

BAB 2 TINJAUAN PUSTAKA DAN DASAR TEORI

Pada bab ini dijelaskan penelitian-penelitian terdahulu mengenai topik penelitian yang diangkat penulis dan perbaikan apa yang ingin dilakukan penulis terhadap penelitian yang sudah ada serta teori-teori yang mendukung penyelesaian masalah.

2.1. Tinjauan Pustaka

Penjadwalan yang efektif adalah pengalokasian beban kerja yang dibutuhkan dikenakan dengan sejumlah kendala dengan tujuan mendayagunakan sumber daya, menghasilkan jadwal dengan beban kerja yang seimbang, dan memuaskan kebutuhan personal sebanyak mungkin (Labidi et al, 2014). Kebutuhan personal, seperti *between days break time*, *on leave*, dan *weekend off day* dalam penjadwalan berbasis beban kerja sangat penting untuk diperhatikan karena mempunyai dampak langsung terhadap kualitas layanannya (Purnama & Yuniartha, 2014). Model penjadwalan yang mampu menyeimbangkan beban kerja dan memuaskan kebutuhan personal memungkinkan tenaga kerja untuk berkonsentrasi pada tugas-tugas yang lebih produktif dan mengurangi *human error* (Rigi & Khoshalhan, 2011; Du et al, 2012). Cara yang dapat dilakukan untuk menyeimbangkan beban kerja dan memuaskan kebutuhan personal adalah pembagian *shift* yang seimbang (Gordon & Erkut, 2004), reduksi/ distribusi waktu lembur, serta waktu libur tenaga kerja dialokasikan secara tepat (Labidi et al, 2014).

Beban kerja fisik maupun psikososial harus diukur terus menerus dengan resolusi tinggi secara *real-time* untuk mengetahui dan memastikan keseimbangan beban kerja terhadap jadwal. Christensen (1991) menjelaskan bahwa salah satu pendekatan untuk mengetahui berat ringannya beban kerja fisik adalah dengan menghitung nadi kerja, konsumsi energi, kapasitas ventilasi paru dan suhu inti tubuh. Pada batas tertentu ventilasi paru, denyut jantung, dan suhu tubuh mempunyai hubungan yang linear dengan konsumsi oksigen atau pekerjaan yang dilakukan. Sedangkan untuk mengukur beban kerja psikososial dapat dilakukan dengan pembagian kuesioner kepada para tenaga kerja lalu dianalisis dengan skala 7 titik dengan angka 7 menunjukkan beban kerja paling tinggi (Vredenburgh et al, 2000). Perataan beban kerja psikososial juga dapat

dilakukan dengan pengukuran preferensi sebelum jadwal dibuat, yaitu tenaga kerja diminta menuliskan keinginan dalam bentuk skor pada *score card*, jika tenaga kerja memberikan skor rendah artinya ia menginginkan hal tersebut (Lim & Mobasher, 2011). Beban kerja kemudian dimasukkan ke dalam parameter penjadwalan berupa input sebagai preferensi tenaga kerja terhadap hari libur dan menjadi *output* yaitu jumlah hari yang mengharuskan tenaga kerja masuk kerja pada saat ia menginginkan libur (Labidi, et al, 2014).

Kriteria penjadwalan atau input model penjadwalan yang sudah diteliti antara lain jam kerja (Lim & Mobasher, 2011; Du et al 2012), jumlah tenaga kerja (Lim & Mobasher, 2011), serta kendala khusus lainnya, yaitu jumlah tenaga kerja yang tersedia (Er et al, 2008; Du et al, 2012), tenaga kerja dengan keterampilan lebih rendah tidak dapat menggantikan tenaga kerja dengan keterampilan yang lebih tinggi, seorang tenaga kerja tidak boleh ditempatkan pada lebih dari 1 level pekerjaan dalam 1 *shift* (Lim & Mobasher, 2011), dan jumlah konsumen (Du et al, 2012; Lim & Mobasher, 2011). Adapun penelitian-penelitian yang sudah memperhatikan kriteria-kriteria beban kerja seperti preferensi (Labidi et al, 2014; Gordon et al, 2004; Rigi & Khoshalhan, 2011), hari libur (Labidi et al, 2014), jumlah minimal tenaga kerja/*shift* (Labidi et al, 2014), aturan tenaga kerja tidak boleh bekerja *shift* pagi setelah *shift* malam di hari sebelumnya, dan aturan tenaga kerja yang boleh masuk kerja pada *shift* tertentu (Labidi et al, 2014). Banyak metode penjadwalan yang dapat digunakan untuk menyelesaikan penjadwalan tenaga kerja, yaitu metode *goal programming* (Eradipa et al, 2014), metode *genetic algorithm* (Rigi & Khoshalhan, 2011; Lim & Mobasher, 2011; Du et al, 2012), metode *multi objective programming* (Labidi et al, 2014), dan metode *integer linear programming* (Bhulai et al, 2008).

