

## BAB II

### MODEL KOMPUTASI *FINITE STATE MACHINE*

Pada Bab II akan dibahas teori dasar matematika yang digunakan dalam pemodelan sistem kontrol elevator ini, yaitu mengenai himpunan, relasi, fungsi, teori graf yang akan digunakan untuk membentuk FSM. Adapun definisi-definisi yang terdapat pada bab ini merujuk [5].

#### 2.1 HIMPUNAN

##### Definisi 2.1

Himpunan adalah koleksi atau kumpulan objek yang tidak berurutan. Objek di dalam himpunan disebut juga elemen atau anggota dari suatu himpunan. Himpunan dikatakan memuat anggotanya.

Contoh himpunan adalah himpunan huruf vokal dalam Bahasa Indonesia yaitu a, i, u, e, dan o. Dalam himpunan ini anggota-anggotanya merupakan huruf vokal dalam bahasa Indonesia.

Dalam hal ini himpunan dinotasikan dengan huruf kapital yang dicetak miring dan semua anggota himpunan tersebut dinyatakan di dalam tanda kurung kurawal. Sebagai contoh  $V$  adalah himpunan huruf vokal dalam bahasa Indonesia, maka himpunan tersebut dinotasikan dengan

$V = \{a, i, u, e, o\}$ . Selanjutnya akan diberikan notasi suatu objek merupakan anggota dari suatu himpunan. Notasi yang digunakan adalah  $x \in V$ , dengan menggunakan contoh sebelumnya,  $a$  anggota di  $V$ , maka dapat ditulis  $a \in V$ .

Anggota dari suatu himpunan dapat merupakan suatu himpunan juga, contohnya adalah himpunan  $A$  yang anggotanya merupakan himpunan  $X = \{1, 2, 3, 4, 5\}$  dan himpunan  $Y = \{a, b, c, d, e\}$ . Himpunan  $A$  dapat ditulis sebagai berikut  $A = \{X, Y\} = \{\{1, 2, 3, 4, 5\}, \{a, b, c, d, e\}\}$ .

Terdapat himpunan yang tidak memiliki anggota. Himpunan tersebut dinamakan himpunan kosong yang dinotasikan dengan  $\emptyset$  atau  $\{\}$ .

### **Definisi 2.2**

Dua himpunan dikatakan sama jika keduanya memiliki anggota yang sama.

Sebagai contoh terdapat dua himpunan  $\{1, 2, 3\}$  dan  $\{3, 1, 2\}$ , dua himpunan adalah sama karena keduanya memiliki anggota yang sama yaitu 1, 2 dan 3. Begitu juga halnya dengan  $\{1, 2, 3\}$  dan  $\{2, 3, 1\}$  atau  $\{1, 2, 3\}$  dan  $\{3, 2, 1\}$ . Kedua hal tersebut juga merupakan dua himpunan yang sama.

Dalam suatu himpunan mungkin saja anggotanya ditulis beberapa kali. Sebagai contoh terdapat himpunan  $\{1, 1, 2, 2, 2, 3\}$  namun himpunan ini sama saja dengan himpunan  $\{1, 2, 3\}$  karena keduanya memiliki anggota yang sama.

### **Definisi 2.3**

Himpunan  $A$  dikatakan subhimpunan dari  $B$  jika setiap anggota di  $A$  juga merupakan anggota di  $B$ .

Subhimpunan dinotasikan dengan  $A \subseteq B$ ,  $A$  subhimpunan dari  $B$  dapat ditulis dengan  $A \subseteq B$ . Sebagai contoh terdapat himpunan  $X = \{a, e\}$  dan himpunan  $V = \{a, i, u, e, o\}$ , karena setiap anggota di  $X$  merupakan semua anggota di  $V$  ( $a$  dan  $e$  merupakan anggota di  $V$ ) maka  $X$  subhimpunan dari  $V$ , oleh karena itu dapat ditulis  $X \subseteq V$ .

#### **Definisi 2.4**

Misalkan  $S$  himpunan. Jika terdapat  $n$  anggota yang berbeda dengan  $n$  adalah bilangan bulat non negatif (lebih besar sama dengan nol), maka  $S$  merupakan himpunan berhingga dan  $n$  merupakan kardinalitas dari  $S$ . Kardinalitas dari  $S$  dinotasikan dengan  $|S|$ .

Sebagai contoh himpunan huruf vokal dalam bahasa Indonesia  $V = \{a, i, u, e, o\}$  memiliki lima anggota yang berbeda, berarti  $V$  merupakan himpunan berhingga dan  $|V| = 5$ .

