PERGAMON

Applied Mathematics Letters 16 (2003) 985–991

Applied Mathematics Letters

www.elsevier.com/locate/aml

A Language Measure for Supervisory Control

A. RAY* AND A. SURANA
Mechanical Engineering Department
The Pennsylvania State University
University Park, PA 16802, U.S.A.
axr2@psu.edu

S. PHOHA
Applied Research Laboratory
The Pennsylvania State University
University Park, PA 16802, U.S.A.

(Received July 2001; revised and accepted February 2003)

Abstract—This paper formulates a signed real measure for sublanguages of regular languages based on the principles of automata theory and real analysis. The measure provides total ordering on the controlled behavior of a finite-state automaton plant under different supervisors. Total variation of the measure serves as a metric for the infinite-dimensional vector space of the sublanguages of a regular language over the finite field GF(2). The computational complexity of the language measure is of polynomial order in the number of plant states.

Keywords—Discrete-event systems, finite-state automata, Formal languages, Measures.

1. INTRODUCTION

The controlled behavior of a finite-state automaton plant could vary under different supervisors if they are designed based on different control specifications. As such, the respective controlled sublanguages of the plant language form a partially ordered set that is not necessarily totally ordered. Since the literature on discrete event system (DES) control does not apparently provide a measure of the plant sublanguages, it may not be possible to quantitatively evaluate the performance of a DES supervisor. Therefore, it is necessary to formulate a mathematically rigorous concept of language measure to quantify performance of individual supervisors such that the measures of partially ordered controlled sublanguages can be structured to form a totally ordered set. From this perspective, the goal of the paper is to construct a signed real measure that can be assigned to any sublanguage of the uncontrolled regular language of the plant to achieve the following objective: given that the relation $\subseteq$ induces a partial ordering on a set of controlled sublanguages $\{L_k\}$ of a regular plant language $\mathcal{L}$, the language measure μ induces a total ordering $\leq$ on $\{\mu(L_k)\}$. That is, the range of the set function μ is totally ordered while its domain could be only partially ordered.

This work was supported in part by Army Research Office (ARO) under Grant No. DAAD19-01-1-0646 and NASA Glenn Research Center under Grant No. NAG3-2448.

*Author to whom all correspondence should be addressed.

 Typeset by $\mathcal{A}_\mathcal{M}\mathcal{S}$-TEX

doi: 10.1016/S0893-9659(03)00133-2

2. CONCEPT OF THE LANGUAGE MEASURE

Let $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$ be a trim (i.e., accessible and coaccessible) deterministic finite-state automaton (DFSA) that represents the discrete-event dynamics of a physical plant [1] where $Q = \{q_1, q_2, \ldots, q_n\}$ is the set of states with q_i being the initial state; $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_m\}$ is the alphabet of events; $Q_m \subseteq Q$ is the (nonempty) set of marked (i.e., accepted) states; $\delta : Q \times \Sigma \to Q$ is the (possibly partial) function of state transitions and $\delta^* : Q \times \Sigma^* \to Q$ is an extension of δ. The (countable) set Σ^* is the Kleene closure of Σ, i.e., the set of all (finite-length) strings made of the events belonging to Σ including the empty string ε that is viewed as the identity element of the monoid Σ^* under the operation of string concatenation; i.e., $\varepsilon s = s = s\varepsilon$, $\forall\, s \in \Sigma^*$.

DEFINITION 1. *A σ-algebra $\mathcal{M}$ of a language $\mathcal{L} \subseteq \Sigma^*$ is a collection of subsets of $\mathcal{L}$ which satisfies the following three conditions:*

(i) $\mathcal{L} \in \mathcal{M}$;
(ii) *if* $L \in \mathcal{M}$, *then* $(\mathcal{L} - L) \in \mathcal{M}$;
(iii) $\bigcup_{k=1}^{\infty} L_k \in \mathcal{M}$ *if* $L_k \in \mathcal{M}$, $\forall\, k$.