Pemetaan dari jadwal-jadwal aktual yang diteliti oleh Dewi et al (2014) telah membagi antara waktu kerja dan waktu libur, namun jadwal aktual belum memperhatikan fleksibilitas untuk mencapai keseimbangan antara beban kerja dan kebutuhan pribadi tenaga kerjanya. Pada penelitian ini akan dilakukan pembangunan model penjadwalan yang memperhatikan keseimbangan beban kerja fisik dan psikososial untuk mencapai kesejahteraan tenaga kerjanya. Model penjadwalan yang akan dijadikan acuan pembangunan model adalah penelitian yang dilakukan Labidi et al (2014). Pemilihan model penelitian Labidi et al (2014) sebagai model acuan karena kesamaan sebagian besar kriteria penjadwalan Labidi et al (2014) dengan kriteria penjadwalan yang digunakan hotel-hotel non

bintang DIY, yaitu periode penjadwalan, jumlah *shift*, alokasi *shift*, *workstretch*, dan pola *workstretch* khusus untuk supervisor. Selain itu, model penjadwalan Labidi et al (2014) memperhatikan beban kerja psikososial yang sesuai dengan tujuan penelitian ini. Penulis akan melakukan perbaikan terhadap jadwal tenaga kerja *security* di hotel non bintang DIY dengan mengacu penelitian Labidi et al (2014) agar memperhatikan beban psikososial dan mengembangkannya dengan menambahkan faktor beban kerja fisik.

2.2. Dasar Teori

Ada beberapa teori yang akan dibahas pada sub bab ini yang dibutuhkan penulis dan mempunyai relevansi untuk menyelesaikan permasalahan penjadwalan *security* berbasis beban kerja.

2.2.1. Penjadwalan Berbasis Beban Kerja (*Workforce Scheduling*)

Setiap instansi menginginkan penjadwalan yang digunakan dapat mengurangi biaya yang harus dikeluarkan dan memperhatikan penggunaan tenaga kerja yang berkualitas agar dapat memenuhi kepuasan pelanggan. Oleh karena itu, penjadwalan tenaga kerja menjadi sangat penting untuk memenuhi kedua tujuan tersebut, yaitu meminimalkan biaya dan mengalokasikan tenaga kerja sesuai dengan kemampuannya di waktu yang tepat. Adapun beberapa tujuan lain dari penjadwalan, yaitu :

a. Efisiensi Jadwal

Efisiensi jadwal yang telah dibuat dihitung efisiensinya dari rumus berikut :

$$\text{Scheduling Efficiency} = \frac{\text{Total Hours Required}}{\text{Total Hours Scheduled}} \quad (2.1.)$$

b. Keadilan dan keseimbangan beban untuk setiap tenaga kerja

c. Lembur

Penjadwalan mencoba untuk menghindari adanya lembur, yang berarti bahwa menghindari adanya periode waktu dengan jumlah orang lebih dari yang dibutuhkan.

d. *Undersupply*

Penjadwalan mencoba untuk menghindari *undersupply*, yang berarti bahwa menghindari adanya periode dengan jumlah orang kurang dari yang dibutuhkan.

Dalam pembuatan jadwal pun harus memperhatikan sejumlah kendala. Pada umumnya persyaratan dan kendala yang diperhatikan, yaitu :

- a. Jumlah personil harus sesuai permintaan
- b. Kemampuan tenaga kerja yang dijadwalkan harus sesuai dengan kebutuhan
- c. Peraturan hukum harus dipatuhi, misalnya tenaga kerja tidak bekerja lebih dari satu *shift* per hari, dan hari libur diperhatikan
- d. Perbedaan *shift*
- e. Istirahat antar *shift* harus diperhatikan

Dalam mendeskripsikan masalah penjadwalan berbasis beban kerja (*workforce scheduling*) perlu memperhatikan 7 hal berikut, yaitu :

- a. *Order*
Adalah kebutuhan tenaga kerja, misalnya profil beban kerja yang diberikan.
- b. *Resources*
Adalah data tenaga kerja dan profil kemampuannya.
- c. *Products*
Adalah layanan yang dapat diberikan oleh tenaga kerja.
- d. *Hard Constraint*
Adalah pembatasan hukum, seperti jam kerja maksimum, maksimum jumlah *shift*, dan tidak boleh bekerja lebih dari 2 *shift* berturut-turut.
- e. *Soft Constraint*
Adalah kendala yang ingin kita capai, misalnya tidak memiliki *overtimes*, tidak memiliki *undersupply*, dan memiliki distribusi beban kerja yang adil untuk setiap tenaga kerja. Kendala tersebut masih diperbolehkan namun seminimal mungkin.
- f. *Goal Functions*
Adalah kualitas jadwal yang diinginkan.
- g. *Events*
Adalah jadwal reaktif yang dialokasikan untuk tenaga kerja yang terlambat atau sakit, dan lain-lain.

