A Rigid Category of DNA Secondary Structures

Andrés Ortiz-Muñoz
Harvard Medical School
Boston, MA, USA
andortiz19@gmail.com

We construct a strict pivotal monoidal category $\mathscr{D}_{\text{DNA}}$ whose objects are DNA sequences (words over $\{A, C, G, T\}$) and whose morphisms are isotopy classes of typed noncrossing planar matchings—composed of through-strands and Watson–Crick-typed arcs—in a rectangle with source and target boundaries. The dual of a sequence is its reverse complement, evaluation and coevaluation are canonical duplex pairings, and the snake identities hold by planar isotopy. A bending correspondence identifies each morphism $x \to y$ with a secondary structure on the combined word $x^\vee y$; in particular, the generalized elements $\varepsilon \to w$ are exactly the non-pseudoknotted secondary structures on w. Composition, viewed in this straightened picture, is computed by a *zip-and-transfer* operation on complementary interfaces—a combinatorial rearrangement of base-pair connectivity of which toehold-mediated strand displacement is a kinetically specific instance. Because $\mathscr{D}_{\text{DNA}}$ is rigid monoidal, it shares the categorical backbone of pregroup grammars and the DisCoCat framework for compositional semantics: a strong monoidal functor from a grammatical category to $\mathscr{D}_{\text{DNA}}$ maps grammatical reductions to Watson–Crick base pairing and sentence meanings to secondary structures. We describe this functor and discuss connections to algorithmic self-assembly, composable strand-displacement circuits, and constructive dynamical systems.

1 Introduction

DNA nanotechnology and molecular computation rest on the predictable interactions governed by Watson–Crick base pairing. In this paper we construct a category $\mathscr{D}_{\text{DNA}}$ (for DNA diagrams) whose objects are words over the nucleotide alphabet $\{A, C, G, T\}$ and whose morphisms are isotopy classes of typed noncrossing arc diagrams (non-pseudoknotted secondary structures). The dual of a sequence is its reverse complement, evaluation and coevaluation are canonical duplex pairings, and the snake identities hold by planar isotopy, making $\mathscr{D}_{\text{DNA}}$ a strict pivotal monoidal category. Composition, in the straightened picture, is computed by a *zip-and-transfer* operation on complementary interfaces—a general mechanism of which toehold-mediated strand displacement [19, 21, 10] is a kinetically specific instance.

The categorical structure underlying $\mathscr{D}_{\text{DNA}}$—a rigid monoidal category—is the same structure that Lambek [8] identified in natural language syntax through pregroup grammars, and that Coecke, Sadrzadeh, and Clark [1] used to build compositional distributional semantics (DisCoCat). This structural coincidence yields a monoidal functor from grammar to DNA in which grammatical reduction maps to Watson–Crick base pairing; we treat this as one application of the rigid structure rather than its sole motivation.

Section 2 recalls the biology; Section 3 reviews rigid monoidal categories; Section 4 constructs $\mathscr{D}_{\text{DNA}}$ and its pivotal structure; Section 5 describes the functor from pregroup grammars to $\mathscr{D}_{\text{DNA}}$; and Section 6 discusses variants, connections, and open questions.

To appear in EPTCS.

2 DNA secondary structures

Deoxyribonucleic acid (DNA) is a polymer whose monomers, called nucleotides or bases, are drawn from the four-letter alphabet

$$\Sigma = \{A, C, G, T\}.$$

A single-stranded DNA molecule is a finite sequence of bases, read by convention from the $5'$ to the $3'$ end. We write such a sequence as a word $w = b_1 b_2 \cdots b_n$ in the free monoid Σ^*, with the empty word ε representing the absence of any bases.

The defining chemical feature of DNA is Watson–Crick complementarity: adenine (A) pairs with thymine (T), and cytosine (C) pairs with guanine (G). We encode this as an involution on the alphabet,

$$\overline{A} = T, \quad \overline{T} = A, \quad \overline{C} = G, \quad \overline{G} = C,$$

which extends to words by reversal and letterwise complementation. The *reverse complement* of $w = b_1 b_2 \cdots b_n$ is

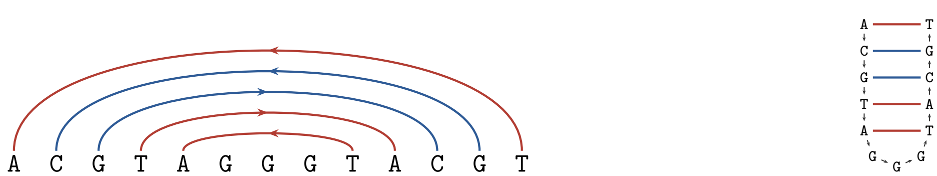
$$w^\vee = \overline{b_n}\,\overline{b_{n-1}} \cdots \overline{b_1}.$$

Reversal reflects the antiparallel orientation of the two strands in a DNA duplex: a strand running $5' \to 3'$ pairs with its complement running $3' \to 5'$. When w and $w^\vee$ are brought together, each base in w can pair with the corresponding base in $w^\vee$, forming a double-helical duplex.

