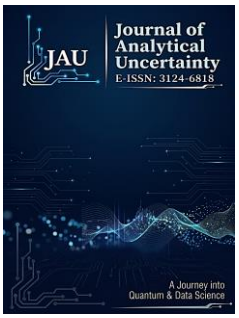
OPEN ACCESS PEER-REVIEWED

JAU Vol.2, No.1, 2026 - Page: 90-98
doi.org/10.63924/jau.v2i1.363
e-ISSN: 3124-6818



Check for updates

DFA Minimization of Fuzzy Multiset Automata

Binod Kumar Sharma[1*], Apostolos Syropoulos[2]

[1]Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Ranchi-834001, India; [2]Independent Scholar, Xanthi, Greece

*Corresponding author: Binod Kumar Sharma, Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Ranchi-834001, India. Email: bksharma@bitmesra.ac.in

Abstract

Deterministic finite automaton (DFA) minimization is the process of converting a DFA into an equivalent DFA with a minimum number of states. Here we apply on such process (i.e., Brzozowski's algorithm) to crisp deterministic fuzzy multiset finite automata. We introduce reverse fuzzy multiset finite automata and establish a relation between their right and left fuzzy multiset languages. In between, we correlate two crisp-deterministic fuzzy multiset finite automata and show that the reachability and observability map of the given crisp deterministic fuzzy multiset finite automata to be isomorphic.

Keywords: DFA Minimization, Fuzzy Multiset Finite Automata, Fuzzy Multiset Language, Reverse Fuzzy Multiset Finite Automata, Crisp-Deterministic Fuzzy Multiset Automata, Brzozoswki,s Algorithm

Introduction

In computer science, finite automata are conceptual machines capable of recognizing whether a string (i.e., a sequence of characters) belongs to some formal language or not [26]. The literature discusses many applications where finite automata prove useful (e.g., see [7, 22, 29]). In automata theory (a branch of theoretical computer science), deterministic finite automaton (DFA) minimization is the process of reducing a given DFA into an equivalent DFA with the minimal number of states. Two DFAs are considered equal if they recognize the same regular language. A well-known example of DFA minimization is Brzozowski's algorithm. The study of fuzzy automata was initiated by Wee [26] and Santos [23] in the 1960s after the introduction of fuzzy set theory by Zadeh [30]. Fuzzy automata and fuzzy language have been introduced and studied as a means of bridging the gap between the precision of formal languages and the vagueness of natural languages (cf. [1, 8, 9, 12, 14, 17, 16, 21]). The last few years, many researchers contributed significantly to the field of fuzzy automata (e.g., see [15, 16, 18]). In particular, these researchers worked on the algebraic aspects of fuzzy automata theory. Fuzzy finite automata have attracted wide attention and have been applied to many domains, such as in pattern recognition, database theory, learning system, and fuzzy discrete event systems (e.g., see [11, 13, 18, 27]).

A multiset (or bag, as it is called sometimes) is like a set except that elements are allowed to appear more than once. The theory of multisets was introduced by Cerf et al. [4] and Peterson [20] in 1970s. After the introduction of the theory of multisets, Yager [28] and Blizard [2] have made significant contributions on it. A multiset is a natural generalization of the notion of a set, and they have been described, among others, by Syropoulos [24]. Yager [28] was the first author who discussed fuzzy multisets. In real life, we often encounter situations involving collections of objects that contain certain elements multiple times. Following the advent of multiset theory, the

Copyright: © 2026 by the author(s). This is an open access article distributed under the CC BY-SA 4.0 license

Received: March 10, 2026 - Accepted: November 12, 2026 - Published online: November 17, 2026

concept of multiset finite automata has been introduced in [6, 10], and their connection to membrane computing was made clear in [5].

Inspired by the generalization of finite automata to fuzzy finite automata, recently Wang, Yin, and Gu [25] introduced the concept of fuzzy multiset finite automata as a generalization of multiset finite automata. Recently, Tiwari et al. [24] associated such an automaton with a deterministic fuzzy multiset finite automaton. Also, Tiwari et al. [24] studied the minimal realization (minimal deterministic fuzzy multiset automata) of a given fuzzy multiset language. In this paper, we study the well-known Brzozowski's algorithm for minimization of a crisp deterministic fuzzy automaton. This algorithm for minimization of finite automata is based on applying the reversal and power-set construction twice in sequence [3, 9]. We introduce the reverse fuzzy multiset finite automata and establish a relation between their right and left fuzzy multiset languages. In between we correlate two crisp and deterministic fuzzy multiset finite automata and show that the reachability and observability maps of the given crisp and deterministic fuzzy multiset finite automata are isomorphic.

This paper is organized as follows. In Sec. 2, we recall some concepts related to multiset and multiset finite automata, while in Sec. 3 we formally present fuzzy multiset finite automata. Furthermore, we introduce the notions of right fuzzy multiset language and left fuzzy multiset language. In Sec. 4, we show that crisp-deterministic fuzzy multiset finite automata are as powerful as fuzzy multiset finite automata in terms of accepting fuzzy multiset languages. In Sec. 5, we correlate two crisp-deterministic fuzzy multiset finite automata and show that the reachability and observability map of the given crisp-deterministic fuzzy multiset finite automata is isomorphic. Finally, in Sec. 6, there is the study of reverse crisp deterministic fuzzy multiset finite automata of a given crisp-deterministic fuzzy multiset finite automata and Brzozowski's algorithm for crisp deterministic fuzzy multiset finite automata.

