

PERTEMUAN 11

HUBUNGAN ER DENGAN OTOMATA BERHINGGA DAN ER KE FINITE STATE AUTOMATA

A. TUJUAN PEMBELAJARAN

Pembahasan pada pertemuan ini secara keseluruhan mengenai hubungan ER dengan otomata berhingga dan ER ke finite state automata. Maka setelah mengikuti perkuliahan pertemuan 10, mahasiswa diharapkan mampu :

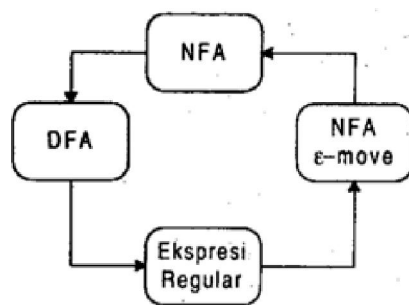
- 1) Menjabarkan hubungan ER dengan otomata
- 2) Menjabarkan ER ke finite state automata

B. URAIAN MATERI

1. Hubungan ER dengan Otomata Berhingga

Suatu ER dikatakan memiliki hubungan dengan otomata berhingga karena suatu otomata berhingga dapat dibentuk dari pola ER, ataupun sebaliknya. Marilah kita lihat berbagai contoh hubungan ER dengan otomata berhingga berikut ini :

Gambaran hubungan ER dengan mesin otomata berhingga



Gambar 6.1 Hubungan ER dengan Finite State Automata

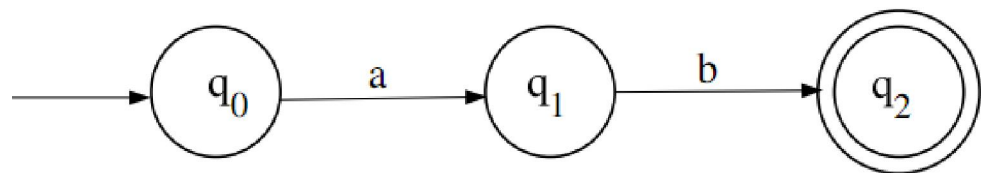
Berikut kita lihat pada contoh penerapannya

1) NFA- ϵ move untuk ER : ab



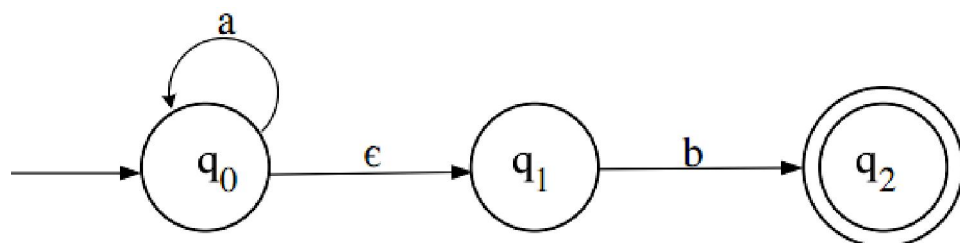
Perhatikan pada input string dari NFA tersebut hanya menerima string ab saja, selain itu tidak diterima. Dengan NFA - ϵ move (Ingat pembahasan pada bab sebelumnya mengenai NFA mampu berpindah transisi walau tanpa menerima input string atau hampa yakni ϵ move atau transisi ϵ) dapat kita rancang 4 (empat) state.

