Finite $\omega$-Automata and Büchi Automata

Done by: Abdulla Eid
eid1@illinois.edu

Supervised by: Prof. Steven M. Lavalle

May 2, 2009
Table of Contents

§1 Introduction ................................................................. 4

§2 Finite \( \omega \)-automata ..................................................... 6
   2.1 Büchi Acceptance ....................................................... 7
   2.2 Muller Acceptance ..................................................... 10
   2.3 Rabin Acceptance .................................................... 11
   2.4 Streett Acceptance .................................................. 12

§3 Deterministic VS Non-deterministic BA ......................... 13

§4 Closure Properties .......................................................... 14
Abstract

In this project of the theory of computation course, we will study and write about the $\omega$-Automata, which is a finite state machine run on words of infinite length, the acceptance condition we have it for DFA or NFA is no longer valid, since we don't have a finite length words. In the first section, the basic notion will be defined, such as $\omega$-Languages, (a set of words of infinite length)

In section 2, the $\omega$-Automata will be introduced formally with 4 type of acceptance conditions, they will be called Büchi, Muller, Rabin and Streett acceptance condition. It will turn out that all of those will be equivalent in power and that will lead to a definition of the regular $\omega$-Automata to be any $\omega$-Language that is recognized by one of those $\omega$-Automata, one exception is the deterministic Büchi Automata which weaker than the others.

In section 3, we will compare the Deterministic Büchi $\omega$-Automata and Non-deterministic Büchi $\omega$-Automata, we will see that non-deterministic is more powerful automata than the deterministic one.

Finally, we will study some of the closure property that a regular $\omega$-languages have such as a union, intersection, concatenation of regular language and regular $\omega$-language and closeness under complementation.

I would like thank Prof.Lavalle to provide us with this opportunity to explore some extra material. It is a worth experience to practice many things at the same time: how to learn independently, write short paper, type it with $\LaTeX$ and present it for 45 minutes in a mini-conference.
\section{Introduction:}

Fix $\Sigma$ to be a finite alphabet.

\textbf{Notation:}

The set of all words of infinite length over $\Sigma$ is denoted by

$$\Sigma^\omega := \Sigma^\infty = \{a_1a_2a_3\cdots | a_i \in \Sigma\} = \{f : \mathbb{N} \to \Sigma\} = \prod_{n \in \mathbb{N}} \Sigma$$

\textbf{Remarks:}

1. Elements of $\Sigma^\omega$ is denoted by Greek letters $\alpha, \beta, \gamma, \cdots$, where each $\alpha = \alpha_0\alpha_1\alpha_2\cdots$, $\alpha_i \in \Sigma$. In this case we have $|\alpha| = \aleph_0$.

2. Any word of infinite length is called $\omega$-word.

3. Let $a \in \Sigma$, then $a^\omega := aaaa\cdots \in \Sigma^\omega$.

4. $\Sigma^* = \bigcup_{n \in \mathbb{N}} \Sigma^n$ is the set of all words of finite length, we also have that $(\Sigma^*, \circ)$ is monoid with identity $\epsilon = \text{empty word}$.

5. $\Sigma^*$ is countable set while $\Sigma^\omega$ is not countable set if $\Sigma$ has more than one element (we use the diagonalization method we used in the first class of the course), we have also $|\Sigma^*| \leq |\Sigma^\omega|$, where $|X|$ is the cardinal number of a set $X$.

6. We can define a concatenation operation $\circ$ as follows

$$\circ : \Sigma^* \times \Sigma^\omega \longrightarrow \Sigma^\omega$$

$$(w, \beta) \longmapsto w\beta$$

7. Let $A \subseteq \Sigma^*$ be a language, then $A^\omega := \{w_1w_2\cdots | w_i \in A\} = \{f : \mathbb{N} \to A\}$.

\textbf{Definition: ($\omega$-Languages)}

A subset $A \subseteq \Sigma^\omega$ is called $\omega$-language

\textbf{Examples:}

1. Let $\Sigma = a, b$. Let $A := \{\alpha \in \Sigma^\omega | \alpha \text{ has infinitely many } b\text{'s}\}$, then $A$ is $\omega$-language and $\alpha := a^*b^* \in A$, $\beta := (ab)^\omega \in A$ and $\gamma := (ba)^\omega \in A$.