Preliminaries

In this section, we recall the concept of multisets and multiset finite automata, which are used in the next sections. We begin with the following concept of multisets, which are fairly standard and can be found in the literature (cf., [2, 6, 25]).

Definition 2.1 Suppose that Σ is a finite alphabet. Then, $A : \Sigma \to \mathbb{N}$ is a *multiset* over Σ, where $\mathbb{N}$ is the set of natural numbers including 0. The cardinality of A, denoted by $|A|$, is equal to $\sum_{a\in\Sigma} A(a)$.

We will denote by 2^{Σ} the set of all multisets over Σ. The multiset $0_{\Sigma} \in 2^{\Sigma}$ is the one that has the property that $0_{\Sigma}(a) = 0$, $\forall a \in \Sigma$. For $b \in \Sigma$, we qill denote by $\langle b\rangle$, a *singleton multiset*, which is defined by

$$\langle b\rangle(a) = \begin{cases} 1, if & \mathrm{b} = \mathrm{a} \\ 0, if & \mathrm{b} \neq \mathrm{a} \end{cases}$$

for all $a \in \Sigma$. For a given set A, let $A = \{\langle a\rangle \mid a \in A\}$. Assume that $\Sigma = \{a, b, c\}$. Then, the multiset $\alpha = (1/a, 2/b, 0/c)$ can be represented by $\langle a\rangle \oplus \langle b\rangle \oplus \langle b\rangle$.

For two multisets $\alpha, \beta \in 2^{\Sigma}$, the operations inclusion $\subseteq$, addition $\oplus$ and difference $\ominus$ are defined as follows:

(i) $\alpha \subseteq \beta$ if $\alpha(a) \leq \beta(a)$,
(ii) $(\alpha \oplus \beta)(a) = \alpha(a) + \beta(a)$, and
(iii) $(\alpha \ominus \beta)(a) = max(0, \alpha(a) - \beta(a))$,

for all $a \in \Sigma$. Furthermore, $\alpha \subset \beta$ if $\alpha \subseteq \beta$ and $\alpha \neq \beta$. Also, for $A, B \subseteq 2^{\Sigma}$, $A \oplus B = \cup_{\alpha\in A,\beta\in B}\alpha \oplus \beta$.

It is trivial to show that 2^{Σ} is a monoid with identity element 0_{Σ} with respect to the $\oplus$ operation.

Definition 2.2 A *multiset finite automaton* (MFA) is a quintet $M = (Q, \Sigma, \delta, q_0, F)$ such that

(i) Q and Σ are nonempty finite sets called the *state-set* and *input-set*, respectively;
(ii) $\delta : Q \times 2^{\Sigma} \to 2^{Q}$ is a map called *transition map*;
(iii) $q_0 \in Q$ is called the *initial state*; and
(iv) $F \subseteq Q$ is called the *set of final states*.

A *configuration* of an *MFA M* is a pair (p, α), where p and α denote the current state and the current multiset, respectively. The transition in a multiset finite automaton described by the set of configurations. The transition from configuration (p, α) leads to configuration (q, β) if there exists a multiset $\gamma \in 2^{\Sigma}$ with $\gamma \subseteq \alpha$, $q \in \delta(p, \gamma)$, and $\beta = \alpha \ominus \gamma$, and is denoted by $(p, \alpha) \rightarrow (q, \beta)$. We shall denote by $\rightarrow *$, the reflexive and transitive closure of this operation.

Definition 2.3 An *MFA* $M = (Q, \Sigma, \delta, q_0, F)$ is said to be a *deterministic multiset finite automaton (DMFA)* if $|\delta(p, \alpha)| \leq 1$ for all $p \in Q$ and for all $\alpha \in 2^{\Sigma}$.

Definition 2.4

(i) For a given set Σ, a *multiset language* L is a subset of 2^{Σ}.

(ii) A multiset language $L \subseteq 2^{\Sigma}$ is *accepted* by an *MFA* $M = (Q, \Sigma, \delta, q_0, F)$ if

$L(M) = \{\alpha \in 2^{\Sigma} : \delta(q_0, \alpha) \cap F \neq \phi\}$,

for all $\alpha \in 2^{\Sigma}$.

Fuzzy Multiset Finite Automata

In this section, we study fuzzy multiset finite automata and fuzzy multiset languages as introduced in [19, 25]. Also, we introduce the notion of a right fuzzy multiset language and a left fuzzy multiset language. Finally, we establish a relation between right fuzzy multiset languages and left fuzzy multiset languages. We begin with the definition of fuzzy multiset finite automata.

Definition 3.1 A *fuzzy multiset finite automaton* (*FMFA*) is a quintuple $(Q, \Sigma, \delta, \sigma, \tau)$ such that

(i) Q and Σ are nonempty finite sets called the *state-set* and *input-set*, respectively;

(ii) $\delta : Q \times 2^{\Sigma} \times Q \rightarrow [0, 1]$ is a map called *fuzzy transition map*;

(iii) $\sigma : Q \rightarrow [0, 1]$ is a map called the *fuzzy set of initial states*; and

(iv) $\tau : Q \rightarrow [0, 1]$ is a map called the *fuzzy set of final states*.