DEFINITION 2. *Let $\mathcal{M}$ be a σ-algebra. An at most countable collection $\{L_k\}$ of members of $\mathcal{M}$ is a partition of a member $L \in \mathcal{M}$ if $L = \bigcup_k L_k$ and $L_i \cap L_j = \varnothing$, $\forall\, i \neq j$.*

DEFINITION 3. *Given a σ-algebra $\mathcal{M}$ of a language $\mathcal{L}$, the set function $\mu : \mathcal{M} \to \Re \equiv (-\infty, \infty)$ is called a signed real measure if the following two conditions are satisfied:*

(i) $\mu(\varnothing) = 0$;
(ii) $\mu(\bigcup_{k=1}^{\infty} L_k) = \sum_{k=1}^{\infty} \mu(L_k)$ *for every partition $\{L_k\}$ of any member $L \in \mathcal{M}$.*

DEFINITION 4. *Total variation measure $|\mu|$ on a σ-algebra $\mathcal{M}$ is defined as $|\mu|(L) = \mathrm{Sup} \sum_k |\mu(L_k)|$, $\forall\, L \subseteq \mathcal{M}$ where the supremum is taken over all partitions $\{L_k\}$ of L.*

DEFINITION 5. *Relative to measure μ, a sublanguage $L \in \mathcal{M}$ is defined to be:*

(i) *null,* $L = 0$, *if* $\mu(L \cap J) = 0$, $\forall\, J \in \mathcal{M}$;
(ii) *positive,* $L > 0$, *if* $\mu(L \cap J) > 0$, $\forall\, J \in \mathcal{M}$;
(iii) *negative,* $L < 0$, *if* $\mu(L \cap J) < 0$, $\forall\, J \in \mathcal{M}$.

REMARK 1. Following standard theorems on complex measures [2], total variation measure $|\mu|$ of a regular language $\mathcal{L}$ is nonnegative and finite, i.e., $|\mu|(\mathcal{L}) \in [0, \infty)$. Hence, $|\mu|(L) \in [0, \infty)$, $\forall\, L \in \mathcal{M}$. ∎

REMARK 2. Based on the Hahn decomposition theorem [2], every sublanguage $L \in \mathcal{M}$ can be partitioned as $L = L^0 \cup L^+ \cup L^-$ where mutually exclusive sublanguages L^0, L^+, and L^- are null, positive, and negative, respectively, relative to a signed real measure μ. ∎

3. FORMULATION OF THE LANGUAGE MEASURE

For a given DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, we now construct a σ-algebra $\mathcal{M}$ as the power set $2^{\mathcal{L}(\mathcal{G}_i)}$ of the regular language $\mathcal{L}(\mathcal{G}_i)$.

PROPOSITION 1. *Total variation measure $|\mu|$ on the σ-algebra $2^{\mathcal{L}(\mathcal{G}_i)}$ is $|\mu|(K) = \sum_{s \in K} |\mu(\{s\})|$, $\forall\, K \subseteq \mathcal{L}(\mathcal{G}_i)$.*

PROOF OF PROPOSITION 1. The proof follows from Definition 4, based on the facts that $\mathcal{L}(\mathcal{G}_i) \subseteq \Sigma^*$ is at most countable and that every singleton set of a legal string belongs to $2^{\mathcal{L}(\mathcal{G}_i)}$. ∎

The marked (i.e., accepted) language $\mathcal{L}_m(\mathcal{G}_i)$ of a trim DFSA $\mathcal{G}_i$ has the following properties: $\varnothing \subset \mathcal{L}_m(\mathcal{G}_i) \subseteq \mathcal{L}(\mathcal{G}_i)$; and $\mathcal{L}_m(\mathcal{G}_i) = \mathcal{L}(\mathcal{G}_i)$ iff $Q_m = Q$. Let the marked states be designated as $Q_m \equiv \{q_{m_1}, q_{m_2}, \ldots, q_{m_\ell}\} \subseteq Q$ where $q_{m_k} = q_j$ for some $j \in \{1, 2, \ldots, n\}$.