2. If $\Sigma = \{a\}$, then $\Sigma^\omega$ is singleton containing only $aaa\cdots$, while $\Sigma^*$ is infinite countable set of words containing $\epsilon, a, aa, aaa, \cdots$.

3. Let $\alpha_1, \alpha_2, \cdots, \alpha_n \in \Sigma^\omega$, then $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ and $\Sigma^\omega \setminus \{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ are both $\omega$-languages.
4. If $A \subseteq \Sigma^*$ be language and $B \subseteq \Sigma^\omega$ is $\omega$-language, then

$$AB := \{ w \circ \alpha \mid w \in A, \alpha \in B \} = \{ wa \mid w \in A, \alpha \in B \}$$

is $\omega$-language.

**Notations:**

Let $a \in \Sigma, \alpha \in \Sigma^\omega$, then

- $|\alpha|_a :=$ number of occurrences of $a$ in $\alpha$.
- $\text{Occ}(\alpha) := \{ b \in \Sigma \mid \exists i \in \mathbb{N}, \alpha_i = b \} \subseteq \Sigma$ is the set of all symbols in $\alpha$.
- $\text{Inf}(\alpha) := \{ a \in \Sigma \mid \forall i \in \mathbb{N}, \exists j > i \in \mathbb{N}, \alpha_j = a \} \subseteq \Sigma$ is the set of symbols that occurring infinitely often in $\alpha$. 
§ 2 - Finite Automata

In the lecture, the notion of acceptance of a finite word by DFA/NFA was well-defined since we have there a finite computation and finite run of an automata on a finite length word.

Here we will consider a finite automata that accepts a word of infinite length and we need to define various acceptance conditions for that.

**Definition:** (Finite $\omega$-Automata)

An non-deterministic finite $\omega$-Automata $M$ is a 5-tuple $M = (Q, \Sigma, \delta, q_0, Acc)$ where:

1. $Q$ is a finite set of states
2. $\Sigma$ is finite alphabet
3. $\delta : Q \times \Sigma \rightarrow \text{Pow}(Q)$ is the transition function
   [with $\delta : Q \times \Sigma \rightarrow Q$ in case of deterministic finite $\omega$-automata]
4. $q_0 \in Q$ is the start state
5. $Acc$ is the acceptance component which can be given by various ways: as a set of states $F \subseteq Q$, as set of state sets $\mathcal{F} \subseteq \text{Pow}(Q)$ or as a set of finite pair of states $\Omega = \{(E_i, F_i) \subseteq Q \times Q \mid i = 1, 2, \cdots, n\}$.

**Definition:** (Run)

Let $M = (Q, \Sigma, \delta, q_0, Acc)$ be a nondeterministic finite $\omega$-automata, let $\alpha \in \Sigma^\omega$. A run of $M$ on $\alpha = a_1a_2a_3 \cdots \in \Sigma^\omega$ is an infinite sequence of states $r = r_0r_1r_2 \cdots \in Q^\omega$ such that

(i) $r_0 = q_0$
(ii) $r_{i+1} \in \delta(q_i, a_{i+1})$, $\forall i = 0, 1, \cdots$ — in case of nondeterministics
$r_{i+1} = \delta(q_i, a_{i+1})$ — in case of deterministic.
Let $\alpha = ba^\omega$, then a run of $M$ is $r = q_0q_0q_1^\omega$

\section*{2.1 - Büchi Acceptance:}

\textbf{Definition:} (Due to Büchi, 1960's)

A non-deterministic finite $\omega$-automata $M = (Q, \Sigma, \delta, q_0, F)$ with accept component $F \subseteq Q$ is called Büchi-Automation if it is used with the following acceptance condition:

\textbf{Büchi Acceptance}

We say $M$ accept a $\omega$-word $\alpha \in \Sigma^\omega$ if and only if there exist a run $r$ of $M$ on $\alpha$ satisfying

$$\text{Inf}(r) \cap F \neq \emptyset$$

i.e at least one accept state in $F$ has to be visited infinitely often during the run $r$, in that case we define the language of $M$ to be the set

$$L(M) := \{ \alpha \in \Sigma^\omega | M \text{ accept } \alpha \}$$

\textbf{Example:}

The following finite $\omega$-automata with $F := \{q_1, q_2\}$ accept the $\omega$-language

$$A := \{ \alpha \in \{a, b\}^\omega | \alpha \text{ ends with either } a^\omega \text{ or } (ab)^\omega \}$$

$$= \Sigma^* a^\omega \cup \Sigma^* (ab)^\omega$$

See Figure 2 on the next page.
**Definition:** (Büchi recognizable language)

A language \( A \subseteq \Sigma^\omega \) is called Büchi recognizable or regular \( \omega \)-language if there exist a Büchi automaton \( M \) such that \( L(M) = A \).

**Examples:**

1. The Language \( A = \Sigma^*a^\omega \cup \Sigma^*(ab)^\omega \) is Büchi recognizable.
2. for any \( a \in \Sigma \), \( \{a^\omega\} \) is Büchi recognizable.

**Note:**

1. In the previous example, if \( \alpha \in \Sigma^*(ab)^\omega \), then if \( r \) is an accepting run, then 
   \[
   \inf(r) \cap F \neq \emptyset \quad \text{and} \quad \inf(r) \cap Q \setminus F \neq \emptyset
   \]
   so in the same run we have an accepting state appear infinitely often and a non-accepting state also appear infinitely often which make Büchi -acceptance condition a weak condition.
2. Because of note 1, we cannot prove the given a Büchi automata, then there exist a Büchi automata that accept the complement of its language by simply changing the accept state to non-accept state.

**Theorem 1:**

Let \( A \subseteq \Sigma^* \) be regular language, then \( A^\omega \subseteq \Sigma^\omega \) is Büchi -recognizable language.

**Proof:**

Assume \( M = (Q, \Sigma, \delta, q_0, F) \) is a DFA that recognize \( A \), i.e. \( L(M) = A \), we might assume that there is no transition from any other state to the initial state (always can be acheived by inserting a new initial state with epsilon transition to \( q_0 \)).

Define \( M' := (Q, \Sigma, \delta', q_0, \{q_0\}) \) such that \( M' \) simulate \( M \) as follows, given \( \alpha \in \Sigma^\omega \), \( M' \) simulate \( M \). Now once we reach a final state in \( M \) say \( q \in Q \), then \( M' \) non-deterministically choose to continue or start \( M \) from its initial state and read the rest.
of the word.

so we have the following:

\[
\delta' : Q \times \Sigma \rightarrow \text{Pow}(Q)
\]

\[
(q, a) \mapsto \begin{cases} 
\delta(q, a), & \text{if } \delta(q, a) \cap F = \emptyset \\
\delta(q, a) \cup \{q_0\}, & \text{if } \delta(q, a) \cap F \neq \emptyset
\end{cases}
\]

Now all what we need to show is \(L(M') = A^\omega\)

Let \(\alpha \in A^\omega\) such that \(\alpha = w_1w_2 \cdots\), with each \(w_i \in A \subseteq \Sigma^*\). Now for each \(w_i\), \(M\) has computations \(r_{0i}, r_{1i}, \cdots, r_{n_i}\) such that:

1. \(r_{0i} = q_0\)
2. \(r_{j+1i} = \delta(r_{ji}, w_{ji})\)
3. \(r_{n_i} \in F\)

so we will get a sequence \(r = r_{01}, r_{11}, \cdots, r_{n_1} q_0 r_{02}, r_{12}, \cdots, r_{n_2} q_0 \cdots\)

so \(\text{Inf}(r) \cap \{q_0\} \neq \emptyset\), hence \(\alpha \in L(M')\) and so \(A \subseteq L(M)\).