A *configuration* of a fuzzy multiset finite automaton M is a pair (p, α), where p and α denote the current state and the current multiset, respectively. The transition of a fuzzy multiset finite automaton is described by the set of configurations. A transition from configuration (p, α) leads to configuration (q, β) with membership degree $k \in [0, 1]$ if there exists a multiset $\gamma \in 2^{\Sigma}$ with $\gamma \subseteq \alpha$, $\delta(p, \gamma, q) = k$ and $\beta = \alpha \ominus \gamma$ and is denoted by

$$(p, \alpha) \overset{k}{\rightarrow} (q, \beta)$$

where $\overset{k}{\rightarrow} *$ denotes the reflexive and transitive closure of $\overset{k}{\rightarrow}$, that is, for $(p, \alpha), (q, \beta) \in Q \times 2^{\Sigma}$, $(p, \alpha) \overset{k}{\rightarrow} * (q, \beta)$, if for some $n \geq 0$, there exists $(n + 1)$ states $q_0, \ldots, q_n$ and $(n + 1)$ multisets $\alpha_0, \ldots, \alpha_n$ such that $q_0 = p$, $q_n = q$, $\alpha_0 = \alpha$, $\alpha_n = \beta$ and $(q_i, \alpha_i) \overset{k_i}{\rightarrow} (q_{i+1}, \alpha_{i+1})$ all $i = 0, \ldots, n - 1$, where $k' = k_0 \wedge k_1 \wedge \cdots \wedge k_{n-1}$. Now, we define

$\mu_M((p, \alpha) \rightarrow *(q, \beta)) = \vee\{\mu_M((p, \alpha) \rightarrow *(r, \alpha \ominus \gamma)) \wedge \mu_M((r, \alpha \ominus \gamma)$

$\rightarrow *(q, \beta)) : r \in Q, \gamma \in 2^{\Sigma}$ and $\gamma \subseteq \alpha\}$,

and

$$\mu_M((p, \alpha) \rightarrow *(q, \alpha)) = \begin{cases} 1, & if\ p = q \\ 0, & \text{if p} \neq \text{q} \end{cases}$$

Definition 3.2

(i) For a given set Σ, a *fuzzy multiset language* is a map $f : 2^{\Sigma} \rightarrow [0, 1]$.

(ii) A fuzzy multiset language $f : 2^{\Sigma} \rightarrow [0, 1]$ is *accepted* by a *FMFA* $M = (Q, \Sigma, \delta, \sigma, \tau)$ if

$f(\alpha) = \vee\{\sigma(q) \wedge \mu_M((q, \alpha) \rightarrow *(p, 0_{\Sigma})) \wedge \tau(p) : q, p \in Q\}$,

$\forall \alpha \in 2^{\Sigma}$. Further, a fuzzy multiset language $f : 2^{\Sigma} \rightarrow [0, 1]$ is called *regular* if there exists a *FMFA M* such that $f = f_M$.

We shall denote a fuzzy multiset language f by f_M, if f is accepted by a *FMFA M*. Now, we introduce the concept of reverse fuzzy multiset finite automata.

Definition 3.3 Let $M = (Q, \Sigma, \delta, \sigma, \tau)$ be a *FMFA*. The *reverse FMFA* of M is a *FMFA* $M_R = (Q, \Sigma, \delta_R, \sigma_R, \tau_R)$, where $\sigma_R = \tau$, $\tau_R = \sigma$ and the map $\delta_R : Q \times 2^{\Sigma} \times$

$Q \rightarrow [0, 1]$ is given by

$\mu_{MR}((q, \alpha) \rightarrow *(p, 0_{\Sigma})) = \mu_M((p, \alpha) \rightarrow *(q, 0_{\Sigma}))$,

for all $q, p \in Q$ and $\alpha \in 2^{\Sigma}$.

Definition 3.4

(i) A *right fuzzy multiset language* $f_q : 2^{\Sigma} \to [0, 1]$ accepted by *FMFA* $M = (Q, \Sigma, \delta, q, \tau)$ with crisp initial state q is given by

$f_q(\alpha) = \vee\{\mu_M((q, \alpha) \to *(p, 0_{\Sigma})) \wedge \tau(p) : p \in Q\}, \ \forall \alpha \in 2^{\Sigma}$.

(ii) A *left fuzzy multiset language* ${}_qf : 2^{\Sigma} \to [0, 1]$ accepted by a *FMFA* $M = (Q, \Sigma, \delta, \sigma, \{q\})$ with crisp final state $\{q\} \in Q$ is given by

$qf(\alpha) = \vee\{\sigma(p) \wedge \mu_M((p, \alpha) \to *(q, 0_{\Sigma})) : p \in Q\}, \ \forall \alpha \in 2^{\Sigma}$.

Proposition 3.1 *The right fuzzy multiset language accepted by an FMFA M is equal to the left fuzzy multiset language accepted by the reverse FMFA M_R.*

Proof: Let right fuzzy multiset language $f_q : 2^{\Sigma} \to [0, 1]$ be accepted by *FMFA* $M = (Q, \Sigma, \delta, q, \tau)$ with crisp initial state $q \in Q$, that is,

$f_q(\alpha) = \vee\{\mu_M((q, \alpha) \to *(p, 0_{\Sigma})) \wedge \tau(p) : p \in Q\}, \quad \forall \alpha \in 2^{\Sigma}$.

Also, let $M_R = (Q, \Sigma, \delta_R, \sigma_R, F)$ be a reverse FMFA of M. Then, for all $\alpha \in 2^{\Sigma}$,

$$\begin{aligned} qf(\alpha) &= \vee\{\sigma_R(p) \wedge \mu_{MR}((p, \alpha) \to (q, 0_{\Sigma})) : p \in Q\} \\ &= \vee \{(\tau(p) \wedge \mu_M((q, \alpha) \to (p, 0_{\Sigma})) : p \in Q\} \\ &= \vee \{(\mu_M((q, \alpha) \to (p, 0_{\Sigma})) \wedge \tau(p)) : p \in Q\} \\ &= f_q(\alpha) \end{aligned}$$

Crisp-Deterministic Fuzzy Multiset Finite Automata

In this section, we introduce and study the concept of crisp-deterministic fuzzy multiset finite automata. These are as powerful as fuzzy multiset finite automata in the sense of acceptance of fuzzy multiset language.

