

BAB II

KAJIAN LITERATUR

Kajian literatur menjelaskan landasan teori yang digunakan dalam penelitian. Landasan teori yang akan dijelaskan dalam penelitian ini adalah mengenai *Supply Chain Management* (SCM), distribusi, optimasi, teori graf, *Vehicle Routing Problem* (VRP), Algoritma *Nearest Neighbor* (ANN) dan Algoritma Genetika (AG).

1.1 Kajian Deduktif

1.1.1 *Supply Chain Management* (SCM)

A.1 Pengertian dan Konsep Dasar *Supply Chain Management* (SCM)

Pada tahun 1990 SCM telah menjadi paradigma dominan bagi lembaga bisnis. Kemunculan SCM dilatar belakangi oleh lingkungan bisnis dan praktek tradisional dalam bisnis (Zabidi.Y, 2001). Menurut Sidarto (2008) produk dan jasa yang telah digunakan pada dasarnya merupakan hasil dari serangkaian proses yang panjang dengan melewati beberapa tahapan fisik maupun non fisik. SCM merupakan rangkaian keseluruhan aktivitas di setiap perusahaan yang diawali dengan kedatangan material dari pemasok (*supplier*), pengolahan material menjadi produk setengah jadi maupun produk jadi hingga kegiatan pendistribusian produk ke konsumen (Miradji M. A., 2014). Menurut Hayati (2014) SCM merupakan serangkaian kegiatan yang mengintegrasikan antara pemasok, produsen, gudang dan toko secara efisien sehingga produk dapat diproduksi dan didistribusikan ke seluruh konsumen dalam jumlah

yang tepat, waktu yang tepat dan meminimalkan biaya. SCM merupakan salah satu solusi terbaik yang dapat diterapkan untuk meningkatkan keunggulan kompetitif di perusahaan (Zabidi, 2001). Keunggulan kompetitif dari SCM di perusahaan dapat dilihat dari bagaimana pengelolaan aliran produk yang dilakukan oleh perusahaan dalam rantai pasok (Watanabe, 2001). SCM memiliki tujuan utama yaitu sebagai proses pengiriman produk secara tepat waktu, meminimasi waktu dan biaya, memusatkan kegiatan perencanaan dan pendistribusian produk, pengelolaan persediaan barang yang baik antara *vendor* dan *buyer* (Pujawan, 2005). Dengan penerapan SCM yang baik, maka kinerja suatu perusahaan semakin terarah sehingga dapat memberikan keuntungan bagi perusahaan, pemasok hingga konsumen (Miradji M. A., 2014).

1.1.2 Distribusi

B.1 Pengertian dan Konsep Distribusi

Distribusi merupakan proses penting di setiap perusahaan guna memperlancar pemasaran produk (Sulistyorini & Mahmud, 2015). Proses distribusi diawali dengan proses pemindahan barang dan penyimpanan barang dari sumber (*source*) hingga sampai ke tujuan (*destination*) dengan biaya yang rendah (Pujawan, 2010). Menurut Tjiptono (2008,185) distribusi dapat dikatakan sebagai suatu kegiatan pemasaran yang bertujuan untuk mempermudah dan memperlancar proses penyampaian produk ke konsumen tepat waktu dan sesuai dengan kebutuhan.

Perusahaan dituntut untuk mampu bersaing dengan perusahaan lain sehingga diperlukan adanya manajemen yang baik agar terciptanya proses kerja yang efektif dan efisien serta perolehan keuntungan yang maksimal. Setiap perusahaan menginginkan peningkatan penjualan produk yang melebihi target, sehingga perusahaan harus ikut berperan aktif dalam aktivitas pendistribusian produk hingga sampai ke tangan konsumen. Untuk dapat mencapai target pasar yang telah ditentukan, perusahaan harus meningkatkan efisiensi dari setiap wilayah distribusi pemasarannya (Prasetyo, 2008).

Willem Siahaya (2013) menyatakan bahwa model distribusi adalah sebagai berikut:

1. Pengiriman langsung

Model distribusi dengan pengiriman langsung dilakukan dengan menggunakan sarana virtual seperti halnya internet sehingga proses pendistribusian dapat berjalan secara langsung tanpa mempertimbangkan peran distributor dan retailer. Tujuan penerapan pengiriman secara langsung adalah perusahaan dapat menekan biaya distribusi.

2. Pengiriman produk melalui transit

Model distribusi ini membutuhkan peran distributor atau retailer untuk mengelola fasilitas transit. Model ini biasa diterapkan untuk produk yang memerlukan komponen lain yang tersedia di berbagai produsen untuk proses perakitan menjadi produk jadi.

3. Distribusi produk melalui distributor

Pada model ini perusahaan memerlukan peran distributor untuk mengantarkan barang hingga ke tangan konsumen. Distributor bertugas mulai dari proses penerimaan produk (*receiving*), penyimpanan (*storage*) hingga proses pendistribusian (*distribution*) terhadap konsumen.

4. Distribusi melalui desentralisasi

Penerapan model distribusi ini memerlukan peran dari distributor. Perusahaan memiliki beberapa distributor di berbagai daerah dengan tujuan untuk mendekatkan produk ke konsumen sesuai dengan segmentasi pasar.

5. Pengambilan langsung oleh konsumen

Model distribusi ini, pengambilan produk dilakukan secara langsung oleh konsumen di suatu tempat yang sudah ditentukan. Perusahaan tidak memerlukan gudang penyimpanan barang karena penyerahan produk menggunakan sistem *cross docking*.

Distribusi merupakan bagian terpenting dalam industri manufaktur terutama industri makanan. Produk makanan merupakan jenis produk yang memiliki umur relatif singkat sehingga industri makanan dituntut untuk menciptakan sistem distribusi yang baik untuk menjamin produk sampai ke tangan konsumen lebih cepat dan mengurangi resiko kerugian. Perkembangan sektor industri yang semakin ketat menuntut sektor industri untuk meningkatkan keandalannya dalam proses pendistribusian agar mencapai pasar tepat waktu (Muma et al., 2014). Dalam proses distribusi, perusahaan disarankan untuk meningkatkan kepedulian terhadap lingkungan dengan meminimalisir polusi serta limbah

(Sutawidjaya et al., 2017). Transportasi merupakan kunci keberhasilan dari proses distribusi. Faktor-faktor pendukung dalam pengembangan transportasi hijau adalah bahan bakar, moda transportasi, infrastruktur dan praktik operasional (Al-Odeh & Smallwood, 2012).

B.2 Saluran Distribusi Pemasaran

Pada prinsipnya, saluran pemasaran merupakan aliran barang dari produsen ke konsumen melalui lembaga pemasaran. Fungsi dari saluran pemasaran adalah untuk melihat serta membandingkan harga dari setiap lembaga pemasaran. Lembaga pemasaran merupakan pedagang yang ikut serta dalam pemasaran produk maupun jasa kekonsumen dengan saluran pemasaran tertentu (Prasetyo, 2008). Menurut Swastha (1990) yang dikutip dari *The American Marketing Assosiation* menyatakan bahwa saluran distribusi merupakan struktur dari organisasi perusahaan yang terdiri dari distributor, dealer, pedagang dan distributor perusahaan yang mengintegrasikan antara pemindahan fisik serta nama suatu produk untuk kepentingan pasar tertentu.

Untuk menentukan jumlah penyalur distribusi, perusahaan dihadapkan dengan tiga alternatif sesuai dengan yang dikemukakan oleh (Swastha,1990) sebagai berikut:

1. Distribusi Intensif

Distribusi intensif dapat dilakukan apabila produsen menjual barang konvenien. Dalam hal ini, produsen menggunakan penyalur terutama pengecer dengan jumlah yang sebanyak-banyaknya untuk mencapai konsumen.

2. Distribusi Selektif

Distribusi selektif ini produsen memilih dan menentukan pedagang besar atau pengecer dalam satu daerah geografis yang sama. Distribusi jenis ini biasa digunakan untuk memperkenalkan produk baru, barang *shooping* atau barang special pemesanan konsumen.

3. Distribusi Eksklusif

Distribusi eksklusif digunakan apabila produsen hanya menggunakan satu pedagang besar atau pengecer di daerah tertentu. Distribusi ini biasanya digunakan pada jenis produk yang memiliki *service* purna jual.