DEFINITION 6. *For a state $q \in Q$ of a given DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, the regular language $L(q_i, q)$ is defined to be the set of all strings that, starting from the initial state q_i, terminate at the state q. Equivalently, $L(q_i, q)$ is the sublanguage of all legal event strings terminating at q starting from q_i.*

The Myhill-Nerode theorem is now applied to construct the following state-based partitions [3,4]: $\mathcal{L}(\mathcal{G}_i) = \bigcup_{q \in Q} L(q_i, q)$ and $\mathcal{L}_m(\mathcal{G}_i) = \bigcup_{q \in Q_m} L(q_i, q)$ where the sublanguage $L(q_i, q_k)$ of all legal event strings starting at the initial state q_i is uniquely labeled by the terminal state q_k, $\forall\, k \in \{1, 2, \dots, n\}$.

In order to obtain a quantitative measure of the marked language $\mathcal{L}_m(\mathcal{G}_i)$, the set of marked states is partitioned as $Q_m = Q_m^+ \cup Q_m^-$ and $Q_m^+ \cap Q_m^- = \varnothing$. The positive set Q_m^+ contains *good* marked states that we desire to reach, and the negative set Q_m^- contains *bad* marked states that we want to avoid, although it may not always be possible to completely avoid the bad states while attempting to reach the good states. In general, the marked language $\mathcal{L}_m(\mathcal{G}_i)$ consists of both good and bad event strings that, starting from the initial state q_i, respectively, lead to Q_m^+ and Q_m^-. Any event string belonging to the language $\mathcal{L}(\mathcal{G}_i) - \mathcal{L}_m(\mathcal{G}_i)$ leads to one of the nonmarked states belonging to $(Q - Q_m)$ and does not contain any one of the good or bad strings.

In view of Definition 5, we proceed to construct a signed real measure $\mu : 2^{\mathcal{L}(\mathcal{G}_i)} \to \Re \equiv (-\infty, \infty)$ to allow state-based decomposition of $\mathcal{L}(\mathcal{G}_i)$ into null, positive, and negative sublanguages such that

(i) $\mu(L(q_i, q)) = 0$, $\forall\, q \notin Q_m$, i.e., a legal event string starting at the initial state q_i and terminating on any nonmarked state has zero measure;
(ii) partitioning of Q_m into Q_m^+ and Q_m^- yields the properties: $\mu(L(q_i, q)) > 0$, $\forall\, q \in Q_m^+$, and $\mu(L(q_i, q)) < 0$, $\forall\, q \in Q_m^-$, which is in agreement with Remark 2 in the sense that $L^0(\mathcal{G}_i) = \bigcup_{q \notin Q_m} L(q_i, q)$; $L_m^+(\mathcal{G}_i) = \bigcup_{q \in Q_m^+} L(q_i, q)$; and $L_m^-(\mathcal{G}_i) = \bigcup_{q \in Q_m^-} L(q_i, q)$.

Partitioning the marked language $\mathcal{L}_m(\mathcal{G}_i)$ into a positive language $L_m^+(\mathcal{G}_i)$ and a negative language $L_m^-(\mathcal{G}_i)$ is equivalent to partitioning Q_m into the positive set Q_m^+ and the negative set Q_m^-. Each state belonging to Q_m^+ is characterized by a positive weight and each state belonging to Q_m^- by a negative weight. These weights are chosen by the designer based on his/her perception of each marked state's role in the system performance.

DEFINITION 7. *The characteristic function $\chi : Q \to [-1, 1]$ that assigns a signed real weight to an event string of $\mathcal{L}(\mathcal{G}_i)$ based on its terminal state q is defined as*

$$\chi(q) \in \begin{cases} [-1, 0), & \text{if } q \in Q_m^-, \\ \{0\}, & \text{if } q \notin Q_m, \\ (0, 1], & \text{if } q \in Q_m^+. \end{cases}$$

Therefore, for any accessible DFSA $\mathcal{G}_i$, the sublanguage $L(q_i, q_k)$ is a nonempty language, $\forall\, k \in \{1, 2, \dots, n\}$. In that case, the implication of the characteristic function is that an event string belonging to a sublanguage $L(q_i, q_k)$, which is labeled by the terminal state q_k, has a zero measure if q_k is not a marked state, a positive measure if q_k is a good marked state, and a negative measure if q_k is a bad marked state.