Definition 4.1 A *crisp-deterministic fuzzy multiset finite automata(CDFMFA)* is a quintuple $M = (Q, \Sigma, \delta, q_0, \tau)$ such that

(i) Q and Σ are nonempty finite sets called the *state-set* and *input-set*, respectively;

(ii) $\delta : Q \times 2^{\Sigma} \to Q$ is a map called *transition map*;

(iii) $q_0 \in Q$ is an *initial states*; and

(iv) $\tau : Q \to [0, 1]$ is a map called the *fuzzy set of final states*.

A *configuration* of a *CDFMFA* M is a pair (p, α), where p and α denote the current state and the current multiset, respectively. The transition of a crisp fuzzy multiset automaton is described by the set of configurations. A transition from configuration (p, α) leads to configuration (q, β) if there exists a multiset $\gamma \in 2^{\Sigma}$ with $\gamma \subseteq \alpha$, $q = \delta(p, \gamma)$ and $\beta = \alpha \ominus \gamma$, and is denoted by $(p, \alpha) \to (q, \beta)$. We shall denote by $\to *$, the reflexive and transitive closure of this operation.

Now, we recall the following from [24], which shows that a *CDFMFA* is as powerful as a *FMFA*.

Proposition 4.1 *For a given fuzzy multiset language $f : 2^{\Sigma} \to [0, 1]$, f is accepted by a FMFA if and only if it is accepted by a CDFMFA.*

Definition 4.2

(i) Given a *CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$, the *fuzzy multiset language* f_q accepted by M in state q is a map $f_q : 2^{\Sigma} \to [0, 1]$ such that

$f_q(\alpha) = \tau(\delta(q, \alpha)), \ \forall \alpha \in 2^{\Sigma}$.

(ii) The fuzzy multiset language accepted by *FMFA* M in state q_0 is called the *fuzzy multiset language* accepted by M.

For a given fuzzy multiset language $f_q : 2^{\Sigma} \to [0, 1]$, we have the following.

Proposition 4.2 *Given a fuzzy multiset language $f_q : 2^{\Sigma} \to [0, 1]$ accepted by the CDFMFA $M = (Q, \Sigma, \delta, q_0, \tau)$, there exist a CDFMFA M', which accepts f_q.*

Proof: Let $f_q : 2^{\Sigma} \to [0, 1]$ be accepted by a *CDFMFA* $(Q, \Sigma, \delta, q_0, \tau)$,

that is, $f_q(\alpha) = \tau(\delta(q, \alpha))$. Define the map $\delta' : F(2^{\Sigma}) \times 2^{\Sigma} \to F(2^{\Sigma})$ and $\tau' : F(2^{\Sigma}) \to [0, 1]$ such that $\delta'(f_q, \alpha) = f_{\delta(q,\alpha)}$ and $\tau'(f_q) = f_q(0_{\Sigma})$, where $F(2^{\Sigma})$ is collection of fuzzy set of 2^{Σ}. The map δ' and τ' are well defined, which can be shown as follows.

Let $f_q, f_p \in F(2^{\Sigma})$ such that $f_q = f_p$. Then

$f_q = f_p \Rightarrow f_q(\alpha) = f_p(\alpha)$,

for all $\alpha \in 2^{\Sigma}$

$\Rightarrow \tau(\delta(q, \alpha)) = \tau(\delta(p, \alpha))$
$\Rightarrow \tau(\delta(\delta(q, \alpha), 0_\Sigma) = \tau(\delta(\delta(p, \alpha), 0_\Sigma)$
$\Rightarrow f\delta(q,\alpha)(0\Sigma) = f\delta(p,\alpha)(0\Sigma)$
$\Rightarrow f\delta(q,\alpha) = f\delta(p,\alpha) \Rightarrow \delta(f_q, \alpha) = \delta(f_p, \alpha)$,
whereby δ' is well defined. Also, let $f_q, f_p \in F(2^\Sigma)$ such that $f_q = f_p$. Then

$$f_q = f_p \Rightarrow f_q(\alpha) = f_p(\alpha),$$

for all $\alpha \in 2^\Sigma$
$\Rightarrow f_q(0_\Sigma) = f_p(0_\Sigma)$
$\Rightarrow \tau'(f_q) = \tau'(f_p)$,
whereby τ' is well defined. Thus $M' = (F(2^\Sigma), \Sigma, \delta', f_q, \tau')$ is a *CDFMFA*. Finally, for all $\alpha \in 2^\Sigma$,
$(f_q)_M{}'(\alpha) = \tau'(\delta(f_q, \alpha))$
$= \tau'(f\delta(q,\alpha))$
$= f\delta(q,\alpha)(0\Sigma)$
$= \tau(\delta(\delta(q, \alpha), 0_\Sigma)$
$= \tau(\delta(q, \alpha)) = f_q(\alpha)$.
Hence $(f_q)_M{}'(\alpha) = f_q(\alpha)$.

We close this section by recalling the concept of homomorphism between two crisp-deterministic fuzzy multiset finite automata from [24].