Kelebihan dari distribusi ini adalah produsen dapat mengurangi biaya penyaluran/distribusi produk.

1.1.3 Optimasi

C.1 Definisi Optimasi

Optimasi menjadi masalah yang penting di kalangan industri untuk meningkatkan keberhasilan dan perkembangan berkelanjutan dari perusahaan. Menurut Soekartawi (1992) optimasi merupakan suatu usaha untuk menghasilkan pencapaian terbaik. Secara umum optimasi dapat diartikan sebagai proses untuk mendapatkan nilai terbaik (minimum atau maksimum) dari beberapa fungsi berdasarkan konteks tertentu. Dalam penerapan di perusahaan, optimasi merupakan suatu upaya untuk meningkatkan kinerja sehingga hasil kerja menjadi lebih baik dan berkualitas tinggi.

Secara matematis optimasi dapat diartikan sebagai upaya dalam memperoleh harga esktrim (nilai minimum atau maksimum) berdasarkan suatu fungsi tertentu dan batasan-batasannya (Sari D. P., 2014). Menurut Sinaga (2014) optimasi merupakan suatu proses yang memiliki tujuan untuk mencapai hasil yang optimal (nilai efektif yang dapat dicapai. Dalam mencapai nilai optimal secara sistematis akan dilakukan pemilihan nilai variable integer maupun riil yang akan menghasilkan solusi optimal (Sinaga, 2014).

C.2 Macam- Macam Permasalahan Optimasi

Permasalahan optimasi dapat terjadi dikehidupan sehari-hari. Perolehan nilai optimal dalam optimasi dapat berupa besaran panjang, jarak, waktu. Menurut Sinaga (2014) permasalahan optimasi yang sering terjadi adalah sebagai berikut:

1. Penentuan lintasan terpendek dari suatu tempat ke tempat yang lain.
2. Penentuan jumlah pekerja pada perusahaan seminimal mungkin untuk melakukan suatu proses produksi agar pengeluaran biaya pekerja dapat diminimalkan dan pendapatan hasil produksi tetap maksimal.
3. Mengatur rute dari kendaraan agar semua lokasi dapat terlewati dengan waktu seminimal mungkin.

4. Mengatur *routing* pada jaringan kabel telepon dengan tujuan meminimasi biaya pemasangan kabel dan tidak terjadi pemborosan saat menggunakannya.

Dalam SCM permasalahan optimasi perusahaan yang sering terjadi adalah menentukan sistem transportasi yang optimal dalam proses distribusi perusahaan. Menurut Sari et al. (2013) ciri-ciri khusus dalam permasalahan sistem transportasi adalah sebagai berikut :

1. Terdapat beberapa sumber dan tujuan tertentu.
2. Ketidakpastian jumlah komoditi yang di distribusikan
3. Jumlah barang yang akan di distribusikan sesuai dengan kapasitas sumber.
4. Besarnya biaya pengangkutan barang yang tidak tentu.

Terdapat beberapa metode yang dapat digunakan dalam mengatasi masalah optimasi. Secara umum metode optimasi terbagi menjadi dua jenis yaitu optimasi kombinatorial dan pemrograman matematika. Optimasi kombinatorial (*combinatorial optimization*) proses pencarian solusi dengan biaya minimum (terkecil) dengan menggunakan ilmu komputer teoritis dan matematika terapan, sedangkan teknik pemrograman matematika adalah metode pencarian titik optimum (Moengin, 2011:2-3). Kombinatorial merupakan bagian dari ilmu matematika untuk mengetahui penyusunan suatu objek. Tujuan dari penyusunan objek adalah untuk mengetahui jumlah cara penyusunan objek-objek dengan cara mengenumerisasi kemungkinan susunan objek-objek (Adhipradhana, 2017).

Beberapa permasalahan yang dapat diselesaikan dengan menerapkan optimasi kombinatorial adalah *minimum spanning tree problem*, *Travelling Salesman Problem* (TSP) dan *Vehicle Routing Problem* (VRP) yang merupakan pencarian rute terpendek. Rute terpendek merupakan salah satu permasalahan dalam sistem distribusi dalam menentukan sisi-sisi dalam sebuah jaringan yang menghasilkan rute terpendek dengan jarak terdekat antara titik asal dengan titik tujuan. Menurut Sinaga (2014) permasalahan optimasi dapat diselesaikan dengan dua metode sebagai berikut:

1. Metode Konvensional

Metode konvensional merupakan metode yang menggunakan perhitungan matematika eksak. Metode konvensional yang biasa digunakan adalah

algoritma Dijkstra, algoritma Floyd-Warshall, Algoritma Bellman-Ford (Muthakiroh et al., 2007).

2. Metode Heuristik

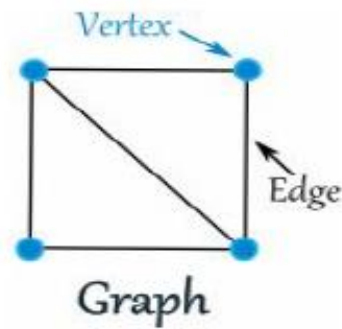
Metode heuristik merupakan metode yang menggunakan suatu system pendekatan untuk menentukan solusi terbaik. Metode yang biasa digunakan dalam penentuan rute terbaik adalah Algoritma Genetika, *Ant Colony Optimization*, logika *Fuzzy*, jaringan syaraf tiruan, *Tabu Search* (Muthakiroh et al., 2007).

1.1.4 GRAF

D.1 Teori Graf

Konsep graf pertama kali diperkenalkan pada tahun 1736 untuk menyelesaikan masalah Jembatan Konigsberg. Salah satu matematikawan Negara Swiss bernama Leonhard Euler berhasil mempelajari permasalahan tersebut dan berhasil mencari solusi dengan menciptakan konsep dari Eulerian Graph. Dengan adanya konsep tersebut, Euler diberi gelar Bapak Teori Graf (Prihatmaja, 2016). Graf merupakan salah satu cabang ilmu matematika dalam kajian matematika diskrit. Graf dapat digunakan sebagai alat untuk merepresentasikan hubungan antara objek-objek diskrit dalam bentuk titik, noktah, bulatan dan adanya garis penghubung antar objek (Saidatuz et al., 2016).

Teori graf merupakan suatu cabang kajian yang mencakup tentang sifat-sifat suatu graf. Graf dapat dikatakan sebagai titik-titik (*vertex* atau *node*) yang terhubung dengan sisi (*edge*) atau busur (*arc*). Simpul dan sisi graf dapat ditunjukkan pada gambar 2.1 berikut.



Gambar 2.1 Simpul dan Sisi Graf

Sumber: Prihatmaja (2016)

Graf menggambarkan sekumpulan titik-titik sebagai simpul yang dihubungkan oleh garis yang melambangkan sisi atau garis berpanah yang melambangkan suatu busur. Dalam teori graf, suatu simpul dapat menghubungkan suatu simpul dengan simpul yang sama dengan istilah gelang (*loop*).

D.2 Jenis Graf

Menurut Sinaga (2014) jenis graf berdasarkan arah pada sisi dapat dibedakan menjadi dua bagian sebagai berikut:

1. Graf tidak berarah (*undirect graf*)

Graf tidak berarah (*undirected graf*) merupakan graf yang tidak memiliki orientasi arah pada sisinya. Graf tersebut tidak memperlihatkan urutan pasangan titik yang dihubungkan oleh sisi. Sehingga dapat dikatakan $(V_j, V_k) = (V_k, V_j)$ dan merupakan sisi yang sama.

Terdapat dua jenis graf tidak berarah yang dapat dibedakan sebagai berikut:

a. Graf tidak berbobot (*unweighted undirected graf*)

Graf tidak berbobot adalah graf yang tidak memiliki bobot untuk setiap sisinya.

b. Graf berbobot (*weighted undirected graf*)

Graf berbobot merupakan suatu graf yang tidak memiliki arah tetapi memiliki bobot di setiap sisinya. Dalam graf tersebut, urutan

pemasangan simpul yang terhubung oleh sisi tidak diperhatikan sehingga $(V_j, V_k) = (V_k, V_j)$ merupakan sisi yang sama. Graf jenis ini sering diaplikasikan pada jaringan saluran telepon karena saluran telepon tidak berarah sehingga dapat beroperasi dengan dua arah.