We now introduce the cost of event strings belonging to $\mathcal{L}(\mathcal{G}_i)$. The cost assignment procedure is conceptually similar to that for state-based conditional probability to events of a string. Since the consecutive events in a string may not be statistically independent, it is necessary to find the joint probability mass functions of arbitrarily large order. This would make the probability space of Σ^* ever expanding because there is no finite upper bound on the length of strings in Σ^*. This problem is circumvented by using the state transition function δ of the DFSA $\mathcal{G}_i$.

DEFINITION 8. *The event cost generated at a DFSA state is defined as $\tilde{\pi} : \Sigma^* \times Q \to [0, 1)$ such that $\forall\, q_j \in Q$, $\forall\, \sigma, \sigma_k \in \Sigma$, $\forall\, s \in \Sigma^*$,*

- $\tilde{\pi}[\sigma_k, q_j] \equiv \tilde{\pi}_{jk} \in [0, 1)$; $\sum_{j=1}^{m} \tilde{\pi}_{jk} < 1$;

- $\tilde{\pi}[\sigma, q_j] = 0$ *if* $\delta(q_j, \sigma)$ *is undefined;* $\tilde{\pi}[\varepsilon \mid q_k] = 1$;
- $\tilde{\pi}[\sigma s, q_j] = \tilde{\pi}[\sigma, q_j]\tilde{\pi}[s, \delta(q_j, \sigma)]$.

The event cost function $\tilde{\pi}$ for an event string $s \in L(q_i, q_k)$ starting from the initial state q_i and terminating at q_k is obtained as the product of respective costs conditioned on the state from which the events are generated. This is conceptually similar to having a product of conditional probabilities. For example, if $s = \sigma_j\sigma_k\sigma_\ell$ for a DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, then $\tilde{\pi}[s, q_i] = \tilde{\pi}_{ij}\tilde{\pi}_{ak}\tilde{\pi}_{b\ell}$ where the state transition function δ defines the (Markov) states $q_a = \delta(q_i, \sigma_j)$ and $q_b = \delta(q_a, \sigma_k)$.

DEFINITION 9. *The signed measure* μ *of every singleton set member of* $2^{\mathcal{L}(\mathcal{G}_i)}$ *is defined as* $\mu(\{s\}) \equiv \tilde{\pi}[s, q_i]\chi(q)$, *where* $s \in L(q_i, q)$.

Definitions 7–9 imply that, for an event string s belonging to an accessible language $\mathcal{L}(\mathcal{G}_i)$,

$$\mu(\{s\}) \begin{cases} = 0, & \text{if } s \in L(q_i, q) \text{ for } q \notin Q_m, \\ > 0, & \text{if } s \in L(q_i, q) \text{ for } q \in Q_m^+, \\ < 0, & \text{if } s \in L(q_i, q) \text{ for } q \in Q_m^-. \end{cases}$$

Therefore, the signed measure μ can be assigned to each event string belonging to $s \in \mathcal{L}(\mathcal{G}_i)$ that is partitioned by the sublanguages $L(q_i, q_k)$, $k \in \{1, 2, \ldots, n\}$, in terms of the nonnegative cost $\tilde{\pi}$ and the signed characteristic function χ.