A *secondary structure* on a sequence $w = b_1 \cdots b_n$ is a set of base pairs—pairs of positions (i, j) with $i < j$ such that b_i and b_j are Watson–Crick complements—subject to two constraints:

1. **Uniqueness.** Each position participates in at most one base pair.
2. **No crossings.** If (i, j) and (k, l) are both base pairs with $i < k$, then either $j < k$ (the pairs are disjoint) or $i < k < l < j$ (one pair is nested inside the other). The forbidden pattern is $i < k < j < l$, which would represent a *pseudoknot*[1].

The standard representation is an *arc diagram*: the bases $b_1, \ldots, b_n$ are arranged along a horizontal line, and each base pair (i, j) is drawn as an arc in the upper half-plane connecting positions i and j. The no-crossing condition means that arcs do not intersect. The diagram below shows a simple hairpin with five nested base pairs closing a three-base loop, shown in both arc and folded views.



(a) Arc view (b) Folded view

These structures are fundamental objects in molecular biology. They determine the geometry and function of single-stranded nucleic acids and have been studied combinatorially since the work of Waterman [17]. Their enumeration is closely related to Motzkin numbers and noncrossing partitions. We will see that they also have a natural categorical interpretation.

[1]The non-pseudoknotted (planar) model is a widely used first approximation for DNA secondary structure and is often sufficient for stochastic modeling of combinatorial DNA dynamics over folded-state energy landscapes. For sufficiently long sequences, however, pseudoknots do occur; categorically, this suggests braided (rather than purely planar pivotal) extensions as a more general framework. Despite this limitation, non-pseudoknotted models have shown strong predictive value in many experimental regimes [17, 19, 21]. In this paper we focus on developing the underlying pivotal structure under the standard non-pseudoknotted assumption.

3 Rigid monoidal categories

We recall the categorical notions used in the DNA construction. Readers familiar with rigid categories may skip to Section 4; we include definitions to fix notation and follow Selinger's conventions [15].

A *monoidal category* $(\mathscr{C}, \otimes, I)$ consists of a category $\mathscr{C}$ with a bifunctor $\otimes \colon \mathscr{C} \times \mathscr{C} \to \mathscr{C}$ (tensor product), a unit object I, and associativity/unitality isomorphisms satisfying coherence conditions. It is *strict* when these isomorphisms are identities; by Mac Lane coherence, every monoidal category is monoidally equivalent to a strict one, and we work strictly throughout.

Definition 3.1. A strict monoidal category $(\mathscr{C}, \otimes, I)$ is *right rigid* if every object A is equipped with a *right dual* A^*, together with morphisms

$$\mathrm{ev}_A \colon A \otimes A^* \to I, \qquad \mathrm{coev}_A \colon I \to A^* \otimes A,$$

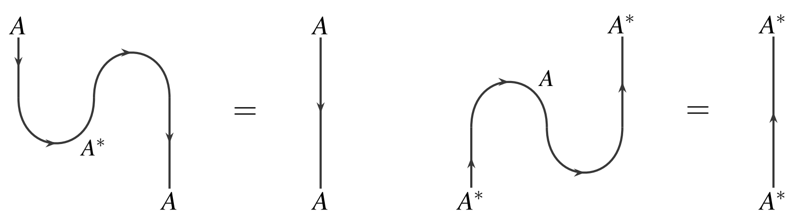
called *evaluation* (or *cup*) and *coevaluation* (or *cap*)[2]. We read string diagrams from top to bottom throughout this paper. With this convention, these maps satisfy the *snake identities*:

$$(\mathrm{ev}_A \otimes \mathrm{id}_A) \circ (\mathrm{id}_A \otimes \mathrm{coev}_A) = \mathrm{id}_A,$$

$$(\mathrm{id}_{A^*} \otimes \mathrm{ev}_A) \circ (\mathrm{coev}_A \otimes \mathrm{id}_{A^*}) = \mathrm{id}_{A^*}.$$

It is *left rigid* if analogous data exist on the other side, and *rigid* (or *autonomous*) if it is both left and right rigid.

The snake identities have a vivid graphical interpretation in the language of string diagrams [6]. Objects are drawn as labeled wires, morphisms as boxes or nodes, and the tensor product as horizontal juxtaposition. The unit object I is represented by the empty diagram. In our pictures, the arrow direction on each wire encodes the duality structure: reversing orientation corresponds to passing between an object and its dual. In this notation, evaluation is a cup (a wire bending up from A and A^* to nothing) and coevaluation is a cap (a wire bending down from nothing to A^* and A). The snake identities then assert that a zig-zag in a wire can be straightened:



Definition 3.2. A rigid monoidal category is *pivotal* if it is equipped with a monoidal natural isomorphism $A \xrightarrow{\sim} A^{**}$ for every object A. It is *strictly pivotal* if this isomorphism is the identity, so that $A^{**} = A$.