## **2.2 RELASI**

Sebelum diberikan definisi relasi, akan diperkenalkan terlebih dahulu definisi-definisi yang akan mengacu pada definisi relasi.

**Definisi 2.5**

*Ordered n-tuple*  $(a_1, a_2, \dots, a_n)$  adalah koleksi berurutan yang memiliki  $a_1$  sebagai anggota ke-1,  $a_2$  sebagai anggota ke-2, ..., dan  $a_n$  sebagai anggota ke- $n$ .

**Definisi 2.6**

Dua *Ordered n-tuple* dikatakan sama jika dan hanya jika setiap anggota dari kedua *Ordered n-tuple* yang bersesuaian sama. Dengan kata lain,  $(a_1, a_2, \dots, a_n) = (b_1, b_2, \dots, b_n)$  jika dan hanya jika  $a_i = b_i$  untuk  $i = 1, 2, \dots, n$ .

Himpunan dan *ordered n-tuple* memiliki persamaan bahwa keduanya merupakan koleksi dari suatu objek, yang menjadi perbedaan adalah dalam keterurutan anggotanya. Sebagai contoh terdapat dua himpunan  $\{a, b\}$  dan  $\{b, a\}$  dan dua *ordered n-tuple*  $(a, b)$  dan  $(b, a)$  maka berdasarkan Definisi 2.3  $\{a, b\} = \{b, a\}$  dan berdasarkan Definisi 2.6  $(a, b) \neq (b, a)$ . Dari contoh tersebut dapat dilihat bahwa keterurutan anggota berpengaruh dalam *ordered n-tuple* sedangkan di dalam himpunan tidak memperhatikan keterurutan anggotanya.

**Definisi 2.7**

Misalkan  $A$  dan  $B$  himpunan. Produk Kartesian dari  $A$  dan  $B$ , dinotasikan dengan  $A \times B$ , himpunan semua pasangan berurut (*2-tuple*)  $(a, b)$  dengan  $a \in A$  dan  $b \in B$ . Sehingga dapat ditulis sebagai berikut

$$A \times B = \{(a, b) \mid a \in A \text{ dan } b \in B\}.$$

Sebagai contoh terdapat dua himpunan  $A = \{1, 2\}$  dan  $B = \{a, b, c\}$ ,

maka

$$A \times B = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\}$$

Sekarang akan didefinisikan produk kartesian untuk banyak himpunan

### Definisi 2.8

Produk kartesian dari himpunan  $A_1, A_2, \dots, A_n$ , dinotasikan dengan

$A_1 \times A_2 \times \dots \times A_n$  adalah himpunan dari ordered  $n$ -tuple  $(a_1, a_2, \dots, a_n)$ , dengan  $a_i$  anggota di  $A_i$  untuk  $i = 1, 2, \dots, n$ . Dengan kata lain

$$A_1 \times A_2 \times \dots \times A_n = \{(a_1, a_2, \dots, a_n) \mid a_i \in A_i \text{ untuk } i = 1, 2, \dots, n\}$$

### Definisi 2.9

Subhimpunan dari produk kartesian dinamakan relasi.

#### Contoh 2.1

Misalnya  $A = \{1, 2, 3\}$  dan  $B = \{a, b, c\}$  maka

$$A \times B = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c), (3, a), (3, b), (3, c)\}$$

Contoh relasi dari produk kartesian tersebut adalah sebagai berikut:

$$\{(1, a), (1, b), (1, c)\}, \{(1, b), (1, c), (2, a)\} \text{ dan } \{(3, b), (3, c)\}.$$

Himpunan-himpunan tersebut masing-masing merupakan relasi karena

$$\{(1, a), (1, b), (1, c)\} \subseteq A \times B, \{(1, b), (1, c), (2, a)\} \subseteq A \times B,$$

$$\text{dan } \{(3, b), (3, c)\} \subseteq A \times B.$$

Relasi dapat dikatakan juga sebagai hubungan antara dua himpunan, sebagai contoh himpunan A yang merupakan himpunan mahasiswa Departemen Matematika Universitas Indonesia dan himpunan B yang merupakan himpunan mata kuliah di Departemen Matematika Universitas

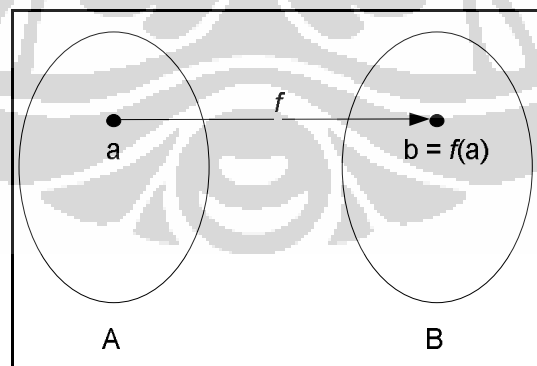
Indonesia. Salah satu contoh relasi dari  $A$  ke  $B$  adalah mahasiswa yang mengambil mata kuliah Matematika Diskrit I.

