BAB II

TINJAUAN PUSTAKA

Sebelum membahas masalah model DM BRT Trans Musi Kota Palembang, pustaka-pustaka yang digunakan dalam menunjang penelitian ini yaitu *Bus Rapid Transit*, *Integer Programming*, *Mixed Integer Programming* dan *Delay Management*.

2.1 *Bus Rapid Transit* (BRT) Trans Musi

Bus Rapid Transit (BRT) merupakan bus yang dirancang dengan fitur lebih layak dibanding dengan bus lainnya. Adanya BRT yang dioperasikan dengan cepat, nyaman, aman dan terjadwal merupakan suatu fasilitas yang memang dibutuhkan oleh masyarakat. Terdapat beberapa kota besar yang telah menerapkan sistem transportasi berjenis BRT ini dengan nama yang berbeda-beda, seperti Trans Musi di Palembang, Transjakarta di Jakarta, Trans Semarang di Semarang, Trans Metro Pekanbaru di Pekanbaru, Trans Jogja di Yogyakarta, Trans Metro Bandung di Bandung, Bus Trans Pakuan di Bogor, dan Batik Solo Trans di Surakarta.

PT. Sarana Pembangunan Palembang Jaya (SP2J) berdasarkan surat Wali Kota Palembang tanggal 22 Oktober 2009 No.551-2/002394 /Dishub ditunjuk untuk pengoperasian BRT Trans Musi Palembang baik pengadaan APBD tahun 2009 maupun bantuan Kementerian Perhubungan, di bawah Pengawasan Dinas Perhubungan Kota Palembang. Berdasarkan SP2J, tujuan dikembangkannya BRT

Trans Musi adalah untuk meningkatkan pelayanan transportasi yang baik kepada masyarakat dengan menciptakan suatu sistem angkutan umum yang efisien, berkualitas, dan berkelanjutan, sehingga dapat mendukung penyelenggaraan lalu lintas dan angkutan jalan yang aman, nyaman, cepat, lancar serta dapat diandalkan. Secara resmi, operasional BRT Trans Musi di-*launching* pada 22 Februari 2010, dengan 2 koridor (trayek), 74 halte, 25 bus (5 bus besar, 20 bus medium). Dengan kesuksesan Trans Musi, pada 1 Mei 2010 pemerintah pusat dalam hal ini Kementrian Perhubungan memberikan 5 buah bus ukuran besar. SP2J pun kemudian menambah 60 bus berukuran sedang dan melakukan *soft launching* pada 23 Februari 2011. Pada tahun 2019, jumlah koridor menjadi 7 koridor dan terdapat 220 bus.

2.2 *Integer Programming*

2.2.1 Definisi *Integer Programming*

Masalah pemrograman *integer programming* (IP) adalah suatu masalah pemrograman linier yang setidaknya salah satu variabel dibatasi untuk nilai *integer* atau bilangan bulat. Dengan istilah lain masalah pemrograman *integer* murni (atau IP murni) lebih menekankan IP yang variabel-variabelnya semua dibatasi untuk nilai bilangan bulat (Chen *et.al.*, 2010).

Tujuan IP adalah untuk mengalokasikan sumber daya bersama, dan tanggung jawab untuk memenuhi persyaratan, untuk semua kegiatan yang bersaing secara optimal atau dengan cara yang terbaik. Istilah "masalah pemrograman" kadang-kadang diganti oleh program, singkatnya masalah pemrograman *integer* juga disebut program *integer*, dan begitu juga program *integer* campuran atau *mixed integer program*, dan program *integer* murni atau *pure integer program*.

2.2.2 Model *Integer Programming*

Menurut (Chen *et.al.*, 2010) bentuk MIP dapat dinyatakan seperti pada Model (2.1).

Maksimumkan /Meminimumkan :

$$z = \sum_{j} c_j \, x_j + \sum_{k} d_k \, y_k \qquad (2.1)$$

dengan kendala :

$$\sum_{J} a_{ij} \, x_j + \sum_{K} g_{ik} \, y_k \le b_i \qquad (i = 1,2, \dots, m) \qquad (2.1.a)$$

$$x_i \ge 0 \qquad (j = 1,2, \dots, n) \qquad (2.1.b)$$

$$y_k = 0,1,2, \dots \qquad (k = 1,2, \dots, p) \qquad (2.1.c)$$

Semua parameter input $(c_j, d_k, a_{ik,} g_{ik}, b_i)$ dapat bernilai positif, negatif, atau nol.