Definition 4.3 Let $M = (Q, \Sigma, \delta, q_0, \tau)$ and $M' = (Q', \Sigma, \delta', q_0', \tau')$ be two *CDFMFA*.

Then, a function $\phi : Q \to Q'$ is called a *homomorphism* from M to M' if

(i) $\phi(\delta(q, \alpha)) = \delta'(\phi(q), \alpha)$; and
(ii) $\tau(q) = \tau'(\phi(q))$ for all $q \in Q$ and $\alpha \in 2^\Sigma$.

Also, M' is called a *homomorphic image* of M, if ϕ is onto and a homomorphism ϕ is called an *isomorphism* if it is both one-one and onto.

Reachability and Observability of Crisp Deterministic Fuzzy Multiset Automata

In this section, we correlate two crisp-deterministic fuzzy multiset finite automata and show that the reachability and observability maps of the given crisp-deterministic fuzzy multiset finite automata are isomorphic.

We begin with by introducing the concept of a reachability map of a crisp deterministic fuzzy multiset finite automaton.

Definition 5.1

(a) Let $M = (Q, \Sigma, \delta, q_0, \tau)$ be a *CDFMFA*. The *reachability map* r of M is a map $r : 2^\Sigma \to Q$ such that

(i) $r(0_\Sigma) = q_0$; and
(ii) $r(\alpha \oplus \langle a \rangle) = \delta(r(\alpha), \langle a \rangle)$, for all $\alpha, \langle a \rangle \in 2^\Sigma$.

(b) M is called *reachable* if r is onto.

Remark 5.1 *It can be easily seen that*

(i) $r(\alpha) = \delta(q_0, \alpha), \forall \alpha \in 2^\Sigma$*; and*
(ii) $r(\alpha \oplus \beta) = \delta(r(\alpha), \beta), \forall \alpha, \beta \in 2^\Sigma$.

A *CDFMFA* is *reachable* when all of its states should be reached from the initial state. In different words, a *CDFMFA* $(Q, \Sigma, \delta, q_0, \tau)$ is said to be *reachable* if for each $q \in Q$ there exists $\alpha \in 2^\Sigma$ such that $\delta(q_0, \alpha) = q$. Which in terms of the reachability map can be characterized by the following result.

Proposition 5.1 *A CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$ *is* reachable *if and only if* r *is onto*.

The reachability of a *CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$, lead us to establish the following relation between a *CDFMFA* M' and a *CDFMFA* $M_f = (Q_f, \Sigma, \delta_f, f_q, \tau_f)$ that have been introduced in proposition 5.6 of [24].

Proposition 5.2 *Let* $M = (Q, \Sigma, \delta, q_0, \tau)$ *be a reachable CDFMFA. Then, CDFMFA* $M_f = (Q_f, \Sigma, \delta_f, f_q, \tau_f)$ *is isomorphic to* $M' = (F(2^\Sigma), \Sigma, \delta', f_q, \tau')$.

Prof: Define a map $\phi : M_f \to M'$ such that $\phi(f_\alpha) = f_{\delta(q_0,\alpha)}$, for all $\alpha \in 2^\Sigma$. The map ϕ is well defined as shown below. Let $\alpha, \beta \in 2^\Sigma$ such that $f_\alpha = f_\beta$. Then,

$f_\alpha = f_\beta \Rightarrow f_\alpha(\gamma) = f_\beta(\gamma)$

for all $\gamma \in 2^{\Sigma}$

$\Rightarrow f_{\alpha}(0_{\Sigma}) = f_{\beta}(0_{\Sigma})$

$\Rightarrow f(\alpha) = f(\beta)$

$\Rightarrow \tau(\delta(q_0, \alpha)) = \tau(\delta(q_0, \beta))$

$\Rightarrow \tau(\delta(q_0, \alpha), 0_{\Sigma}) = \tau(\delta(q_0, \beta), 0_{\Sigma})$

$\Rightarrow f\delta(q_0,\alpha)(0\Sigma) = f\delta(q_0,\beta), (0\Sigma)$

$\Rightarrow f\delta(q_0,\alpha) = f\delta(q_0,\beta)$

$\Rightarrow \phi(f_{\alpha}) = \phi(f_{\beta})$,

whereby ϕ is well defined. Since M is reachable, that is, for all $q \in Q$ there exists an $\alpha \in 2^{\Sigma}$ such that $\delta(q_0, \alpha) = q$, whereby $f_q = f_{\delta(q0},\alpha)$. Then, for given $f_q \in F(2^{\Sigma})$, there exists an $f_{\alpha} \in Q_f$ such that $\phi(f_{\alpha}) = (f_q)$. Thus, ϕ is onto. Obviously, the defined map ϕ is one-one. Finally, for all $\alpha, \beta \in 2^{\Sigma}$,

$\phi(\delta_f(f_{\alpha}, \beta)) = \phi(f_{\alpha\oplus\beta})$

$= f\delta(q_0,\alpha\bigoplus\beta)$

$= f\delta(\delta(q_0,\alpha),\beta)$

$= \delta'(f_{\delta(q0},\alpha), \beta) = \delta'(\phi(f_{\alpha}), \beta)$,

whereby ϕ is homomorphism. Hence M_f is isomorphic to M'.

Definition 5.2 Given a *CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$, the *observability map* σ_M is a map $\sigma_M : Q \to F(2^{\Sigma})$ such that $\sigma_M(q) = f_q$. M is called observable if σ_M is one-one.