2. Graf Berarah (*directed graf*)

Graf berarah merupakan salah satu jenis graf yang memiliki arah pada setiap sisinya sehingga $(V_j, V_k) = (V_k, V_j)$ merupakan dua buah sisi yang berbeda. Untuk sisi (V_j, V_k) titik asal merupakan V_j dan titik terminal merupakan V_k

Terdapat dua jenis graf berarah diantaranya adalah sebagai berikut:

a. Graf berarah tidak berbobot

Graf berarah tidak berbobot merupakan jenis graf yang memiliki arah tetapi tidak memiliki bobot.

b. Graf berarah berbobot

Graf berarah berbobot merupakan jenis graf yang setiap sisinya memiliki anak panah (arah) dan juga memiliki bobot. Pada graf tersebut, pasangan simpul yang terhubung oleh sisi dianggap sama, dengan kata lain $(V_j, V_k) = (V_k, V_j)$ merupakan sisi yang sama.

Menurut Saidatuz et al. (2016) jenis graf berdasarkan ada tidaknya gelang maupun sisi ganda pada suatu graf dapat dibedakan menjadi dua sebagai berikut:

1. Graf sederhana (*simple graph*)

Graf sederhana merupakan jenis graf yang tidak memiliki gelang maupun sisi ganda.

2. Graf tak sederhana (*unsimple graph*)

Graf tak sederhana merupakan jenis graf yang memiliki gelang dan mengandung sisi ganda. Graf jenis ini terbagi menjadi dua yaitu sebagai berikut:

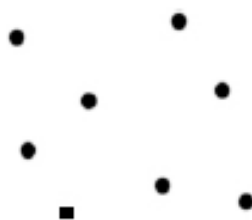
a. Graf ganda (*multigraph*)

b. Graf semu (*pseudograph*)

2.1.5 *Vehicle Routing Problem (VRP)*

VRP merupakan salah satu permasalahan transportasi dalam proses pendistribusian produk kepada konsumen dengan menggunakan kendaraan dengan tujuan meminimasi fungsi tujuan distribusi yang telah ditetapkan (Perwitasari & Subanar, 2013). VRP dapat dikatakan sebagai perumunan dari TSP atau disebut dengan m-TSP, dimana m menggambarkan jumlah *salesman* yang mengunjungi sejumlah kota. Dengan kata lain, VRP merupakan permasalahan dalam penentuan rute terbaik dengan menggunakan kendaraan lebih dari satu dan memuat produk sesuai kapasitas yang telah ditentukan. Rute yang dilalui harus dimulai dan diakhiri di tempat yang sama (depot). Setiap pelanggan yang akan dikunjungi hanya boleh dikunjungi sebanyak satu kali dengan kapasitas yang tidak melebihi kemampuan kapasitas kendaraan (Perwitasari & Subanar, 2013).

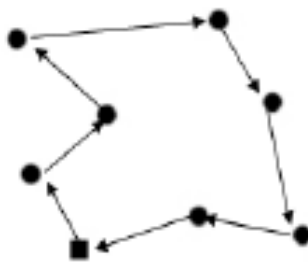
Menurut Kallehauge dan Marsen (2001) VRP dapat menjadi TSP apabila perusahaan hanya memiliki satu kendaraan yang menangkut produk yang kapasitasnya tak terhingga. Dalam proses pencarian rute terpendek, permasalahan kasus TSP direpresentasikan kedalam bentuk graf. Gambar 2.2 merupakan beberapa titik yang mewakili sejumlah kota yang akan dilewati. Dalam penentuan rute terpendek, jarak-jarak tiap kota harus diketahui terlebih dahulu sehingga jarak dapat dihitung berdasarkan koordinat pada masing-masing titik.



Gambar 2.2 Koordinat Titik

Sumber: (Hasibuan & Lusiana, 2015)

Apabila jarak yang menghubungkan antar kota telah diketahui, maka rute terpendek dapat diketahui dengan menghubungkan antar kota dan menjumlahkan jarak dari hasil kombinasi hingga menghasilkan rute yang optimal. Gambar 2.3 menyajikan pembentukan rute antar kota.



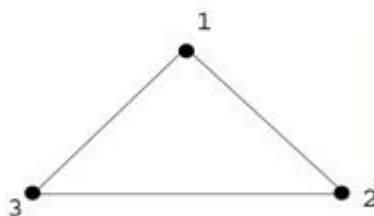
Gambar 2.3 Pembentukan Rute Antar Kota

Sumber: (Hasibuan & Lusiana, 2015)

Penerapan TSP dengan tujuan beberapa kota adalah sebagai berikut:

1. TSP dengan 3 kota

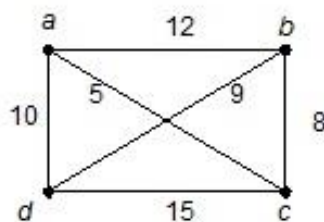
TSP dengan jumlah 3 kota disajikan pada Gambar 2.4 berikut.



Gambar 2.4 TSP Rute 3 Kota

Gambar 2.4 merupakan TSP dengan rute 3 kota sehingga kombinasi jalur yang dilalui tidak perlu diselesaikan menggunakan komputer karena hanya memiliki satu jalur seperti pada gambar.

2. TSP dengan 4 kota



Gambar 2.5 TSP Rute 4 Kota

Sumber: (Wardhani, 2014)

Gambar 2.5 merupakan TSP dengan rute 4 kota sehingga terdapat beberapa kombinasi rute yang dapat dijadikan sebagai jalur optimal. Berdasarkan perhitungan

faktorial terdapat 3 kombinasi rute. Untuk graf yang memiliki n buah kota, secara umum dapat menggunakan rumus sebagai berikut (Wardhani, 2014) :

$$\frac{(n-1)!}{2} = \dots(2.1)$$

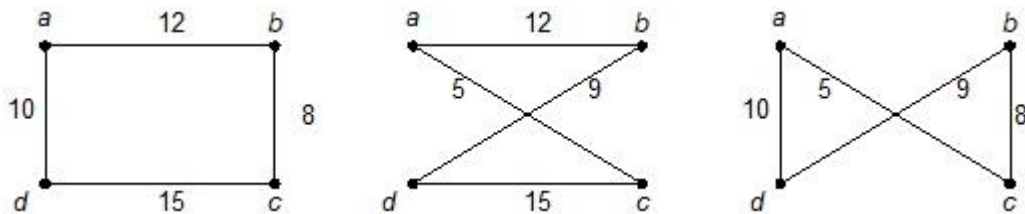
Keterangan:

n = jumlah kota.

Berikut merupakan contoh perhitungan jumlah kombinasi rute dengan 4 kota:

$$\text{Jumlah rute} = \frac{(4-1)!}{2} = 3$$

Sehingga, kombinasi dari semua rute disajikan pada Gambar 2.6 sebagai berikut:



Gambar 2.6 Kombinasi Rute dengan 4 Kota

Sumber : (Wardhani, 2014)

- 1) $L_1 = (a - b - c - d - a)$ atau $(a - d - c - b - a)$,
Panjang rute yang harus dilalui sebesar = 45, $(10 + 12 + 8 + 15 = 45)$
- 2) $L_2 = (a - c - d - b - a)$ atau $(a - b - d - c - a)$,
Panjang rute yang harus dilalui sebesar = 41, $(12 + 5 + 9 + 15 = 41)$
- 3) $L_3 = (a - c - b - d - a)$ atau $(a - d - b - c - a)$
Panjang rute yang harus dilalui sebesar 32, $(10 + 5 + 9 + 8 = 32)$

Berdasarkan perhitungan tersebut, rute optimal adalah pada jalur L_3 dengan panjang lintasan sebesar 32.