DEFINITION 10. *Given a DFSA* $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, *the cost* ν *of a sublanguage* $K \subseteq \mathcal{L}(\mathcal{G}_i)$ *is defined as the sum of the event cost* $\tilde{\pi}$ *of individual strings belonging to* K,

$$\nu(K) = \sum_{s \in K} \tilde{\pi}[s, q_i],$$

and the signed measure of $K \subseteq \mathcal{L}(\mathcal{G}_i)$ *is defined as the sum of the signed measures of equivalence classes of* K *as*

$$\mu(K) = \sum_j \nu(L(q_i, q_j) \cap K)\chi(q_j).$$

In view of Remark 1, Definition 10 assigns a total variation measure $|\mu|$ to each event string $s \in \mathcal{L}(\mathcal{G}_i)$, and hence, to every sublanguage $K \subseteq \mathcal{L}(\mathcal{G}_i)$ as $|\mu|(K) = \sum_j \nu(L(q_i, q_j) \cap K)|\chi(q_j)|$.

4. CONVERGENCE OF THE LANGUAGE MEASURE

The previous section formulated the real signed measure μ based on Definition 10. This section establishes convergence of the measure μ in view of Remark 1 and Proposition 1 by showing that $|\mu|(L(q_i, q_k)) < \infty$, $\forall\, q_i, q_k \in Q$, which is equivalent to $\mu(\mathcal{L}_m(\mathcal{G}_i)) = \mu(\mathcal{L}(\mathcal{G}_i)) \le |\mu|(\mathcal{L}(\mathcal{G}_i)) < \infty$.

The following definitions and propositions are introduced to compute $\mu(L)$ and $|\mu|(L)$ for any $L \subseteq \mathcal{L}(\mathcal{G}_i)$ and establish the convergence.

DEFINITION 11. *Given* $q_i, q_k \in Q$, *a nonempty string* p *of events (i.e.,* $p \neq \varepsilon$*) starting from* q_i *and terminating at* q_k *is called a path. A path* p *from* q_i *to* q_k *is said to pass through* q_j *if* $\exists$ *strings* $s \neq \varepsilon$ *and* $t \neq \varepsilon$ *such that* $p = st$; $\delta^*(q_i, s) = q_j$ *and* $\delta^*(q_j, t) = q_k$ *where* $\delta^* : Q \times \Sigma^* \to Q$.

DEFINITION 12. *A path language* p_{ik}^j *is defined to be the set of all paths from* q_i *to* q_k, *which do not pass through any state* q_ℓ *for* $\ell > j$. *The path language* p_{ik} *is defined to be the set of all paths from* q_i *to* q_k. *Thus, the language* $L(q_i, q_k)$ *is obtained in terms of the path language* p_{ik} *as*

$$L(q_i, q_k) = \begin{cases} p_{ii} \cup \{\varepsilon\}, & \text{if } k = i, \\ p_{ik}, & \text{if } k \neq i, \end{cases} \quad \Rightarrow \quad \nu(L(q_i, q_k)) = \begin{cases} \nu(p_{ii}) + 1, & \text{if } k = i, \\ \nu(p_{ik}), & \text{if } k \neq i. \end{cases}$$

Based on the above definitions, we present the following propositions and lemmas to quantify the language measure μ.

REMARK 3. Let $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$ be a DFSA with $|Q| = n$. Then, $p_{ik}^j = p_{ik}$, $\forall j \geq n$, because no string passes through a state numbered higher than n. ∎

REMARK 4. Every path language is regular for a DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$. Since p_{ik}^0 is a finite language and hence regular, it follows from the proof of Kleene's theorem [4, p. 123] by the induction hypothesis that p_{ik}^{j+1} is regular if p_{ik}^j is regular for all $1 \leq j \leq n$. ∎

PROPOSITION 2. *For a given DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, the following recursive relation holds for $0 \leq \ell \leq n-1$: $p_{ik}^0 = \{\sigma \in \Sigma : \delta(q_i, \sigma) = q_k\}$ and $p_{ik}^{\ell+1} = p_{ik}^\ell \cup p_{i,\ell+1}^\ell (p_{\ell+1,\ell+1}^\ell)^* p_{\ell+1,k}^\ell$.*

PROOF OF PROPOSITION 2. The proof is given by Martin [4, p. 124]. ∎

PROPOSITION 3. *The following recursive relations hold for $0 \leq \ell \leq n-1$:*

$$\nu\left(p_{ik}^{\ell+1}\right) = \nu\left(p_{ik}^\ell\right) + \frac{\nu\left(p_{i,\ell+1}^\ell\right)\nu\left(p_{\ell+1,k}^\ell\right)}{1-\nu\left(p_{\ell+1,\ell+1}^\ell\right)} \in [0, \infty).$$

PROOF OF PROPOSITION 3. We need three lemmas to prove the proposition.

