

BAB II

LANDASAN TEORI

2.1. Sistem Pendukung Keputusan (SPK)

2.1.1. Jenis Keputusan

Jenis keputusan terbagi atas 2 (dua) buah, yang pertama adalah keputusan terstruktur dan yang kedua adalah keputusan tidak terstruktur. Keputusan terstruktur mempunyai aturan aturan yang jelas dan teliti. Dipakai berulang dapat diprogramkan sehingga Keputusan ini dapat didelegasikan kepada orang lain atau komputerisasi. Jenis yang kedua yaitu keputusan tidak terstruktur mempunyai ciri kemunculan yang kadang sifat keputusan yang harus diambil mempunyai bersifat sehingga sifat analisisnya pun baru, tidak dapat didelegasikan, kadang alat analisisnya tidak lengkap dan bahkan keputusan lebih didominasi oleh intitusi [Umar, 2002].

2.1.2. Keterampilan Pengambilan Keputusan

Nilai keterampilan didalam pengambilan keputusan yang dimiliki oleh seorang pengambil keputusan misalnya manajer, tergantung dari beberapa faktor seperti faktor intelegensi, kapabilitas, kapasitas dan tanggung jawab. [Umar 2002].

2.2. Teori Keputusan

Keputusan adalah tindakan yang dilakukan oleh setiap orang pada umumnya dari waktu ke waktu. Baik hal itu dilakukan secara sadar atau tidak. Untuk mengambil keputusan yang dilakukan secara sadar memerlukan perhitungan yang cermat serta diperlukan pertimbangan, persiapan yang matang serta membuat analisis. Dan sering kali pengambilan keputusan tersebut memerlukan banyak bahan, keterangan dan pendapat orang lain yang memiliki peran dalam pengambilan keputusan tersebut.

Keputusan berkaitan dengan kegiatan yang ditujukan pada pencapaian suatu keputusan. Keputusan yang terlalu cepat diambil mungkin akan menghasilkan sesuatu yang tidak diinginkan dikarenakan ada beberapa faktor yang dipikirkan oleh manusia. Oleh karena itu yang menjadi masalah adalah kapan keputusan tersebut harus diambil agar yang merupakan ketidakpastian dapat dihindari dan dengan penuh harapan suatu keberhasilan dapat dicapai sebesar mungkin.

Keputusan dalam suatu organisasi merupakan hasil suatu proses komunikasi dan partisipasi dari keseluruhan organisasi. Hasil dari keputusan tersebut dapat berupa pernyataan alternatifnya untuk mencapai tujuan tertentu. Persoalan pengambilan keputusan dasarnya adalah bentuk pemilihan berbagai alternatif tindakan yang dipilih melalui proses tertentu dengan harapan akan menghasilkan keputusan yang terbaik. Dengan kata lain keputusan merupakan sebuah kesimpulan yang dicapai sesudah dilakukan pertimbangan, yang terjadi setelah satu kemungkinan terpilih sementara yang lain dikesampingkan. Dalam hal ini yang dimaksud dengan pertimbangan adalah menganalisa beberapa kemungkinan atau alternatif, lalu memilih satu diantaranya [Suryadi, 1998].

2.2.1. Pengertian Pengambilan Keputusan

Persoalan pengambilan keputusan adalah bentuk pemilihan berbagai alternatif tindakan yang mungkin dipilih prosesnya melalui mekanisme tertentu. Dengan harapan akan menghasilkan sebuah hasil yang baik. Penyusunan model keputusan adalah suatu cara untuk mengembangkan hubungan logis yang mendasari persoalan keputusan kedalam bentuk matematis yang mencerminkan hubungan yang terjadi antara faktor yang terlibat. Hal yang paling sulit dilakukan setelah keputusan didapat adalah segi penerapannya karena perlu menyakinkan semua orang yang terlibat, bahwa keputusan adalah yang terbaik [Suryadi, 1998].

Jadi pengambilan keputusan adalah bentuk-bentuk pemilihan berbagai alternatif tindakan yang mungkin dipilih

prosesnya melalui mekanisme tertentu dengan harapan akan menghasilkan keputusan yang terbaik, [Suryadi, 1998].

2.2.2. Tahap Pengambil Keputusan

Tahap pengambilan keputusan melalui beberapa fase yaitu :

1. Intelegence

Tahap ini merupakan proses pelurusan dan pendekatan dari lingkup problematika serta proses pengenalan masalah. Data masukan diperleh, diproses, dan diuji dalam rangka mengidentifikasi masalah.

