

BAB 2

LANDASAN TEORI

2.1 Konsep Sistem Pendukung Keputusan (SPK)

Sebelum membahas tentang definisi SPK, perlu diketahui definisi dari beberapa istilah yang berkaitan dengan SPK, antara lain sebagai berikut:

2.1.1 Klasifikasi Keputusan

Keputusan diklasifikasikan menjadi tiga (O'Brien, 2005:438), yaitu:

1. Keputusan terstruktur

Keputusan terstruktur melibatkan situasi dimana prosedur yang diikuti ketika keputusan diperlukan, dapat disebutkan lebih awal. Contoh: Keputusan pemesanan ulang persediaan yang dihadapi oleh kebanyakan bisnis.

2. Keputusan tak terstruktur

Keputusan tak terstruktur melibatkan situasi keputusan dimana tidak mungkin menentukan lebih awal mengenai prosedur keputusan yang harus diikuti.

3. Keputusan semi terstruktur

Beberapa prosedur keputusan dapat ditentukan, namun tidak cukup untuk mengarah ke suatu keputusan yang direkomendasikan.

2.1.2 Definisi SPK

Sistem Pendukung Keputusan (*Decision Support System*) adalah sistem informasi berbasis komputer yang menyediakan dukungan informasi yang interaktif bagi manajer dan praktisi bisnis selama proses pengambilan keputusan (O'Brien, 2005: 448).

SPK dibangun tentunya mempunyai tujuan yang ingin dicapai oleh seorang pembuat keputusan. Menurut Aji Supriyanto (2005:260) tujuan SPK adalah sebagai "*second opinion*" atau "*information sources*" sebagai bahan pertimbangan seorang manajer sebelum memutuskan kebijakan.

2.1.3 Komponen SPK

Menurut Aji Supriyanto (2005:260) SPK dibangun oleh tiga komponen, yaitu:

a. Database

Sistem *Database* adalah kumpulan semua data yang dimiliki oleh perusahaan baik data dasar maupun transaksi sehari-hari yang dilakukan.

b. Model base

Model base adalah suatu model yang merepresentasikan permasalahan dalam format kuantitatif.

c. Software System

Software System adalah paduan antara database dan model base, setelah sebelumnya direpresentasikan ke dalam bentuk model yang dimengerti oleh sistem komputer.

Sedangkan menurut Tata Sutabri (2005:200) SPK terdiri dari 4 komponen, yaitu:

a. Dialog

Alat untuk berinteraksi antara komputer dengan pemakainya. Pemakai harus bisa mengerti apa arti informasi yang dihasilkan. Ini berarti, sistem (komputer beserta programnya) mudah dipakai (*user friendly*). Ditinjau dari sudut pemakainya, pemakai harus pula belajar dan berlatih cara penggunaannya serta arti yang dihasilkan.

b. Model

Model serta sistem yang membolehkan pemakai memilih model yang cocok. Tiga macam model yang biasa digunakan adalah:

1. Optimalisasi: mencari yang terbaik. Contohnya membuat jadwal, membuat perbandingan *linear programming*, simulasi, dan lain sebagainya.
2. Statistik / matematis: menggambarkan masalah dengan standar kuantifikasi yang ada. Contohnya *forecasting*, fungsi

kemungkinan (probabilitas), proyeksi penjualan, dan lain sebagainya.

3. Financial: mencari kesempatan yang lebih menguntungkan.
Contohnya: investasi, *cash flow* dan lain sebagainya.

c. Database

Kumpulan dari item data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di hardware komputer dan dengan software untuk melakukan manipulasi untuk kegunaan tertentu.

d. Data

suatu angka atau kelompok angka yang mempunyai arti atau nilai

Dari uraian mengenai komponen SPK diatas, untuk mengembangkan SPK dengan metode *Tsukamoto*, dipilih komponen SPK sebagai berikut: *Model base, Database, dan Software system*.

2.1.4 Validitas SPK

Validitas SPK digunakan untuk mengetahui SPK valid atau tidak. Pengujian Validitas SPK dilakukan dengan membandingkan hasil perhitungan SPK dengan hasil perhitungan manual. Misalkan ada n buah data yang akan digunakan untuk menguji tingkat validitas SPK seperti disajikan dalam Tabel 2.1.