Conversely, let \(\alpha \in L(M')\), then there exist a run with infinitely many \(q_0\), the words between any two \(q_0\)'s are in fact computations done by \(A\), \(\alpha \in A^\omega\) and so \(L(M) = A\) \(\square\)

Consider a Büchi Automata \(M = (Q, \Sigma, \delta, q_0, F)\), let \(p, q \in Q\), let \(W(p, q)\) be the set of all words obtained from state \(p\) to state \(q\), now \(W(p, q) \subseteq \Sigma^*\) is a regular language.

\(W(p, q) := \{ w = a_1 a_2 \cdots a_n \in \Sigma^* | \text{there exist a run from } p \text{ as start state on } w \text{ ending in } q \}\)

now a \(\omega\)-word \(\alpha \in \Sigma^*\) is accepted by \(M\) if and only if there exist a state \(q \in Q\) that has been visited infinitely often, so \(\alpha \in W(q_0, q) W(q, q)^\omega\), so

\[
L(M) = \bigcup_{q \in F} W(q_0, q) W(q, q)^\omega
\]

now this leads us to the important result know as Büchi Characterization Theorem.

**Theorem 2:**

Every Büchi recognizable language \(A\) is of the form

\[
A = \bigcup_{i=1}^{n} A_i B_i^\omega, \text{where } A_i, B_i \in \Sigma^* \text{ are regular languages } (i = 1, 2, \cdots, n)
\]

**proof:**
Let $A$ be Büchi recognizable language, then there is a Büchi automata $M$ such that $L(M) = A$, let $\alpha \in A = L(M)$, then $\exists q \in Q$ such that there exist a run of $M$ on $\alpha$ that visits $q$ infinitely often, so $\alpha \in W(q_0, q)W(q, q)\omega$, so

$$A = \bigcup_{q \in F} W(q_0, q)W(q, q)\omega = \bigcup_{i=1}^{n} A_iB_i\omega, A_i, B_i \text{ are regular languages}$$

Remarks:

1. In section 4, we will show that theorem 2 is actually "if and only if" statement.
2. Every $\omega$-language (Büchi recognizable) contains a periodic word.
3. In theorem 2, we can decide the emptiness problem as follows:
   
   Given $M$, we compute the set of reachable states from $q_0$, now for each reachable state $q$ which is in $F$, we check if there is a nonempty loop from $q$ to $q$, if such a loop exist, then $\exists \alpha \in \Sigma^\omega$ such that a run of $M$ on $\alpha$ will visit $q$ infinitely often.
   
   The algorithm is summarized as follows:
   
   (a) $\forall p \in Q$ find a reachable $q \in F$.
   (b) find a non-empty loop from $q$ to $q$.

2.2 - Muller Acceptance:

Definition: (Due to Muller)

A non-deterministic finite $\omega$-automata $M = (Q, \Sigma, \delta, q_0, F)$ with accept component $F \subseteq \text{pow}(Q)$ is called Muller Automation if it is used with the following acceptance condition:

Muller Acceptance

We say $M$ accept a $\omega$-word $\alpha \in \Sigma^\omega$ if and only if there exist a run $r$ of $M$ on $\alpha$ satisfying

$$\text{Inf}(r) \in F$$

i.e the set of infinitely often visited states are exactly one of the set in $F$

Example:

Recall the automata in the previous example, let $F = \{\{q_1, q_2\}\}$. Also let $A = \Sigma a^\omega \cup \Sigma (ab)^\omega$, then $A$ is the language for the following deterministic Muller automata with

$F = \{\{q_0\}, \{q_a, q_b\}\}$
Note:

1. The Muller condition is sharper condition than Büchi condition.
2. The deterministic Muller Automata has the same power as non-deterministic Muller Automata and both have the same power as non-deterministic Büchi automata.
3. The Muller condition will be used at the end of this project to show that the class of regular $\omega$-languages are closed under complementation.