2.1.6 Karakteristik *Vehicle Routing Problem* (VRP)

VRP memiliki beberapa karakteristik yang perlu diperhatikan dalam penentuan rute pendistribusian (Toth & Vigo, 2002). Yang pertama merupakan komponen-komponen dalam VRP adalah sebagai berikut:

1. Pelanggan
2. Depot
3. Pengemudi
4. Rute kendaraan

Karakteristik lain dari VRP adalah dalam tujuan fungsi kendala yang telah ditentukan dalam kasus VRP. Menurut Wirasambada dan Handayani (2016) berdasarkan kendala yang ditentukan, VRP dapat dibagi menjadi beberapa tipe seperti berikut:

a. CVRP (*Capacitated Vehicle Routing Problem*)

CVRP merupakan salah satu model dasar VRP dengan menggunakan kapasitas angkut kendaraan sebagai kendala yang dihadapi. Jumlah permintaan setiap pelanggan sudah diketahui di awal. Kendaraan yang bertugas untuk mengantarkan produk adalah identik dan hanya memiliki satu depot sebagai lokasi awal dan akhir setiap kendaraan.

b. VRPTW (*Vehicle Routing Problem with Time Windows*)

VRPTW merupakan salah satu model dasar VRP dengan memperhatikan ketersediaan waktu pelayanan dari pelanggan. Pelanggan memiliki ketentuan jangka waktu tertentu untuk dilayani yang disebut dengan *time windows*. Setiap kendaraan yang melayani sejumlah pelanggan harus memperhitungkan waktu kendaraan meninggalkan depot, waktu menuju lokasi pelanggan dan waktu pelayanan yang dimiliki pelanggan.

c. VRPB (*Vehicle Routing Problem with Backhauls*)

Konsep dari VRPB adalah membagi pelanggan dalam dua kondisi. Kondisi pertama dimana pelanggan memiliki permintaan untuk dikirimkan barang pesannya. Kondisi kedua adalah pelanggan memiliki permintaan untuk diambil barangnya. Rute yang ditentukan dalam permasalahan tersebut adalah dengan melakukan proses pengantaran terlebih

dahulu sebelum proses pemasukan barang. Hal ini dilakukan untuk menghindari pemuatan ulang barang-barang di kendaraan.

d. VRPPD (*Vehicle Routing Problem Pick Up and Delivery*)

Konsep VRPPD dihadapkan dengan pelanggan yang terasosiasi pada dua jenis permintaan sekaligus. Pelanggan menginginkan untuk kirimkan barang ke lokasinya dan diambilkan barang dari lokasinya. Kegiatan mengantarkan barang dilakukan terlebih dahulu sebelum kegiatan pengambilan barang. Hal tersebut dilakukan karena mengingat pelanggan hanya memiliki satu kesempatan untuk dikunjungi.

e. MDVRP (*Multi-Depot Vehicle Routing Problem*)

Konsep dari MDVRP adalah pengembangan dari CVRP dengan mempertimbangkan kapasitas angkut dari kendaraan dan menambahkan multi depot dalam permasalahan CVRP.

f. OVRP (*Open Vehicle Routing Problem*)

Konsep dari OVRP adalah suatu permasalahan yang sama dengan CVRP. Akan tetapi, dalam OVRP kendaraan tidak perlu kembali ke depot setelah selesai melakukan pengantaran produk.

2.1.7 Definisi Permasalahan dan Notasi VRPTW

Menurut Wirasambada dan Handayani (2016) Permasalahan penentuan rute kendaraan pada proses pendistribusian produk di suatu perusahaan tergolong dalam permasalahan *Vehicle Routing Problem with Time Windows* (VRPTW). Model matematis dari VRPTW telah dikembangkan oleh Kallehauge et al. (2002) dengan fungsi tujuan meminimalkan total *cost* dengan batasan bahwa pelanggan hanya dapat dilayani satu kali dengan batasan waktu pelayanan pelanggan (*time windows*) dan kapasitas, melalui rute yang berawal dan berakhir dengan depot yang sama. *Time windows* pada setiap pelanggan dapat berbeda satu sama lain yang dinyatakan dengan selang waktu berupa batas awal dan batas akhir waktu pelayanan pada pelanggan tersebut.

Secara umum, VRPTW dinyatakan dalam suatu bentuk graf $G = (N, A)$. N dinyatakan sebagai himpunan yang terdiri dari kumpulan dari himpunan pelanggan C dan depot. Himpunan C terdiri dari simpul 1 sampai dengan simpul n . Sementara itu, depot

dinyatakan sebagai simpul 0 sebagai awalan dan $n+1$ sebagai depot akhir (Salaki & Rindengan, 2010). Rute jalan yang digunakan selama perjalanan merupakan himpunan sisi berarah yang dinyatakan sebagai A (penghubung antar simpul). Setiap kendaraan yang digunakan dalam pengantaran produk dinyatakan sebagai himpunan kendaraan dengan notasi V dengan kapasitas q (Salaki & Rindengan, 2010).

VRPTW memiliki tujuan untuk menentukan himpunan rute yang memiliki jarak tempuh minimal atau biaya perjalanan minimal dengan syarat setiap pelanggan hanya dilayani sebanyak satu kali yang berawal dari simpul 0 dan diakhiri dengan simpul $n+1$ serta dapat memenuhi kendala kapasitas dan *time windows*.

Notasi model matematis Kallehauge et al. (2002) diantaranya adalah sebagai berikut:

V	= kumpulan kendaraan dengan kapasitas yang sama
C	= kumpulan pelanggan (<i>customer</i>)
A	= himpunan sisi berarah $\{(i,j) \mid i,j \in N, i \neq j\}$
N	= himpunan simpul
c_{ij}	= jarak/biaya dari simpul i ke simpul j (bilangan tak negatif)
t_{ij}	= waktu tempuh dari simpul i ke simpul j (bilangan tak negatif)
d_i	= jumlah permintaan pelanggan (bilangan integer dan tak negatif)
q	= kapasitas kendaraan (bilangan integer dan tak negatif)
x_{ij}	= kendaraan k dari pelanggan i menuju pelanggan j
s_{ik}	= waktu kendaraan k mulai melayani pelanggan i
$[\alpha_i, \beta_i]$	= <i>time windows</i> dari simpul i (bilangan tak negatif)
$[\alpha_0, \beta_0]$	= <i>time windows</i> depot (bilangan tak negatif)

Index yang digunakan dalam kasus VRPTW adalah

i	= indeks yang menyatakan pelanggan awal, $i = 1,2,3,\dots,N$ ($i \in C$)
j	= indeks yang menyatakan pelanggan tujuan, $j = 1,2,3,\dots,N$
k	= indeks yang menyatakan kendaraan, $k = 1,2,3,\dots,N$

Dalam setiap $(i,j) \in A$, $i \neq n+1$, $j \neq 0$ dan setiap kendaraan k didefinisikan sebagai berikut:

$$x_{ijk} = \begin{cases} 0, & \text{kendaraan } k \text{ tidak melakukan perjalanan dari titik } i \text{ ke titik } j \\ 1, & \text{kendaraan melakukan perjalanan dari titik } i \text{ ke titik } j \end{cases}$$

Formulasi matematis menurut Kallehauge et al. (2002) adalah sebagai berikut:

$$Z_{VRPTW} = \text{minimize} \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk} \quad \dots(2.2)$$

Batasan:

$$\sum_{k \in V} \sum_{i \in N} x_{ijk} = 1 \quad \forall_i \in C \quad \dots(2.3)$$

$$\sum_{i \in C} d_i \sum_{j \in N} x_{ijk} \leq q \quad \forall_k \in V \quad \dots(2.4)$$

$$\sum_{j \in N} x_{0jk} = 1 \quad \forall_k \in V \quad \dots(2.5)$$

$$\sum_{i \in N} x_{ik} - \sum_{j \in N} x_{jk} = 0 \quad \forall_h \in C, \forall_k \in V \quad \dots(2.6)$$

$$\sum_{i \in N} x_{i,n+1,k} = 1 \quad \forall_k \in V \quad \dots(2.7)$$

$$s_{ik} + t_{ij} - K(1 - x_{ijk}) \leq s_{jk} \quad \forall_{i,j} \in N, \forall_k \in V \quad \dots(2.8)$$

$$a_i \leq s_{ik} \leq b_i \quad \forall_i \in N, \forall_k \in V \quad \dots(2.9)$$

$$x_{ijk} \in \{0,1\} \quad \forall_{i,j} \in N, \forall_k \in V \quad \dots(2.10)$$

Berdasarkan fungsi turunan tersebut, Batasan (2.2) menyatakan bahwa tujuan dari model adalah untuk meminimasi biaya perjalanan (Kallehauge, et al., 2002). Batasan (2.3) menyatakan setiap pelanggan yang akan dilayani hanya dikunjungi sebanyak satu kali. Batasan (2.4) menyatakan bahwa jumlah kapasitas tidak diperbolehkan lebih dari batas kapasitas kendaraan. Batasan (2.5) menyatakan setiap kendaraan berawal dari depot. Batasan (2.6) menyatakan apabila kendaraan telah selesai mengunjungi satu pelanggan, maka kendaraan akan melanjutkan ke pelanggan selanjutnya. Batasan (2.7) menyatakan setiap kendaraan akan berakhir di depot. Batasan (2.8) menyatakan bahwa kendaraan k tidak diperbolehkan sampai di pelanggan j sebelum $s_{ik} + t_{ij}$ atau sebelum waktu perjalanan dari i ke j dan waktu dimulai pelayanan dengan K merupakan bilangan riil bernilai besar. Batasan (2.9) menyatakan bahwa *time windows* dari masing-masing pelanggan terpenuhi.