LEMMA 1. $\nu((p_{kk}^0)^* \bigcup_{j \neq k} p_{kj}^0) \in [0, 1)$.

PROOF OF LEMMA 1. Following Definitions 8 and 10, $\nu(p_{kk}^0) \in [0, 1)$. Therefore, by convergence of a geometric series,

$$\nu\left(\left(p_{kk}^0\right)^* \bigcup_{j \neq k} p_{kj}^0\right) = \frac{\sum_{j \neq k} \nu\left(p_{kj}^0\right)}{1-\nu\left(p_{kk}^0\right)} \in [0, 1)$$

because

$$\sum_j \nu\left(p_{kj}^0\right) < 1 \Rightarrow \sum_{j \neq k} \nu\left(p_{kj}^0\right) < 1 - \nu\left(p_{kk}^0\right).$$ ∎

LEMMA 2. $\nu(p_{j+1,j+1}^j) \in [0, 1)$.

PROOF OF LEMMA 2. The path $p_{j+1,j+1}^j$ may contain at most j loops, one around the states $q_1, q_2, \ldots, q_j$. If the path $p_{j+1,j+1}^j$ does not contain any loop, then $\nu(p_{j+1,j+1}^j) \in [0, 1)$ because it is a product of $\tilde{\pi}_{ik}$s, each of which is a nonnegative fraction. Next, suppose there is a loop around q_ℓ that does not contain any other loop; this loop must be followed by one or more events σ_k generated at q_ℓ and leading to some other states q_m where $m \in \{1, \ldots, j+1\}$ and $m \neq \ell$. By Lemma 1, $\nu(p_{j+1,j+1}^j) \in [0, 1)$. The proof follows by starting from the innermost loop and ending with all loops at q_j. ∎

LEMMA 3. $\nu((p_{j+1,j+1}^j)^*) \in [1, \infty)$.

PROOF OF LEMMA 3. Since $\nu(p_{j+1,j+1}^j) \in [0, 1)$ from Lemma 2,

$$\nu\left(\left(p_{j+1,j+1}^j\right)^*\right) = \frac{1}{1-\nu\left(p_{j+1,j+1}^j\right)} \in [1, \infty).$$ ∎

Now we proceed to prove Proposition 3. Since the languages $p_{j,\ell+1}^\ell$, $p_{\ell+1,\ell+1}^\ell$, and $p_{\ell+1,k}^\ell$ are mutually disjoint, it follows from Proposition 2 that $\nu(p_{ik}^{\ell+1}) = \nu(p_{ik}^\ell) + \nu(p_{i,\ell+1}^\ell)\nu((p_{\ell+1,\ell+1}^\ell)^*) \times \nu(p_{\ell+1,k}^\ell)$. The proof follows by applying Lemmas 1–3 to the above expression. ∎

COROLLARY TO PROPOSITION 3. *For a given DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$, the measure and total variation of every sublanguage $K \subseteq \mathcal{L}(\mathcal{G}_i)$ are finite. Specifically, $|\mu(K)| \leq |\mu|(K) < \infty$.*

PROOF OF COROLLARY TO PROPOSITION 3. The proof follows from Propositions 1 and 3 using Definitions 10 and 12. ∎

REMARK 5. In view of Definition 12 and Proposition 3, the algorithm for numerically solving $\nu(p_{ik})$ requires three nested for-loops. Hence, for an n-state automaton, the computation time for the language measure in Definition 10 is in the order of n^3. ∎

5. VECTOR SPACE OF FORMAL LANGUAGES

This section makes use of the language measure to construct a metric space of sublanguages of a regular language representing the DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$ where the total variation measure $|\mu|$ induces a metric on this space, which quantifies the distance function between any two sublanguages of $\mathcal{L}(\mathcal{G}_i)$.