2. Design

Tahap ini merupakan proses menentukan, mengembangkan dan menganalisa alternatif tindakan yang bisa dilakukan. Tahap ini meliputi proses untuk mengerti masalah. Menurunkan solusi dan menguji kelayaaan sistem.

3. Choice

Tahap ini dilakukan proses pemilihan diantara berbagai alternative yang mungkin dijalankan. Hasil pemilihan ini tersebut kemudian diimplementasikan dalam proses pengambilan keputusan [Suryadi, 1998].

2.3. Penelitian Sebelumnya

Penelitian tentang sistem pendukung keputusan dengan menggunakan metode *Promethee* telah banyak dilakukan oleh peneliti-peneliti, akademisi maupun praktisi-praktisi Teknologi Informasi (TI) terdahulu salah satunya dalam skripsi dengan judul “Aplikasi Sistem Pendukung Keputusan Dengan Metode *Promethee* Untuk Pemilihan Jurusan Di Sekolah Menengah Atas Nu I” (A. Saifudin Hajar, 2011). Dalam penelitian tersebut terlihat bahwa penerapan metode *Promethee* dapat membandingkan berbagai macam kriteria untuk perankingan penjurusan kelas XI, jika dibandingkan dengan cara konvensional maka perankingan tidak optimal, hal disebabkan karena tidak seluruh kriteria dilibatkan

untuk mengurangi perhitungan yang kompleks dan lama.

Penelitian tentang system pendukung keputusan dengan menggunakan metode *Promethee* telah banyak dilakukan oleh peneliti-peneliti, akademisi maupun praktisi-praktisi Teknologi Informasi (TI) terdahulu salah satunya dalam skripsi dengan judul “*Aplikasi Sistem pendukung keputusan (SPK) untuk seleksi siswa baru kelas unggulan dengan metode Promethee di SMP Al-Azhar Menganti*”. (Sukron, 2012). Dalam penelitian tersebut terlihat bahwa penerapan metode *Promethee* Metode yang diterapkan di SMP Al-Azhar Menganti masih menggunakan cara manual dan belum memperhatikan perbedaan antar siswa dalam hal kecerdasan dan kompetensi yang dimiliki antar siswa. Atas dasar inilah yang mendorong dilakukannya pembenahan atau bahkan pembuatan sebuah sistem yang dapat menunjang dalam mengambil suatu keputusan.

Preference Ranking Organization Method for Enrichment Evaluation (Promethee) adalah salah satu metode penentuan urutan atau prioritas dalam MCDM (*Multi Criterion Decision Making*). Penggunaan *Promethee* adalah menentukan dan menghasilkan keputusan dari beberapa alternatif. Metode *Promethee* dipilih karena metode ini merupakan suatu bentuk model pendukung keputusan yang digunakan untuk pengambilan keputusan dengan kriteria beragam, sehingga sangat sesuai jika digunakan dalam pengambilan keputusan yang memiliki lebih dari satu kriteria (multi kriteria).

2.4. PHP

Hypertext Preprocessor atau yang bisaa dikenal dengan *PHP* adalah sebuah bahasa *web server side* yang bersifat *open source*. *Script PHP* tersebut dimasukkan ke dalam dokumen *HTML* untuk diproses *web server* ketika ada *request* dari *user*. *PHP* juga didesain untuk dapat bekerja dengan kebanyakan *SQL server* termasuk *open source SQL server*, seperti *MySQL*.

PHP pada awalnya bernama *PHP/FI*, yakni singkatan dari *Personal Home Page/Form Interface*. Yang pertama kali diciptakan oleh Rasmus Lerdoff pada tahun 1994. Semula *PHP* diciptakan untuk menyimpan data

dari orang-orang yang telah berkunjung ke sebuah *website*, serta untuk mengetahui berapa jumlah orang yang telah berkunjung ke *website* tersebut. Namun, karena *software* ini disebarluaskan sebagai *software open source* sehingga dalam pertumbuhannya banyak sekali mendapatkan kontribusi atau masukan dari pengguna.

Pada dasarnya *PHP* dapat mengerjakan semua yang dapat dikerjakan oleh program *CGI (Common Gateway Interface)*, seperti menyimpan data yang dimasukkan melalui sebuah *form* dalam *website*, menampilkan isi *website* yang dinamis serta menerima *cookies*. Selain itu, kemampuan *PHP* yang paling menonjol adalah dukungan kebanyakan *database*. Adapun daftar *database* yang dapat diakses melalui *script PHP*, antara lain : *dBase, DBM, FilePro, mSQL, MySQL, ODBC, Oracle, Postgres, Sybase, Velocis*.