Tabel 2.1 Hasil uji validitas SPK

No.	Data ke-	SPK	Perhitungan manual	KET (T/F)
1	1	Hasil SPK-1	Hasil manual-1	T
2	2	Hasil SPK-2	Hasil manual-2	F
3	-	-	-	-
4	-	-	-	-
5	-	-	-	-
6	n	Hasil SPK-n	Hasil manual-n	T

Keterangan:

T = True. Terjadi apabila hasil perhitungan SPK sama dengan hasil perhitungan manual.

F = False. Terjadi apabila hasil perhitungan SPK tidak sama dengan hasil perhitungan manual.

Menurut Teddy Rismawan (2008:6) berdasarkan pengujian validitas yang telah dilakukan, tingkat validitas SPK dapat dicari dengan persamaan

$$\text{Tingkat validitas SPK} = \frac{\text{banyaknyahasilpengujianbernilaiT}}{\text{banyaknyadatasample}} \times 100 \% \dots (2.1)$$

2.2 Himpunan dan Logika Fuzzy

2.2.1 Dari Himpunan Klasik ke Himpunan Samar (*fuzzy*)

Misalkan U sebagai semesta pembicaraan (himpunan semesta) yang berisi semua anggota yang mungkin dalam setiap pembicaraan atau aplikasi. Misalkan himpunan tegas A dalam semesta pembicaraan U. Dalam matematika ada tiga metode atau bentuk untuk menyatakan himpunan, yaitu metode pencacahan, metode pencirian dan metode keanggotaan. Metode pencacahan digunakan apabila suatu himpunan didefinisikan dengan mancacah atau mendaftar anggota-anggotanya. Sedangkan metode pencirian, digunakan apabila suatu himpunan didefinisikan dengan menyatakan sifat anggota-anggotanya. (Setiadji, 2009: 8). Dalam kenyataannya, cara pencirian lebih umum digunakan, kemudian setiap himpunan A ditampilkan dengan cara pencirian sebagai berikut:

$$A = \{x \in U \mid x \text{ memenuhi suatu kondisi}\} \dots (2.2)$$

Metode ketiga adalah metode keanggotaan yang mempergunakan fungsi keanggotaan nol-satu untuk setiap himpunan A yang dinyatakan sebagai $\mu_A(x)$.

$$\mu_A(x) = \begin{cases} 1, & \text{jika } x \in A \\ 0, & \text{jika } x \notin A \end{cases} \dots (2.3)$$

Menurut Nguyen dkk (2003: 86) fungsi pada persamaan (2.3) disebut fungsikarakteristik atau fungsi indikator. Suatu himpunan *fuzzy* A di dalam semesta pembicaraan U didefinisikan sebagai himpunan yang bercirikan

suatu fungsi keanggotaan μ_A , yang mengawankan setiap $x \in U$ dengan bilangan real di dalam interval $[0,1]$, dengan nilai $\mu_A(x)$ menyatakan derajat keanggotaan x di dalam A .

Dengan kata lain jika A adalah himpunan tegas, maka nilai keanggotaannya hanya terdiri dari dua nilai yaitu 0 dan 1. Sedangkan nilai keanggotaan di himpunan *fuzzy* adalah interval tertutup $[0,1]$.

2.2.2 Atribut

Himpunan fuzzy memiliki 2 atribut (Sri Kusumadewi dan Hari Purnomo, 2004:6), yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: Muda, Parobaya, Tua.
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 25, 50, dsb.

2.2.3 Istilah-istilah dalam logika fuzzy

Ada beberapa istilah yang perlu diketahui dalam memahami sistem *fuzzy*, yaitu:

2.2.3.1 Variabel *fuzzy*

Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy* (Sri Kusumadewi dan Hari Purnomo, 2004: 6). Contoh: Umur, Temperatur, Permintaan, Persediaan, Produksi, dan sebagainya.

2.2.3.2 Himpunan fuzzy

Misalkan X semesta pembicaraan, terdapat A di dalam X sedemikian sehingga:

$$A = \{ x, \mu_A[x] \mid x \in X, \mu_A : x \rightarrow [0,1] \} \dots\dots\dots (2.4)$$

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Misalkan $X=Umur$ adalah variabel *fuzzy*. Maka dapat didefinisikan himpunan “Muda”, “Parobaya”, dan “Tua” (Ginanjar Abdurrahman, 2011: 10).