PROPOSITION 4. *Let $\mathcal{L}(\mathcal{G}_i)$ be the language of a DFSA $\mathcal{G}_i \equiv \langle Q, \Sigma, \delta, q_i, Q_m \rangle$. Let the binary operation of symmetric difference $\oplus : 2^{\mathcal{L}(\mathcal{G}_i)} \times 2^{\mathcal{L}(\mathcal{G}_i)} \to 2^{\mathcal{L}(\mathcal{G}_i)}$ be defined as $(K_1 \oplus K_2) \equiv (K_1 \cup K_2) - (K_1 \cap K_2)$, $\forall K_1, K_2 \subseteq \mathcal{L}(\mathcal{G}_i)$. Then, $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, \oplus \rangle$ is a vector space over the Galois field* GF(2).

PROOF OF PROPOSITION 4. We notice that $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, \oplus \rangle$ is an Abelian group where $\varnothing$ is the zero element of the group and the unique inverse of every element $K \in 2^{\mathcal{L}(\mathcal{G}_i)}$ is K itself because $K_1 \oplus K_2 = \varnothing$ if and only if $K_1 = K_2$. The associative and distributive properties of the vector space follows by defining the scalar multiplication of vectors as $0 \otimes K \equiv \varnothing$ and $1 \otimes K \equiv K$. ∎

REMARK 6. The collection of singleton languages made from each element of $\mathcal{L}(\mathcal{G}_i)$ forms a basis set of the vector space $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, \oplus \rangle$ over GF(2). Thus, $\mathcal{L}(\mathcal{G}_i)$ is bijective to any basis set of $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, \oplus \rangle$. ∎

DEFINITION 13. *Let $\mathcal{L}(\mathcal{G}_i)$ be the regular language for a DFSA $\mathcal{G}_i$. The distance function $d : 2^{\mathcal{L}(\mathcal{G}_i)} \times 2^{\mathcal{L}(\mathcal{G}_i)} \to [0, \infty)$ is defined in terms of the total variation measure $|\mu|$ as*

$$\forall K_1, K_2 \subseteq \mathcal{L}(\mathcal{G}_i), \qquad d(K_1, K_2) = |\mu| (K_1 \oplus K_2) = |\mu| ((K_1 \cup K_2) - (K_1 \cap K_2)).$$

The above distance function $d(\bullet, \bullet)$ quantifies the difference between two supervisors relative to the controlled performance of the DFSA plant.

PROPOSITION 5. *The distance function $d : 2^{\mathcal{L}(\mathcal{G}_i)} \times 2^{\mathcal{L}(\mathcal{G}_i)} \to [0, \infty)$ is a pseudometric on the space $2^{\mathcal{L}(\mathcal{G}_i)}$.*

PROOF OF PROPOSITION 5. We notice that $\forall K_1, K_2 \in 2^{\mathcal{L}(\mathcal{G}_i)}$, $d(K_1, K_2) = |\mu|(K_1 \oplus K_2) \geq 0$, and $d(K_1, K_2) = d(K_2, K_1)$. The remaining property of the triangular inequality follows from the inequality $|\mu|(K_1 \oplus K_2) \leq |\mu|(K_1) + |\mu|(K_2)$ because $(K_1 \oplus K_2) \subseteq (K_1 \cup K_2)$ and $|\mu|(K_1) \leq |\mu|(K_2)$, $\forall K_1 \subset K_2$. ∎