## 2.3 FUNGSI

### Definisi 2.10

Misalkan  $A$  dan  $B$  himpunan. Fungsi  $f$  dari  $A$  ke  $B$  yang dinotasikan dengan  $f : A \rightarrow B$  adalah relasi dari  $A$  ke  $B$  dengan setiap anggota di  $A$  dihubungkan dengan tepat satu anggota di  $B$ . Jika  $b$  adalah anggota di  $B$  yang unik dihubungkan oleh  $f$  dengan  $a$  anggota di  $A$  maka dapat dinotasikan dengan  $f(a) = b$ .

Gambar 2.1 merupakan cara penyajian fungsi secara diagram



Gambar 2.1 Penyajian fungsi secara diagram

Fungsi merupakan relasi yang mempunyai dua syarat yang harus dipenuhi yaitu:

1. Setiap anggota di  $A$  akan mempunyai pasangan di  $B$ .

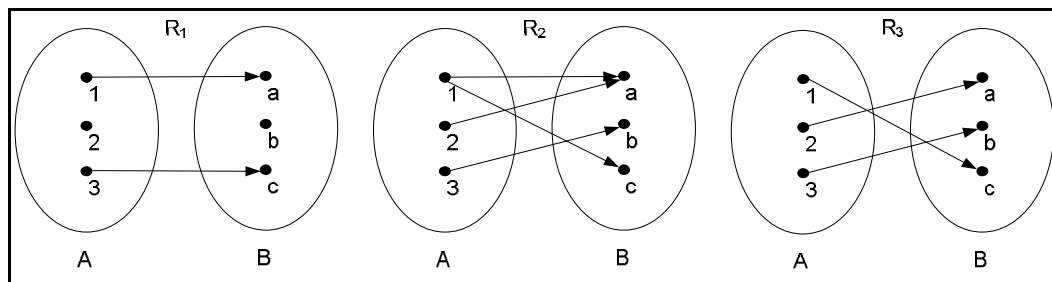
Dengan kata lain, untuk setiap  $a \in A$  terdapat  $b \in B$  sehingga  $(a, b)$ .

2. Setiap anggota di  $A$  tidak akan mempunyai dua pasangan atau lebih di  $B$ . Dengan kata lain, misalkan  $a \in A$  dan  $b_1, b_2 \in B$  jika  $(a, b_1)$  dan  $(a, b_2)$  maka  $b_1 = b_2$ .

### Contoh 2.2

Dengan menggunakan himpunan  $A$ ,  $B$  dan  $A \times B$  pada Contoh 2.1 diberikan tiga contoh relasi:

- a.  $R_1 = \{(1, a), (3, c)\}$  bukan merupakan fungsi, karena terdapat anggota di  $A$  yang tidak mendapat pasangan yaitu 2, dengan ini melanggar syarat pertama.
- b.  $R_2 = \{(1, a), (1, c), (2, a), (3, b)\}$  bukan merupakan fungsi, karena terdapat anggota di  $A$  yang mempunyai dua pasangan atau lebih yaitu 1 dengan  $a$  dan  $c$ , dengan ini melanggar syarat kedua.
- c.  $R_3 = \{(1, c), (2, a), (3, b)\}$  ini merupakan fungsi karena relasi ini memenuhi kedua syarat fungsi, yaitu setiap anggota di  $A$  mendapatkan pasangan di  $B$  dan tidak ada anggota  $A$  yang mempunyai dua pasangan atau lebih.



Gambar 2.2 Relasi  $R_1$ ,  $R_2$  dan  $R_3$

Berikut ini akan diberikan mengenai dasar-dasar pada teori graf. Dengan graf dapat digambarkan model sistem kontrol elevator secara diagram.