Batasan (2.10) menyatakan bahwa x_{ij} bernilai biner. Kendaraan yang tidak melakukan pelayanan memiliki rute kosong 0, $n+1$.

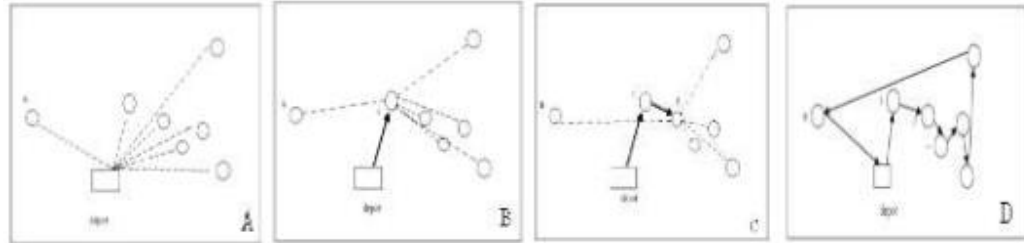
2.1.8 Metode *Nearest Neighbor*

Metode *Nearest Neighbor* pertama kali diperkenalkan pada tahun 1983 dan merupakan metode yang sederhana. Dalam proses iterasinya dengan mencari jarak terdekat dari pelanggan yang terakhir untuk ditambahkan dalam rute pengantaran. Pencarian rute akan dilakukan dengan pencarian rute baru apabila terdapat kendala kapasitas permintaan dan *time windows* yang menyebabkan tidak terdapat posisi yang fisibel untuk menempatkan pelanggan dalam rute tersebut (Braysy & Gendreau, 2005). Dalam proses penentuan rute, langkah pertama yang dilakukan adalah dengan mencari jarak pelanggan paling dekat dengan depot lalu dilanjutkan dengan memasukan satu persatu pelanggan terdekat kedalam satu rute. Dalam memasukan pelanggan yang saling berdekatan dalam satu rute yang sama, kendaraan tidak boleh melanggar batasan kapasitas maksimum kendaraan atau melewati batasan-batasan VRP yang telah ditentukan (Amri et al., 2014). Proses yang sama terus dilakukan hingga kendaraan menemukan rute distribusi tercepat dan semua pelanggan telah dikunjungi (Gunawan, 2012).

Langkah-langkah dalam metode *Nearest Neighbor* adalah sebagai berikut (Pop, 2011) :

1. Dimulai dari suatu depot yang menjadi pusat utama rute pengantaran, kemudian pencarian lokasi pelanggan yang paling dekat dengan depot.
2. Lanjutkan pencarian lokasi pelanggan terdekat dari lokasi pelanggan yang telah terpilih sebelumnya dan tidak melebihi kapasitas kendaraan.
 - a. Apabila terdapat lokasi pelanggan yang terpilih sebagai tujuan berikutnya dan tidak melebihi kapasitas dari kendaraan maka diulang kembali pada langkah kedua (2).
 - b. Apabila melebihi kapasitas maka diulang kembali seperti pada langkah pertama (1)
 - c. Apabila tidak terdapat lokasi yang terpilih karena jumlah pengiriman melebihi kapasitas kendaraan, maka diulang kembali seperti langkah pertama (1) dimulai dari gung dan dilanjutkan dengan pelanggan yang belum dikunjungi dengan jarak paling dekat.

3. Apabila keseluruhan pelanggan telah dikunjungi maka algoritma berakhir dan dapat dihentikan. Penjelasan proses metode *Nearest Neighbor* dijelaskan seperti Gambar 2.7 di bawah ini:



Gambar 2.7 Langkah Metode *Nearest Neighbor*

Sumber: (Amri et al., 2014)

Berdasarkan Gambar 2.7 langkah A menunjukkan depot dengan beberapa lokasi pelanggan, kemudian proses pencarian lokasi pelanggan terdekat dengan depot yang ditunjukkan dengan gambar B. Langkah selanjutnya pencarian pelanggan terdekat dengan pelanggan yang sudah ditentukan sebelumnya seperti gambar C. Proses terus berjalan hingga menemukan rute terdekat seperti gambar D. Pada VRP, metode *Nearest Neighbor* dapat digunakan dalam pembentukan *cluster* pelanggan. *Cluster* merupakan suatu kelompok yang terdiri dari suatu objek berdasarkan kesamaan karakteristik tertentu (Medriosa, 2014).

2.1.9 Algoritma Genetika (AG)

E.1 Pengertian Algoritma Genetika (AG)

AG merupakan salah satu algoritma pencarian heuristik dengan menerapkan evolusi biologi berupa variasi dari kromosom yang dimiliki setiap individu organisme variasi dari kromosom berperan penting dalam laju reproduksi dan tingkat bertahan hidup organisme (Kusumadewi & Purnomo, 2005). Terdapat 4 kondisi yang dapat berpengaruh pada proses evaluasi yaitu sebagai berikut (Kusumadewi & Purnomo, 2005) :

1. Kemampuan organisme dalam melakukan reproduksi
2. Keberadaan populasi organisme yang dapat bereproduksi

3. Keberadaan organisme di dalam suatu populasi
4. Perbedaan kemampuan untuk *survive*.

Individu dikatakan lebih kuat apabila individu tersebut memiliki kemampuan *survival* dan reproduksi yang lebih tinggi dibandingkan dengan individu lainnya.

Awal mulai AG dikembangkan oleh John Holland dari Universitas Michigan (1975) yang menyatakan bahwa setiap masalah yang dapat beradaptasi (alam maupun buatan) dapat ditransformasikan kedalam bentuk terminologi genetika. AG merupakan sebuah simulasi dari teori evolusi Darwin dan operasi genetika dalam kromosom. AG berasal dari suatu himpunan solusi yang dihasilkan secara acak dan dari himpunan acak tersebut akan menghasilkan populasi (Sundarningsih et al., 2017).

Populasi terdiri dari individu yang disebut dengan kromosom. Di dalam kromosom akan terjadi proses iterasi yang akan menghasilkan generasi baru. Hal itu merupakan cara kromosom untuk berevolusi (Sundarningsih et al., 2017). Menurut Gen dan Cheng (1997) setiap generasi, kromosom yang terbentuk akan di evaluasi sesuai dengan suatu fungsi evaluasi. Setelah mengalami beberapa iterasi, hasil generasi dari kromosom akan mengalami konvergen pada kromosom terbaik sehingga dihasilkan solusi yang optimal (Goldberg, 1989).

Penerapan algoritma harus disesuaikan dengan permasalahan yang terjadi. Permasalahan tersebut harus memiliki fungsi tujuan yang disebut sebagai fungsi *fitness*. Hasil perolehan nilai fungsi *fitness* yang menunjukkan nilai semakin besar maka solusi yang dihasilkan akan semakin baik (Sundarningsih et al., 2017). Menurut Basuki (2003), berikut merupakan ciri-ciri dari permasalahan yang dapat diselesaikan dengan metode AG:

1. Memiliki fungsi tujuan dengan kendala dalam bentuk non linear.
2. Memiliki hasil solusi yang berkombinasi bahkan tak terhingga.
3. Solusi yang dihasilkan dapat diimplementasikan dari permasalahan untuk perubahan yang lebih baik.
4. Terdiri dari beberapa kriteria atau *multi-objective* yang dapat diterima oleh semua pihak.

