Forbidding-Enforcing Properties in DNA Self-Assembly of Graphs

Giuditta Franco\(^{1}\) and Nataša Jonoska\(^{2}\)

1 Department of Computer Science, University of Verona, Italy.
   e-mail: franco@sci.univr.it
2 Department of Mathematics, University of South Florida, Tampa, Florida, USA
   e-mail: jonoska@math.usf.edu

Summary. Starting from a set of strands (or other types of building blocks) a variant of forbidding-enforcing systems for graphs is proposed which models the DNA self-assembly process. The possible outcomes of the self-assembly process comply with necessary constraints arising from the physical and chemical properties of DNA. A set of forbidding and enforcing rules that describe these constraints are presented.

1 Introduction

Guided constructions of complex structures at nano scale are one of the key challenges involving science and technology in the twenty-first century. This challenge is at the core of an emerging discipline of Nanoscience. One of the first synthesized DNA molecule with a structure deviant from the standard double helix is the stable four-junction molecule (now known as J1) that was designed in the late 80’s in the laboratory of N.C. Seeman [18]. This molecule now is one of the basic building blocks used for design and assembly of various different constructions in the rapidly growing field of DNA nanotechnology. It has been used as a basis for a more complex building blocks of double and triple cross-over molecules [23, 9] as well as for junction molecules with more than four branches [22]. These armed branched molecules were employed in a construction of two-dimensional array [10], suggested for growing a DNA fractal-like molecule [1], for assembling arbitrary three dimensional graphs [15] even for obtaining DNA Borromean rings [12].

The powerful molecular recognition of Watson-Crick complementarity employed in DNA base pairing is also used in various models of biomolecular computing and information processing to guide the assembly of complex DNA structures. The DNA strands have a natural orientation that is maintained by the concatenation through the phosphodiester bonds, while the Watson-Crick hydrogen bonds anneal two strands with opposite orientation following the
base complementarity of adenine ↔ thymine (a ↔ t) and citosine ↔ guanine
(c ↔ g). Two and three-dimensional DNA assemblies have been suggested
and demonstrated for information processing as well as for computing (see for ex.
[24, 2]). An experimental demonstration for some of these ideas were ob-
tained through the construction of the Sierpinski triangle [14] as well as by the
linear assembly of TX molecules encoding an XOR computation [11]. Using
brunched junction molecules for computation through assembling three di-
mensional structures was suggested in [7] by demonstrating how NP complete
problems can be solved by one step assembly.

This process in which substructures, driven by their selective affinity, are
spontaneously self-ordered into superstructures, is now widely referred to as
self-assembly. Although there have been some notable successes in the self-
assembly process, there are still lack of consistent methods for constructing
complex structures out of pool of individual molecular components and in gen-
eral, understanding the process of self-assembly [19]. There are initial theoreti-
cal investigations dealing with complexity of the self-assembled structures and
the computational power (see for ex. [6, 13, 20, 21]), however understanding
how the molecular architecture works has been a challenge. This necessitates
much more theoretical and experimental investigations.

In this paper we present a theoretical model for the generation of DNA
self-assembled forms as a family of graph structures that comply to certain
“forbidden” constraints and follow some chemically predetermined “enforc-
ing” properties. This model is a variant of the forbidding-enforcing systems,
introduced in [5] as a model of chemical processes. We elevate this original
model to the construction of three-dimensional structures, in particular, we
concentrate here on structures obtained by DNA self-assembly. On the other
hand, as a systematic way of describing classes of graphs, our model can be
considered as a starting point of developing new ways to investigate graphs
in classical graph theory.

2 Forbidding-Enforcing Systems

The forbidding-enforcing (f-e shortly) systems are a non-standard device to
generate formal languages, alternative to the grammar systems from the clas-
sical formal language theory. It was inspired by chemical processes and was
used to simulate certain DNA based computations [5], and afterwards it was
introduced in the context of the membrane computing [3].