## 2.4 GRAF

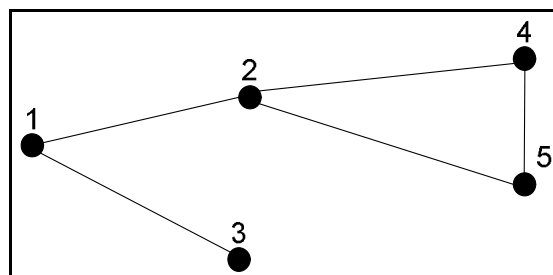
Dasar teori graf yang digunakan yaitu graf sederhana dan multigraf berarah yang akan dijelaskan sebagai berikut:

### Definisi 2.11

Graf sederhana  $G = (V, E)$  terdiri dari himpunan  $V$  dan himpunan  $E$ .  $V$  merupakan himpunan tidak kosong yang anggotanya dinamakan verteks.  $E$  merupakan himpunan pasangan yang tak berurut dari anggota yang berbeda di  $V$ , anggota di  $E$  dinamakan *edge*.

Sebagai contoh terdapat graf  $G = (V, E)$  dengan  $V = \{1, 2, 3, 4, 5\}$  dan  $E = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{4, 5\}, \{2, 5\}\}$ . 1, 2, 3, 4, dan 5 merupakan verteks sedangkan  $\{1, 2\}, \{1, 3\}, \{2, 4\}, \{4, 5\}, \{2, 5\}$  merupakan *edge*.

Graf  $G = (V, E)$  tersebut dapat digambarkan sebagai berikut





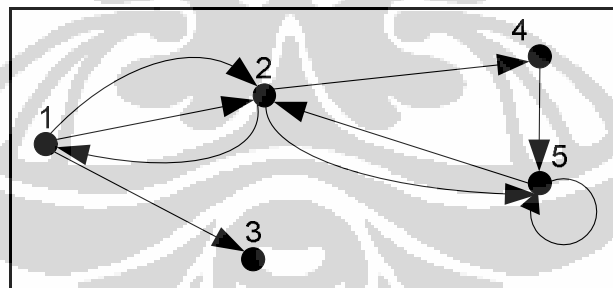
Gambar 2.3 Penyajian secara diagram graf sederhana  $G = (V, E)$

Berikutnya akan diberikan definisi multigraf berarah, yaitu graf yang dapat memuat dua atau lebih *edge* berarah pada sepasang verteks dan juga dapat memuat *loop*. *Loop* adalah *edge* yang pasangan verteksnya merupakan verteks yang sama.

**Definisi 2.12**

Multigraf berarah  $G = (V, E)$  terdiri dari himpunan verteks  $V$  dan himpunan *edge*  $E$  dan sebuah fungsi  $f$  dari  $E$  ke  $\{(u, v) \mid u, v \in V\}$ . *Edge*  $e_1$  dan  $e_2$  dikatakan *multiple edges* atau *parallel edges* jika  $f(e_1) = f(e_2)$ . *Edge* dikatakan *loop* jika  $f(e) = (u, u)$  untuk  $u \in V$ .

Sebagai contoh diberikan graf sebagai berikut



Gambar 2.4 Penyajian secara diagram multigraf berarah  $G = (V, E)$

Graf pada Gambar 2.4 dapat dinyatakan dengan  $G = (V, E)$  dengan  $V = \{1, 2, 3, 4, 5\}$ ,  $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}$  dan sebuah fungsi  $f$  dengan  $f(e_1) = (1, 2)$ ,  $f(e_2) = (1, 2)$ ,  $f(e_3) = (2, 1)$ ,  $f(e_4) = (1, 3)$ ,  $f(e_5) = (2, 4)$ ,  $f(e_6) = (2, 5)$ ,  $f(e_7) = (5, 2)$ ,  $f(e_8) = (4, 5)$ ,  $f(e_9) = (5, 5)$ . *Edge*  $e_1$ ,  $e_2$  dan  $e_3$  dinamakan *multiple edges* karena  $f(e_1) = f(e_2) = (1, 2)$ , dan  $e_9$  dinamakan *loop* karena  $f(e_9) = (5, 5)$ .

Berikutnya akan dibahas mengenai jenis model yang akan digunakan dalam skripsi ini.

## **2.5 FINITE STATE MACHINE**

FSM adalah model yang memperlihatkan perilaku dari sistem dengan keadaan-keadaan. Model ini memberikan keadaan-keadaan yang mungkin dijalani oleh sistem, apa yang harus dilakukan sistem pada keadaan-keadaan tersebut, serta jika diberikan suatu input maka sistem akan melakukan atau menghasilkan sesuatu.