2.2.3.3 Semesta Pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Contoh: semesta pembicaraan untuk variabel umur: $[0,+\infty)$. (Sri Kusumadewi dan Hari Purnomo, 2004:7). Sehingga semesta pembicaraan dari variable umur adalah $0 \leq umur < +\infty$. Dalam hal ini, nilai yang diperbolehkan untuk dioperasikan dalam variable umur adalah lebih besar dari atau sama dengan 0, atau kurang dari positif tak hingga.

2.2.3.4 Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif. Contoh domain himpunan *fuzzy*: Muda $= [0,45]$ (Sri Kusumadewi dan Hari Purnomo, 2004: 8).

2.2.4 Fungsi Keanggotaan

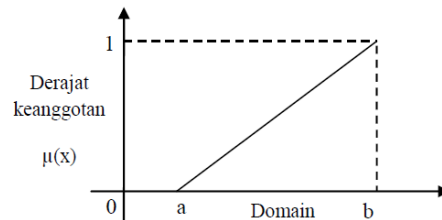
Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik input ke dalam nilai keanggotaan yang memiliki interval 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan

adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan, yaitu :

2.2.4.1 Representasi linier

Pada representasi linier, pemetaan input ke derajat keanggotaannya digambarkan sebagai garis lurus. Ada 2 keadaan himpunan fuzzy yang linier.

Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi. Ditunjukkan pada gambar 2.1

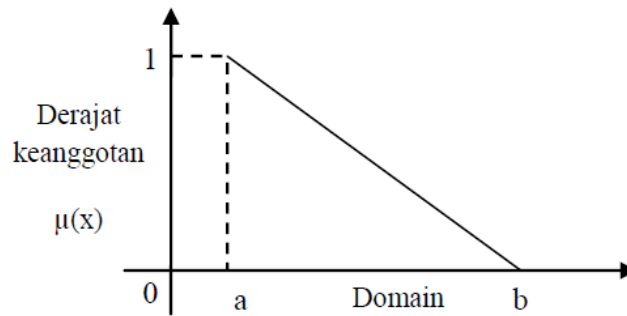


Gambar 2.1 Representasi linier naik (Kusumadewi dan Purnomo, 2010)

Fungsi Keanggotaan :

$$\mu[x] = \begin{cases} 0 & , x < a \\ (x - a)/(b - a) & , a \leq x \leq b \\ 1 & , x > b \end{cases} \dots\dots\dots(2.5)$$

Kedua, garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah, lihat pada gambar 2.2



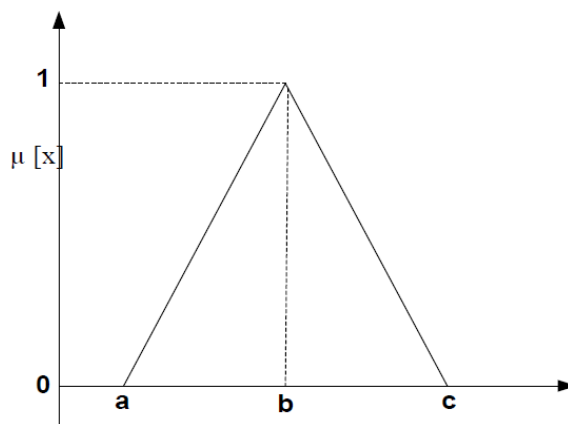
Gambar 2.2Representasi linier turun (Kusumadewi dan Purnomo, 2010)

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 1 & , x < a \\ (b - x)/(b - a) & , a \leq x \leq b \\ 0 & , x > b \end{cases} \dots\dots\dots(2.6)$$

2.2.4.2 Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier) seperti terlihat pada gambar 2.3.



Gambar 2.3Representasi Kurva Segitiga (Kusumadewi dan Purnomo, 2010)

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0 & , x \leq a \text{ atau } x \geq b \\ (x - a)/(b - a) & , a \leq x \leq b \\ (b - x)/(c - b) & , b \leq x \leq c \end{cases} \dots\dots\dots(2.7)$$

2.2.5 Teori Operasi Himpunan

Menurut Lin dan Lee (1996: 27) Ada dua operasi pokok dalam himpunan *fuzzy*, yaitu:

2.2.5.1 Konjungsi fuzzy

Konjungsi *fuzzy* dari A dan B dilambangkan dengan $A \wedge B$ dan didefinisikan oleh:

$$\mu_{A \wedge B} = \mu_A(x) \cap \mu_B(y) = \min(\mu_A(x), \mu_B(y)) \dots\dots\dots(2.8)$$