Remark 5.2 *The observability of a CDFMFA M tells us which different fuzzy languages are assigned to distinct states.*

Proposition 5.3 *The observability map* $\sigma_M : Q \to F(2^{\Sigma})$ *of a CDFMFA* $M = (Q, \delta, \Sigma, q_0, \tau)$ *is a homomorphism from M to* $M' = (F(2^{\Sigma}), \Sigma, \delta', f_q, \tau')$.

Proof: Given a *CDFMFA* $M = (Q, \delta, \Sigma, q_0, \tau)$, the observability map $\sigma_M : Q \to F(2^{\Sigma})$ is given by $\sigma_M(q) = f_q$. Then, for all $q \in Q$ and $\alpha \in 2^{\Sigma}$,

$\sigma M(\delta(q, \alpha)) = f\delta(q,\alpha)$

and

$\delta_1(\sigma_M(q), \alpha) = \delta_1(f_q, \alpha) = f_{\delta(q,\alpha)}$,

that is, $\sigma_M(\delta(q, \alpha)) = \delta_1(\sigma_M(q), \alpha)$. Also, for all $q \in Q$,

$\tau_1(\sigma_M(q)) = \tau_1(f_q)$

$= f_q(0_{\Sigma})$

$= \tau(\delta(q, 0_{\Sigma})) = \tau(q)$.

Definition 5.3 A *CDFMFA* $M = (Q, \Sigma, \delta, q_o, \tau)$ is *minimal* if M is both reachable and observable.

Brzozowski's Algorithm for Crisp Deterministic Fuzzy Multiset Automata

In this section, we introduce and study the concept of reverse crisp deterministic fuzzy multiset finite automaton of a given crisp-deterministic fuzzy multiset finite automaton. It is shown that if the given crisp deterministic fuzzy multiset finite automaton accepts a fuzzy multiset language, the reverse crisp deterministic fuzzy multiset finite automata accepts the reverse fuzzy multiset language. Finally, we introduce the Brzozowski's algorithm for crisp deterministic fuzzy multiset finite automata.

Let $M = (Q, \Sigma, \delta, q_0, \tau)$ be crisp deterministic fuzzy multiset finite automaton. Define a map $\tau_{\alpha} : Q \to [0, 1]$ such that $\tau_{\alpha}(q) = \tau(\delta(q, \alpha))$, and let

$Q_r = \tau_{\alpha} : \alpha \in 2^{\Sigma}$.

We define two maps $\delta_r : Q_r \times 2^{\Sigma} \to Q_r$ and $\nu_r : Q_r \to [0, 1]$ such that $\delta_r(\tau_{\alpha}, \beta) = \tau_{\alpha\oplus\beta}$, $\forall \alpha, \beta \in 2^{\Sigma}$ and $\nu_r(\tau_{\alpha}) = \tau_{\alpha}(q_0)$. Both these maps are well defined and here are the relevant details. Let $\tau_{\alpha}, \tau_{\alpha}' \in 2^{\Sigma}$ such that $\tau_{\alpha} = \tau\alpha'$. Then,

$\tau_{\alpha} = \tau_{\alpha}' \Rightarrow \tau_{\alpha}(\delta(q, \beta)) = \tau_{\alpha}'(\delta(q, \beta))$

$\Rightarrow \tau(\delta(\delta(q, \alpha), \beta)) = \tau(\delta(\delta(q, \alpha'), \beta))$

$\Rightarrow \tau(\delta(q, \alpha \oplus \beta)) = \tau(\delta(q, \alpha' \oplus \beta))$

$\Rightarrow \tau_{\alpha\oplus\beta}(q) = \tau_{\alpha}'\bigoplus\beta(q)$

$\Rightarrow \tau\alpha\bigoplus\beta = \tau\alpha'\bigoplus\beta$

$\Rightarrow \delta_r(\tau_\alpha,\beta) = \delta_r(\tau_\alpha', \beta)$,

whereby δ_r is well defined. Also, let $\tau_\alpha = \tau_\alpha'$. Then,

$\tau_\alpha(q_0) = \tau_\alpha'(q_0) \Rightarrow \nu_r(\tau_\alpha) = \nu_r(\tau_\alpha')$.

Thus, ν_r is well-defined.

Now, we introduce the concept of a reverse *CDFMFA* of a given *CDFMFA*.

Definition 6.1 Let $M = (Q, \Sigma, \delta, q_0, \tau)$ be a crisp-deterministic fuzzy multiset automaton. Then, the reverse crisp-deterministic fuzzy multiset automata of M is the automaton $M_r = (Q_r, \Sigma, \delta_r, \tau_{0\Sigma}, \nu_r)$, where

(i) $\delta_r : Q_r \times 2^\Sigma \to Q_r$ is a map such that $\delta_r(\tau_\alpha, \beta) = \tau_{\alpha\oplus\beta}$; and

(ii) $\nu_r : Q_r \to [0, 1]$, such that $\nu_r(\tau_\alpha) = \tau_\alpha(q_0)$.

Definition 6.2 For a fuzzy multiset language $f : 2^\Sigma \to [0, 1]$, the *reverse fuzzy multiset language* of f is a fuzzy multiset language $f : 2^\Sigma \to [0, 1]$ such that

$$\overline{f(\alpha)} = f(\alpha), \ \forall \alpha \in 2^\Sigma,$$

where α is the reverse of α. From the above definition, it is clear that $\overline{(f(\bar{\alpha}))} = f(\bar{\alpha})$,as $\alpha = \alpha$, for all $\alpha \in 2^\Sigma$.