The basic idea is to simulate a molecular reaction where “everything that
is not forbidden is allowed”. This assumes a totally different perspective with
respect to the basic axiom underlying the computation of grammars and au-
tomata, where “everything that is not allowed is forbidden”. In fact, while in
a typical formal language theory model, a set of rewriting productions estab-
lishes how to generate (or recognize) the words of a language, in a f-e system
a family of languages is generated by some enforcing conditions, dictating
certain evolving rules for the system and some forbidding conditions, given as a group of patterns which cannot occur together at the same time. The enforcing rules ensure that if a certain group of strings is present in the system, then some other strings will eventually be present too.

More formally, given an alphabet $\Sigma$, we have the following definitions as introduced in [5].

**Definition 1.** A forbidding set $F$ is a family of finite nonempty subsets of $\Sigma^+$, and an enforcing set $E$ is a family of ordered pairs $(X, Y)$, where $X$ and $Y$ are finite subsets of $\Sigma^+$ and $Y \neq \emptyset$.

We call a forbider any element of $F$ and an enforcer any element of $E$.

**Definition 2.** A forbidding-enforcing system (f-e system) is a triple $\Gamma = (\Sigma, F, E)$, where $F$ is a forbidding set and $E$ is an enforcing set (over $\Sigma$).

As usual, given a language $L$ we denote with $\text{sub}(L)$ the set of all subwords of $w$ for some $w \in L$.

**Definition 3.** A language $L$ over $\Sigma^*$ is generated by an f-e system $\Gamma$ if $F \not\subseteq \text{sub}(L)$ for every $F \in F$, and $X \subseteq L \Rightarrow Y \cap L \neq \emptyset$ for every $(X, Y) \in E$. The family of all languages generated by $\Gamma$ is indicated by $L(\Gamma)$ and is called f-e family.

In order to generate a language, the evolution of an f-e system proceeds according to the “molecular reactions” specified by $E$ (for every forbider $(X, Y)$, the presence of all the strings contained in $X$ produces at least one of the strings contained in $Y$), but it is constrained by $F$, that is, the evolution cannot lead to any group of patterns specified by a forbidder from $F$. Note that the forbidding set $F$ contains patterns, for example $x$ and $y$, that may not be in the system, or that may not be simultaneously in the system, it depends on whether $\{x\}, \{y\} \in F$ or $\{x, y\} \in F$, respectively.

**Definition 4.** A f-e system $\Gamma = (\Sigma, F, E)$ is finitary if for any finite language $Z$ there are at most finite number of elements $(Z, Y)$ in $E$.

In other words, a system is finitary if at one instance, presence of a finite set of strings in the system enforces presence of only a finite number of additional strings. Any family of languages that can be specified by an enforcing set can be also specified by a finitary enforcing set [5], thus there exists a sort of “finitary normal form” of f-e systems.

We conclude the introduction of the basic notions about f-e systems with a simple example. Let $\Sigma = \{a, b\}$, $F = \{\{aa, bb\}\}$, and $E = \{(\emptyset, bb)\}$. Then the subsets of $\Sigma^* \setminus \Sigma^* aa \Sigma^*$ containing $\{bb\}$ make up the f-e family of the given system. To see this one can observe that the forbidder is satisfied by subsets from $\Sigma^* \setminus \Sigma^* aa \Sigma^*$ or $\Sigma^* \setminus \Sigma^* bb \Sigma^*$, since at east one of $aa$ or $bb$ cannot appear as a subword in the language. But since the $\emptyset$ is a subset of any
language, the enforcer ensures that $bb$ is in every language of the family. Now $\Sigma^* \setminus \Sigma^* \{bb\} \Sigma^*$ excludes $bb$, so all languages that satisfy the f-e system are subsets of $\Sigma^* \setminus \Sigma^* \{aa\} \Sigma^*$ and contain $\{bb\}$.

In what follows we concentrate on annealing DNA strands guided by Watson-Crick complementarity. This will be performed by imposing forbidding-enforcing constraints guided by the physical and chemical constraints of the molecule. Starting from a system containing some given DNA filaments that can be partially annealed, we model the formation of further bonds guided by Watson-Crick complementarity, without destroying or changing any of the bonds that are already present. Each DNA strand is represented as a directed path or a directed cycle (in the case of circular molecules) and the Watson-Crick connections are represented with undirected edges. This idea allows potentially larger number of “possible” products by the same DNA strands initially present in the pot. Therefore, instead of increasing the number of strings, we model the increase of annealing bonds among initial DNA sub-structures. In this way, we extend the basic idea of f-e systems to graphs and suggest another way to look into DNA self-assembly.