2.2.5.2 Disjungsi fuzzy

Disjungsi *fuzzy* dari A dan B dilambangkan dengan $A \vee B$ dan didefinisikan oleh:

$$\mu_{A \vee B} = \mu_A(x) \cup \mu_B(y) = \max(\mu_A(x), \mu_B(y)) \dots\dots\dots(2.9)$$

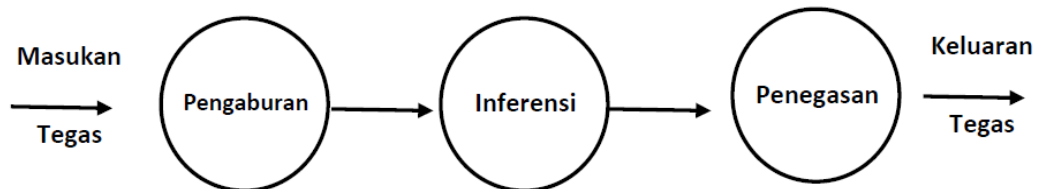
2.2.6 Sistem Inference Fuzzy

Sistem inferensi fuzzy merupakan kerangka komputasi yang didasarkan pada teori himpunan fuzzy, aturan fuzzy berbentuk IF-THEN, dan penalaran fuzzy. Sistem inferensi fuzzy menerima input crisp. Input ini kemudian dikirim ke basis pengetahuan yang berisi n aturan fuzzy dalam bentuk IF-THEN. Fire strength akan dicari pada setiap aturan. Apabila jumlah aturan lebih dari satu, maka akan dilakukan agregasi dari semua aturan. Selanjutnya, pada hasil agregasi akan dilakukan *defuzzy* untuk mendapatkan nilai *crisp* sebagai output sistem.

Pada gambar 2.3 merupakan penerapan logika fuzzy dapat meningkatkan kinerja sistem kendali dengan menekan fungsi-fungsi liar pada keluaran yang disebabkan oleh fluktuasi pada variabel masukannya. Pendekatan logika fuzzy secara garis besar diimplementasikan dalam tiga tahapan yang dapat dijelaskan sebagai berikut :

1. Tahap pengaburan (*fuzzification*) yakni pemetaan dari masukan tegas ke himpunan kabur.
2. Tahap inferensi, yakni pembangkitan aturan kabur.

3. Tahap penegasan (*defuzzification*), yakni transformasi keluaran dari nilai kabur ke nilai tegas



Gambar 2.4 Tahapan proses dalam logika kabur (Fanoeel Thamrin : 2012)

2.2.6.1 Operasi Himpunan Fuzzy

Seperti pada himpunan konvensional, ada operasi himpunan juga pada himpunan fuzzy. Hasil operasi 2 himpunan disebut juga fire strength atau α -predikat. Ada 3 operator pada operasi himpunan fuzzy, yaitu :

2.2.6.1.1 Operator AND (Interseksi / irisan)

Berhubungan dengan operasi irisan himpunan. Diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan. Misalnya operasi AND nilai keanggotaan himpunan fuzzy A dan B, $\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$.

2.2.6.1.2 Operator OR (Union / gabungan)

Berhubungan dengan operasi gabungan himpunan. Diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan. Misalnya operasi OR nilai keanggotaan himpunan fuzzy A dan B, $\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$.

2.2.6.1.3 Operator NOT (komplemen)

Berhubungan dengan operasi komplemen. Misalnya operasi NOT pada nilai keanggotaan $\mu_A[x]$ menjadi: $\mu_A[x]^c = 1 - \mu_A[x]$.

2.2.7 Metode *Fuzzy Inference System* (FIS)

Inferensi adalah proses penggabungan banyak aturan berdasarkan data yang tersedia. Komponen yang melakukan inferensi dalam sistem pakar disebut mesin inferensi. Dua pendekatan untuk menarik kesimpulan pada *IF-THEN rule* (aturan jika-maka) adalah *forward chaining* dan *backward chaining* (Turban dkk, 2005:726).