2.3 - Rabin Acceptance:

Definition: (Due to Rabin)

A non-deterministic finite $\omega$-automata $M = (Q, \Sigma, \delta, q_0, \Omega)$ with accept component $\Omega = \{(E_1, F_1), (E_2, F_2), \cdots, (E_n, F_n) \mid E_i, F_i \subseteq Q\}$ is called Rabin Automata if it is used with the following acceptance condition:

Rabin Acceptance

We say $M$ accept a $\omega$-word $\alpha \in \Sigma^\omega$ if and only if there exist a run $r$ of $M$ on $\alpha$ such the $\exists (E, F) \in \Omega$ with

$$\text{Inf}(r) \cap E = \phi \text{ and } \text{Inf}(r) \cap F \neq \phi$$

Example:
Let $\Omega_1 = \{ (\{ q_a \}, \{ q_b \}) \}$ then the following automata accept the words with infinitely many $a$’s and only finite $b$’s, i.e. $L(M) = \Sigma^* a^w$.

Now if $\Omega_2 = \{ \phi, \{ q_a \} \}$, then
$L(M) = A = \{ \alpha \in \Sigma^\omega | \alpha \text{ contains infinitely many } a \}$

\[ \text{\textbullet \textbullet \textbullet} \]

2.4 - Street Acceptance:

**Definition:** (Due to Streett)

A non-deterministic finite $\omega$-automata $M = (Q, \Sigma, \delta, q_0, \Omega)$ with accept component $\Omega = \{ (E_1, F_1), (E_2, F_2), \cdots, (E_n, F_n) \mid E_i, F_i \subseteq Q \}$ is called Streett Automata if it is used with the following acceptance condition:

**Streett Acceptance**

We say $M$ accept a $\omega$-word $\alpha \in \Sigma^\omega$ if and only if there exist a run $r$ of $M$ on $\alpha$ such the $\exists (E, F) \in \Omega$ with

\[ \text{if } \text{Inf}(r) \cap F \neq \phi \text{ then } \text{Inf}(r) \cap E \neq \phi \]

**Example:**

![Finite automata](Figure5.png)

Figure 5: Finite automata

If $\Omega_1 = \{ (\{ q_a \}, \{ q_b \}) \}$ then $L(M) =$ the set of $\omega$-words that if it contains infinitely many $b$’s, then it contains infinitely many $a$’s.

\[ \text{\textbullet \textbullet \textbullet} \]

**Remark:**

1. Büchi automata is a special case of
   - Muller Automata with $F := \{ q \in Q \mid q \cap F \neq \phi \}$
   - Rabin Automata with $\Omega := \{ (\phi, F) \}$
   - Streett Automata with $\Omega := \{ (F, Q) \}$

2. Deterministic and non-deterministic Rabin and Streett Automata are equivalent to each other and they equivalent to non-deterministic Büchi automata.
§ 3 - Deterministic Büchi Automata versus non-deterministic Büchi automata

In this section, we will show that non-deterministic Büchi automata is stronger than the deterministic Büchi automata by providing a regular $\omega$-language $A \subseteq \Sigma^\omega$ that can be recognized by non-deterministic Büchi automata but with no deterministic Büchi automata.

Example:

Let $A := \Sigma^*a^\omega \subseteq \Sigma^\omega$ be $\omega$-language, now the following non-deterministic Büchi automata accept $A$. where $F = \{q_2\}$.

![Büchi automata recognize $A$](image)

**Theorem 1:**

Non-deterministic Büchi automata is stronger than deterministic Büchi automata

**Proof:**

by the previous example, we found a non-deterministic Büchi automata that recognize $A$. Now assume also there exist a deterministic Büchi automata $M$ that recognize $A = \Sigma^*a^\omega$, then $M$ accept the $\omega$-word $a^\omega$, after a finite prefix, $M$ will visit a $F$-state, say after $n_1$-letter of $a^\omega$. Now since $M$ recognize $A$, then it should accept $a^{n_1}ba^\omega$, so again after reading $b$, $M$ will visit a $F$-state, say after $n_2$-letter from $b$. so again $M$ must accept $a^{n_1}ba^{n_2}ba^\omega$ and continue this way, we reach that $M$ should accept $a^{n_1}ba^{n_2}ba^{n_3}ba^{n_4} \ldots$, i.e. $M$ will accept a $\omega$-word that contain infinitely many $b$’s which is contradiction. \hfill \Box