3 A Model for DNA Self-Assembly

Regardless of the biochemical and topological properties of the structures seen in different aspects of DNA nanostructures, such forms can be seen as complex structures made of single strands connected (or attached respectively) to each other by two kinds of “bonds”: phosphodiester bonds, i.e., concatenation and Watson-Crick complementarity.

We describe three-dimensional (3D) DNA forms by means of graphs $(V, P, E, \lambda)$ where $V$ is a set of vertices labeled by elements of $\{a,c,g,t\}^k$ with $k$ a fixed positive integer, $P$ is a set of directed paths, possibly cycles, on the vertices of $V$, and $E$ a matching set of undirected edges such that two vertices in $V$ are incident with the same edge only if they have Watson-Crick complementary labels$^3$. The labeling function is $\lambda: V \to \{a,c,g,t\}^k$.

The labels of the paths (concatenation of the labels of the vertices) represent the given DNA filaments, while the edges of $E$ represent the Watson-Crick complementarity.

$^3$ Given an involution $\varphi$ on the alphabet $Z = \{a,t,g,c\}$ (a mapping from $Z$ to $Z$ such that $\varphi^2$ is equal to the identity mapping), and the usual reverse operation on strings $\text{rev}(a_1 \ldots a_{n-1} a_n) = a_n a_{n-1} \ldots a_1$, where $a_1, \ldots, a_n \in Z$, we call $\text{corev}$ the composition of the reverse operation with $\varphi$ extended to a morphic involution on $Z^*$. The order is not relevant, because it holds that $\varphi \circ \text{rev} = \text{rev} \circ \varphi$.

In the case of DNA complementarity, the involution $c$ is the correspondence: $a \to t$, $c \to g$, $t \to a$, $g \to c$, and $\text{corev}$ is the Watson-Crick complementarity on DNA strings, that pairs strings such that the “reverse” of a given string is the image of the other one by $\varphi$.

As is customary, in this paper $\text{corev}(\alpha)$ will be denoted by $\overline{\alpha}$, where $\alpha \in Z^*$.
bonds generated to form the structure. In Figure 1 one can see a simple DNA form described by such graph.

![Diagram](image)

**Fig. 1.** Example of a self-assembled structure. Here we note that: \( \delta_1 = \beta_4 \), \( \delta_2 = \phi_1 \), \( \beta_3 = \phi_2 \), \( \eta_1 = \phi_3 \), \( \gamma_2 = \beta_2 \) and \( \eta_2 = \gamma_1 \).

A similar idea was used in [21] for describing a DNA complex on which self-assembly rules were defined. In that case, a DNA complex was considered as a connected directed graph with vertices labeled by symbols from \( \{a, t, c, g\} \) and edges from \( \{\text{backbone}, \text{basepair}\} \), with at most one incoming and one outgoing edge of each type at each node. Here we consider strings having a fixed length \( k \) as labels of the vertices, by abstracting the experimental fact that there exists a lower bound on these lengths that will provide Watson-Crick pairing. This bound depends on the temperature, salt concentration, and on other parameters of the experiment. It is quite intuitive that starting from the same filaments the number of possible self-assembled DNA forms increases as the \( k \) value decreases, which in our case represents the (minimum) length of the attached portions. Here we focus on the graph structures corresponding to self-assembled forms, by fixing the value for \( k \) (for example \( k = 5 \), which is approximately the length of a half helical turn) and the complementarity between two strings of length \( k \).

Another possibility that can appear experimentally but is ignored in this paper is the overlapping of strings. Consider the ten symbol string \( w = \text{actactacta} \). For \( k = 5 \), we can write \( w = uv \) with \( u = \text{actac} \) and \( v = \text{tacta} \). However there are two occurrences of \( u \) as a substring of \( w \), i.e., \( w = \text{act} \cdot u \cdot \text{ta} \). In practice, a complementary string of \( u \) can anneal to both of these occurrences. Our model assumes that the strings representing DNA strands do not have such labeling.