FSM terdiri dari dua jenis yaitu FSM beroutput dan FSM tidak beroutput. FSM tidak beroutput (biasanya disebut automata hingga) digunakan untuk pengenalan bahasa dalam komputer, dengan input-input yang dimasukkan akan diperoleh apakah input tersebut dikenal oleh bahasa komputer atau tidak. Salah satu penggunaan FSM tidak beroutput adalah program *compiler*, yaitu program untuk memeriksa apakah perintah yang diberikan pengguna benar atau salah. Sementara untuk FSM beroutput digunakan untuk merancang sebuah sistem atau mesin. FSM yang digunakan dalam skripsi ini adalah FSM yang beroutput, dan untuk selanjutnya FSM yang beroutput akan dituliskan FSM saja.

Berikut ini adalah definisi dari FSM:

**Definisi 2.13**

FSM  $M = (S, I, O, f, g, s_0)$  terdiri dari:

1. Himpunan berhingga keadaan  $S$ .
2. Himpunan berhingga input  $I$ .
3. Himpunan berhingga output  $O$ .
4. Fungsi transisi  $f$  yang memasangkan pasangan keadaan dan input pada keadaan. Dinotasikan  $f : S \times I \rightarrow S$ .
5. Fungsi output  $g$  yang memasangkan keadaan pada output. Dinotasikan  $g : S \rightarrow O$ .
6.  $s_0$  merupakan keadaan awal,  $s_0 \in S$ .

Fungsi transisi merupakan fungsi yang menunjukkan jika sistem pada suatu keadaan dan diberikan suatu input maka sistem tersebut akan bertransisi menjadi keadaan yang baru. Sedangkan fungsi output menunjukkan sistem melakukan suatu aksi pada suatu keadaan.

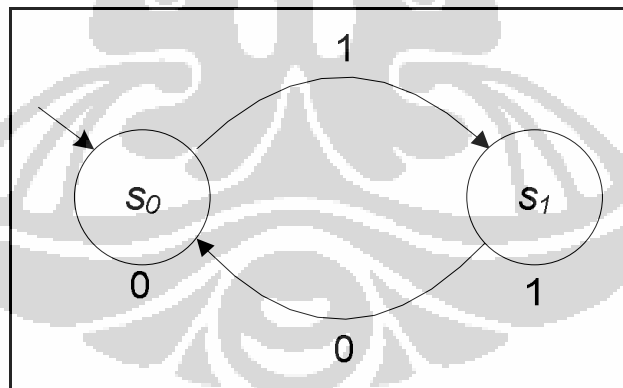
**Contoh 2.2**

Sebagai contoh diberikan FSM  $M$  dengan  $S = \{s_0, s_1\}$ ,  $I = \{0, 1\}$ ,  $O = \{0, 1\}$  dengan fungsi transisi  $f(s_0, 1) = s_1$  dan  $f(s_1, 0) = s_0$  dan fungsi output  $g(s_0) = 0$  dan  $g(s_1) = 1$ . Contoh 2.2 merupakan FSM untuk mesin saklar lampu, keadaannya adalah lampu mati ( $s_0$ ) dan lampu hidup ( $s_1$ ). Keadaan awalnya adalah lampu mati. Saat lampu dalam keadaan mati kemudian ditekan saklar lampu untuk hidup (input 1) maka terjadi transisi keadaan

menjadi lampu hidup ( $f(s_0,1) = s_1$ ) dan menyalakan lampu tersebut ( $g(s_1) = 1$ ). Saat lampu dalam keadaan hidup kemudian ditekan saklar lampu untuk mati (input 0) maka terjadi transisi keadaan menjadi lampu mati ( $f(s_1,0) = s_0$ ) dan mematikan lampu yang sedang hidup ( $g(s_0) = 0$ ).

FSM pada contoh 2.2 dapat digambarkan dengan diagram transisi. Diagram tersebut merupakan multigraf berarah yang vertexnya merupakan keadaan sedangkan *edgenya* merupakan representasi dari fungsi transisi, dan label dari *edgenya* merupakan input yang menyebabkan transisi. Sedangkan outputnya adalah tulisan di bawah keadaan. Tanda panah yang menunjuk ke  $s_0$  merupakan  $s_0$  adalah keadaan awal.

Diagram transisi dari FSM  $M$  pada contoh 2.2 adalah sebagai berikut:



Gambar 2.5 Diagram transisi keadaan untuk mesin M