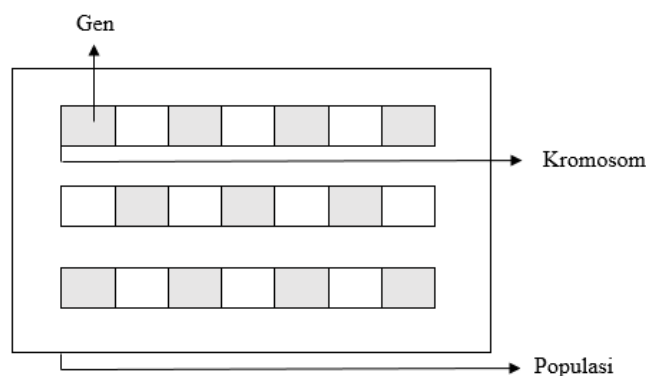
Dalam AG variabel dan parameter-parameter yang digunakan adalah sebagai berikut:

1. Fungsi *fitness* yang dimiliki oleh setiap individu sebagai nilai kesesuaian dengan kriteria yang diinginkan.
2. Populasi.
3. Jumlah generasi.
4. Probabilitas *crossover*.
5. Probabilitas mutasi.

E.2 Struktur Algoritma Genetika (AG)

Teknik pencarian AG berdasarkan sejumlah solusi yang dikatakan dengan populasi. Bagian dari populasi terdiri dari individu-individu yang dinamakan dengan kromosom. Penentuan populasi awal dapat dilakukan secara acak, sedangkan populasi berikutnya merupakan hasil dari proses iterasi kromosom-kromosom yang disebut dengan generasi. Generasi hasil dari iterasi dinyatakan sebagai anak (*off-spring*) yang terbentuk dari dua kromosom induk (*parent*) pada proses persilangan (*crossover*). Proses modifikasi kromosom selanjutnya adalah mutasi (Kusumadewi & Purnomo, 2005).

Kualitas dari kromosom dapat ditentukan dengan fungsi *fitness*. Populasi kromosom baru dapat ditentukan dengan menyeleksi kromosom induk (*parent*) dan kromosom anak (*offspring*) dan kromosom yang memiliki kualitas rendah tidak digunakan untuk menghasilkan populasi yang konstant dan kromosom yang terbaik (konvergen) (Kusumadewi & Purnomo, 2005). Struktur dari kromosom seperti pada Gambar 2.8 berikut.



Gambar 2.8 Struktur Kromosom

E.3 Langkah-langkah Algoritma Genetika (AG)

Berikut merupakan langkah-langkah dari metode AG:

1. Teknik Pengkodean

Teknik pengkodean merupakan langkah dalam mengkodekan gen dari suatu kromosom meliputi pengkodean dalam bentuk *string bit*, *tree*, *array* bilangan *real*, daftar aturan, elemen permutasi, elemen program dan lainnya yang merepresentasikan operator genetika (Merry, 2011).

2. Inisialisasi

Pada tahap inisialisasi menghasilkan jumlah individu secara acak (*random*). Proses pembentukan populasi awal secara *random* sejumlah n-kromosom sesuai dengan permasalahan terkait dan penentuan jumlah generasi. Setelah itu, tahap selanjutnya melakukan inisialisasi kromosom (Merry, 2011).

3. Fungsi Evaluasi (Penentuan nilai *Fitness*)

Penentuan fungsi tujuan dan kendala dari permasalahan lalu mengevaluasi setiap nilai *fitness* $f(x)$ dari masing-masing kromosom x pada populasi. Perhitungan nilai *fitness* dilakukan untuk mengetahui ukuran kinerja dari kromosom. Kromosom dengan hasil nilai *fitness* tinggi akan dipertahankan, sedangkan kromosom dengan perolehan *fitness* rendah akan diganti.

4. Populasi baru

Tahap selanjutnya adalah menciptakan populasi baru untuk mendapatkan solusi optimal. Adapun langkah-langkahnya adalah sebagai berikut (Obitko, 1998):

1) Seleksi

Pemilihan kromosom berdasarkan hasil nilai *fitness* terbesar sehingga dapat ditentukan kromosom induk. Nilai *fitness* yang semakin besar menunjukkan bahwa kromosom tersebut tepat untuk dijadikan solusi (Mahmudy, 2015). Menurut Karas dan Umit (2011) *fitness* merupakan fungsi yang dimiliki oleh masing-masing individu untuk menentukan tingkat kesesuaian individu dengan kriteria yang ingin dicapai.

Dalam penerapan AG, nilai yang akan dijadikan acuan dalam penentuan solusi optimal adalah nilai *fitness* (Basuki, 2003). Berikut merupakan fungsi untuk menghitung nilai *fitness* pada persamaan 2.12.

$$fitness = \frac{1}{(fx)} \quad \dots(2.12)$$

Dimana: $fx = \Sigma(\alpha Dij) + \Sigma(\alpha P)$

Keterangan:

Dij adalah jarak tempuh dari titik i ke titik j .

P merupakan *Penalty* apabila *customer* dilayani diluar jadwal.

Terdapat beberapa metode dalam melakukan proses seleksi kromosom yaitu sebagai berikut:

a. *Roulette Wheel Selection*

Proses penyeleksian dengan metode *roulette wheel selection* dengan menyeleksi *parent* bertujuan untuk mendapatkan kromosom dengan nilai terbaik sebagai kromosom induk. Kromosom dengan perolehan nilai *fitness* terbesar memiliki peluang untuk dipilih beberapa kali. Keunggulan dari metode *roulette wheel selection* adalah proses kalkulasi yang mudah dan sederhana.

b. Seleksi *Good Fitness*

Proses penyeleksian dengan metode seleksi *good fitness* dengan menghilangkan setengah dari jumlah populasi yang memiliki nilai *fitness* rendah dan mempertahankan populasi dengan perolehan nilai *fitness* tinggi. Solusi hasil perolehan seleksi tersebut dinyatakan sebagai populasi induk (*parent*). Jumlah populasi harus tetap sehingga perlu dilakukan proses *generate* sebanyak setengah dari jumlah populasi yang ada dengan tujuan menemukan alternatif solusi yang lebih baik. Terdapat dua cara dalam proses *generate* solusi baru yaitu dengan reproduksi kromosom baru dan melakukan mutasi dari solusi induk.

2) *Crossover* (Persilangan)

Crossover merupakan suatu proses dalam tahap AG dengan mengkombinasikan dua kromosom induk (*parent*) untuk menghasilkan kromosom keturunan (*offspring*) pada waktu tertentu. Proses pengkombinasian ini disebut juga dengan proses persilangan kromosom. Proses *crossover* dapat dilakukan dengan cara

menunjukkan nilai gen yang dihasilkan dari dua *parent* secara *random*. Probabilitas *crossover* digunakan dalam setiap generasi dalam populasi yang akan mengalami proses *crossover*. Nilai yang disarankan adalah 80%-95% (Suhartono, 2011). Secara umum, langkah dalam proses *crossover* adalah sebagai berikut:

- a. Memilih dua buah kromosom yang dipilih untuk menjadi induk.
- b. Memilih secara *random* posisi dalam kromosom, biasa disebut dengan titik *crossover*, sehingga masing-masing kromosom induk terpecah menjadi dua segmen.
- c. Melakukan pertukaran antar gen dari kromosom induk (*parent*) untuk menghasilkan kromosom anak (*offspring*). Berikut merupakan contoh proses *crossover*:

Diketahui jumlah populasi sebanyak 20 kromosom dengan probabilitas *crossover* (p_c) = 0,2 maka jumlah kromosom yang akan mengalami proses *crossover* sebanyak 5 kromosom. Dalam pemilihan kromosom untuk proses *crossover* dengan cara menggunakan bilangan *random* antara 0-1 sebanyak 20 buah. Contoh perolehan bilangan *random* seperti pada Tabel 2.1 berikut.