On the other hand, we suppose that the correspondence from vertices to (labeling) strings may not necessarily be injective, in fact more than one occurrence of a string may be located along the filaments forming the structure. Therefore we consider a labeling function \( \lambda : V \to \{a, c, g, t\}^k \) that assigns
a string from \{a, t, c, g\}^k to each vertex from \(V\). Further, in order to keep
the model more realistic, all our graphs are finite graphs, where \(V\) and \(P\) are
(given) finite sets.

The description of a self-assembly structure by means of such a graph
simplifies the representation and emphasizes the interrelations between the
substructures, for example a loop (a cycle with only one non-directed edge)
corresponds to a hairpin formation [16], and a connected component of the
graph corresponds to one DNA structure.

Consider the triple cross over molecule TX [9] designed in the N. Seeman’s
laboratory (see Fig. 2 top). This complex structure is made of six strands,
although there are examples of similar TX molecules with fewer strands. It
can be presented as a graph in the following way. The length of the molecule
is about 4.5 helical turns which corresponds to roughly 48 bases. For this, we
consider 36 vertices, each labeled with a string of length 8. In Fig. 2 (a) the
TX molecule is simplified by ignoring the helical turns and the Watson-Crick
pairing corresponding to 8 consecutive nucleotides is identified with a group of
five short bars. In Figure 2 (b) - (d) the process of obtaining a graph structure
 corresponding to the TX molecule is presented. The directed edges follow the
5' - 3' direction of the strand, and the undirected edges indicate the sequence
of 8 base pairs.

Regardless the size, shape of any form can be self-assembled by spontaneous
local processes from just a few types of initial pieces. In fact, the

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig2}
\caption{Triple cross over molecule (TX) as a self-assembly graph.}
\end{figure}
Kolmogorov complexity of a shape provides upper and lower bounds for the number of tile types necessary to self-assemble a given shape (at some scale) [20]. Nevertheless, to find strands that generate given structures remains a difficult design problem.

One can consider the converse question, what kind of forms are obtained by adding in a pot some given DNA substructures (or filaments). In other terms, assume a labeled graph structure \((V, P, \lambda, E)\) where \(V\) is the set of vertice, \(P\) the set of paths, \(\lambda\), the labeling function and \(E\) a partial matching on the vertices is given. What are the edges that can be added in the (possibly empty) matching set \(E\) such that there exists a DNA structure corresponding to the obtained graph. It is clear that there exist graphs \((V, P, \lambda, E)\) that do not represent DNA structures, for example, they do not take into account some physical constraints. Figure 3 shows an example of two such structures.

![Fig. 3. Examples of graph non-corresponding to a DNA form.](image)

Thus, given a collection of directed paths and cycles with vertices labeled by strings, we are going to consider a set of valid graphs, where “forbidden” structures are not present. In particular, in order to obtain a graph which represents a self-assembled DNA structure, the matching set must respect certain constraints defined by means of a set of forbidden subgraphs which on the other hand follow physical and chemical restrictions of the DNA molecule. Such restrictions about the interrelations between DNA strings can be formulated only locally [8]. Moreover, a set of enforcing structures is considered in order to describe the parallelism intrinsic to the nature of self-assembly. This includes the considerations that further pairing of partially annealed molecules is preferred over strands that are far apart.

Molecular self-assembly is an inherently parallel process which begins anywhere it is energetically favored. Here we assume all thermodynamical conditions necessary for self-assembly are present such that assembly is obtained wherever it is structurally possible. Another assumption coming from the chemical structure of DNA is its non-flexibility. For example for a given \(k\) the model forbids formation of double stranded DNA circular molecules with length less than \(nk\) nucleotides. We can assume that \(nk \geq 100\) for example, in which case \(n\) would depend on our choice of \(k\).

First we consider the theoretical model for forbidding-enforcing systems in graphs.
4 Graph-Forbidding-Enforcing Systems

Consider graphs of the type $G = (V, \mathcal{P}, E)$, where $V$ is a finite set of vertices, $\mathcal{P}$ is a set of oriented paths, possibly cycles, on vertices of $V$, and $E$ is partial matching, that is, a set of undirected edges such that any vertex of $V$ is incident to at most one other vertex. We denote this family of graphs with $\mathcal{G}$. If $p \in \mathcal{P}$ we indicate with $A(p)$ the set of arcs included in the path $p$, and call $A(\mathcal{P})$ the set $\bigcup_{p \in \mathcal{P}} A(p)$.

