Set of Gene Languages is a Semi-Topology

Gene Moo Lee, Hoh Peter In[†]

Department of Computer Science and Engineering
Korea University
1,5-ka, Anam-dong, Sungbuk-ku
Seoul 136-701, Korea
{gm0707, hoh_in}@korea.ac.kr

Abstract
This paper presents an innovative connection from the language and automata theory into topology. Automata called "Gene automata" are proposed by relaxing the constraints of finite set of states and alphabets in regular language into arbitrary set of states and alphabets. Languages called "Gene languages" are proposed as languages accepted by the Gene automata. Semi-topology is a new concept proposed by relaxing a constraint of the definition of topological space. We prove that the set of Gene languages is a semi-topological space on set of strings.

Keywords –Gene languages, Gene automata, semi-topology, closure under set operations.

[†] Hoh Peter In is the corresponding author.

1. Introduction

Theory of automata and language is essential in the theory of computation and modeling of computer systems. Meanwhile, topology theory is useful for studying geometric properties. The connection between these two theories is useful since theories, lemma, and tools of a theory can be applicable for another, or vice versa.

Semi-topology is a new concept proposed by relaxing a constraint of the definition of topological space. In order to be a semi-topology, a set of languages should be closed under countably infinite union. Regular language is closed under finite union and intersection [Sip96], but not closed under countably infinite union. Thus, a new language is proposed by relaxing the constraints of finite set of states and alphabets in regular language. Languages called "Gene languages" are proposed as languages accepted by the Gene automata. We prove that the set of Gene languages is a semi-topological space on set of strings.

The paper is organized as follows: basic knowledge of semi-topology and new concepts in the automata and language theory are reviewed in Section 2. The connection of the two theories is explored in Section 3. Related works and future research challenges are presented in Section 4 and 5.

2. Background

Semi-topology, Gene automata and Gene languages are defined in this section.

2.1. Semi-Topology

The definition of topological space is presented in [Mun99]. A new concept called *semi-topology* is proposed by releasing a condition in the definition of topology.

Definition 1

A *semi-topology* or *semi-topological space* on a set X is a collection T of subsets of X having the following properties

1. An empty set, {}, and X are in T.
2. Union of the elements of *countable* subcollection of T is in T.
3. Intersection of the elements of any finite subcollection of T is in T.

The second property makes a semi-topology to be closed under countably infinite union operation, and the third is for the finite intersection closure. The elements of a semi-topology are called *open sets*. For example, let $\Re$ be the set of real numbers and T be the set of all open intervals. Then T is a semi-topology on $\Re$, which is also called the standard topology.

2.2. Gene Automata and Gene Language

The formal definition of Gene automata and Gene language is given in this subsection. The following concepts are adapted from finite automata and regular language [Sip96].

Definition 2

A Gene automaton M is a 5-tuple (Q, ∑, δ, q_0, F), where

1. Q is a set of states,
2. ∑ is a set of alphabets,
3. δ: Q × ∑ → Q is a transition function,
4. $q_0 \in Q$ is the initial state, and
5. $F \subseteq Q$ is a set of final or accept states

The "finiteness" constraint in the definition of traditional finite automata is relaxed in this definition, and the number of states in the Gene automaton may be infinite.

Definition 3

A language $L \subseteq \sum^*$ is called *a Gene language* if a Gene automaton accepts it.
Let the class of Gene languages be **GL**.

The concept of *accepting* a language is the same as the traditional definition in the theory of computation [Sip96]. That is, a Gene automaton *recognizes* a Gene language. In Section 3, the connection between Gene languages and semi-topological space is discussed.

3. Integration of Semi-Topology, Automata, and Language Theory

All the Gene automata are constructed over the same set of alphabets ∑ for convenience. The definitions of set operations, such as intersection, union, and complement, are given in [Mun99].

We will prove that set of Gene languages, **GL**, is a semi-topology on ∑* by showing that **GL** satisfies three conditions given in Definition 1. First, Lemma 1 and Theorem 1 are to show that GL meets the first condition, that is, closure under countable set union.

Lemma 1

GL is closed under a union operation.