Proposition 6.1 *If a CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$ *accepts the fuzzy multiset language* $f : 2^\Sigma \to [0, 1]$, *then the reverse CDFMFA* $M_r = (Q_r, \Sigma, \delta_r, \tau_{0\Sigma}, \nu_r)$ *accepts the fuzzy multiset language* f.

Proof: Let the fuzzy multiset language $f : 2^\Sigma \to [0, 1]$ be accepted by the *CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$, that is, $f(\alpha) = \tau(\delta(q_0, \alpha))$, for all $\alpha \in 2^\Sigma$. Now,

$\nu_r(\delta_r(\tau_{0\Sigma}, \alpha) = \nu_r(\tau_\alpha)$

$= \tau_\alpha(q_0)$

$= \tau(\delta(q_0, \alpha))$

$= f(\alpha) = \overline{f(\alpha)}$,

for all $\alpha \in 2^\Sigma$. Thus, the fuzzy multiset language f is accepted by M_r.

Proposition 6.2 *Let CDFMFA* $M = (Q, \Sigma, \delta, q_0, \tau)$ *be reachable. Then,* $M_r=(Q_r, \Sigma, \delta_r, \tau_{0\Sigma}, \nu_r)$ *is observable.*

Proof: Let $M = (Q, \Sigma, \delta, q_0, \tau)$ be reachable. Then, for each $q \in Q$ there exists a $\gamma \in 2^\Sigma$ such that $\delta(q_0, \gamma) = q$. Also, to show the observability map $\sigma_{Mr} : Q_r \to F(2^\Sigma)$ is one-one, we assume $\tau_\alpha, \tau_\beta \in Q_r$ such that $\tau_\alpha \neq \tau_\beta$, that is, for all $q \in Q$,

$\tau_\alpha(q) \neq \tau_\beta(q)$. Then, for all $\gamma \in 2^\Sigma$

$f_{\tau\alpha}(\gamma) = \nu_r(\delta_r(\tau_\alpha, \gamma)$

$= \nu r(\tau\gamma\oplus\alpha)$

$= \tau\gamma\oplus\alpha(q0)$

$= \tau(\delta(q_0, \gamma \oplus \alpha)$

$= \tau(\delta(\delta(q_0, \gamma), \alpha))$

$= \tau(\delta(q, \alpha)) = \tau_\alpha(q)$,

whereby $f_{\tau\alpha}(\gamma) = \tau_\alpha(q)$. Similarly, $f_{\tau\beta}(\gamma) = \tau_\beta(q)$. Thus, for $\tau_\alpha(q) \neq \tau_\beta(q)$, we have $f_{\tau\alpha}(\gamma) \neq f_{\tau\beta}(\gamma)$. Hence M_r is observable.

The following is the Brzozowski's algorithm for obtaining the minimal *CDFMFA* accepting the fuzzy multiset language f for a given *CDFMFA* accepting a fuzzy multiset language f.

Proposition 6.3 *Given a crisp-deterministic fuzzy multiset finite automaton accepting a fuzzy multiset language* f*:*

(i) apply the reverse construction;

(ii) take the reachable part;

(iii) again, apply the reverse construction; and

(iv) again, take the reachable part.

The resulting crisp-deterministic fuzzy multiset finite automaton is a minimal crisp-deterministic fuzzy multiset finite automata accepting f.

Proof: The crisp-deterministic fuzzy multiset automaton obtain after step (ii) is reachable and accepts f. After step (iii), the achieved crisp-deterministic fuzzy multiset automaton is observable (cf., Proposition 6.2) and accepts $\bar{\bar{f}} = f$. Finally, the crisp-deterministic fuzzy multiset automaton obtained after step (iv) is both reachable and observable, and hence minimal.

Conclusion

In this paper, we tried to correlate two crisp deterministic fuzzy multiset finite automata and showed that the reachability and observability maps of the given crisp deterministic fuzzy multiset finite automaton are isomorphic. We introduced and studied the concept of the reverse crisp deterministic fuzzy multiset finite automaton based on a given crisp deterministic fuzzy multiset finite automaton. We have shown that if the given crisp deterministic fuzzy multiset finite automaton accepts a fuzzy multiset language, the reverse crisp deterministic fuzzy multiset finite automaton accepts the reverse fuzzy multiset language. Finally, we introduced Brzozowski's algorithm for crisp deterministic fuzzy multiset finite automata.

Ethics approval

Not required.

Acknowledgments

No Acknowledgements.

Competing interests

All the authors declare that there are no conflicts of interest.

Funding

This study received no external funding.

Underlying data

Derived data supporting the findings of this study are available in article

CRediT authorship contribution statement

B. K. Sharma: Conceptualization, Methodology, Formal analysis, Investigation, Supervision, Data curation, Writing – original draft, Project administration. Apostolos Syropoulos: Methodology, Validation, Data curation, Writing – review & editing, Visualization, Funding acquisition.

Declaration of artificial intelligence use

Artificial intelligence (AI-Gemini)-assisted technologies were employed exclusively to enhance the clarity and grammatical accuracy of the text during the drafting phase. The author conducted a rigorous critical evaluation of all resulting outputs, affirming that the conceptual framework, data interpretation, and scholarly conclusions remain entirely original and independent.