**Definition 5.** Given a positive integer $m$ a $m$-local g-forbidder is a graph $(V, \mathcal{P}, E)$ in $\mathcal{G}$ with $|A(p)| < m$ for every $p \in \mathcal{P}$. A g-enforcer is an ordered pair $(X, Y)$, where $X = (V, \mathcal{P}, E) \in \mathcal{G}$ and $Y = (V', \mathcal{P}', E') \in \mathcal{G}$ are such that $V = V'$, $\mathcal{P} = \mathcal{P}'$ and $E \subseteq E'$.

The constant $m$ is included to ensure that all forbidders act locally, i.e., one only needs to concentrate on paths with not more than $m$ vertices. This in general may depend on the experimental conditions. In what follows, we assume that $m$ is fixed and all $m$-local g-forbidders are referred to as simply g-forbidders.

The set $\mathcal{F}$ of g-forbidders is called **forbidding set**, and the set $\mathcal{E}$ of g-enforcers is called the **enforcing set** of a family of graphs.

As in the original definition, a forbidding set may be infinite and the only requirement is that each forbidder is finite [5]. A g-forbidder is finite if it has a finite number of arcs and edges. Moreover, regardless of the presence of the other forbidders, each g-forbidder cannot appear as subgraph of a graph satisfying that forbidder.

**Definition 6.** A g-f-e system is a structure $\Gamma = (V, \mathcal{P}, E, \mathcal{F}, \mathcal{E})$, where $V$ is a finite set of vertices, $\mathcal{P}$ is a set of directed paths, possibly cycles, on vertices of $V$, $E$ is a partial matching on $V$, $\mathcal{F}$ is a set of g-forbidders and $\mathcal{E}$ is a set of g-enforcers.

Given a graph $G = (V, \mathcal{P}, E) \in \mathcal{G}$, we call $\text{sub}(G)$ the set of all subgraphs $(V_0, \mathcal{P}_0, E_0)$ of $G$, where $V_0 \subseteq V$, every $p_0 \in \mathcal{P}_0$ is a path on vertices from $V_0$ such that $p_0 \subseteq p$ for some $p \in \mathcal{P}$ and $E_0 \subseteq E$ is a matching set on $V_0$. We write $G' \leq G$ for $G' \in \text{sub}(G)$. Similarly, $G' < G$ if $G' \leq G$ but $G' \neq G$.

**Definition 7.** A graph $G = (V, \mathcal{P}, E^*)$ is generated by the g-f-e system $\Gamma = (V, \mathcal{P}, E, \mathcal{F}, \mathcal{E})$ if, $E \subseteq E^*$, $F \notin \text{sub}(G^*)$ for every $F \in \mathcal{F}$, and for every $(X, Y) \in \mathcal{E}$, if $X \in \text{sub}(G)$ then there is $Y' \in \text{sub}(G)$ such that $X < Y' \leq Y$.

The family of all graphs generated by a graph-forbidding-enforcing system $\Gamma$ is indicated by $\mathcal{G}(\Gamma)$. The elements of $\mathcal{G}(\Gamma)$ are called **assembled graphs**.

Similarly as in the original definition of forbidding-enforcing systems, the evolution of a g-f-e system proceeds according to the molecular reactions specified through $\mathcal{E}$ by increasing the elements of the matching set $E$, but not allowing subgraphs that are forbidden by $\mathcal{F}$. 
Now we concentrate on the model of g-f-e systems that simulate the self-assembly process of DNA. In this case the vertices of the graphs are labeled by strings from alphabet \( \{ a, g, c, t \}^k \). Hence, all graphs belong to the class of graphs \((V, P, E, \lambda)\) where \( V, P, \) and \( E \) are the same as in the previous section, and \( \lambda : V \to \{ a, g, c, t \}^k \) is the labeling of the vertices. All definitions for g-forbidders, g-enforcers and g-f-e systems are transferred to this class of graphs in a straightforward way. We note that the labeling of the vertices of every subgraph of a graph is preserved.