2.2.2. Jenis-jenis Penjadwalan Tenaga Kerja

Nanda & Browne (1992) menjelaskan bahwa penjadwalan berbasis beban kerja diklasifikasikan menjadi penjadwalan hari libur (*days-off scheduling*), penjadwalan *shift* (*shift scheduling*), dan *tour scheduling*. Penjadwalan tenaga kerja dikategorikan menjadi 3 jenis, yaitu :

- a. *Days-off scheduling*
Penjadwalan hari libur adalah masalah yang timbul dalam instansi yang memiliki 7 hari kerja dalam seminggu, seperti restoran, rumah sakit, dan

hotel. Masalahnya adalah permintaan kerja yang kontinyu namun tenaga kerja tidak dapat bekerja kontinyu atau terus menerus. Oleh karena itu, pola hari libur setiap tenaga kerja harus dijadwalkan untuk memenuhi semua tuntutan, baik tuntutan pekerjaan maupun tenaga kerja (Alfares, 2002).

b. *Shift scheduling*

Shift scheduling adalah masalah *days-off scheduling* yang juga melibatkan *shift*, yaitu perbedaan waktu saat memulai bekerja dan/lama bekerja.

c. *Tour scheduling*

Tour scheduling menggabungkan permasalahan penjadwalan *shift* dan *days-off* dengan menentukan jam kerja harian dan hari kerja mingguan untuk setiap tenaga kerja.

2.2.3. *Influence Diagram*

Influence diagram adalah suatu diagram yang menyatakan hubungan keterkaitan antara input dengan komponen, komponen dengan komponen, dan komponen dengan output. Keterkaitan itu digambarkan melalui garis dengan ujung anak panah. Jika garis hanya memiliki satu anak panah maka jenis interaksi yang dinyatakan bersifat uni-direksional, sedangkan bila garis memiliki dua anak panah maka interaksi yang dinyatakan bersifat mutual. Beberapa simbol yang digunakan dalam *influence diagram* adalah:

- Kotak untuk input yang terkontrol ()
- Awan untuk parameter/input tidak terkontrol ()
- Lingkaran untuk proses ()
- Oval untuk output ()
- Tanda panah untuk menunjukkan hubungan keterkaitan ()

Influence diagram sangat mempengaruhi kemudahan pemodelan sistem sehingga dapat mengetahui aspek input, proses, dan output yang ada di dalamnya, serta keterkaitan antar aspek.

2.2.4. *Optimasi dan Pemodelan*

Optimasi merupakan cara mencari nilai terbesar atau terkecil sesuai dengan fungsi tujuan yang ingin dicapai. Untuk dapat mencari nilai optimal secara sistematis dilakukan pemilihan nilai variabel *integer* atau nyata yang akan memberikan solusi optimal (Wardy, 2007). Teknik dalam penyelesaian sistem optimasi secara garis besar dapat dibagi dalam 3, yaitu teknik pemrograman matematika, teknik proses stokastik, dan metode statistik.

Penyelesaian suatu permasalahan optimasi akan lebih mudah bila masalah ini diubah dalam bentuk model matematika dan kemudian diselesaikan dengan menggunakan teknik pemrograman matematika. Model adalah suatu representasi atau formalisasi dalam bahasa tertentu dari suatu sistem nyata. Pemodelan adalah tahap pembuatan model dari sebuah sistem. Adapun tahapan proses studi dan simulasi sistem, yaitu :

a. Formulasi Masalah dan Tujuan Studi

Formulasi masalah yang sukses membutuhkan penilaian yang akurat terhadap tujuan-tujuan yang disertai oleh suatu deskripsi dari sistem riil.

b. Membangun Model Simulasi

Langkah penting dalam simulasi adalah membangun model yang merepresentasikan kondisi riil masalah yang akan disimulasikan. Setelah membangun model maka dibutuhkan suatu skenario yang akan diterapkan sesuai dengan tujuan yang ingin dicapai oleh studi simulasi tersebut.