2.2.7.1 *Forward chaining*

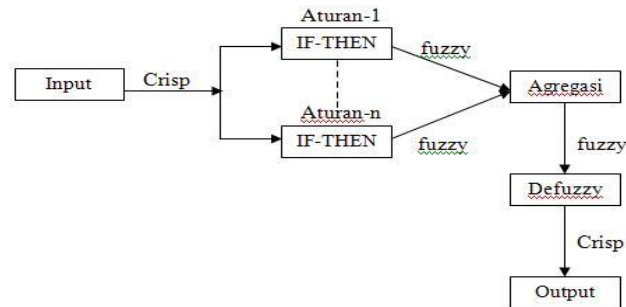
Forward chaining mencari bagian JIKA terlebih dahulu. Setelah semua kondisi dipenuhi, aturan dipilih untuk mendapatkan kesimpulan. Jika kesimpulan yang diambil dari keadaan pertama, bukan dari keadaan yang terakhir, maka ia akan digunakan sebagai fakta untuk disesuaikan dengan kondisi JIKA aturan yang lain untuk mendapatkan kesimpulan yang lebih baik. Proses ini berlanjut hingga dicapai kesimpulan akhir.

2.2.7.2 *Backward chaining*

Backward chaining adalah kebalikan dari *forward chaining*. Pendekatan ini dimulai dari kesimpulan dan hipotesis bahwa kesimpulan adalah benar. Mesin inferensi kemudian mengidentifikasi kondisi JIKA yang diperlukan untuk membuat kesimpulan benar dan mencari fakta untuk menguji apakah kondisi JIKA adalah benar. Jika semua kondisi JIKA adalah benar, maka aturan dipilih dan kesimpulan dicapai. Jika beberapa kondisi salah, maka aturan dibuang dan aturan berikutnya digunakan sebagai hipotesis kedua. Jika tidak ada fakta yang membuktikan bahwa semua kondisi JIKA adalah benar atau salah, maka mesin inferensi terus mencari aturan yang kesimpulannya sesuai dengan kondisi JIKA yang tidak diputuskan untuk bergerak satu langkah ke depan memeriksa kondisi tersebut. Proses ini berlanjut hingga suatu set aturan didapat untuk mencapai kesimpulan atau untuk membuktikan tidak dapat mencapai kesimpulan.

Menurut Sri Kusumadewi dan Sri Hartati (2006:34) sistem inferensi *fuzzy* merupakan suatu kerangka komputasi yang didasarkan

pada teori himpunan *fuzzy*, aturan *fuzzy* yang berbentuk IF-THEN, dan penalaran *fuzzy*. Secara garis besar, diagram blok proses inferensi *fuzzy* terlihat pada Gambar 2.5.



Gambar 2.5 Diagram Blok Sistem Inferensi *Fuzzy* (Sri Kusumadewi dan SriHartati, 2006: 34)

Sistem inferensi *fuzzy* menerima input *crisp*. Input ini kemudian dikirim ke basis pengetahuan yang berisi n aturan *fuzzy* dalam bentuk IF-THEN. *Firestrength* (nilai keanggotaan anteseden atau α) akan dicari pada setiap aturan. Apabila aturan lebih dari satu, maka akan dilakukan agregasi semua aturan. Selanjutnya pada hasil agregasi akan dilakukan *defuzzy* untuk mendapatkan nilai *crisp* sebagai *output* sistem.

2.2.8. Fuzzy Tsukamoto

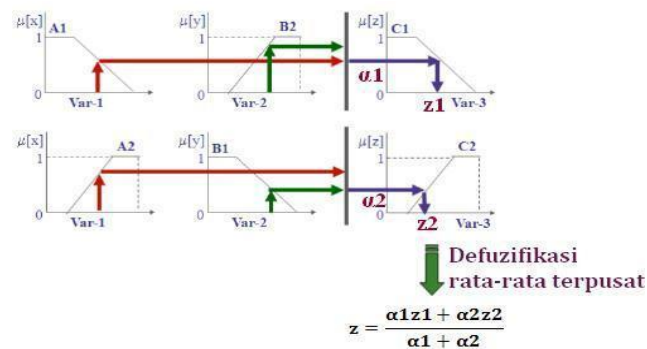
Dalam membangun sebuah sistem fuzzy dikenal beberapa metode penalaran, antara lain : metode tsukamoto, metode mamdani dan metode sugeno. Pada metode *Tsukamoto*, implikasi setiap aturan berbentuk implikasi “Sebab-Akibat”/Implikasi “*Input-Output*” dimana antara anteseden dan konsekuen harus ada hubungannya. Setiap aturan direpresentasikan menggunakan himpunan-himpunan *fuzzy*, dengan fungsi keanggotaan yang monoton. Kemudian untuk menentukan hasil tegas (*Crisp Solution*) digunakan rumus penegasan (defuzifikasi) yang disebut “Metode rata-rata terpusat” atau “Metode defuzifikasi rata-rata terpusat (*Center Average Deffuzzyfier*)” (Setiadji, 2009: 200). Untuk lebih memahami metode *Tsukamoto*, perhatikan Contoh 2.1.