REMARK 7. The pseudometric $|\mu| : 2^{\mathcal{L}(\mathcal{G}_i)} \to [0, \infty)$ can be converted to a metric of the space $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, \oplus \rangle$ by clustering all languages that have zero total variation measure as the null equivalence class $\mathcal{N} \equiv \{K \in 2^{\mathcal{L}(\mathcal{G}_i)} : |\mu|(K) = 0\}$. This procedure is conceptually similar to what is done for defining norms in the L_p spaces [2]. In that case, $\mathcal{N}$ contains all sublanguages of $\mathcal{L}(\mathcal{G}_i)$, which terminate on nonmarked states starting from the initial state; i.e., $\mathcal{N} = \{\varnothing\} \cup (\bigcup_{q \notin Q_m} L(q, q_i))$. Therefore, $|\mu|(\bullet)$ can be treated as a metric of the space $2^{\mathcal{L}(\mathcal{G}_i)}$ and can be generated from the distance function $d(\bullet, \bullet)$ in Definition 13 as $|\mu|(K) = d(K, J)$, $\forall K \in 2^{\mathcal{L}(\mathcal{G})}$, $\forall J \in \mathcal{N}$. ∎

REMARK 8. The metric space $\langle 2^{\mathcal{L}(\mathcal{G}_i)}, d \rangle$ can be modified by augmenting the state set Q with the additional dump state q_{n+1}. In that case, the state transition function δ becomes a total function with $\chi(q_{n+1}) = 0$ following Definition 8. As the domain of the language measure μ is extended from $2^{\mathcal{L}(\mathcal{G}_i)}$ to 2^{Σ^*}, nonzero values of μ remain unchanged and the null equivalence class is expanded as $\mathcal{N} = \{K \in 2^{\Sigma^*} : |\mu|(K) = 0\}$. ∎

6. SUMMARY AND CONCLUSIONS

This paper presents a signed real measure of formal languages, which is based on an event cost matrix and a characteristic vector. The language measure provides a tool for performance analysis and comparison of the unsupervised plant automaton and supervised plant automata. The total variation of the language measure induces a metric on the vector space of sublanguages of the regular language, which is defined over the Galois field GF(2).

The language measure μ can be constructed from the perspectives of mission objectives for design and analysis of controlled plant automata under different DES supervisors. To this end, the role of the language measure is explained below.

A discrete-event nonmarking supervisor $\mathcal{S}$ restricts the marked behavior of an uncontrolled plant $\mathcal{G}_i$ such that $\mathcal{L}_m(\mathcal{S}/\mathcal{G}_i) \subseteq \mathcal{L}_m(\mathcal{G}_i)$. The uncontrolled marked language $\mathcal{L}_m(\mathcal{G}_i)$ consists of good strings leading to Q_m^+ and bad strings leading to Q_m^-. A controlled language $\mathcal{L}_m(\mathcal{S}/\mathcal{G}_i)$ based on a given specification of the supervisor $\mathcal{S}$ may disable some of the bad strings and retain some of the good strings enabled as much as possible. Different supervisors $\mathcal{S}_j : j \in \{1, 2, \ldots, n_s\}$ for a DFSA $\mathcal{G}_i$ achieve this goal in different ways and generate a partially ordered set of controlled sublanguages $\{\mathcal{L}_m(\mathcal{S}_j/\mathcal{G}_i) : j \in \{1, 2, \ldots, n_s\}\}$. The real signed measure μ provides a precise quantitative comparison of the controlled plant behavior under different supervisors because the set $\{\mu(\mathcal{L}_m(\mathcal{S}_j/\mathcal{G}_i)) : j \in \{1, 2, \ldots, n_s\}\}$ is totally ordered.

Computational complexity of the language measure is polynomial in the number of states of the deterministic finite-state automaton that is a minimal realization of the regular language.

REFERENCES

1. P.J. Ramadge and W.M. Wonham, Supervisory control of a class of discrete event processes, *SIAM Journal of Control and Optimization* **25** (1), 206–230, (1987).
2. W. Rudin, *Real and Complex Analysis*, Third Edition, McGraw-Hill, New York, (1988).
3. H.E. Hopcroft, R. Motwani and J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Second Edition, Addison-Wesley, Reading, MA, (2001).
4. J.C. Martin, *Introduction to Languages and the Theory of Computation*, Second Edition, McGraw-Hill, New York, (1997).
5. V. Drobot, *Formal Languages and Automata Theory*, Computer Science Press, (1989).