2.5. Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Fitur Android:

- *Framework* aplikasi, memungkinkan daur ulang dan penggantian komponen.
- *Browser* terintegrasi berbasis *engine Open Source WebKit* yang juga digunakan di *browser iPhone* dan *Nokia S60v3*.
- Rancangan *handset. Platform* disesuaikan dengan kebutuhan *VGA (Video Graphics Adapter)* yang lebih besar, *library* grafik 2D dan 3D yang berdasarkan pada spesifikasi *OpenGL ES 1.0* serta

layout smartphone yang tradisional.

- Konektivitas. Android mendukung berbagai teknologi konektivitas seperti GSM (*Global System for Mobile Communications*) EDGE (*Enhanced Data rates for GSM Evolution*), CDMA (*Code Division Multiple Access*), EV-DO (*Evolution-Data Optimized*), UMTS (*Universal Mobile Telecommunications System*), Bluetooth dan Wi-Fi (*Wireless Fidelity*).
- Pesan. Android mendukung pengiriman pesan dalam bentuk SMS (*Short Message Service*) dan MMS (*Multimedia Messaging Service*).
- Dukungan Java. *Software* yang ditulis dalam bahasa Java dapat dikompilasi dan akan dieksekusi pada mesin virtual Dalvik, yang merupakan implementasi dari VM (*Virtual Machine*) yang dirancang khusus untuk penggunaan perangkat bergerak.
- Dukungan media. Android mendukung beberapa format audio/video seperti: H.263, H.264 (dalam kontainer 3GP atau MP4), MPEG-4 SP, AMR, AMR-WB (dalam kontainer 3GP), AAC, HE-AAC (dalam kontainer MP4 atau 3GP), MP3, MIDI, OGG Vorbis, WAV, JPEG, PNG, GIF dan BMP.
- Dukungan *hardware* tambahan. Android mendukung penggunaan kamera, layar sentuh, GPS (*Global Positioning System*), pengukur kecepatan, *magnetometer*, akselerasi 2D *bit blits* (dengan orientasi *hardware, scaling*, konversi format piksel) dan akselerasi grafis 3D.
- *Market*. Mirip dengan *App Store* pada iPhone OS, Android Market adalah sebuah katalog aplikasi yang dapat di-*download* dan diinstal pada telepon seluler secara online, tanpa menggunakan PC (*Personal Computer*). Awalnya hanya aplikasi gratis saja yang didukung. Dan sejak tanggal 19 Februari 2009 aplikasi berbayar telah tersedia di *Android Market* untuk Amerika Serikat.
- *Multi-touch*. Android memiliki dukungan bawaan untuk *multi-touch* yang tersedia pada *handset* terbaru seperti HTC Hero. Pada awalnya fitur tersebut dinonaktifkan pada *level* kernel (mungkin untuk

menghindari pelanggaran paten terhadap teknologi layar sentuh Apple). Sejak Google merilis *update* untuk Nexus One dan berencana juga untuk merilis *update* untuk Motorola Droid yang memungkinkan *multi-touch*.

- Lingkungan pengembangan yang kaya, termasuk *emulator*, peralatan *debugging*, dan *plugin* untuk Eclipse IDE.

Arsitektur Android :

a. *Linux Kernel*

Android bergantung pada Linux Versi 2.6 untuk inti sistem pelayanan seperti keamanan, manajemen memori, proses manajemen, susunan jaringan, dan driver model. Linux kernel menyediakan driver layar, kamera, keypad, kamera, WiFi, memori flash, audio, dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan keamanan. Kernel juga bertindak sebagai lapisan yang abstrak antara *hardware* dan *software stack*-nya.

b. *Library Dasar*

Layer kedua di atas kernel terdiri dari *library* dasar Android. *Library* ini disusun menggunakan bahasa C++, dicompile sesuai dengan arsitektur pada *handphone*, dan sudah dimasukkan oleh manufaktur *handphone*.