Proof) Let L_1 and L_2 be Gene languages. It is needed to prove that $L_1 \cup L_2$ is also a Gene language. Let $M_1 = (Q_1, \sum, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \sum, \delta_2, q_2, F_2)$ be the Gene automata that accept L_1 and L_2, respectively. Then a Gene automaton M = (Q, ∑, δ, q_0, F) may be constructed where

(1) $Q = Q_1 \times Q_2$ (Cartesian product),

(2) $q_0 = (q_1, q_2)$,

(3) $F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$, and

(4) $\delta : Q \times \sum \rightarrow Q$ defined by $\delta(q_a, q_b) = (\delta_1(q_a), \delta_2(q_b))$.

By its definition, M accepts $L_1 \cup L_2$, so $L_1 \cup L_2$ is also a Gene language.

Hence, **GL** is closed under the union operation. □

Theorem 1

GL is closed under countable union operation.

Proof) If the number of union operations is finite, induction is used for this proof. Let *n* be the number of union operation. First, for the base case (*n* = 2), it is trivial that **GL** is closed under one-time union by Lemma 1.

Next, for the inductive case, suppose that this theorem holds for the case *n* = k. Then if there are k Gene languages $\{L_1, L_2 \ldots L_k\}$, then $\cup^{k}_{i=1} L_i$ is also a Gene language by the hypothesis. By definition of a Gene language, there exists a Gene automaton M that accepts $\cup^{k}_{i=1} L_i$.

Let L' be another Gene language. By Lemma 3, $(\cup^{k}_{i=1} L_i) \cup L'$ is also a Gene language, which shows that this theorem holds for the case *n* = k+1. Hence, **GL** is closed under finite union operation.

For the countably infinite case, let $L = \{L_1, L_2, L_3 \ldots\}$ be a collection of Gene languages. Then $M = \{M_1, M_2, M_3 \ldots\}$ be the collection of Gene automata that accepts languages in L. That is, M_i accepts L_i, where $M_i = (Q_i, \sum, \delta_i, q_i, F_i)$. Now, a new Gene automaton $M = (Q, \sum, \delta, q_0, F)$ can be constructed in the same way of Lemma 1, where

(1) $Q = \Pi^{\infty}_{i=1} Q_i$,

(2) $q_0 = (q_1, q_2, \ldots)$,

(3) $F = \cup^{\infty}_{i=1} (\Pi^{i-1}_{j=1} \times Q_i \times \Pi^{\infty}_{k=i+1})$, and

(4) $\delta : Q \times \sum \rightarrow Q$ defined by $\delta(q_1, q_2, \ldots) = (\delta_1(q_1), \delta_2(q_2), \ldots)$.

Then by its definition, the Gene automaton M accepts $\cup^{\infty}_{i=1} L_i$, and $\cup^{\infty}_{i=1} L_i$ is also a Gene language. In other words, **GL** is closed under countably infinite union operation. Therefore, **GL** is closed under countable union operation. □

Secondly, Lemma 2, 3, and Theorem 2 show that **GL** is closed under set intersection.

Lemma 2

If L is a Gene language, then the complement set, L', is also a Gene language.

Proof) Let L be a Gene language, and $M = (Q, \sum, \delta, q_0, F)$ be the Gene automaton that accepts L. Now M' that will accept L' can be made, i.e., M' does not accept strings or elements in L but accepts strings that are not contained in L.

Now $M' = (Q, \sum, \delta, q_0, F')$ is a Gene automaton where the set of states, set of alphabets, transition function and the initial state are the same as those in M and let $F' = Q - F$.

Then clearly, M’ accepts L’, and L’ is also a Gene language. Hence, **GL** is closed under complement operation. □

Lemma 3

GL is closed under intersection operation.

Proof) Let L_1 and L_2 be Gene languages. It is needed to prove that $L_1 \cap L_2$ is also a Gene language.

Notice that by De Morgan’s law, $L_1 \cap L_2 = \neg (\neg L_1 \cup \neg L_2)$, where $\neg$ represents a complement operation. $\neg L_1$ and $\neg L_2$ are Gene languages by Lemma 2. Thus, $\neg L_1 \cup \neg L_2$ is a Gene language by Lemma 1.

Lastly, $\neg (\neg L_1 \cup \neg L_2)$ is also a Gene language by Lemma 2, and it is concluded that **GL** is closed under intersection operation. □

Theorem 2

GL is closed under finite intersection operation.

Proof) Induction is used for this proof. Let n be the number of intersection operation. First, for the base case ($n = 2$), it is trivial that **GL** is closed under one-time intersection by Lemma 3.