Consider a g-f-e system \( G(\Gamma) \) with \( \Gamma = (V, P, E, \lambda, F, E) \) where the DNA strings are associated to the paths \( P \) are given by the (finite number of) initial DNA filaments in the pot, the set \( E \) is given by the Watson-Crick bonds present in the initial DNA substructures. We specify a set of forbidders and enforcers that ensures construction of DNA structures.

The forbidding set \( F \) forbids constructions that are “impossible” by the physical and chemical properties of DNA. We list three g-forbidders that are most straightforward observations and should be included in every g-f-e system that simulates DNA self-assembly.

1. **proper annealing** (a pair of vertices is matched only if they have complementary labels)
   \[
   F_0 = (V = \{ v_1, v_2 \}, P = \emptyset, E = \{ e = \{ v_1, v_2 \} \}, \lambda(v_1) = \alpha, \lambda(v_2) = \bar{\alpha})
   \]
2. **hairpin constraint** (a strand with a string \( \alpha \bar{\alpha} \) without any distance between \( \alpha \) and \( \bar{\alpha} \) cannot form a hairpin)
   \[
   F_1 = (V = \{ v_1, v_2 \}, P = \{ p = (v_1, v_2) \}, E = \{ e = \{ v_1, v_2 \} \}, \lambda(v_1) = \alpha, \lambda(v_2) = \bar{\alpha})
   \]
   To ease notation, we describe the forbidders just by listing the labels of the vertices in the paths and the set \( E \). This assumes that all vertices appearing in the listed paths are distinct. Hence, the above forbidder is written \( F_1 = (\{\alpha \bar{\alpha}\}, \{\alpha, \bar{\alpha}\}) \).
3. **non-crossing, i.e. orientation preserving constraint**
   \[
   F_2 = (\{a_1 \bar{a}_1, a_2 \bar{a}_2\}, \{\{a_1, \bar{a}_1\}, \{a_2, \bar{a}_2\}\}).
   \]
   The forbidder \( F_1 \) says that a positive length (between the attached portions) is necessary to allow a strand to turn back and attach to itself [16]. The forbidder \( F_2 \) avoids physically impossible situations as those ones in Figure 3. Note that both structures presented in Figure 3 are forbidden by \( F_2 \).

Due to the experimental conditions, the purpose of the design as well as the length of the labeling strings the complete set of g-forbidders for the self-assembly system may include additional structures, for example the one shown in Figure 4.

The basic set \( E \) contains the following g-enforcers. Arising from experimental evidence [27] it is clear that DNA strands prefer pairing with com-
complete complements. Also, all of the DNA nanostructures are obtained by using “stick-end” cohesion. Hence, we have the following enforcers.

1. **annealing**

   $E_0 = (X, Y)$ where $X = (V = \{v_1, v_2\}, \emptyset, \emptyset, \lambda(v_1) = \bar{\lambda(v_2)})$ and $Y = (V = \{v_1, v_2\}, \emptyset, \emptyset, \lambda(v_1) = \lambda(v_2))$.

   This enforcer can be seen as the brute-force enforcer that ensure annealing of the complementary edges. If left without any changes (say for example requiring that vertices $v_1$ and $v_2$ belong to paths with certain lengths) $E_0$ will ensure that all structures in the g-f-e system have all possible complementary vertices connected.

2. **one side context rules, complete complements**

   $E_1 = (X, Y)$ where (by the simplified notation)
   
   $X = (\{\alpha \beta, \bar{\beta} \bar{\alpha}\}, \{\{\alpha, \bar{\alpha}\}\})$,

   $Y = (\{\alpha \beta, \bar{\beta} \bar{\alpha}\}, \{\{\alpha, \bar{\alpha}\}, \{\beta, \bar{\beta}\}\})$

   and $E'_1 = (X, Y)$ is essentially the same as $E_1$ except the initial annealing has occurred at the other side of the molecule:
   
   $X = (\{\alpha \beta, \bar{\beta} \bar{\alpha}\}, \{\beta, \bar{\beta}\}), Y = (\{\alpha \beta, \bar{\beta} \bar{\alpha}\}, \{\{\alpha, \bar{\alpha}\}, \{\beta, \bar{\beta}\}\})$

   The enforcers $E_1$ and $E'_1$ ensure that full complements of the strands are preferred. This is in accordance with the experimental finding [27] and this fact is one of the basis for several DNA based molecular devices [25, 27].