2) NFA untuk ER ab



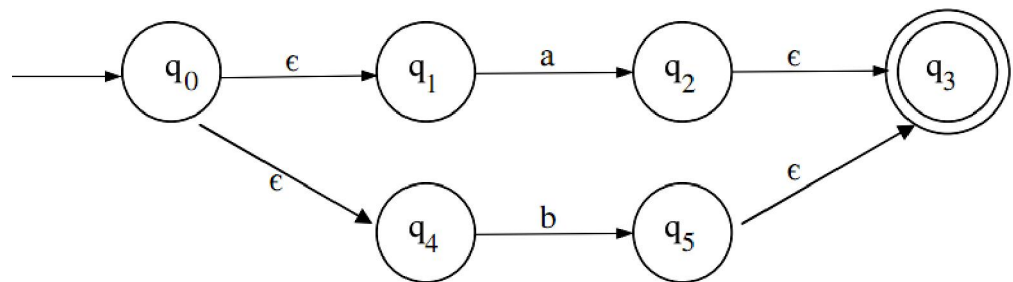
Berbeda dengan NFA pada contoh no 1, pada rancangan ini tidak menggunakan NFA ϵ - move. Sehingga jika diperhatikan akan lebih sederhana dibandingkan dengan contoh di atas karena hanya terdapat 3 (tiga) state q_0 , q_1 , dan q_2 . Hal ini artinya saat ingin merancang mesin otomata berhingga non deterministik, Kita dapat memberikan state sesuai dengan kebutuhan rancangan tanpa mengganggu bentuk bahasa yang dibutuhkan.

3) NFA- ϵ move untuk ER : a^*b



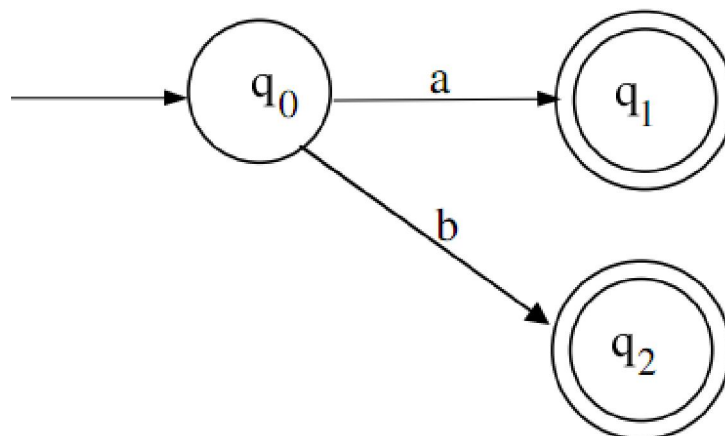
Input string yang mungkin dikenali oleh mesin NFA tersebut adalah dapat dengan hanya karakter b saja, ab, aab, aaaab, dst untuk selalu diakhiri dengan satu karakter b dan dapat muncul sekali atau lebih dengan banyak berhingga karakter a di awal.

4) NFA - ϵ move untuk ER : $a \cup b$



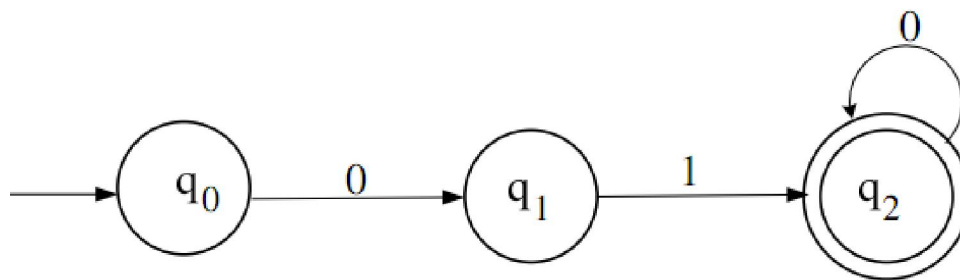
Dengan menggunakan 6 (enam) state dapat kita rancang mesin NFA - ϵ move untuk menghasilkan mesin yang mampu menerima string a, b saja.