Keterangan :

m adalah sejumlah kendala

n adalah jumlah variabel kontinu

p adalah jumlah *integer* kontinu

2.3 *Mixed Integer Programming*

Beberapa notasi untuk menggambarkan jaringan transportasi diasumsikan $\mathcal{V}$ sebagai himpunan halte dan $\mathcal{F}$ sebagai himpunan kendaraan. Jika dimungkinkan untuk beralih dari kendaraan i ke kendaraan j di halte k, maka kejadian beralih disebut koneksi dan seluruh rangkaian koneksi dimana $\mathcal{U}$ merupakan unit himpunan bagian ketika dua kendaraan datang ke satu halte diberikan oleh :

$$\mathcal{U} \subseteq \mathcal{F} \times \mathcal{F} \times \mathcal{V} \qquad (2.2)$$

Selain itu, diasumsikan bahwa T adalah periode waktu untuk semua kendaraan, dan bahwa dalam periode waktu berikutnya semua kendaraan berada dalam waktu, *delay* atau penundaan awal diberikan sebagai jumlah waktu V, seperti kendaraan yang ditentukan tiba di halte yang telah ditentukan, dan diasumsikan sebagai $V \leq T$.

2.4 *Delay Management* (DM)

Masalah DM adalah menemukan keputusan untuk menunggu, tidak hanya untuk satu bus tunggal, tetapi untuk semua kendaraan di jaringan transportasi umum, sedemikian rupa sehingga ketidaknyamanan atas semua pelanggan dapat diminimalkan (Schobel, 2001). Untuk memodelkan masalah DM dan mendapatkan prosedur solusi, maka digunakan konsep jaringan *event-activity* atau peristiwa-aktivitas jaringan.

Jaringan aktivitas $N = (E, A)$ diarahkan oleh graf yang *vertex* nya disebut peristiwa dan ujung-ujungnya diarahkan disebut aktivitas. Dalam kasus non periodik, suatu kegiatan yang menghubungkan dua peristiwa dengan memodelkan kendala dan diutamakan peristiwa awal kegiatan harus terjadi terlebih dahulu. Setiap kegiatan memiliki penetapan batas bawah pada durasinya, sehingga waktu yang dijadwalkan untuk suatu kegiatan harus lebih besar dari atau sama dengan waktu yang dijadwalkan pada mulainya suatu kegiatan ditambah batas bawah. Berbeda dengan kasus non periodik, dalam jaringan aktivitas periodik (digunakan misalnya untuk pengaturan waktu berkala), setiap aktivitas telah menetapkan batasan waktu pemodelan batas bawah dan batas atas.

2.4.1 *Event-Activity*

Dalam kasus BRT Trans Musi, ε merupakan suatu peristiwa yang terdiri dari *arrival events* atau peristiwa kedatangan bus di halte $\varepsilon_{\mathrm{arr}}$, dan *departure events* atau peristiwa keberangkatan bus dari halte $\varepsilon_{\mathrm{dep}}$.

a. *Driving activities* $\mathcal{A}_{\mathrm{drive}} \subset \varepsilon_{\mathrm{dep}} \times \varepsilon_{\mathrm{arr}}$, merupakan model antara dua halte berturut-turut, sehingga aktivitas mengemudi menghubungkan peristiwa keberangkatan bus dengan kedatangan di halte berikutnya. Batas bawah yaitu $L_{\mathrm{a}} > 0$ dari aktivitas mengemudi $a \in \mathcal{A}_{\mathrm{drive}}$ yaitu waktu mengemudi minimum antara kedua halte.

b. *Waiting activities* $\mathcal{A}_{\mathrm{wait}} \subset \varepsilon_{\mathrm{arr}} \times \varepsilon_{\mathrm{dep}}$, merupakan waktu tunggu bus di halte, misalnya untuk menaikkan penumpang atau untuk perubahan halte penumpang. Aktivitas menunggu menghubungkan kedatangan bus di halte dengan keberangkatannya dari halte yang sama. Batas bawah $L_{\mathrm{a}} > 0$ dari aktivitas menunggu $a \in \mathcal{A}_{\mathrm{wait}}$ menggambarkan waktu minimal yang diperlukan untuk membiarkan penumpang naik atau turun dan juga memperhitungkan waktu untuk berubah atau tindakan lainnya. Setiap aktivitas pada $\mathcal{A}_{\text{drive}}$ dan $\mathcal{A}_{\text{wait}}$ berhubungan dalam satu jaringan, sehingga dapat ditulis $\mathcal{A}_{\text{bus}} = \mathcal{A}_{\text{drive}} \cup \mathcal{A}_{\text{wait}}$.

c. *Changing activities* $\mathcal{A}_{\text{change}} \subset \varepsilon_{\text{arr}} \times \varepsilon_{\text{dep}}$, memungkinkan penumpang untuk berpindah dari satu bus ke bus yang lain di halte yang sama, sehingga aktivitas *change* menghubungkan kedatangan beberapa bus di beberapa halte dengan keberangkatan dari bus lainnya di halte yang sama. Batas bawah yaitu $L_{\text{a}} > 0$ mengacu pada waktu minimum yang dibutuhkan penumpang saat berpindah antara kedua bus. Itu merupakan salah satu tugas DM untuk memutuskan setiap aktivitas yang berubah jika sesuai koneksi harus diperhatikan atau tidak.