3. **sticky end cohesion** (see Fig. 6 (a))
Forbidding-Enforcing Properties in DNA Self-Assembly of Graphs

$E_2 = (X,Y)$ where (by the simplified notation)

$X = \{\{\alpha\beta, \gamma\beta\}, \{\alpha, \bar{\alpha}\}, \{\gamma, \bar{\gamma}\}\}$

$Y = \{\{\alpha\beta, \gamma\beta\}, \{\alpha, \bar{\alpha}\}, \{\gamma, \bar{\gamma}\}, \{\beta, \bar{\beta}\}\}$

4. joining (see Fig. 6 (b))

$E_3 = (X,Y)$ where (by the simplified notation)

$X = \{\{\alpha\beta\gamma\delta, \bar{\delta}\bar{\gamma}, \bar{\beta}\bar{\alpha}\}, \emptyset\}$

$Y = \{\{\alpha\beta\gamma\delta, \bar{\delta}\bar{\gamma}, \bar{\beta}\bar{\alpha}\}, \{\alpha, \alpha\bar{\alpha}\}, \{\beta, \beta\bar{\beta}\}, \{\gamma, \gamma\bar{\gamma}\}, \{\delta, \delta\bar{\delta}\}\}$

Fig. 6. The enforcing rules for sticky end cohesion (a) and “gluing” two molecules by a complement to both (b).

The enforcers that ensure sticky end cohesion are depicted in Figure 6. The left portion of the figure is the partial annealing of the molecules and the right portion depicts the graphs corresponding to these two cases. Note that the enforcer $E_3$ adds one, two, three or four new undirected edges. If only one edge is added, there is no guarantee that the full annealing will happen, but by the enforcers $E_1$ and $E'_1$ there will be at least one more edge added.

As in the case of forbidders, according to the experimental conditions and the initial designs of the molecules, additional enforcers may be added.
However, we believe that the above set of enforcers should be included in every model of g-f-e system that describes DNA self-assembly.

We conjecture that the family $\mathcal{G}(\Gamma)$ defined by the g-f-e system described above exhibits all graph structures corresponding to the DNA forms that can be obtained from the initial substructures (given by $(V, P, E, \lambda)$) by means of self-assembly. Clearly it contains any DNA construction generated from the initial substructures, but it remains an open problem to show that the g-forbidders and the g-enforcers proposed above guarantee that this family contains only those graphs corresponding to possible DNA structures.

6 Conclusion

This paper suggests new directions in both graph theory and DNA self-assembly. The general problem faced here is the following: given a set $P$ of paths and cycles, a set $F$ of forbidden structures, and a set $E$ of enforced structures, what are the graphs included in the set $\mathcal{G}(\Gamma)$ for $\Gamma = (V, P, E, \lambda, \mathcal{F}, \mathcal{E})$? The presented model focuses in particular to DNA self-assembly and the set of structures obtained through this process. However, the idea of graph forbidding enforcing systems, can certainly be extended to other self-assembly processes in nature as well as in the pure theoretical methods to study mathematical properties of graphs. In the case of DNA self-assembly, the evolution process is described in a very natural way as an increase of the cardinality of the matching set between vertices with complementary labels. For other types of applications the concept of g-f-e systems may need to be adjusted in a different way that will be more suitable for simulating the evolution in those particular processes.

Taking in account that the labels of the vertices are strings over a finite alphabet, one can consider theoretical questions in the context of formal language theory. It may be interesting to investigate the classes of graphs generated by a g-f-e system where the labels of $V$ belong to a given language taken from one of the Chomsky classes. On the other hand, considering finite languages and investigating how the structure of generated graphs depends on the g-f-e system could be useful in study of cellular processes, where for example the function of signal transduction nets is fairly well understood.

Acknowledgment. Many ideas presented in this chapter are a result of numerous discussions with Nadrian C. Seeman. This work was done during the visit of G. Franco to University of South Florida in the Spring of 2005 and is in part supported by the grants EIA-0086015, CCF-0523928 and CCF-0432009 from the National Science Foundation, USA.
References