c. Desain Eksperimen

Dalam kerangka konseptual model simulasi diperlukan suatu desain eksperimen yang akan menyediakan arahan dalam pengumpulan data, analisis dan bagaimana hasil simulasi akan digunakan untuk menjawab pertanyaan yang sedang dipecahkan.

d. Verifikasi dan Validasi Model

Proses verifikasi dilakukan untuk menentukan apakah model simulasi berjalan sesuai keinginan pembuat model, misalnya dengan melakukan proses *debug* program komputer, sedangkan validasi dilakukan untuk menentukan apakah model simulasi mampu mewakili sistem riil secara akurat (Nasution & Baihaqi, 2007 dalam Ekoanindiyo, 2011). Validasi merupakan proses perbandingan parameter antara model simulasi dengan sistem yang disimulasikan (Pidd, 1992 dalam Ekoanindiyo, 2011). Pendekatan yang biasa digunakan dalam melakukan uji validasi adalah:

1. Validasi kotak hitam

Validasi kotak hitam (*black box validation*) dilaksanakan dengan melakukan observasi perilaku sistem riil pada suatu kondisi tertentu dan menjalankan model pada kondisi yang sedapat mungkin mendekati kondisi sistem riil. Model dianggap valid jika tidak ada perbedaan yang signifikan antara observasi model dengan sistem riil.

Metodologi yang dapat dilakukan untuk membandingkan dengan menetapkan suatu hipotesis awal dan kemudian melakukan pengujian statistik terhadap nilai rata-rata sistem riil dan hasil observasi model.

2. Validasi Kotak Putih

Validasi kotak putih (*white box validation*) dilakukan dengan mengamati cara kerja *interval* model simulasi, misalnya *input* distribusi dan logika sistem, baik statis maupun dinamis.

e. Mengevaluasi Hasil Simulasi untuk Pengambilan Keputusan

Dengan memformulasikan dan menguji asumsi yang berbeda pada perilaku sistem maka kita dapat mengevaluasi kebijakan atau aturan keputusan tertentu sehingga manajemen dapat memilih satu keputusan terbaik untuk memenuhi tujuan organisasi.

Model yang dibuat dapat berfungsi sebagai pembantu untuk berpikir, pembantu untuk berkomunikasi, alat dan latihan, alat prediksi, dan pembantu dalam percobaan. Beberapa kriteria model yang baik adalah mudah dimengerti pemakainya, harus mempunyai tujuan yang jelas, dinyatakan secara jelas dan lengkap, mudah dikontrol dan dimanipulasi pemakai, mengandung pemecahan masalah yang penting dan jelas, mudah diubah dan mempunyai prosedur modifikasi.

2.2.5. Jenis-jenis Metode Penjadwalan Tenaga Kerja

Ada berbagai macam metode yang digunakan untuk melakukan penjadwalan tenaga kerja, yaitu :

a. Metode *goal programming*

Metode *goal programming* digunakan untuk meminimalkan deviasi pada tujuan ganda atau jamak secara bersamaan. Contohnya adalah untuk meminimasi kebutuhan dan meminimasi pelanggaran peraturan pada penjadwalan *room boy* (Eradipa et al,2014).

b. Metode *genetic algorithm*

Metode *genetic algorithm* digunakan untuk menyelesaikan masalah multikriteria dan multi-objektif, seperti masalah penyusunan jadwal pelajaran yang melibatkan aspek guru, mata pelajaran, jumlah ruangan, dan kapasitas ruangan. Metode GA mampu menyelesaikan masalah dengan waktu penyusunan jadwal yang cukup singkat.

c. Metode *multi objective programming*

Metode *multiobjective programming* digunakan dalam pengambilan keputusan multi-kriteria, seperti untuk mencari hasil optimasi maksimasi pendapatan laba penjualan dan minimasi biaya produksi pada UKM jipang ketan Banjarnegara (Mahendra, 2015). Metode ini juga digunakan untuk mencari hasil optimasi minimasi jumlah lembur dan maksimasi preferensi tenaga kerja di departemen IT (Labidi et al, 2014).

d. Metode *integer linear programming*

Metode *integer linear programming* digunakan untuk masalah yang mengharuskan hasilnya dalam bentuk bilangan bulat, seperti masalah penjadwalan perawat pada rumah sakit berupa pengalokasian hari kerja dengan mempertimbangkan kebutuhan perawat, *shift* malam dan kebutuhan hari libur dari tiap-tiap perawat serta beberapa kendala teknis lain yang perlu diperhitungkan berdasarkan pada aturan dari pihak manajemen rumah sakit. Adapun metode ILP digunakan dalam penyusunan *timetabling* untuk angkutan umum dengan tujuan meminimumkan kepadatan penumpang dalam kendaraan umum (Wiyanto, 2012). Metode *integer programming* sudah lama digunakan untuk meminimasi biaya dan maksimasi preferensi (Bard & Purnomo, 2005).