5) NFA dengan ER $a \cup b$



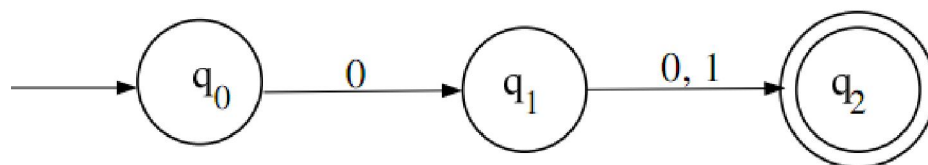
NFA ini adalah tampilan sederhana dari mesin otomata berhingga yang mampu menerima string a, b saja. Jika dibandingkan dengan NFA pada contoh no 4 (empat) yang memiliki 6 (enam) state, NFA ini hanya ada 3 (tiga) state saja

6) NFA dengan ER 010*



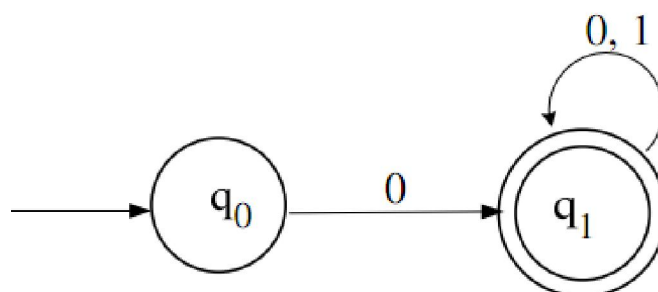
Input string yang mampu dibentuk adalah 01, 010, 0100, 01000, dst (string 0 di akhir boleh ada atau lebih banyak berhingga). Mesin NFA ini menampilkan inputan bilangan biner pada 3 (tiga) state saja.

7) NFA dengan ER $0(1 \cup 0)$



Pada mesin NFA ini dapat menerima input string 00 dan 01 saja. Sesuai ER, rancangan mesin NFA ini hanya membutuhkan tiga state.

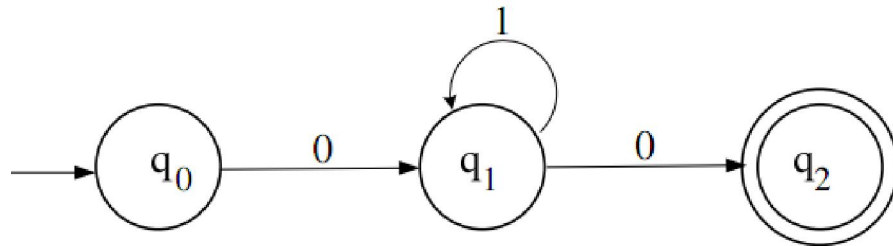
8) NFA dengan ER $0(0 \cup 1)^*$



Input string yang diterima adalah 0, 00, 01, 000, 0000, 01111, 01001, dst. Pada NFA ini dapat kita lihat input string yang diterima lebih banyak variatif nya, yakni

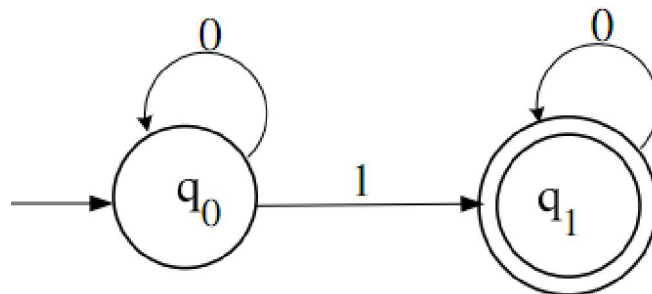
setiap 0 atau 1 di akhir boleh tidak dimunculkan atau lebih dari banyak berhingga dengan kombinasi 0 dan 1.

9) NFA untuk ER 01^*0



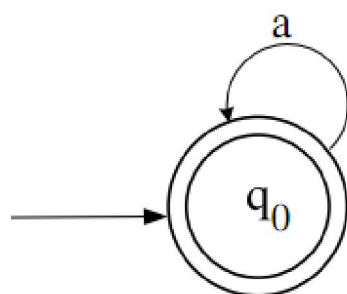
Input string yang mampu dibentuk adalah 00, 010, 01110, 0111110 dst. Perhatikan pola input string yang diterima, string 1 di tengah boleh tidak dimunculkan atau dimunculkan lebih banyak berhingga.