Next, for the inductive case, suppose that this theorem holds for the case $n = k$. Then if there are k Gene languages $\{L_1, L_2 \ldots L_k\}$, then $\cap^{k}_{i=1} L_i$ is also a Gene language by the hypothesis. By definition of a Gene language, there exists a Gene automaton M that accepts $\cap^{k}_{i=1} L_i$.

Let L be another Gene language. By Lemma 3, $(\cap^{k}_{i=1} L_i) \cap L$ is also a Gene language, which shows that this theorem holds for the case $n = k+1$. Hence, **GL** is closed under finite intersection operation. □

Theorem 3

GL is a semi-topological space on Σ^*.

Proof) Recall that to be a semi-topology, a subset of power set should include an empty set {} and the universal set Σ^*. Define $M_0 = (Q, \Sigma, \delta, q_0, \{\})$. Then the Gene automaton M_0 accepts no inputs, which means that $L(M_0) = \{\}$.

Define $M_\forall = (Q, \Sigma, \delta, q_0, Q)$. Then this Gene automaton $M_\forall$ always accepts all the inputs, in which case $L(M_\forall) = \Sigma^*$. Therefore, $\{\}, \Sigma^* \in$ **GL**, and **GL** satisfies the first condition to be a semi-topology.

By Theorem 1, **GL** is closed under countable union operation. By Theorem 2, **GL** is closed under finite intersection operation. Therefore, **GL** is a semi-topology on Σ^*. □

4. Related Work

The related work takes advantage of topology to make the hierarchies of temporal properties. Manna and Pnueli [MP90] proved that, in the topological view, the hierarchy of temporal properties coincides with the two lower levels of the Borel hierarchy, starting with the closed sets and open sets.

Baier and Kwiatkowska [BK00] showed that the usual topology on strings, Mazurkiewicz traces and pomsets arises as special cases. They have been able to obtain extensional, topological and temporal characterizations of classes of temporal properties including safety and liveness. Extensional characterization often admits a topological characterization with respect to the natural topologies of the domain of computations.

Alpern and Scheider [AS85] worked with the Cantor topology in the domain of infinite sequences of states. Meanwhile, Chang, Manna and Pnueli [CMP92] worked with the same domain but focused on the syntactic classes of properties expressed in Linear-time Temporal Logic (LTL). When considering a partial order semantic domain of computations together with a partial order temporal logic, the picture complicates further, as the natural topologies of such domains (the relativised Scott topology) are coarser than their metric topologies.

5. Future Work

This paper presents the connection between Gene languages and semi-topological space. It is a good contribution to make this new connection because there are many powerful tools in topology theory to help studying language theory, and vice versa. However, there are two questions to be solved. First, how can we use Gene languages in the theory of computation? The concept of the Gene language is just introduced, and we should find the location of Gene language in the whole hierarchy of languages. Second, how can we take advantage of the theory of topology to language and automata theory? A semi-topology T is a topology if union of the elements of *any* subcollection of T is in T. This paper only considered *countable* subcollections. We plan to study the closure property of Gene languages in the *uncountable* cases.

References

[Mun99] J. Munkres, *Topology*, Prentice Hall, 1999.

[Sip96] M. Sipser, *Introduction to the Theory of Computation*, Brooks Cole, 1996.

[HU79] J. Hopcroft, J. Ullman, *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley, 1979.

[MP90] Z. Manna, A. Pnueli, *A Hierarchy of Temporal Properties*, Proceedings of 9th ACM Symposium on Principles of Distributed Computing, pp 377-408, ACM Press,

1990.
[CMP92] E. Chang, Z. Manna, A. Pnueli, *The Safety-Progress Classification*. Proceedings Computer and System Science, NATO Advanced Science Institute Series, Springer-Verlag, 1992.
[BK00] C. Baier, M. Kwiatkowska, *On Topological Hierarchies of Temporal Properties*, Fundamenta Informaticae Vol. 41, pp 259-294, 2000.
[KPP94] M. Kwiatkowska, D. Peled, W. Penczek, *A Hierarchy of Partial Order Temporal Properties*, Proceedings Temporal Logic, LNCS vol. 827, pp 398-414, Springer-Verlag, 1994.
[Kwi91] M. Kwiatkowska, *On Topological Characterization of Behavioral Properties,* Topology and Category Theory in Computer Science, pp 153-177, Oxford Press, 1991.
[AS85] B. Alpern, F. Schneider, *Defining Liveness*, Information Processing Letters 21, pp 181-185, 1985.