Dari keempat macam metode penjadwalan tenaga kerja, penulis memilih menggunakan metode *integer programming*. Berikut adalah penjelasan mengenai metode *integer linier programming*. Program *integer* merupakan pengembangan dari program linear yang beberapa atau semua variabel keputusannya harus berupa *integer*. Adapun beberapa jenis ILP, yaitu :

- a. Jika hanya sebagian variabel keputusannya merupakan *integer* maka disebut program *integer* campuran (*mixed integer progamming*).
- b. Jika semua variabel keputusannya bernilai *integer* disebut program *integer* murni (*pure integer progamming*).
- c. Jika semua variabel keputusannya harus bernilai *integer* 0 atau 1 disebut program *integer* 0-1 (*binary*).

Ada dua tipe masalah yaitu masalah minimasi dan maksimasi. Beberapa metode solusi yang dapat digunakan, yaitu :

a. Metode *Trial and Error*

Suatu metode yang serupa dengan pendekatan pembulatan adalah prosedur coba-coba (*trial and eror*). Dengan menggunakan cara ini,

pengambil keputusan mengamati solusi *integer* dan memilih solusi yang mengoptimalkan nilai fungsi tujuan. Metode ini sangat tidak efektif jika masalahnya melibatkan sejumlah besar kendala dan variabel. Terlebih lagi, memeriksa kelayakan setiap solusi yang dibulatkan akan banyak memakan waktu.

b. Pendekatan Grafik

Pendekatan grafik digunakan untuk menyelesaikan masalah *integer programming* yang melibatkan hanya dua variabel. Pendekatan dilakukan dengan menggunakan kertas grafik dan menggambarkan sekumpulan titik-titik *integer* dalam ruang solusi layak.

c. Pendekatan Gomory (*Cutting Plane Algorithm*)

Suatu prosedur sistematis untuk memperoleh solusi *integer* optimum terhadap *pure integer programming* pertama kali dikemukakan oleh R.E. Gomory. Ia kemudian memperluas prosedur ini untuk menangani kasus yang lebih sulit yaitu *mixed integer programming*. Langkah-langkah prosedur Gomory diringkas seperti berikut:

1. Selesaikan masalah *integer programming* dengan menggunakan metode simpleks. Jika masalahnya sederhana maka dapat diselesaikan dengan pendekatan grafik, sehingga pendekatan Gomory kurang efisien.
2. Periksa solusi optimum. Jika semua variabel basis memiliki nilai *integer*, solusi optimum *integer* telah diperoleh dan proses solusi telah berakhir. Jika satu atau lebih variabel basis masih memiliki nilai pecah, teruskan ke tahap 3.
3. Buatlah suatu skala Gomory (suatu bidang pemotong atau *cutting plane*) dan cari solusi optimum melalui prosedur dual simpleks.

d. Metode *Branch and Bound*

Metode *Branch and Bound* merupakan kode komputer standar untuk *integer programming*, dan penerapan-penerapan dalam praktek tampaknya menyarankan bahwa metode ini lebih efisien dibanding dengan pendekatan Gomory. Teknik ini dapat diterapkan baik untuk masalah *pure* maupun *mixed integer programming*.

Branch and bound adalah algoritma umum untuk mencari solusi optimal dari berbagai masalah optimasi yang diperkenalkan pertama kali oleh A.H. Land dan A.G. Doig pada tahun 1960. Prinsip dasar metode ini adalah memecah daerah feasible suatu masalah *linier programming*-relaksasi

dengan membuat sub problem - sub problem. Ada dua konsep dasar dalam algoritma *branch and bound*.

1. *Branching*

Branching adalah proses membagi-bagi permasalahan menjadi sub problem-sub problem yang mungkin mengarah ke solusi.