10) NFA dengan ER 0^*10^*



Input string yang dapat dibentuk adalah string 1, 010, 0001000, 000100000, dst. NFA ini mampu menerima kombinasi setiap string 0 di awal dan akhir yang dimunculkan lebih banyak berhingga atau tidak dimunculkan.

11) NFA dengan ER a^*

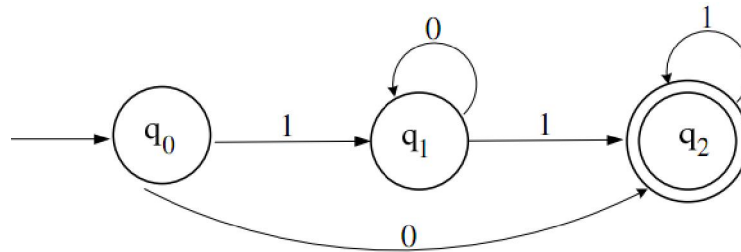


NFA ini sebagai penerima input string bahkan tanpa string pun dapat dikenali, atau a, aaa, aaaaa, aaaaa dst. Karena state awal adalah state penerima

Contoh soal latihan :

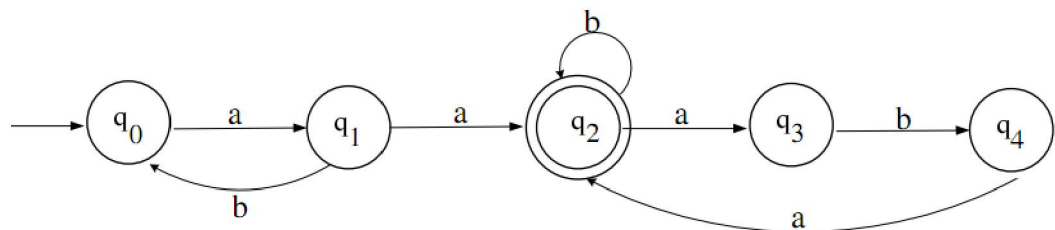
Jika diketahui bentuk mesin otomata berhingga berikut ini, deskripsikan ER nya untuk mengetahui himpunan string yang mampu dikenali NFA tersebut :

a.



Dan

b.



Penyelesaian

Untuk menjawab soal a dan b, maka kita harus menganalisa himpunan input string atau Σ dengan cermat. Dengan menggunakan notasi ER yang sesuai kondisi mesin, maka akan kita bentuk ER sebagai berikut :

- a. Himpunan input string nya adalah 0 dan 1 dengan untai string yang mampu dikenali mesin adalah berawalan 0 atau 1 dimana setiap diawali 0 maka boleh dimunculkan 1 lebih banyak berhingga atau tanpa 1. Sedangkan jika diawali 1 akan ada konkatenasi 11 dengan disisipi 0 yang bisa dimunculkan atau tidak, serta dapat dimunculkan juga akhiran 1 lebih banyak berhingga atau pun tidak dimunculkan. Dirumuskanlah ER sebagai berikut:

$$01^* \cup 10^*11^*$$

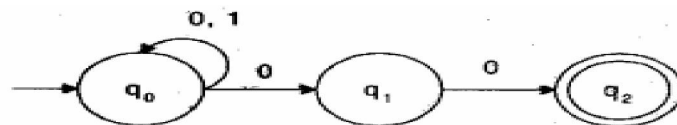
- b. Sama dengan langkah pada (a), kita tentukan awalan, tengah ataupun akhiran string yang mungkin diterima. Awalan dapat menerima string a atau aba, dengan string akhiran ab (b boleh muncul lebih banyak berhingga atau tidak muncul) atau abab (untuk b akhir boleh tidak muncul atau muncul banyak berhingga), karena kondisi state penerima ada pada posisi tengah (q2). Dirumuskanlah ER sebagai berikut :