Beberapa library-library penting dari Android adalah sebagai berikut:

- *Surface Manager* : Android menggunakan manajemen interface berbasis *composite*, seperti yang digunakan oleh Windows Vista atau Compiz, tetapi dalam bentuk yang lebih sederhana. Hal ini menjadikan sistem mampu membuat efek-efek tampilan menarik pada layar.
- *2D dan 3D Graphics* : Elemen dua dan tiga dimensi dapat digabungkan dalam satu tampilan Android. Library akan menggunakan perangkat 3D jika *handphone* memiliki hardware untuk akselerasi 3D.
- *SQLite* : Mesin database yang kuat dan ringan.

c. *Android Runtime*

Android terdiri dari satu set *core libraries* yang menyediakan sebagian besar fungsi yang tersedia pada *core libraries* bahasa pemrograman Java. Setiap aplikasi menjalankan sendiri prosesnya dalam Android, dengan masing-masing *instance* dari *Dalvik Virtual Machine*. Dalvik dirancang agar perangkat dapat menjalankan banyak aplikasi secara efisien.

d. *Application Framework*

Pengembang memiliki akses penuh terhadap *framework* API yang sama yang digunakan oleh aplikasi inti. Arsitektur aplikasi dirancang agar komponen dapat digunakan kembali (*reuse*) dengan mudah. Semua aplikasi yang merupakan rangkaian layanan dan sistem, termasuk:

- *View set* kaya dan *extensible* yang dapat digunakan untuk membangun aplikasi, termasuk daftar, kotak teks, tombol, dan bahkan sebuah *embeddable web*.
- *Content provider* yang memungkinkan aplikasi untuk mengakses data (seperti dari daftar kontak telepon) atau dari data mereka sendiri.
- *Resource manager*, yang menyediakan akses ke kode sumber non-lokal seperti string, gambar, dan tata letak file.
- *Notification manager* yang memungkinkan aplikasi untuk ditampilkan dalam *alert* status bar.
- *Activity manager* yang mengelola siklus hidup aplikasi dan menyediakan navigasi umum.

e. *Application*

Android telah menyertakan aplikasi inti seperti *email client*, SMS, kalender, peta, browser, kontak, dan lain-lain. Semua aplikasi tersebut ditulis dengan menggunakan bahasa pemrograman Java. Pada *layer* inilah developer menempatkan aplikasi yang dibuat.

2.6. Promethee

2.6.1. Karakteristik Metode Promethee

Promethee merupakan salah satu ranking dalam Multiple Criteria Decision Making (MCDM). Pengertian dari metode Promethee adalah sebagai berikut:

“*Promethee* adalah suatu metode menentukan urutan (prioritas) dalam analisa multikriteria. Masalah pokoknya adalah kesederhanaan, kejelasan, dan kestabilan. Dugaan dari dominasi kriteria yang digunakan dalam *promethee* adalah penggunaan nilai dalam hubungan *outranking* (*Metode yang dapat menangani kriteria kualitatif dan kuantitatif*). Semua parameter yang dinyatakan mempunyai pengaruh nyata menurut pandangan ekonomi [Suryadi,2002].

Prinsip yang digunakan adalah penetapan prioritas alternatif yang telah ditetapkan berdasarkan pertimbangan $\forall i | f_i(,) \rightarrow \mathfrak{R}$ [*real world*]), dengan kaidah dasar:

$$\text{Max}\{f_1(x), f_2(x), f_3(x) \dots, f_j(x) \dots, f_k(x) | x \in \mathfrak{K}\}$$

Dimana K adalah sejumlah kumpulan alternatif dan f_i ($i = 1,2,3,\dots, k$) merupakan nilai atau ukuran relatif kriteria untuk masing-masing alternatif. Dalam aplikasinya sejumlah kriteria telah ditetapkan untuk menjelaskan K yang merupakan penilaian dari \mathfrak{R} (*real world*). *Promethee* termasuk dalam keluarga dari metode *outranking* yang dikembangkan oleh B. Roy dan meliputi 2 fase:

1. Membangun hubungan *outranking* dari K.
2. Eksploitasi dari hubungan ini memberikan jawaban optimasi kriteria dalam paradigma permasalahan multikriteria.

Pada tahap pertama, nilai hubungan *outranking* berdasarkan pertimbangan dominasi masing-masing kriteria. Indeks preferensi ditentukan dan nilai *outranking* secara grafis disajikan berdasarkan preferensi dari pembuat keputusan.

Pada tahap kedua, eksploitasi dilakukan dengan mempertimbangkan nilai *Leaving Flow* dan *Entering Flow* pada grafik nilai outranking. Urutan parsial untuk *Promethee I* dan urutan lengkap pada *Promethee II*. Data dasar untuk evaluasi dengan metode *Promethee* terdapat pada Tabel 2.1

