I define and study models arising from Biomolecular Computing in the context of Formal Language Theory, Graph Theory, Topology and Algebra. I use these models to define and study new classes of graphs, languages and other structures.

Inspired by biomolecular reactions and processes Rozenberg and Ehrenfeucht define forbidding and enforcing systems (fe-systems) on languages and study their language theoretical and computational properties (see [8–10, 41]). The general idea of the forbidding-enforcing paradigm is that a class of structures is defined by a pair of constraints, called an fe-system. In their model, an fe-system defines a class of languages called an fe-family. The fe-family consists of all languages that satisfy both the forbidding and enforcing constraints of the fe-system. The precise definitions of fe-systems are lengthy and thus omitted (see [8, 41, 18, 15]).

In [18] we study fe-families topologically and algebraically. Our work was the first to suggest that the set of formal languages can be studied as a topological space. This research has inspired further investigation on the space of formal languages under different metrics in [32]. We define a metric on the set of formal languages, which was implicitly used in [10, 41] and follows a similar approach discussed in Symbolic Dynamics (see [33]) and study the resulting metric space. We prove that this space is homeomorphic to the Cantor Space. We show that every infinite language is a limit of a sequence of finite languages and that every family of languages that contains the class of finite languages FIN is not closed. Since fe-families are closed (see [9, 10, 41]), we conclude that the well-known Chomsky families cannot be defined by an fe-system. Thus, we prove that fe-systems define entirely new classes of languages. A series of other topological results were presented including the necessary and sufficient conditions for fe-families to be open. We also study fe-systems algebraically by characterizing the homomorphisms that extend to continuous functions on languages, which naturally corresponds to the continuous maps on infinite sequences in [40]. A series of theorems in [18] investigate the necessary and sufficient conditions for homomorphisms to map fe-families into fe-families.

In [15] we show how fe-systems can define valid DNA sequences. We, also, show how fe-systems can define the process of splicing of DNA by an enzyme and recombination. This is an equivalent definition for splicing systems initially defined as string rewriting types of rules in [22]. We propose a generalization of the forbidding-enforcing paradigm and define it in a categorical sense. We show how fe-systems on the category of sets can define solutions to computational problems like the k-colorability problem. This suggests that fe-systems can be very useful for information processing.

Membrane systems (P systems) were introduced in 1998 by G. Paun (see [35] and [36]) as a theoretical distributed computational model inspired by the structure and the
functioning of the living cell. In [4] and [5], we introduce a new type of membrane system: a P system with symport/antiport of rules. Unlike traditional P systems where objects and evolution rules are associated with hierarchical arrangement of compartments (called membranes) in such a way that the rules “direct” the objects to move from one membrane to another, in our model we propose that evolution rules evolve the objects in a classical way, but without moving them through membranes. Instead, symport/antiport rules are used to move the evolution rules across membranes and thus present an entirely new way of looking at P systems. We prove a number of results. In particular, we show that using non-cooperative rules and antiports of unbounded weight makes it possible to obtain at least the Parikh set of ET0L languages. On the other hand, using catalytic rules (one catalyst) and antiports of weight 2, these systems become universal.

In [16] we carry the ideas of fe-systems to graph theory. We define new classes of graphs based on the boundary conditions of forbidding and enforcing. The resulting fe-family of graphs contains all connected graphs that satisfy both the forbidding and enforcing constraints. In the past, only forbidden induced subgraphs have been considered and only in the “strict” case of singleton forbidders. They have been used primarily to prove necessary and sufficient conditions for Hamiltonian graphs (see [21]). Our fe-systems model provides a much more general platform to study classes of graphs. We characterize well-known classes of graphs like trees, bipartite graphs, paths, cycles, complete graphs, Eulerian, and k-regular graphs. We define and investigate properties of connecting graphs and show that every forbidding set can be replaced entirely by an enforcing set. However, such a characterization does not make forbidding sets obsolete as the corresponding enforcing sets are infinite. We also show why the converse, i.e. replacing enforcing sets with forbidding sets, cannot be done in general. Inspired by experimental and theoretical work on self-assembly of graphs (see for example [30]), we show how graph fe-systems can model vertex coloring and solutions to the k-colorability problem, and hence, they have a potential for information processing.

In [17] we investigate ways to remove redundancy from fe-systems and to simplify them. We state and prove several normal forms for forbidding constraints and enforcing constraints and provide a variety of examples.

In [13] I define and study a model of fe-systems, in which an fe-system defines a single language, rather than a family of languages. This work was supported by UNF Summer Research Grant 2008. The connection between fe-languages and fe-families was investigated. A series of results establish the link between Chomsky’s classes of languages and fe-languages. The classes of finite and factorial languages were characterized by language fe-systems and for all of the Chomsky’s classes of languages it was shown that there exists a language that can be defined by a language fe-system. This contrasts the fe-family language model for which we showed in [18] that none of the Chomsky’s classes can be defined by a fe-family fe-system. The paper ends with showing that this model is suitable for describing computation. More specifically, language fe-systems are shown to be capable of defining the solutions to the k-colorability problem and modeling splicing by an enzyme. Some of these results were presented at the Biomathematical Computing
conference at Binghamton University in 2008 and the full paper was presented at CiE 2011.

Simplification and eliminating redundancy of language fe-systems was investigated in [14]. This investigation resulted in a series of normal forms. The relationship between fe-family fe-systems and fe-language fe-systems investigated in [13] was used to prove that every normal form for forbidding sets for families of languages is also a normal form for forbidding sets for single languages. However, the unique minimal normal form for families is not at all minimal or unique for the single language model. Hence, new normal forms were proposed that are minimal and unique for single language forbidding sets. Forbidden subwords have well been extensively studied in combinatorics of words. However, to my knowledge, they have only been studied in the strict sense. Forbidding finite sets of subwords with parts of the sets allowed was first proposed in [13] and the corresponding normal forms in [14]. This paper was presented at LATA 2012.

In [19] we use the fe-family fe-systems as defined in [8] to study DNA codes. In order to avoid undesirable Watson-Crick hybridization of DNA strings (words), a valid DNA code has to satisfy certain coding properties. These properties have been widely studied (see for example [24, 31, 25–28]). It is well-known that generating a large set of DNA code words is a difficult problem (see for example [12]). In this paper we show that one fe-family fe-systems can define an entire class of DNA codes, e.g. $\theta$-$k$-codes, as opposed to studying DNA codes with classical language theory where one grammar (or an automaton) generates (or accepts) one DNA code. We characterize and study the properties of $\theta$-subword-$k$-codes, $\theta$-infix, $\theta$-intercode, $\theta$-comma-free, $\theta$-$k$-codes, and several other DNA codes by fe-systems. We confirm some known properties of DNA codes by fe-systems and provide a connection with experimental results from [28]. We also show how the computational nature of fe-system as discussed in [8] can be used for generating good DNA code words. This work was partially funded by UNF Research Grant 2010 and presented at DNA 17 at Caltech, CiE 2012 at Cambridge and will be presented at the 1st Int’l Conf. on Theory and Practice of Natural Computing 2012 in Tarragona, Spain.

Our current work on DNA codes [20] focuses on investigating maximal sets of DNA code words through single language fe-systems.

References