$$a(ba)^*ab^*(abab^*)^*$$

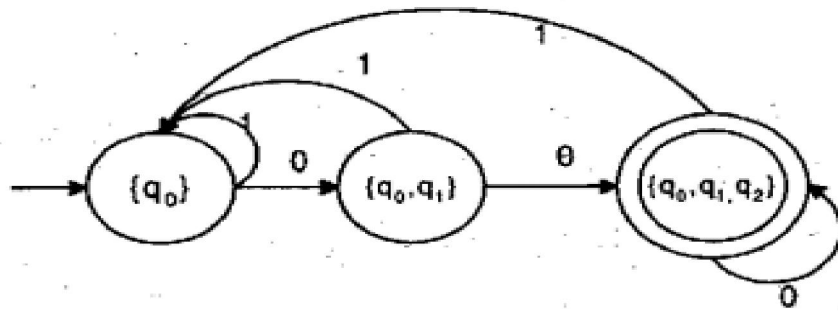
2. ER ke Finite State Automata

Contoh :

Ingin dibuat mesin Deterministic Finite Automata yang menerima bahasa yang berupa semua string yang berakhiran dengan 00. Diketahui VT = {0,1}. Pertama kita buat ekspresi regular-nya: $(0+1)^*00$ Dari ekspresi regular tersebut lebih mudah dibuat NFA-nya lebih dahulu, daripada langsung ke DFA. Mesin NFA-nya dapat di gambarkan seperti gambar berikut



Akhirnya di buat DFA yang ekivalen dengan NFA tersebut. Bisa dicek dengan untai yang harus bisa diterima oleh mesin itu, seperti: 00, 100, 000, 0100, 0000, 1100

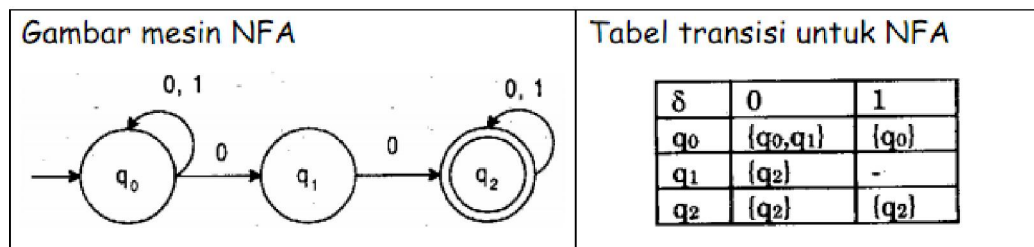


Gambar Mesin DFA

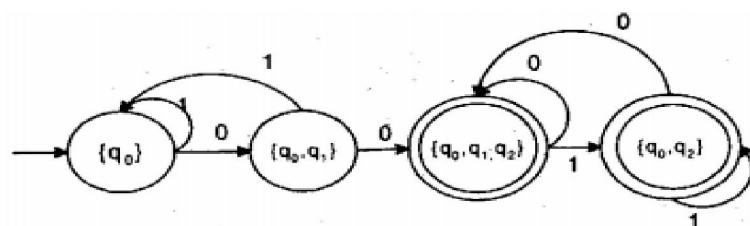
Contoh lainnya :

Ingin dibuat mesin DFA yang menerima bahasa yang berupa semua string yang memuat minimal dua nol berturutan ('00') Diketahui $S = (0, 1)$, Maka ekspresi regular-nya: $(0+1)^*00(0+1)^*$

Diperoleh hasil sebagai berikut :