Table 2.1 Data Dasar Analisa *Promethee*

| | $f_1(.)$ | $f_2(.)$ | ... | $f_i(.)$ | ... | $f_k(.)$ |
|-------|------------|------------|-----|------------|-----|------------|
| a_1 | $f_1(a_1)$ | $f_2(a_1)$ | ... | $f_i(a_1)$ | ... | $f_k(a_1)$ |
| a_2 | $f_1(a_2)$ | $f_2(a_2)$ | ... | $f_i(a_2)$ | ... | $f_k(a_2)$ |
| ... | ... | ... | ... | ... | ... | ... |
| a_i | $f_1(a_i)$ | $f_2(a_i)$ | ... | $f_i(a_i)$ | ... | $f_k(a_i)$ |
| ... | ... | ... | ... | ... | ... | ... |
| a_n | $f_1(a_n)$ | $f_2(a_n)$ | ... | $f_i(a_n)$ | ... | $f_k(a_n)$ |

Keterangan :

1. a_1, a_2, a_i, a_n : a alternative potensial.
2. f_1, f_2, f_i, f_k : f criteria evaluasi.

2.6.2. Dominasi Kriteria

Penyampaian intensitas (P) dari preferensi alternatif a terhadap alternatif b sedemikian rupa sehingga:

- f. $P(a,b) = 0$, berarti tidak ada beda (*indifferent*) antara a dan b, atau tidak ada preferensi dari a lebih baik dari b.
- g. $P(a,b) \sim 0$, berarti lemah preferensi dari a lebih baik dari b.
- c. $P(a,b) \sim 1$, berarti kuat preferensi dari a lebih baik dari b kuat.
- d. $P(a,b) = 1$, berarti mutlak preferensi dari a lebih baik dari b.

Dalam metode ini, fungsi preferensi sering kali menghasilkan nilai fungsi yang berbeda antara dua evaluasi sehingga:

$$P(a,b) = P(f(a) - f(b))$$

Untuk semua kriteria, suatu alternatif akan dipertimbangkan memiliki nilai kriteria yang lebih baik ditentukan oleh nilai f dan nilai akumulasi dari nilai ini menentukan nilai preferensi atas

masing-masing alternatif yang akan dipilih.

2.6.3. Rekomendasi fungsi preferensi untuk keperluan aplikasi

Menurut Kadarsah [1998:148], dalam metode Promethee terdapat enam fungsi preferensi kriteria, yaitu:

- a. Kriteria biasa (*Usual Criterion*)
- b. Kriteria quasiasia (*Quasi Criterion*)
- c. Kriteria *linear*
- d. Kriteria *level*
- e. Kriteria dengan preferensi *linear* dan area yang tidak berbeda
- f. Kriteria *Gaussian*

Hal ini tentu saja tidak mutlak, tetapi bentuk ini cukup baik untuk beberapa kasus. Untuk memberikan gambaran yang lebih baik terdapat area yang tidak sama, digunakan fungsi selisih nilai kriteria antara alternatif H(d) dimana hal ini mempunyai hubungan langsung pada fungsi preferensi. Penjelasan masing-masing kriteria tersebut adalah sebagai berikut:

1. Kriteria biasa (*Usual Criterion*)

$$H(d) = \begin{cases} 0 & \text{jika } d \leq 0 \\ 1 & \text{jika } d > 0 \end{cases} \dots\dots\dots (2.1)$$

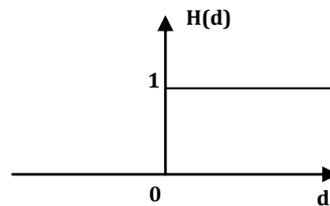
dimana:

d : selisih nilai kriteria { $d = f(a) - f(b)$ }

H(d) : fungsi selisih kriteria antar alternatif.

Pada kasus ini tidak ada beda (sama penting) antara a dan b jika $f(a) = f(b)$; apabila nilai kriteria pada masing-masing alternatif memiliki nilai berbeda, pembuatan keputusan membuat preferensi mutlak untuk alternatif yang memiliki nilai lebih baik. Untuk memilih kasus preferensi pada kriteria biasa, ilustrasinya dapat dilihat dari perlombaan renang, seorang peserta dengan peserta lainnya akan memiliki peringkat yang mutlak berbeda walaupun hanya dengan selisih nilai (waktu) teramat kecil, dan dia akan memiliki peringkat yang sama jika dan hanya jika

waktu tempuhnya sama atau selisih nilai diantara keduanya sebesar nol. Fungsi $H(d)$ untuk fungsi preferensi ini ditunjukkan pada Gambar 2.1



Gambar 2.1. Kriteria biasa.