Contoh 2.1:

Misalkan ada 2 variabel input, Var-1 (x) dan Var-2(x), serta variabel output, Var-3(z), dimana Var-1 terbagi atas 2 himpunan yaitu A1 dan A2. Var-2 terbagi atas 2 himpunan B1 dan B2, Var-3 juga terbagi atas 2 himpunan yaitu C1 dan C2 (C1 dan C2 harus monoton). Ada 2 aturan yang digunakan, yaitu:

[R1] IF (x is A1) and (y is B2) THEN (z is C1) [R2] IF (x is A2) and (y is B1) THEN (z is C2)

Pertama-tama dicari fungsi keanggotaan dari masing-masing himpunan *fuzzy* dari setiap aturan, yaitu himpunan A1, B2 dan C1 dari aturan *fuzzy* [R1], dan himpunan A2, B1 dan C2 dari aturan *fuzzy* [R2]. Aturan *fuzzy* R1 dan R2 dapat direpresentasikan dalam Gambar 2.6 untuk mendapatkan suatu nilai crisp Z.



Gambar 2.6 Inferensi dengan menggunakan

Metode *Tsukamoto* (SriKusumadewi dan Hari Purnomo, 2004:34).

Karena pada metode *Tsukamoto* operasi himpunan yang digunakan adalah konjungsi (*AND*), maka nilai keanggotaan anteseden dari aturan *fuzzy* [R1] adalah irisan dari nilai keanggotaan A1 dari Var-1 dengan nilai keanggotaan B2 dari Var-2. Menurut teori operasi himpunan pada persamaan 2.8, maka nilai keanggotaan anteseden dari operasi konjungsi (*And*) dari aturan *fuzzy* [R1] adalah nilai minimum antara nilai keanggotaan A1 dari Var-1 dan nilai keanggotaan B2 dari Var-2. Demikian pula nilai keanggotaan anteseden dari aturan *fuzzy* [R2] adalah nilai minimum antara nilai keanggotaan A2 dari Var-1 dengan nilai keanggotaan B1 dari Var-2. Selanjutnya, nilai keanggotaan anteseden dari aturan *fuzzy* [R1]

dan [R2] masing-masing disebut dengan α_1 dan α_2 . Nilai α_1 dan α_2 kemudian disubstitusikan pada fungsi keanggotaan himpunan C1 dan C2 sesuai aturan *fuzzy*

[R1] dan [R2] untuk memperoleh nilai z_1 dan z_2 , yaitu nilai z (nilai perkiraan produksi) untuk aturan *fuzzy* [R1] dan [R2]. Untuk memperoleh nilai output *crisp*/nilai tegas Z , dicari dengan cara mengubah input (berupa himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*) menjadi suatu bilangan pada domain himpunan *fuzzy* tersebut. Cara ini disebut dengan metode defuzifikasi (penegasan). Metode defuzifikasi yang digunakan dalam metode *Tsukamoto* adalah metode defuzifikasi rata-rata terpusat (*Center Average Defuzzifier*) yang dirumuskan pada persamaan 2.10.

$$Z = \frac{\sum_{i=1}^n \alpha_i z_i}{\sum_{i=1}^n \alpha_i} \text{ (Defuzifikasi rata-rata terpusat) } \dots\dots\dots(2.10)$$

2.3 Arum Clean Laundry

Arum clean laundry pertama kali didirikan di kota Gresik pada tahun 2010. Berawal dari usaha rumahan sederhana sampai menjadi seperti sekarang. Pelayanan yang diberikan oleh laundry ini yaitu pelayanan antar jemput cucian sehingga konsumen tidak perlu repot untuk mengantar cucian ke gerai laundry. Jenis jasa cucian yang diberikan yaitu cuci kering, cuci setrika dan pelayanan setrika cucian dimana biaya jasa dihitung berdasarkan berat cucian. Berbekal kepercayaan dari para konsumen usaha laundry ini tetap eksis sampai 5 tahun atau sampai sekarang. Penggunaan mesin di laundry tersebut masih belum menggunakan otomatis tapi tingkat kebersihannya bisa disejajarkan dengan para pelaku usaha laundry yang modern. Karena ada faktor proses pra pencucian yang manual sehingga untuk proses melihat kotoran bisa lebih teliti lagi. Faktor ini merupakan inovasi dari pemilik laundry untuk tetap bisa menjaga kualitas cucian yang baik.