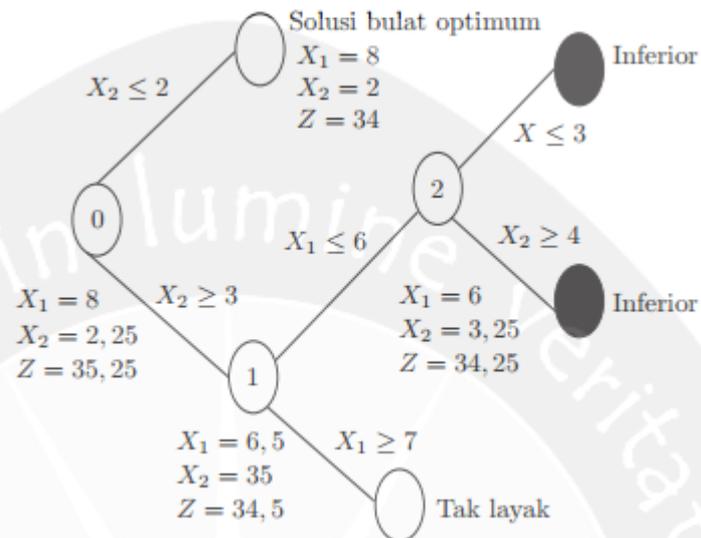
2. *Bounding*

Bounding adalah suatu proses untuk mencari/menghitung batas atas (dalam masalah minimasi) dan batas bawah (dalam masalah maksimasi) untuk solusi optimal pada sub problem yang mengarah ke solusi, lalu dilakukan LP-relaksasi pada ILP.

Metode *branch and bound* diawali dengan menyelesaikan LP-relaksasi dari suatu masalah *integer programming*. Jika semua nilai variabel keputusan solusi optimal sudah *integer*, maka solusi tersebut merupakan solusi optimal ILP. Jika tidak dilakukan pencabangan dan penambahan batasan pada LP-relaksasi kemudian diselesaikan. Langkah-langkah metode *Branch and Bound* untuk masalah maksimasi dapat dilakukan seperti berikut:

1. Selesaikan masalah LP dengan metode simpleks biasa tanpa pembatasan bilangan bulat.
2. Teliti solusi optimumnya. Jika variabel basis yang diharapkan bulat adalah bulat, solusi optimum bulat telah tercapai. Jika satu atau lebih variabel basis yang diharapkan bulat ternyata tidak bulat, lanjutkan ke langkah 3.
3. Nilai solusi pecah yang layak dicabangkan ke dalam sub-sub masalah. Tujuannya adalah untuk menghilangkan solusi kontinyu yang tidak memenuhi persyaratan bulat dalam masalah itu. Pencabangan itu dilakukan melalui kendala-kendala *mutually exclusive* yang perlu untuk memenuhi persyaratan bulat dengan jaminan tidak ada solusi bulat layak yang tidak diikuti sertakan.
4. Untuk setiap sub-masalah, nilai solusi optimum kontinyu fungsi tujuan ditetapkan sebagai batas atas. Solusi bulat terbaik menjadi batas bawah. Sub-sub masalah yang memiliki batas atas kurang dari batas bawah yang ada, tidak diikuti sertakan pada analisa selanjutnya. Suatu solusi bulat layak adalah sama baik atau lebih baik dari batas atas untuk setiap sub masalah yang dicari. Jika solusi yang demikian terjadi,

suatu sub masalah dengan batas atas terbaik dipilih untuk dicabangkan. Contoh pemecahan masalah dengan metode *Branch and bound* ditunjukkan pada Gambar 2.1.



Gambar 2.1. Contoh Pemecahan Masalah dengan Metode *Branch and Bound*
(Sumber : Pane, 2012)

2.2.6. Beban Kerja

Beban kerja adalah kemampuan tubuh tenaga kerja dalam menerima pekerjaan. Dari sudut pandang ergonomi, setiap beban kerja yang diterima seseorang harus sesuai dan seimbang terhadap kemampuan fisik maupun psikologis tenaga kerja yang menerima beban kerja tersebut (Manuaba, 2000). Dalam sebuah organisasi perlu penataan beban kerja untuk setiap pegawai yang menjadi tanggung jawab pelaksana tugas pejabat yang ada di dalamnya, sehingga dengan adanya pemerataan beban kerja pegawai dapat meningkatkan kinerja pegawai yang dapat meningkatkan kinerja organisasi secara keseluruhan.

Beban kerja dipengaruhi oleh 2 faktor, yaitu faktor eksternal dan faktor internal. Menurut Manuaba (2000), faktor-faktor yang mempengaruhi beban kerja antara lain :

- a. Faktor eksternal, yaitu beban yang berasal dari luar tubuh tenaga kerja, seperti:
 - i. Tugas-tugas yang bersifat fisik, seperti stasiun kerja, tata ruang, tempat kerja, alat dan sarana kerja, kondisi kerja, sikap kerja, dan tugas-tugas

yang bersifat psikologis, seperti kompleksitas pekerjaan, tingkat kesulitan, dan tanggung jawab pekerjaan.