(Sumber :J.P.Brans, dan B. Marsechal,1997)

2. Kriteria quasi (*Quasi Criterion*)

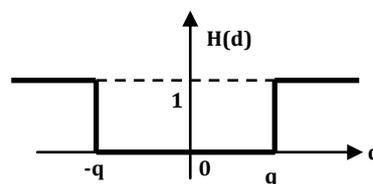
$$H(d) = \begin{cases} 0 & \text{jika } -q \leq d \leq q \\ 1 & \text{jika } d < -q \text{ atau } d > q \end{cases} \quad \dots\dots\dots (2.2)$$

dimana:

d : selisih nilai kriteria { $d = f(a) - f(b)$ }

$H(d)$: fungsi selisih nilai kriteria antara alternatif.

Parameter (q) : harus merupakan nilai yang tetap, $q = 0$.



Gambar 2.2 Kriteria quasi.

(Sumber :J.P.Brans, dan B. Marsechal,1997)

Kriteria ini memiliki alternatif preferensi yang sama penting selama selisih atau nilai $H(d)$ dari masing-masing alternatif untuk kriteria tertentu tidak melebihi nilai q , dan apabila selisih hasil evaluasi untuk masing-masing alternatif melebihi nilai q maka akan terjadi bentuk preferensi mutlak. Fungsi $H(d)$ untuk fungsi preferensi ini ditunjukkan pada Gambar 2.2.

Misalnya seorang akan dipandang mutlak lebih kaya apabila selisih nilai kekayaannya lebih besar dari Rp10.000.000 dan apabila selisih kekayaannya kurang dari Rp 10.000.000

dipandang sama kaya.

3. Kriteria linear

$$H(d) = \begin{cases} 0 & \text{jika } d \leq 0 \\ \frac{d}{p} & \text{jika } 0 < d \leq p \\ 1 & \text{jika } d > p \end{cases} \quad \dots\dots\dots(2.3)$$

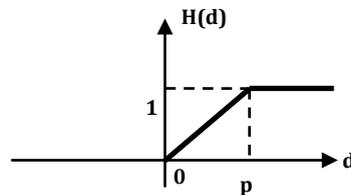
dimana:

d : selisih nilai kriteria { $d = f(a) - f(b)$ }

p : nilai kecenderungan atas.

$H(d)$: fungsi selisih nilai kriteria antar alternatif.

Kriteria ini menjelaskan bahwa selama nilai selisih memiliki nilai yang lebih rendah dari p , preferensi dari pembuat keputusan meningkat secara linear dengan nilai d . Jika nilai d lebih besar dibandingkan dengan nilai p , maka terjadi referensi mutlak. Fungsi $H(d)$ untuk fungsi preferensi ini ditunjukkan pada Gambar 2.3.



Gambar 2.3 Kriteria linear.

(Sumber :J.P.Brans, dan B. Marsechal,1997)

Misal akan terjadi preferensi dalam hubungan linear kriteria kecerdasan seseorang dengan cara lain apabila nilai seseorang berselisih dibawah 30, apabila diatas 30 poin maka mutlak dikatakan orang itu lebih cerdas dibandingkan dengan orang lain.

4. Kriteria level

$$H(d) = \begin{cases} 0 & \text{jika } |d| \leq q \\ 0,5 & \text{jika } q < |d| \leq p \\ 1 & \text{jika } d > p \end{cases} \quad \dots\dots\dots(2.4)$$

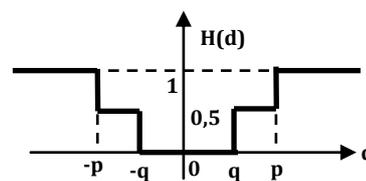
dimana:

p : nilai kecenderungan atas.

$H(d)$: fungsi selisih kriteria antar alternatif.

Parameter (q) : harus merupakan nilai yang tetap, $q = 0$

Dalam kasus ini kecenderungan tidak berbeda q dan kecenderungan preferensi p ditentukan secara simultan. Jika d berada di antara nilai q dan p , hal ini berarti situasi preferensi yang lemah ($H(d) = 0.5$).



Gambar 2.4 Kriteria level.

(Sumber :J.P.Brans, dan B. Marsechal,1997)

Gambar 2.4 menjelaskan pembuat keputusan telah menentukan kedua kecenderungan untuk kriteria ini. Bentuk kriteria level ini dapat dijelaskan misalnya dalam penetapan nilai preferensi jarak tempuh antar kota. Misalnya jarak antara Surabaya-Bromo sebesar 60 km, Bromo-Kalibaru sebesar 68 km. Kalibaru-Ijen sebesar 45 km, Bromo-Ijen 133 km. Dan telah ditetapkan bahwa selisih dibawah 10 km maka dianggap jarak antar kota tersebut adalah tidak berbeda, selisih jarak sebesar 10-30 km relative berbeda dengan preferensi yang lemah, sedangkan selisih diatas 30 km diidentifikasi memiliki preferensi mutlak berbeda.

