism selection* dengan melibatkan *offspring* (λ).

4. $(\mu/r + \lambda)$

Siklus $(\mu/r + \lambda)$ reproduksi menggunakan rekombinasi dan mutasi. Seleksi menggunakan metode *elitism selection* melibatkan *parent* (μ) dan *offspring* (λ).

Penggunaan siklus *evolution strategies* didasarkan pada jenis permasalahan yang diselesaikan. Untuk permasalahan kombinatorial Herdy (1990) dan Beyer (1992) merekomendasikan penggunaan siklus $(\mu + \lambda)$ (Beyer & Schwefel, 2002). Distribusi barang dua tahap merupakan salah satu contoh permasalahan kombinatorial dengan banyak kemungkinan solusi penyelesaian $n! \times m!$, sehingga

pada penelitian ini siklus ES yang digunakan adalah $(\mu+\lambda)$. Siklus $(\mu+\lambda)$ digunakan dalam beberapa kasus kombinatorial seperti *Travelling Salesman Problem* dan penjadwalan (Whitley, 2001). Siklus $(\mu+\lambda)$ merupakan *elitist strategy* (Eiben & Smith, 2004). Siklus $(\mu+\lambda)$ menggunakan *elitism selection* yang menjamin pemilihan individu terbaik dapat berasal dari *parent* maupun *offspring* (Hansen, Arnold, & Auger, 2013). Jika $P(t)$ dan $C(t)$ merupakan populasi *parent* dan *offspring* pada generasi ke t , maka *pseudocode* dari siklus $(\mu+\lambda)$ ditunjukkan pada Gambar 2.3.

```
procedure EvolutionStrategies
begin
     $t = 0$ 
    inialisasi  $P(t)$ : generate random  $\mu$  individu
    while (bukan kondisi berhenti) do
        mutasi  $C(t)$  untuk menghasilkan  $\lambda$  dari  $P(t)$ 
        seleksi  $P(t+1)$  dari  $P(t)$  dan  $C(t)$ 
         $t = t + 1$ 
    end while
end
```

Gambar 2.3 Pseudocode proses ES $(\mu+\lambda)$

Sumber : (Mahmudy, 2013)

Dari *pseudocode* yang ditunjukkan Gambar 2.3 terlihat bahwa pada siklus $(\mu+\lambda)$ inialisasi populasi *parent* dilakukan secara *random*. Proses reproduksi hanya menggunakan proses mutasi dan seleksi yang melibatkan populasi *parent* dan populasi *offspring*.

2.4.2 Representasi Kromosom

Kromosom tersusun dari sejumlah gen yang merepresentasikan variabel-variabel solusi (Lange, 2007). Representasi kromosom pada algoritma *evolution strategies* disertai dengan fungsi *fitness* yang menyatakan kebaikan dari solusi dan *strategy parameters* yang menyatakan level mutasi (Hansen, Arnold, & Auger, 2013).

Beberapa bentuk representasi kromosom, yaitu representasi biner, integer, *real*, dan permutasi (Mahmudy, 2013). Distribusi dua tahap merupakan salah satu

permasalahan kombinatorial yang solusinya dapat direpresentasikan dengan menggunakan representasi permutasi. Setiap gen pada representasi kromosom permutasi adalah angka integer yang menyatakan urutan dari setiap *node* (Lange, 2007). Gambar 2.4 merupakan contoh representasi permutasi.

Node	1	2	3	4
Kromosom	2	3	1	4

Gambar 2.4 Representasi Permutasi

2.4.3 Reproduksi

Proses reproduksi dalam algoritma *evolution strategies* merupakan proses untuk membentuk suatu individu baru. Reproduksi pada algoritma *evolution strategies* dengan siklus $(\mu+\lambda)$ hanya menggunakan proses mutasi. Proses mutasi yang digunakan pada penelitian ini adalah *exchange mutation*.

Mutasi dengan *exchange mutation* diawali dengan memilih dua posisi gen (*exchange point/ XP*) secara *random*, kemudian menukarkan nilai gen pada posisi tersebut (Mahmudy, 2013). Proses mutasi dengan metode *exchange mutation* dijelaskan pada Gambar 2.5.