How to cite

Sharma, B. K., & Syropoulos, A. (2026). DFA Minimization of Fuzzy Multiset Automata. *Journal of Analytical Uncertainty*, *2*(1), 90–98. https://doi.org/10.63924/jau.v2i1.363

References

[1] R. Bělohlávek. "Determinism and fuzzy automata". In: Information Sciences 143 (2002), pp. 205–209.
[2] W.D. Blizard. "Multiset Theory". In: Notre Dame Journal of Formal Logic 30 (1989), pp. 36–66.
[3] J. A. Brzozowski. "Canonical regular expression and minimal state graphs for definite events". In: Mathematical Theory of automata, MRI Symposia Series, Polytechnic Institute of Brooklyn, NY 12 (1962), pp. 529–561.
[4] V. Cerf, E. Fernández, K. Gostelow, and S. Volansky. Formal Control-Flow Properties of a Model of Computation. Tech. rep. Report ENG. 7178. Los Angeles: Department of Computer Science, University of California, 1971.
[5] G. Ciobanu and M. Gontineac. "Mealy Multiset Automata". In: International Journal of Foundations of Computer Science 17 (2006), pp. 111–126.

[6] E. Csuhaj-Varjú, C. Martín-Vide, and V. Mitrana. “Multiset Automata”. In: Multiset processing-mathematical, computer science and molecular computing points of view. Ed. by Calude C.S., G. Păun, G. Rozenberg, and A. Salomaa. Berlin: Springer, 2001, pp. 69–83.
[7] J. E. Hopcroft, R. Motwani, and J. D. Ullman. Introduction to Automata Theory, Languages, and Computation. 3rd. London: Pearson Education Ltd, 2014.
[8] J. Ignjatović, M. Ćirić, and S. Bogdanović. “Determinization of fuzzy automata with membership values in complete residuated lattices”. In: Information Sciences 178 (2008), pp. 164–180.
[9] Z. Jančić and M. Ćirić. “Brzozowski type determinization for fuzzy automata”. In: Fuzzy Sets and Systems 249 (2014), pp. 73–82.
[10] M. Kudlek, C. Martín-Vide, and G. Păun. “Toward a formal macroset theory”. In: Multiset processing-mathematical, computer science and molecular computing points of view. Ed. by Calude C. S., G. Păun, G. Rozenberg, and A. Salomaa. Berlin: Springer, 2001, pp. 123–133.
[11] Y. Li. “Approximation and robustness of fuzzy finite automata”. In: International Journal of Approximate Reasoning 47.2 (2008), pp. 247–257.
[12] Y. Li and W. Pedrycz. “Minimization of lattice finite automata and its application to the decomposition of lattice languages”. In: Fuzzy Sets and Systems 158 (2007), pp. 1423–1436.
[13] Z. Lia, P. Lia, and Y. Li. “The relationships among several types of fuzzy automata”. In: Information Sciences 176.15 (2006), pp. 2208–2226.
[14] W. Lihua and D. Qiu. “Automata theory based on complete residuated lattice-valued logic: Reduction and minimization”. In: Fuzzy Sets and Systems 161 (2010), pp. 1635–1656.
[15] D. S. Malik and J. N. Mordeson. “On fuzzy regular languages”. In: Information Science 88 (1996), pp. 263–273.
[16] D. S. Malik and J. N. Mordeson. Fuzzy discrete structures. New York: Physica Verlag, 2000.
[17] D. S. Malik, J. N. Mordeson, and M. K. Sen. “Submachines of fuzzy finite state machine”. In: Journal of Fuzzy Mathematics 2 (1994), pp. 781–792.
[18] J. N. Mordeson and D. S. Malik. Fuzzy automata and languages, theory and applications. London: Champan and Hall/CRC, 2002.
[19] Tiwari S. P., V. Gautam, and M. K. Dubey. “On fuzzy multiset automata”. In: Journal of Applied Mathematics and Computing 51 (2016), pp. 643–657.
[20] J. Peterson. “Computation sequence sets”. In: Journal of Computer System Science 13 (1976), pp. 1–24.
[21] D. W. Qiu. “Supervisory control of fuzzy discrete event systems: a formal approach”. In: Transactions on Systems, Man, and Cybernetics: Part B (Cybernetics) 35.1 (2005), pp. 72–88.
[22] A. Salomaa. Theory of Automata. Oxford: Pergamon Press, 1969.
[23] E. S. Santos. “Maximin automata”. In: Information and Control 13.4 (1968), pp. 363–377.
[24] A. Syropoulos. “Mathematics of Multisets”. In: Multiset Processing. Ed. by Cristian S. Calude, Gheorghe Păun, Grzegorz Rozenberg, and Arto Salomaa. Lecture Notes in Computer Science 2235. Berlin: Springer-Verlag, 2001, pp. 347–358.
[25] J. Wang, M. Yin, and W. Gu. “Fuzzy multiset finite automata and their languages”. In: Soft Computing 17 (2013), pp. 381–390.
[26] W. G. Wee. “On generalizations of adaptive algorithm and application of the fuzzy sets concepts to pattern classification”. PhD thesis. Purdue University, 1967.
[27] W. G. Wee and K. S. Fu. “A formulation of fuzzy automata and its application as a model of learning systems”. In: Systems Science and Cybernetics, IEEE Transactions 5 (1969), pp. 215–223.
[28] R. R. Yager. “On the theory of bags”. In: International Journal of General Systems 13 (1986), pp. 23–37.
[29] S. Yu. “Regular Languages”. In: Handbook of Formal Languages. Ed. By Grzegorz Rozenberg and Arto Salomaa. Berlin: Springer, 1997, pp. 41–110.
[30] L.A. Zadeh. “Fuzzy Sets”. In: Information and Control 8 (1965), pp. 338–353.