Dalam kasus ini, selisih jarak antara Surabaya-Bromo dan Bromo-Kalibaru dianggap tidak berbeda ($H(d) = 0$) karena selisih jaraknya dibawah 10 km, yaitu $(68-60) \text{ km} = 8 \text{ km}$, sedangkan preferensi jarak antara Bromo-kalibaru dan Kalibaru-Ijen dianggap berbeda dengan preferensi lemah ($H(d)=0.5$) karena memiliki selisih yang berada pada interval 10-30 km, yaitu $(68-45) \text{ km} = 23 \text{ km}$. Dan terjadi preferensi mutlak ($H(d) = 1$) antara

jarak Bromo_Ijen dan KalibaruIjen karena memiliki selisih jarak lebih dari 30 km.

5. Kriteria dengan preferensi *linear* dan area yang tidak berbeda

$$H(d) = \begin{cases} 0 & \text{jika } |d| \leq q \\ (|d| - q)/(p - q) & \text{jika } q < |d| \leq p \\ 1 & \text{jika } p < |d| \end{cases} \dots\dots\dots(2.5)$$

dimana:

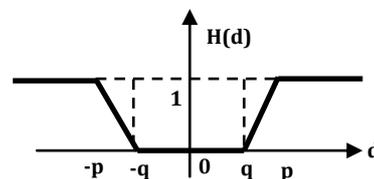
d : selisih nilai kriteria { d = f(a) - f(b) }

H(d) : fungsi selisih nilai kriteria antar alternatif.

Parameter (p) : nilai kecendrungan atas.

Parameter (q) : harus merupakan nilai yang tetap, q = 0

Pada kasus ini, pengambil keputusan mempertimbangkan peningkatan preferensi secara linear dan tidak berbeda hingga preferensi mutlak dalam area antara dua kecenderungan q dan p. Dua parameter tersebut telah ditentukan dimana fungsi H adalah hasil perbandingan antar alternatif, seperti yang ditunjukkan pada Gambar 2.5.

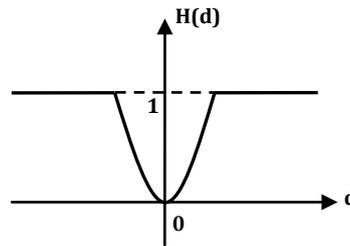


Gambar 2.5 Kriteria preferensi linear dan area yang tidak berbeda (Sumber :J.P.Brans, dan B. Marsechal,1997)

6. Kriteria gaussian

$$H(d) = 1 - \exp\{-d^2/2\sigma^2\} \dots\dots\dots (2.6)$$

Fungsi ini bersyarat apabila telah ditentukan nilai σ , dimana dapat dibuat berdasarkan distribusi normal dalam statistic. Fungsi ini ditunjukkan seperti pada Gambar 2.6.



Gambar 2.6 Kriteria gaussian.

(Sumber :J.P.Brans, dan B. Marsechal,1997)

2.6.4. Penentuan Tipe Preferensi

Seperti telah disebutkan diatas, maka proses penentuan preferensi merupakan langkah yang penting sehingga saat perhitungan indeks preferensi dapat representatif terhadap permasalahan. Dalam membantu penentuan tingkat preferensi dapat ditunjukkan pada Tabel 2.2.

| pertimbangan | Tingkat fungsi preferensi | | | | | |
|--------------------------------------|---------------------------|---------|---------|---------|---------|---------|
| | I | II | III | IV | V | VI |
| Akurasi | Kasar | Kasar | Akurat | Kasar | Akurat | Akurat |
| Kecendrungan tidak berbeda $ d < q$ | Tidak | Ya | Tidak | Ya | Ya | Tidak |
| Kecendrungan kokoh mutlak $ d < q$ | Tidak | Tidak | Tidak | Ya | Ya | Tidak |
| Distribusi normal | mungkin | mungkin | mungkin | mungkin | mungkin | mungkin |

2.6.5. Indeks Preferensi Multikriteria

Tujuan pembuatan keputusan adalah menetapkan fungsi preferensi P_i dan π_i untuk semua kriteria f_i ($i = 1,2,3,\dots,k$) dari masalah optimasi kriteria majemuk. Bobot (weight) x_i merupakan ukuran relatif dari kepentingan kriteria f_i , jika semua kriteria memiliki nilai kepentingan yang sama dalam pengambilan maka semua nilai bobot adalah sama.

Indeks preferensi multikriteria ditentukan berdasarkan rata-

rata bobot dari fungsi preferensi P_i .