Parent	2	3	1	4
Offspring	1	3	2	4

Gambar 2.5 Proses *Exchange Mutation*

Individu yang dibangkitkan disertai dengan nilai *strategy parameters* yang menyatakan level mutasi. *Strategy parameters* digunakan untuk menentukan banyaknya proses mutasi yang dilakukan untuk menghasilkan satu *offspring*. *Strategy parameter* dibangkitkan secara *random* dalam range nilai tertentu. Mekanisme *self adaptation* digunakan untuk mengontrol nilai *strategy parameters* menggunakan aturan 1/5, dimana nilai σ (*strategy parameter*) dinaikkan jika terdapat paling sedikit 1/5 atau 20% *offspring* yang dihasilkan memiliki *fitness* yang lebih baik dari induknya, jika tidak ada *offspring* yang lebih baik maka nilai σ diturunkan. Aturan 1/5 ditunjukkan pada Persamaan 2.7 (Beyer & Schwefel, 2002).

$$\sigma = \begin{cases} \sigma/a, & \text{if } P_s > 1/5 \\ \sigma \cdot a, & \text{if } P_s < 1/5 \\ \sigma, & \text{if } P_s = 1/5 \end{cases} \quad 2.7$$

Nilai parameter a adalah *constant* yang ditentukan untuk menaikkan atau menurunkan nilai σ (*strategy parameter*). Jika jumlah generasi lebih besar dari 30, maka nilai parameter a yang direkomendasikan antara $0.85 \leq a \leq 1$ (Beyer & Schwefel, 2002).

Banyak *offspring* (λ) yang dihasilkan dalam proses algoritma *evolution strategies* diperoleh dari perkalian populasi awal (μ) dengan suatu *constant* dalam rentang nilai $[0, 10]$, seperti yang ditunjukkan pada Persamaan 2.8 (Mahmudy, 2013).

$$\lambda = C * \mu \quad 2.8$$

2.4.4 Fungsi *Fitness*

Fungsi *fitness* digunakan untuk mengukur kebaikan solusi yang dibawa oleh suatu individu. Individu terbaik dengan nilai *fitness* yang baik cenderung menghasilkan solusi yang terbaik. Fungsi *fitness* yang digunakan untuk masalah optimasi distribusi barang dua tahap adalah fungsi *fitness* untuk minimasi, dimana nilai *fitness* terbesar dari $F(x)$ yang paling kecil. Fungsi *fitness* untuk minimasi ditunjukkan pada Persamaan 2.9 (Mahmudy, 2013).

$$\text{fitness} = \frac{C}{F(x)} \quad 2.9$$

C adalah *constant* yang ditentukan untuk menghindari pembagian oleh nol. $F(x)$ merupakan total biaya distribusi barang dua tahap sesuai dengan Persamaan 2.1.

2.4.5 Seleksi

Berdasarkan siklusnya seleksi dalam algoritma *evolution strategies* terdiri dari *comma selection* dan *plus selection* (Beyer & Schwefel, 2002). *Comma selection* adalah seleksi yang hanya dilakukan pada *offspring*. Sedangkan *plus selection* adalah seleksi yang dilakukan pada *parent* dan *offspring*. Penelitian ini siklus $(\mu+\lambda)$ dengan proses seleksi melibatkan *parent* (μ) dan *offspring* (λ). Metode

seleksi yang digunakan adalah *elitism selection*. *Elitism selection* dilakukan dengan mengurutkan individu-individu yang memiliki nilai *fitness* tinggi ke rendah. *Elitism selection* menjamin individu yang lolos seleksi adalah individu dengan *fitness* terbaik (Mahmudy, 2013). Individu yang lolos seleksi akan menjadi *parent* pada generasi berikutnya. Gambar 2.6 menunjukkan *pseudocode* seleksi *elitism*.

```
PROCEDURE ElitismSelection
Input:
    POP: himpunan individu pada populasi
    pop_size: ukuran populasi
    OS: himpunan individu anak (offspring) hasil
        reproduksi menggunakan crossover dan mutasi
Output :
    POP: himpunan individu pada populasi setelah proses
        seleksi selesai
    /* gabungkan individu pada POP dan OS ke dalam TEMP */
    TEMP <- Merge (POP,OS)
    /* urutkan individu berdasarkan fitness secara ascending */
    OrderAscending (Temp)
    /* copy pop_size individu terbaik ke POP */
    POP <- CopyBest (Temp, pop_size)
END PROCEDURE
```

Gambar 2.6 *Pseudocode* Seleksi *Elitism*

Sumber: (Mahmudy, 2013)