d. *Headway activities* $\mathcal{A}_{\text{head}} \subset \varepsilon_{\text{dep}} \times \varepsilon_{\text{dep}}$, merupakan model kapasitas yang terbatas dari jalur sistem. Jika $(i, j) \in \mathcal{A}_{\text{head}}$, maka $(j, i) \in \mathcal{A}_{\text{head}}$. Berbeda dengan aktivitas lainnya, satu kali h*eadway* tidak memodelkan satu kendala, tetapi bersama dengan lainnya, yaitu model sepasang kendala disjungtif. Sebagai *event-activity* model kendala prioritas, tidak mungkin untuk memenuhi dua kendala yang dihasilkan dari sepasang aktivitas *headway* pada saat yang sama. Batas bawah bawah $L_{\text{ij}} > 0$ dari aktivitas *headway* $(i, j) \in \mathcal{A}_{\text{head}}$, terdapat jarak. Untuk dua waktu keberangkatan i dan j, merupakan minimal *headway* antara keberangkatan bus yang sesuai, yaitu waktu minimum untuk bus j dan harus menunggu setelah keberangkatan bus i untuk memastikan bahwa perjalanan lancar. Waktu *headway* tidak perlu simetris. Secara umum $L_{\text{ij}} \neq L_{\text{ji}}$. Sehingga dapat disimpulkan $\mathcal{A} = \mathcal{A}_{\text{drive}} \cup \mathcal{A}_{\text{wait}} \cup \mathcal{A}_{\text{change}} \cup \mathcal{A}_{\text{head}}$

Timetable $\pi : \varepsilon \rightarrow N$, disebut *feasible* jika batas bawah dari *driving, waiting*, dan *changing* tepat satu *headway* dari setiap aktivitas *headway*.

$$\mathcal{A}_{head}^{\text{forw}} := \{(i, j) \in \mathcal{A}_{\text{head}} : \pi_i < \pi_j\} \tag{2.3}$$

$$\mathcal{A}_{head}^{\text{back}} := \{(i, j) \in \mathcal{A}_{\text{head}} : \pi_i > \pi_j\} \tag{2.4}$$

Untuk memodelkan keputusan *delay*, maka digunakan variabel biner

$$z_a = \begin{cases} 0 \text{ jika terdapat aktivitas } changing\ a \\ 1 \text{ untuk lainnya.} \end{cases}$$

Untuk semua aktivitas *changing* $\in \mathcal{A}_{\text{change}}$ maka digunakan variabel biner

$$g_{ij} = \begin{cases} 0 \text{ jika } event\ i \text{ sebelum } event\ j \\ 1 \text{ untuk lainnya.} \end{cases}$$

2.4.2 Model *Delay Management*

Menurut Schachtebeck (2009), model DM dapat dilihat pada Model(2.5)

$$\text{(DM) Minimumkan } f(x, z, g) = \sum_{i \in \varepsilon} (x_i - \pi_i) + \sum_{a \in \mathcal{A}_{change}} z_a T \tag{2.5}$$

dengan kendala :

$$x_i \geq \pi_i + d_i \qquad \forall\, i \in \varepsilon \tag{2.5.a}$$

$$x_j - x_i \geq L_a + d_a \qquad \forall\, a = (i, j) \in \varepsilon\, \mathcal{A}_{\text{bus}} \tag{2.5.b}$$

$$M z_a + x_j - x_i \geq L_a \qquad \forall\, a = (i, j) \in \varepsilon\, \mathcal{A}_{\text{change}} \tag{2.5.c}$$

$$M g_{ij} + x_j - x_i \geq L_{ij} \qquad \forall\, (i, j) \in \mathcal{A}_{\text{head}} \tag{2.5.d}$$

$$g_{ij} + g_{ji} = 1 \qquad \forall\, a = (i, j) \in \varepsilon\, \mathcal{A}_{\text{change}} \tag{2.5.e}$$

$x_i \ \epsilon \, N$ $\forall \, i \ \in \ \varepsilon$ (2.5.f)

$z_a \ \epsilon \, \{0,1\}$ $\forall \, a \ \varepsilon \, \mathcal{A}_{\text{change}}$ (2.5.g)

$g_{ij} \ \epsilon \, \{0,1\}$ $\forall \, (ij) \ \varepsilon \, \mathcal{A}_{\text{head}}$ (2.5.h)

Keterangan :

x_i adalah waktu pada halte

π_i adalah waktu i yang sudah terjadwal

z_a adalah model *wait-depart* dengan variabel jika *changing activity maintended*

T adalah lama perjalanan yang ditempuh

d_i adalah waktu *delay* yang terjadi pada *event*

d_a adalah waktu *delay* yang terjadi pada *activitiy*

L_a adalah batas bawah (*lower bound*) waktu tempuh minimal antara 2 halte

L_{ij} adalah waktu minimum *headway activity*

M adalah konstanta dengan nilai yang cukup besar, berupa bilangan bulat positif

g_{ij} adalah variabel 0 jika *event* i terjadi sebelum *event j*

g_{ji} adalah variabel 1 jika *event* j terjadi sebelum *event* i