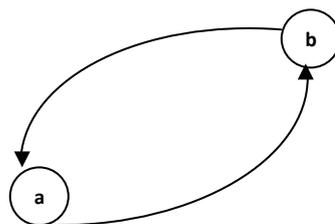
$$\wp(a, b) = \sum_{i=1}^n \pi P_i(a, b): \forall a, b \in A$$

dimana:

1. $\wp(a, b)$, merupakan intensitas preferensi pembuat keputusan yang menyatakan bahwa alternatif a lebih baik dari alternatif b dengan pertimbangan secara simultan dari seluruh kriteria.
2. $\wp(a, b) = 0$, menunjukkan preferensi yang lemah untuk alternatif a lebih dari alternatif b berdasarkan semua kriteria.
3. $\wp(a, b) = 1$, menunjukkan preferensi yang kuat untuk alternatif a lebih dari alternatif b berdasarkan semua kriteria.

Indek preferensi ditentukan berdasarkan nilai hubungan outranking pada sejumlah kriteria dari masing-masing alternative.

Hubungan ini dapat ditunjukkan sebagai grafik nilai outranking, nodenodenya merupakan alternatif berdasarkan penilaian kriteria tertentu. Diantaranya dua node (alternatif) a dan b, merupakan garis lengkung yang mempunyai nilai $\wp(a, b)$ dan $\wp(b, a)$ (tidak ada hubungan khusus antara $\wp(a, b)$ dan $\wp(b, a)$). Hal ini ditunjukkan pada Gambar 2.7.



Gambar 2.7 Hubungan antar Node

2.6.6. Arah Preferensi Multikriteria

Arah preferensi terbagi menjadi dua arah yaitu Leaving Flow (LF) dan Entering Flow (EF). Leaving Flow merupakan ukuran dari karakter outranking a, sedangkan Entering Flow merupakan ukuran karakter a yang di outrank. Secara sistematis

dapat ditentukan Leaving Flow dengan persamaan :

$$\Phi^+(a) = \frac{1}{n-1} \sum_{x \in A} \wp(a, x) \quad \dots\dots\dots (2.7)$$

Adapun persamaan Entering Flow adalah:

$$\Phi^-(a) = \frac{1}{n-1} \sum_{x \in A} \wp(a, x) \quad \dots\dots\dots (2.8)$$

Adapun persamaan Net Flow adalah:

$$\Phi(a) = \Phi^+(a) - \Phi^-(a) \quad \dots\dots\dots (2.9)$$

2.6.7. Perankingan dalam Promethee

Dalam metode Promethee proses perankingan dilakukan melalui dua perankingan yaitu Promethee I (Promethee parsial) dan Promethee II (Promethee complete). Perankingan Promethee I didasarkan pada masing-masing nilai Leaving Floe dan Entering Floe. Semakin besar nilai Leaving Flow dan semakin kecil nilai Entering Floe maka alternatif semakin baik. Jika nilai ranking Leaving Floe dan Entering Floe sama maka hasil ranking Promethee I menjadi solusi metode Promethee. Tetapi, jika sebaliknya maka proses harus dilanjutkan ke Promethee II. Promethee II didasarkan pada nilai dari Net Flow-nya. Semakin besar nilai Net Flow maka semakin tinggi rankingnya.

Adapun Perbedaan Promethee dan AHP seperti pada tabel 2.3

| Metode | Refrensi | Kelebihan | Kekurangan |
|---|--|---|--|
| <i>Analytical Hierarchy process (AHP)</i> | <ul style="list-style-type: none"> • Saaty (1986) • Nurul Hastuti (2005) | <ul style="list-style-type: none"> • Kemampuannya memecahkan masalah dalam objective dan multikriteria yang berdasar pada perbandingan preferensi dari setiap elemen dalam hirarki. • Dalam hal kesatuan AHP membuat permasalahan yang luas dan tidak terstruktur menjadi fleksibel dan | <ul style="list-style-type: none"> • Tidak dapat digunakan untuk problem keputusan yang diketahui adanya interaksi dan ketergantungan level atas pada elemen level bawah sehingga tidak bisa dibuat struktur hierarki. • Tidak mampu menangani masalah |

| | | | |
|--|--|--|---|
| | | <p>mudah dipahami.</p> <ul style="list-style-type: none">• AHP tidak mengharuskan adanya consensus tapi menggabungkan hasil judgment yang berbeda. | <p>ptimasi terhadap kendala yang sangat mungkin ada dalam permasalahan pemilihan alternative optimal.</p> |
|--|--|--|---|