**Note:**

The key observation is that in the deterministic Büchi automata, we don’t allow the universe to die, it must continue after reading $a$ and go to $F$-state. in our example of the non-deterministic Büchi automata, if $a^\omega$ accepted by non-deterministic Büchi automata, then it is still true that $\exists n_1 \in \mathbb{N}$ such that $a^{n_1}ba^\omega$, but in this case if we arrive $F$-state after $n_1$-letters and we read $b$, then the universe die and we reject, so whenever we arrive $F$-state, then no $b$ is allowed.
§ 4 - Closure Properties

Theorem 1:

If $A_1, A_2 \subseteq \Sigma^\omega$ are two regular $\omega$-languages, then $A_1 \cup A_2$ is regular $\omega$-language.

Proof:
The idea is to use the cartesian product as we did in the class.
Assume $A_1, A_2 \subseteq \Sigma^\omega$ be two regular $\omega$-languages, then there exist a Büchi automaton $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $A_1 = L(M_1)$ and $A_2 = L(M_2)$, now let:

1. $Q := Q_1 \times Q_2$
2. $q_0 := (q_1, q_2) \in Q$
3. $F := \{ (p_1, p_2) \in Q \mid p_1 \in F_1 \text{ or } p_2 \in F_2 \}$
4. The transition function is

\[
\delta : Q \times \Sigma \rightarrow Pow(Q) \\
((p_1, p_2), a) \mapsto \{ (r, s) \in Q \mid r \in \delta(p_1, a), s \in \delta(p_2, a) \}
\]

Then we have the Büchi automaton $M := (Q, \Sigma, \delta, q_0, F)$ such that $L(M) = A_1 \cup A_2$
To see this, let $\alpha \in A_1 \cup A_2$, assume for instance that $\alpha \in A_1$, then there exist a run $r_1$ of $M_1$ on $\alpha$ such that $\text{Inf}(r_1) \cap F_1 \neq \emptyset$.
Now if we simulate $\alpha$ on $M$, then there will be a run $(r, s) = (r_0 r_1 \cdots, s_0 s_1 \cdots)$ with $\text{Inf}(r, s) \cap F \neq \emptyset$, because $\exists q \in F_1$ such $q \in \text{Inf}(r_1)$, so $(q, q') \in \text{Inf}(r, s)$ and $(q, q') \in F$, so $\text{Inf}(r, s) \cap F \neq \emptyset$, hence $\alpha \in L(M)$, i.e. $A_1 \cup A_2 \subseteq L(M)$

Conversely, let $\alpha \in L(M)$, then there exist a run $(r, s) = (r_0 r_1 \cdots, s_0 s_1 \cdots)$ such that $\text{Inf}(r, s) \cap F \neq \emptyset$, so $\exists (q, q') \in \text{Inf}(r, s)$ such that $(q, q') \in F$, so either $q \in \text{Inf}(r) \cap F_1$ or $q' \in \text{Inf}(s) \cap F_2$, so either $\text{Inf}(r) \cap F_1 \neq \emptyset$ or $\text{Inf}(s) \cap F_2 \neq \emptyset$, hence $\alpha \in A_1 \cup A_2$, so $A_1 \cup A_2 = L(M)$

Note:

To prove that regular $\omega$-languages are closed under intersection require a little bit of work and cannot simply modify the proof of theorem 2 by letting $F := F_1 \times F_2$. to see this, assume a run $(r, s)$ such that $r$ visits a state $q \in F_1$ infinitely many and $s$ visits $q' \in F_2$ infinitely many, but $(r, s)$ as a pair don’t necessarily visit $(q, q')$ infinitely many, it might be the case that $(q, p) \in F_1 \times Q_2 \setminus F_2$ and $(p', q') \in Q_1 \setminus F_1 \times F_2$ with $(q, p), (p', q') \in \text{Inf}(r, s)$.

So the idea to prove the intersection is the following:

1. We simulate $M_1$ and $M_2$ simultaneously.
2. Assume \( M_1 \) go into \( F_1 \)-state.
3. We continue simulation and wait for \( M_2 \) to enter \( F_2 \)-state.
4. We continue simulation and wait for \( M_1 \) to enter again in \( F_1 \)-state.