Tabel 2.1 Bilangan *Random* Kromosom

<i>Chr</i>	Bil.	<i>Chr</i>	Bil.	<i>Chr</i>	Bil.	<i>Chr</i>	Bil.
k-1	0,38	k-6	0,96	k-11	0,90	k-16	0,05
k-2	0,10	k-7	0,59	k-12	0,10	k-17	0,12
k-3	0,04	k-8	0,6	k-13	0,85	k-18	0,11
k-4	0,49	k-9	0,84	k-14	0,48	k-19	0,96
k-5	0,47	k-10	0,46	k-15	0,80	k-20	0,85

Sumber: (Kusumadewi & Purnomo, 2005)

Tentukan bilangan *random* (r_s) kurang dari nilai p_c untuk menentukan kromosom yang akan mengalami *crossover*. Terdapat enam kromosom terpilih yang dapat melakukan *crossover* yaitu kromosom k-2, k-3, k-12, k-

16,k-17,k-18. Proses *crossover* dilakukan oleh dua kromosom (k-2 dengan k-3), (k-12 dengan k-16), (k-17 dengan k-18). Untuk memilih posisi *crossover* digunakan bilangan *random* 1 hingga (1-L), dimana L jumlah gen kromosom (Kusumadewi & Purnomo, 2005). Dimisalkan $L=18$, dan bilangan *random* = 4 maka proses *crossover* kromosom k-2 dengan k-3 ditunjukkan pada Gambar 2.9.



Gambar 2.9 Proses *Crossover*.

Keterangan :

k-2 = kromosom induk (*parent*)

k-3 = kromosom induk (*parent*)

k'-2 = kromosom anak (*offspring*)

k'-3 = kromosom anak (*offspring*)

3) Mutasi

Tahap selanjutnya adalah proses mutasi. Setelah kromosom mengalami proses *crossover*, langkah berikutnya dapat dilakukan mutasi. Mutasi dilakukan dengan cara melakukan perubahan pada satu gen atau lebih dari individu. Akan tetapi tidak semua keturunan dari *crossover* mengalami mutasi. Banyaknya *parent* yang mengalami mutasi akan bergantung pada probabilitas mutasi yang ditentukan. Mutasi berfungsi untuk meningkatkan variasi populasi karena dapat menggantikan gen yang hilang dari populasi selama proses seleksi dan dapat menghasilkan gen yang tidak terdapat dalam populasi awal.

Probabilitas mutasi ditentukan dalam proses mutasi untuk digunakan dalam setiap generasi. Peluang mutasi lebih kecil dibandingkan dengan peluang *crossover* dengan nilai probabilitas yang disarankan antara 0,5%-1%

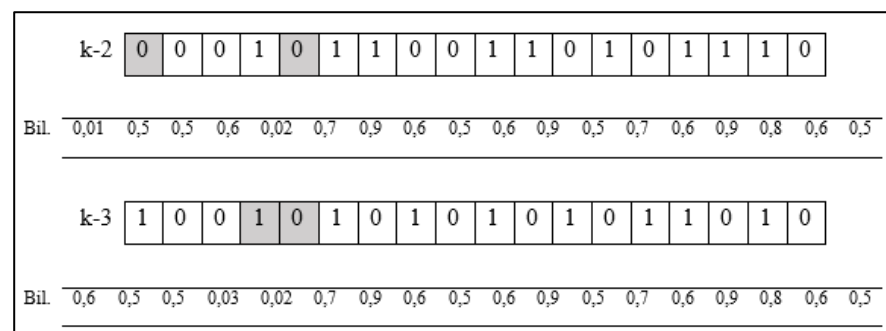
(Suhartono, 2011). Menurut Kusumadewi dan Purnomo (2005) probabilitas mutasi yang terlalu kecil dapat menyebabkan gen yang mungkin berguna tidak terevaluasi, sedangkan apabila probabilitas terlalu besar akan memunculkan gangguan acak sehingga anak akan kehilangan kemiripan dari induknya. Berikut merupakan contoh proses mutasi pada kromosom:

Diketahui jumlah populasi 15 kromosom dengan $L = 18$ disetiap kromosom dan peluang mutasi (p_m) adalah 0,01 yang berarti jumlah bit yang akan mengalami mutasi sebesar 1% dari jumlah keseluruhan bit ($2,7=3$) (Kusumadewi & Purnomo, 2005). Untuk menghitung jumlah bit dapat menggunakan Persamaan 2.13.

$$\text{Jumlah bit} = \text{popsize} \times L \quad \dots(2.13)$$

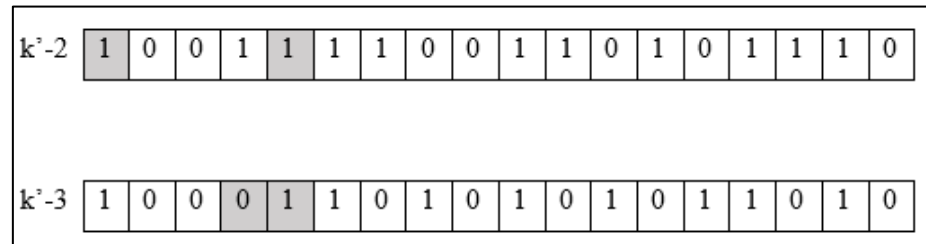
$$270 = 15 \times 18$$

Pemilihan bit yang akan melakukan mutasi dengan cara membangkitkan bilangan *random* sebanyak 270. Diketahui nilai $p_m = 0,01$. Proses pembangkitan bilangan *random* ditunjukkan pada Gambar 2.10.



Gambar 2.10 Pembangkitan Bilangan *Random*

Pada proses mutasi akan menghasilkan anak kromosom seperti pada Gambar 2.11 berikut.



Gambar 2.11 Proses Mutasi.

Bit yang memiliki nilai bilangan random (r_s) yang memenuhi nilai probabilitas mutasi 0,01 adalah bit 1 dan 5 dalam kromosom 2, bit 4 dan 5 pada kromosom 3. Sehingga menghasilkan kromosom anak (*offspring*).

5. Penerimaan

Beberapa proses sebelumnya akan menghasilkan kromosom baru, sehingga populasi yang dapat dijadikan solusi penyelesaian masalah akan bertambah.

6. Penggunaan populasi baru

Populasi baru yang dihasilkan dapat digunakan untuk proses algoritma berikutnya.

7. Tes atau pengujian

Apabila kondisi akhir terpenuhi, dan proses berhenti dengan menunjukkan nilai *fitness* tertinggi maka hasil tersebut merupakan solusi terbaik.

E.4 Kelebihan dan Kekurangan Algoritma Genetika (AG)

Menurut Wang et al. (2014) dan Tanujaya et al. (2011) metode AG memiliki beberapa kelebihan dan kekurangan diantaranya adalah sebagai berikut:

1. Kelebihan AG

- Dapat mengoptimasi parameter yang optimal dengan ruang lingkup yang besar.
- Solusi yang buruk tidak mempengaruhi solusi akhir karena AG langsung menghilangkan solusi yang tidak optimal.
- Dapat memberikan informasi mengenai stabilitas dari solusi.
- Dapat digunakan untuk menghasilkan solusi optimasi yang multidimensional.

- e. Proses pengolahan AG tidak memerlukan kriteria khusus seperti pada metode heuristik lainnya sehingga proses komputasi lebih cepat dan mendapatkan solusi optimal dengan nilai obyektif yang sama.

2. Kekurangan AG

- a. Menghasilkan solusi yang berbeda-beda karena menggunakan bilangan *random*.
- b. Nilai solusi optimal bergantung pada parameter-parameter yang digunakan dalam AG.

AG merupakan salah satu metode numerik dengan menggunakan system pendekatan untuk menentukan solusi terbaik dan dapat dikatakan sebagai metode *heuristic* (Muthakiroh et al., 2007). Tabel 2.2 dan Tabel 2.3 akan menjelaskan kelebihan dan kekurangan dari metode numerik, metode analitik dan metode simulasi (Ekoanindyo, 2011).