- ii. Organisasi kerja, seperti lamanya waktu bekerja, waktu istirahat, *shift* kerja, kerja malam, sistem pengupahan, model struktur organisasi, pelimpahan tugas dan wewenang.
- iii. Lingkungan kerja adalah lingkungan kerja fisik, lingkungan kimiawi, lingkungan kerja biologis dan lingkungan kerja psikologis.

b. Faktor internal

Faktor internal adalah faktor yang berasal dari dalam tubuh itu sendiri akibat dari reaksi beban kerja eksternal. Faktor internal meliputi faktor somatis (jenis kelamin, umur, ukuran tubuh, status gizi, dan kondisi kesehatan) dan faktor psikis (motivasi, persepsi, kepercayaan, keinginan dan kepuasan).

Beban kerja yang akan dibahas pada penelitian ini adalah beban kerja fisik dan psikososial.

i. Beban Kerja Fisik

Beban kerja fisik yang terlalu berat atau kemampuan fisik yang terlalu lemah dapat mengakibatkan seorang tenaga kerja menderita gangguan atau penyakit akibat kerja. Penilaian beban kerja fisik dapat dilakukan dengan dua metode yaitu secara objektif (penelitian secara langsung) dan metode tidak langsung. Metode pengukuran langsung yaitu dengan mengukur oksigen yang dikeluarkan (*energy expenditure*) melalui asupan energi selama bekerja. Semakin berat kerja semakin banyak energi yang dikeluarkan. Salah satu pendekatan untuk mengetahui berat ringannya beban kerja adalah dengan menghitung nadi kerja, konsumsi energi, kapasitas ventilasi paru dan suhu inti tubuh. Nurmiyanto (2003) mengemukakan bahwa denyut jantung adalah suatu alat estimasi laju metabolisme yang baik, kecuali dalam keadaan emosi. Pada penelitian ini, beban fisik diukur berdasar nilai denyut jantung yang dikonversi menjadi nilai RPE (*Rating of Perceived Exertion*). *Rating of Perceived Exertion* adalah cara untuk mengukur tingkat intensitas aktivitas fisik. Hal ini didasarkan pada sensasi fisik pengalaman orang selama aktivitas fisik, termasuk peningkatan denyut jantung, peningkatan respirasi atau tingkat pernapasan, peningkatan dalam berkeringat, dan kelelahan otot. Meskipun ini adalah ukuran subjektif, RPE dapat memberikan perkiraan yang cukup baik dari denyut jantung yang sebenarnya selama aktivitas fisik (Borg, 1998).

Skala Borg dengan skala 6-20 mengikuti denyut jantung orang dewasa yang sehat dengan mengalikan 10. Pengukuran denyut jantung berkisar antara 60 hingga 200 denyut per menit. Interpretasi masing-masing ranking ditunjukkan pada Tabel 2.1.

Tabel 2.1. Interpretasi Skala RPE

Rating	Interpretation of Rating
6	<i>No exertion et all</i>
7	
8	<i>Extremely light</i>
9	<i>Very light</i>
10	
11	<i>Light</i>
12	
13	<i>Somewhat hard</i>
14	
15	<i>Hard</i>
16	
17	<i>Very hard</i>
18	
19	<i>Extremely hard</i>
20	<i>Maximal exertion</i>

(Sumber : Bridger, 2003)

ii. **Beban Kerja Psikososial**

Aspek psikososial harus diperhatikan dalam lingkungan kerja karena menyangkut kesejahteraan tenaga kerja dan tingkat konsentrasi tenaga kerja sehingga membahayakan keselamatan dalam bekerja (Caplan, 1984). Johansson & Rubenowitz (1994) menjelaskan faktor-faktor psikososial dalam lingkungan kerja yang memiliki pengaruh dalam kinerja sebagai berikut :

1. **Pengaruh dan kontrol pekerjaan**

Dalam hal ini ada beberapa hal yang bisa dilihat antara lain seperti pengaruh tingkatan kerja, pengaruh metode kerja, pengaruh alokasi kerja dan kontrol teknis serta pengaruh peraturan kerja

2. **Rangsang dari kerja itu sendiri**

Hal-hal yang diperhatikan adalah apakah pekerjaan tersebut menarik dan menstimulasi individu untuk bekerja atau tidak, apakah pekerjaan tersebut bervariasi dan terbagi-bagi atau tidak, kesempatan untuk

mempergunakan bakat dan keterampilan, dan perasaan keseluruhan tentang pekerjaan yang dilakukan

3. Hubungan dengan rekan kerja

Hal-hal yang diperhatikan antara lain adalah hubungan dan kontak dengan rekan kerja, pembicaraan tentang hal-hal yang berkaitan dengan pekerjaan dengan rekan kerja, perluasan pengalaman dalam suasana kerja yang menyenangkan, dan diskusi tentang masalah yang berkaitan dengan pekerjaan.