Akhirnya dibuat Deterministic Finite Automata yang ekivalen dengan Non-deterministic Finite Automata tersebut. Bisa dicek dengan untai yang harus bisa diterima oleh mesin itu, seperti: 00, 100, 001, 000, 0100, 1001, 0000, 1100, 0011, 1010011, 110011

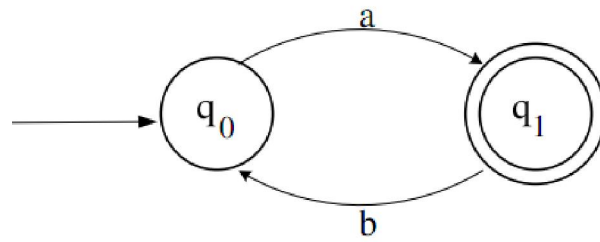


Mesin DFA

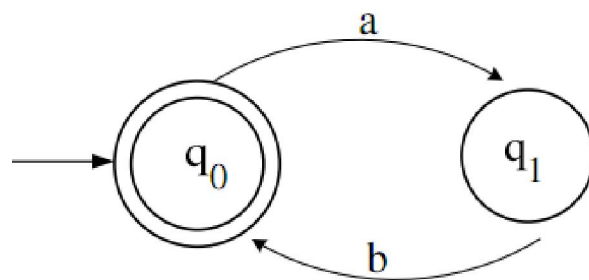
C. SOAL LATIHAN 7 DAN TUGAS

Tentukan ER untuk diagram transisi DFA sertakan input string yang mungkin diterima mesin berikut

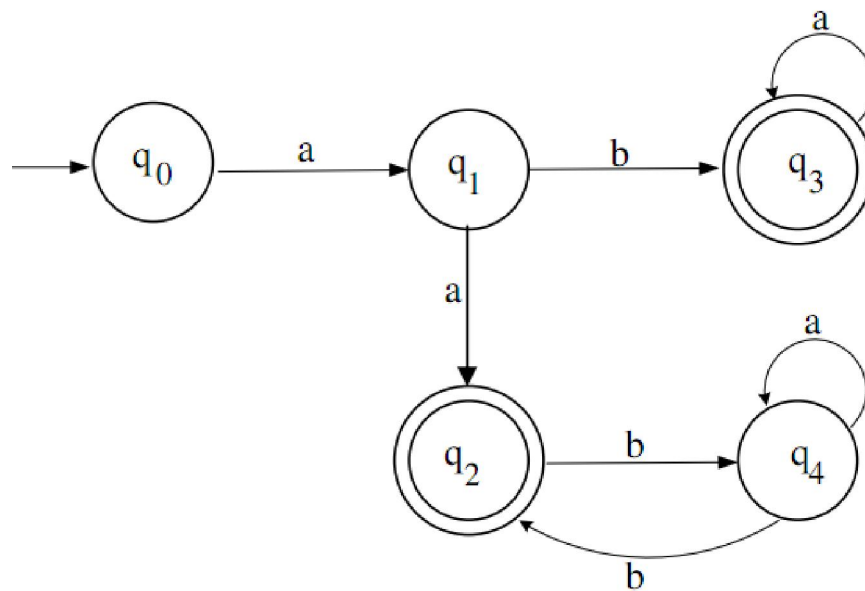
a.



2.



3.



D. DAFTAR PUSTAKA

Hopcroft, John. E., etc. 2001. Second edition. Introduction to Automata Theory, Languages, and Computation. US America : Pearson

Martin, John C. 2010. Fourth Edition. Introduction to Language and The Theory of Computation. United State America : McGraw-Hill

Modul Teori bahasa Automata. (www.ibbi.ac.id/ibbiacid/bahan/teori-bahasa-dan-otomata). Diakses pada tanggal 20 November 2015

Santosa, Kussigit. Modul Teori Bahasa Otomata. Universitas Pamulang Fakultas Teknik Program Studi Teknik Informatika. Pamulang