2.4. Penelitian Sebelumnya

Pada tahun 2012, Muh. Hasan Fikri dari Jurusan Teknik Informatika, Universitas Muhammadiyah Gresik, menulis sebuah skripsi yang berjudul tentang “Sistem Pendukung Keputusan Penentuan Takaran Detergen di Mandiri Laundry Dengan Menggunakan Metode Fuzzy Tsukamoto”. Pada penelitian ini menjelaskan perhitungan takaran deterjen di Mandiri Laundry masih menggunakan sistem manual dengan perhitungan berdasarkan berat cucian. Penelitian ini menggunakan 3 variabel input untuk perhitungan, yaitu berat kain, warna kain dan tingkat kotor. Dengan hasil pengujian sistem yang dilakukan sebanyak 25 kali, rata-rata takaran detergen yang bisa dihemat sebesar 15,54 gram. Tingkat kevalidan sistem mencapai 88% dengan asumsi dari 25 kali pengujian terdapat 3 data yang hasilnya lebih besar dari takaran manual.

Pada tahun 2013, Lincahya dari Jurusan Teknik Informatika, Universitas Pembangunan Nasional “Veteran”, melakukan penelitian tentang “Sistem Pendukung Keputusan Untuk Menentukan Jumlah takaran Detergen Pada Laundry Menggunakan Metode Fuzzy Tsukamoto, Studi Kasus “Mister Londree””. Pada penelitian ini menjelaskan kecenderungan orang memilih layanan cuci kering dibandingkan mencuci sendiri membuat usaha laundry berkembang pesat saat ini. Dari proses pencucian komponen yang sangat diperlukan adalah detergen. Bagi pihak laundry, untuk pemakaian deterjen dapat diminimalisir tanpa mengurangi tingkat kebersihan cucian setelah dicuci. Untuk meminimalisir jumlah detergen harus menggunakan perhitungan yangn tepat, bukan hanya perkiraan saja. Penelitian ini menggunakan menggunakan 3 variabel yaitu variabel bobot, warna dan tingkat ketebalan kain. Rule yang digunakan dalam sistem ini ada 8 rule untuk menentukan aturan penggunaan detergen sesuai aturan pada laundry. Hasil uji coba yang dilakukan, user dapat melakukan proses penakaran detergen secara otomatis dari sistem dan dapat mengetahui seberapa banyak detergen yang telah digunakan. Diharapkan sistem yang dibuat dapat

membantu pihak laundry dalam memnetukan takaran detergen dan membantu perhitungan penggunaan detergen dengan lebih akurat.

Pada tahun 2013, Siti Nur Hidayati dari Jurusan Teknik Informatika, Universitas Pembangunan Nasional “Veteran”, melakukan penelitian tentang “Sistem Pendukung Keputusan Untuk Menentukan Jumlah takaran Detergen Pada Laundry Menggunakan Metode Fuzzy Mamdani, Studi Kasus Laundry “Fresh” Di Rungkut Alang-Alang Surabaya”. Pada penelitian ini menjelaskan kecenderungan orang memilih layanan cuci kering dibandingkan mencuci sendiri membuat usaha laundry berkembang pesat saat ini. Dari proses pencucian komponen yang sangat diperlukan adalah detergen. Bagi pihak laundry, untuk pemakaian deterjen dapat diminimalisir tanpa mengurangi tingkat kebersihan cucian setelah dicuci. Untuk meminimalisir jumlah detergen harus menggunakan perhitungan yangn tepat, bukan hanya perkiraan saja. Penelitian ini menggunakan menggunakan 4 variabel yaitu variabel bobot, warna, tingkat ketebalan kain dan tingkat kekotoran pakaian. Rule yang digunakan dalam sistem ini ada 16 rule untuk menentukan aturan penggunaan detergen sesuai aturan pada laundry. Hasil uji coba yang dilakukan, user dapat melakukan proses penakaran detergen secara otomatis dari sistem dan dapat mengetahui seberapa banyak detergen yang telah digunakan.