So if we change the waiting state infinitely many, then we accept, otherwise we reject, here is the formal proof:

**Theorem 2:**

If \( A_1, A_2 \subseteq \Sigma^\omega \) are two regular \( \omega \)-languages, then \( A_1 \cap A_2 \) is regular \( \omega \)-language.

**Proof:**

\( M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1) \) and \( M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2) \) such that \( A_1 = L(M_1) \) and \( A_2 = L(M_2) \), now let:

1. \( Q := Q_1 \times Q_2 \times \{1, 2, 3\} \)
2. \( q_0 := (q_1, q_2, 1) \in Q \)
3. \( F := F_1 \times F_2 \times 3 \)
4. 

\[
\delta : Q \times \Sigma \rightarrow \text{Pow}(Q)
\]

\[
((p_1, p_2, 1), a) \mapsto \begin{cases} 
(\delta_1(q_1, a), \delta_2(q_2, a), 1), & \text{if } \delta_1(q_1, a) \notin F_1 \\
(\delta_1(q_1, a), \delta_2(q_2, a), 2), & \text{if } \delta_1(q_1, a) \in F_1 
\end{cases}
\]

\[
((p_1, p_2, 2), a) \mapsto \begin{cases} 
(\delta_1(q_1, a), \delta_2(q_2, a), 2), & \text{if } \delta_2(q_2, a) \notin F_2 \\
(\delta_1(q_1, a), \delta_2(q_2, a), 3), & \text{if } \delta_2(q_2, a) \in F_2 
\end{cases}
\]

\[
((p_1, p_2, 3), a) \mapsto (\delta_1(q_1, a), \delta_2(q_2, a), 1)
\]

Then we have the Büchi automaton \( M := (Q, \Sigma, \delta, q_0, F) \) such that \( L(M) = A_1 \cup A_2 \)

\( \square \)

Now if we allow the idea of \( \epsilon \)-transition in Büchi automata, we immediately have the following result.

**Proposition 3:**

If \( A \subseteq \Sigma^* \) is regular language, \( B \subseteq \Sigma^\omega \) is regular \( \omega \)-language, then \( AB \) is regular \( \omega \)-language.

**Theorem 4:** (Büchi Characterization Theorem)

A \( \omega \)-language \( A \subseteq \Sigma^\omega \) is regular if and only if \( A = \bigcup_{i=1}^{n} A_i B_i^\omega \), where \( A_i, B_i \) are regular languages (\( \forall i = 1, 2, \cdots, n \))
**Proof:**

$\Rightarrow$) Theorem 2 section 2.1.

$\Leftarrow$) Assume $A = \bigcup_{i=1}^{n} A_i B_i^\omega$, where $A_i, B_i$ are regular languages ($\forall i = 1, 2, \cdots, n$), then we know that $B_i^\omega$ is regular $\omega$-language by theorem 1, section 2.1, now by proposition 3, we have $A_i B_i^\omega$ is regular $\omega$-language and by theorem 1, finite union of regular $\omega$-language is regular $\omega$-language, hence $A = \bigcup_{i=1}^{n} A_i B_i^\omega$ is regular $\omega$-language.

Finally we show that regular $\omega$-language is closed under complementation.

**Theorem 5:**

Let $A \subseteq \Sigma^\omega$ be regular $\omega$-language, then $\Sigma^\omega - A$ is regular $\omega$-language.

**Proof:**

Let $A \subseteq \Sigma^\omega$ be regular $\omega$-language, then there exist a Muller automaton that accept it, i.e. there exist $M = (Q, \Sigma, \delta, q_0, F)$ with $F \subseteq \text{pow}(Q)$ with $L(M) = A$, now consider the Muller automata $M' = (Q, \Sigma, \delta, q_0, \text{pow}(Q) - F)$, then clearly $L(M') = \Sigma^\omega - A$. $\square$
References


[5] Michael O. Rabin, Automata on infinite objects and Church’s problem, Providence, R.I., Published for the Conference Board of the Mathematical Sciences by the American Mathematical Society [1972].