4. Beban kerja secara psikologis

Beberapa hal yang dipertimbangkan adalah stres kerja, beban kerja, perasaan lelah dan kejenuhan sehabis bekerja yang meningkat, ada atau tidaknya kemungkinan untuk relaksasi dan beristirahat saat bekerja dan beban mental yang ditimbulkan oleh pekerjaan itu sendiri.

2.2.7. LINGO

LINGO merupakan bahasa pemrograman tingkat tinggi yang digunakan untuk menyelesaikan masalah optimisasi berskala besar. Model program yang dapat diselesaikan antara lain *Linear Programming* (LP), *Integer Programming* (IP), *Quadratic Programming* (QP), dan *Non Linear Programming* (NLP) (Kurniawan, 2011). Solusi model akan dicari menggunakan *software* LINGO 13.0 secara simultan/sekaligus dengan metode *Branch and Bound*. *Software* LINGO 13.0 yang digunakan dalam penelitian ini adalah khusus untuk edukasi dengan batas maksimum kendala, variabel, integer variabel, non linear variabel, dan global variabel. *Software* tersebut hanya maksimal mampu menjalankan 4000 kendala, 8000 variabel, 800 integer variabel, 800 non linear variabel, dan 20 global variabel. Model yang dikembangkan masih dalam batasan maksimum LINGO 13.0 sehingga model masih bisa dikerjakan dengan menggunakan *software* tersebut.

LINGO memiliki 4 *solver* yang digunakan untuk menyelesaikan berbagai tipe model yang berbeda, yaitu *direct solver*, *linear solver*, *nonlinear solver*, dan *branch and bound*. *Direct solver* akan menghitung nilai-nilai untuk sebanyak mungkin variabel. Jika *direct solver* menemukan sebuah kendala dengan satu variabel yang tidak diketahui, itu menentukan nilai untuk variabel yang memenuhi kendala. *Direct solver* berhenti ketika variabel yang tidak diketahui sudah tidak ditemukan lagi atau tidak ada lagi sisa kendala dengan variabel

yang tidak diketahui. *Linier solver* di LINGO menggunakan metode simpleks yang direvisi dengan bentuk produk terbalik. *Nonlinier Solver* menggunakan algoritma *successive linear programming* (SLP) dan *generalized reduced gradient* (GRG). Metode *branch-and-bound* digunakan untuk menyelesaikan model bilangan bulat (*integer*). Untuk model linier *integer*, LINGO sangat memperhatikan *pre-processing*, yaitu menambahkan kendala “*cuts*” untuk membatasi daerah layak non *integer*. Pemotongan ini akan meningkatkan solusi untuk model *integer programming*. Model class yang akan ditemukan saat menyelesaikan masalah dengan menggunakan LINGO ditunjukkan pada Gambar 2.2.

<i>Abbreviation</i>	<i>Class</i>	<i>Description</i>
LP	Linear Program	All expressions are linear and the model contains no integer restrictions on the variables.
QP	Quadratic Program	All expressions are linear or quadratic, the model is convex, and there are no integer restrictions.
CONE	Conic Program	The model is a conic (second-order cone) program and all variables are continuous.
NLP	Nonlinear Program	At least one of the relationships in the model is nonlinear with respect to the variables.
MILP	Mixed Integer Linear Program	All expressions are linear, and a subset of the variables is restricted to integer values.
MIQP	Mixed Integer Quadratic Program	All expressions are either linear or quadratic, the model is convex, and a subset of the variables has integer restrictions.
MICONE	Mixed Integer Conic Program	The model is a conic (second-order cone) program, and a subset of the variables is restricted to integer values.
MINLP	Integer Nonlinear Program	At least one of the expressions in the model is nonlinear, and a subset of the variables has integer restrictions. <i>In general, this class of model will be very difficult to solve for all but the smallest cases.</i>
PILP	Pure Integer Linear Program	All expressions are linear, and all variables are restricted to integer values.
PIQP	Pure Integer Quadratic Program	All expressions are linear or quadratic, the model is convex, and all variables are restricted to integer values.
PICONE	Pure Integer Conic (Second-Order Cone) Program	The model is a conic (second-order cone) program, and all the variables are restricted to integer values.
PINLP	Pure Integer Nonlinear Program	At least one of the expressions in the model is nonlinear, and all variables have integer restrictions. <i>In general, this class of model will be very difficult to solve for all but the smallest cases.</i>

Gambar 2.2. Model Class
(Sumber : Lingo 13 Online Users Manual)