Tabel 2.2 Perbandingan Kelebihan Metode

Metode Numerik	Metode Analitik	Metode Simulasi
1. Komputasi dan perhitungan tidak membutuhkan waktu yang lama.	1. Operasi perhitungan yang sudah baku. 2. Nilai yang dihasilkan adalah <i>exact</i> (tepat)	1. Dapat memberikan gambaran dan memodelkan permasalahan secara <i>random</i> . 2. Model simulasi memberikan evaluasi strategi <i>improvement</i> . 3. Dapat mengatasi permasalahan yang kompleks. 4. Tidak membutuhkan waktu yang cukup lama.
2. Dapat dikomputasikan dengan bantuan computer.		
3. Menghasilkan solusi dari persoalan.		
4. Dapat mengatasi permasalahan yang kompleks.		

Tabel 2.3 menyajikan perbandingan kekurangan metode seperti berikut.

Tabel 2.3 Perbandingan Kekurangan Metode

Metode Numerik	Metode Analitik	Metode Simulasi
1. Perubahan parameter akan mengubah hasil solusi.	1. Memakan banyak waktu. 2. Terkadang tidak menemukan solusi.	1. Simulasi bukan merupakan presisi dan proses optimasi.
2. Tanpa bantuan alat hitung, perhitungan membutuhkan waktu lama dan beberapa iterasi.		2. Simulasi tidak menghasilkan solusi tetapi menghasilkan cara untuk menilai solusi. 3. Tidak semua permasalahan dapat dinilai melalui simulasi.

Penelitian ini menggunakan salah satu metode numerik yaitu Algoritma Genetika (AG). Algoritma Genetika dinilai memiliki hasil yang optimal untuk banyak masalah dengan banyak obyektif sehingga hal ini membuktikan AG dapat menghasilkan himpunan solusi optimal dan kekuatan utama AG adalah kemampuannya untuk menyelesaikan permasalahan kompleks dalam waktu relatif cepat (Mahmudy et al., 2014). Pada dasarnya permasalahan dalam optimasi dapat diselesaikan dengan menggunakan metode model MILP, *robust optimization*, serta *branch and bound* akan tetapi metode tersebut jumlah perhitungan yang dilakukan cukup besar dan lama.

2.2 Kajian Induktif

Terdapat beberapa penelitian terahulu mengenai kasus optimasi diantaranya adalah penelitian Pradhana et al. 2012 melakukan penelitian dengan menerapkan Algoritma *Tabu Search* untuk mengatasi permasalahan VRP. Penelitian dilakukan di IT COMM Bantul yang memiliki subdistribusi. Simpulan yang diperoleh dalam penelitian ini adalah rute optimum dengan jarak tempuh 79 km dari rute awalan dengan jarak tempuh 88 km. Nugroho (2015) melakukan penelitian untuk menentukan rute kendaraan dengan

pemerataan beban menggunakan AG. Penelitian ini menggunakan populasi sebanyak 100, p_c sebesar 0,5 dan p_m sebesar 0,01 dan menghasilkan nilai *fitness* sebesar 822995,48. Penghematan yang terjadi pada biaya solar sebesar Rp 170.625,00 dan Rp 171.640,00.

Yulia dan Singgih 2012 menerapkan metode Linear Programming dan Algoritma *Cross Entropy* pada permasalahan optimasi produk dan distribusi perusahaan gas cair. Hasil pengolahan optimasi produk yang dilakukan didapatkan hasil peningkatan profit manufaktur perusahaan dari bulan Januari 2010 hingga bulan Mei 2010. Peningkatan bulan Januari 2010 sebesar 0,002096%, Februari 2010 sebesar 0,018349%, Maret 2010 sebesar 0,00019%, April 2010 sebesar 0,000942% dan Mei 2010 sebesar 0,001806%.

Penelitian selanjutnya Rizky et al. (2017) menerapkan AG pada kasus optimasi *multi travelling* pada industri tekstil. Jumlah populasi optimal yang digunakan sebanyak 120, jumlah generasi yang digunakan sebanyak 800. Terdapat 2 *salesman* dengan data 26 tempat sebagai agen yang harus di layani setiap *salesman*. Input data yang digunakan adalah jarak antar agen maupun depot. Hasil dari pengolahan tersebut nilai *fitness* terbesar adalah 2,9964 dengan $p_c = 0,4$ dan $p_m = 0,6$.

Suyudi et al. (2015) melakukan penelitian pada permasalahan rute distribusi menggunakan metode Algoritma *Nearest Neighbor* dan *Local Search*. Hasil dari pengolahan data jarak menghasilkan jarak yang lebih pendek dengan selisih 13,1 km. Penelitian Utami et al., (2014) menggunakan metode AG untuk pencarian rute terpendek untuk pemadam kebakaran wilayah kota Pontianak. P_c yang digunakan sebesar 0,6 dan p_m yang digunakan sebesar 0,01. Penelitian ini memperhatikan kondisi jalan berdasarkan titik awal, titik tujuan dan waktu keberangkatan. Hasil dari pengolahan data tersebut menunjukkan nilai *fitness* tertentu pada pukul 06.00-07.59 dan 16.00-17.59.

Hasibuhan dan Lusiana (2015) melakukan penelitian untuk menentukan rute terbaik pada Dinas Kebersihan Pekanbaru menggunakan AG. Input data yang digunakan lebih jarak dari masing-masing kota. Penelitian ini menunjukkan jumlah populasi sebesar 100, generasi sebanyak 1000 nilai p_c sebesar 0,5 dan p_m sebesar 0,1. Nilai *fitness* tertentu yang dihasilkan sebesar 0,0242390 dengan jarak minimum sebesar 41,2558 pada generasi 914. (Basriati & Aziza, 2017) melakukan penelitian penentuan rute distribusi menggunakan

metode *Insertion Heuristic* pada perusahaan laundry. Input data yang digunakan adalah jarak tempuh rute perjalanan. Rute optimal yang dihasilkan adalah rute dengan jarak tempuh sebesar 2,3 km (rute 1), jarak 1,6 km (rute 2), jarak 12,5 km (rute 3), jarak 5,2 km (rute 4) dan jarak 5,6 km (rute 5).

Supriyana (2017) melakukan penelitian dalam pencarian rute distribusi dengan menggunakan metode AG. Pengujian dilakukan dengan 10 populasi, maksimum generasi sebanyak 30, nilai p_m sebesar 0,25, *tournament selection size* sebesar 30 dan jumlah elite kromosom sebanyak 1. Nilai *fitness* tertinggi yang dihasilkan sebesar 2,0606 dengan jarak minimal sebesar 4853 km. Penelitian Anka et al. 2017 dalam penentuan rute distribusi untuk meminimasi biaya transportasi dengan menggunakan AG. Hasil yang diperoleh terjadi penurunan total jarak tempuh sebesar 35%, penurunan total biaya transportasi sebesar 31% dan peningkatan pemenuhan permintaan menjadi 94,89%.

Tabel 2.4 *State of The Art*

Tahun	Penulis	Fokus Penelitian			Metode			Input Data	
		Distrbusi Produk	Rute Perjalanan	Lainnya	AG	Algoritma Tabu Search	Algoritma <i>Nearest Neighbor</i>	Lainnya	Waktu
2012	Pradhana et al	v				v			v
2012	Yulia dan Singgih	v					v		v
2014	Utami et al	v			v			v	
2015	Suyudi et al	v				v			v
2015	Hasibuhan dan Lusiana		v		v				v
2017	Rizky et al	v			v				v
2017	Basriati dan Aziza	v					v		v
2017	Supriyana	v			v				v
2018	Penelitian	v			v	v		v	

Berdasarkan beberapa penelitian terdahulu bahwa permasalahan optimasi jalur distribusi dapat diselesaikan dengan menerapkan metode Algoritma *Nearest Neighbor* namun hasil yang diperoleh belum optimal sehingga perlu melakukan pengembangan dengan metode AG. Pada penelitian terdahulu AG dapat menyelesaikan permasalahan distribusi. Sehingga pada penelitian ini, penulis melakukan pengembangan berdasarkan hasil algoritma *Nearest Neighbor* untuk hasil yang optimal. Di dalam penelitian ini, kasus yang akan diselesaikan adalah *Vehicle Routing Problem with Time Windows* (VRPTW). Tabel 2.2 menunjukkan posisi penelitian yang dilakukan terhadap penelitian terdahulu.