In a pivotal category, left and right duals coincide (up to the pivotal isomorphism), and the graphical calculus acquires full rotational invariance within the plane: diagrams can be bent and turned without changing the morphism they represent. This is the diagrammatic setting natural for planar arc diagrams.

4 The category $\mathscr{D}_{\mathrm{DNA}}$ (DNA diagrams)

We now construct the category. The reader should keep in mind two pictures simultaneously: the algebraic data (objects, morphisms, composition) and their diagrammatic representations as typed arc diagrams.

[2] Different texts adopt different orientation and reading conventions for string diagrams (for example, bottom-to-top or left-to-right). Accordingly, the labels "cup" and "cap" are convention-dependent and may be interchanged in other references.

4.1 Objects

The objects of $\mathscr{D}_{\mathrm{DNA}}$ are the words in the free monoid Σ^*. The tensor product on objects is concatenation:

$$x \otimes y := xy,$$

and the monoidal unit is the empty word ε. This is strictly associative and unital.

4.2 Duality

The dual of a word $w = b_1 \cdots b_n$ is its reverse complement:

$$w^{\vee} = \overline{b_n}\,\overline{b_{n-1}} \cdots \overline{b_1}.$$

Three properties are immediate from the definitions and are worth recording:

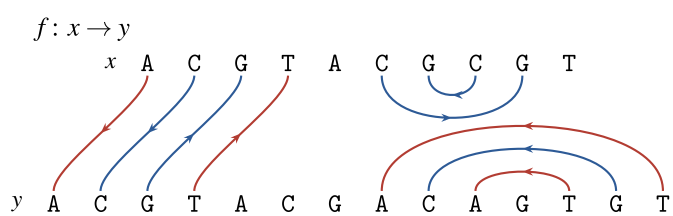
1. $\varepsilon^{\vee} = \varepsilon$.
2. $(uv)^{\vee} = v^{\vee} u^{\vee}$ for all words u, v.
3. $(w^{\vee})^{\vee} = w$ for every word w.

Property (3) implies that the double dual is the identity on objects.

4.3 Morphisms

A morphism $f : x \to y$ in $\mathscr{D}_{\mathrm{DNA}}$ is an isotopy class of typed noncrossing planar partial matchings, drawn in a rectangle with the bases of x arranged along the top boundary (the source) and the bases of y arranged along the bottom boundary (the target). The matching consists of three components:

1. **Through-wires**: wires connecting a position of x to a position of y, representing a base that passes from source to target.
2. **Source arcs**: noncrossing arcs among positions of x, representing internal base pairings on the source side.
3. **Target arcs**: noncrossing arcs among positions of y, representing internal base pairings on the target side.



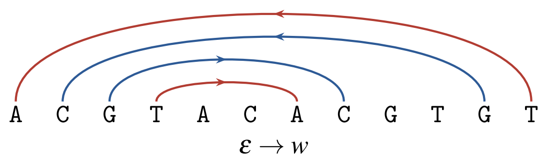
The global constraints are:

- *Degree*: every position participates in at most one edge or arc.
- *Planarity*: the union of all strands and arcs is noncrossing in the rectangle.
- *Pairing and wire conventions*: every arc connects Watson–Crick complementary bases, while through-strands are identity wires (the same base on source and target). We draw A–T pairs in red and C–G pairs in blue; A and C wires go down, while T and G wires go up.

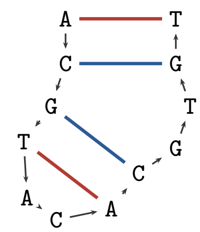
Two such diagrams represent the same morphism if they are related by planar isotopy within the rectangle, keeping the boundary points fixed.

4.4 Generalized elements: secondary structures as terms

A morphism $s\colon \varepsilon \to w$ is a generalized element of the object w: it is a diagram with no source boundary and the bases of w on the target boundary. Concretely, it consists entirely of target arcs—that is, it is a noncrossing arc diagram on the positions of w with complementary pairings. This is exactly a non-pseudoknotted secondary structure on w.[3]



(a) Arc view



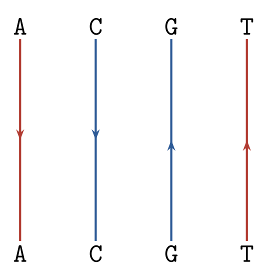
(b) Folded view

In the folded view, backbone arrows indicate the $5' \to 3'$ direction (left to right, matching the arc-view ordering), and arcs are rendered as base-pair bonds.[4]

Conversely, every non-pseudoknotted secondary structure on w defines a morphism $\varepsilon \to w$ in $\mathscr{D}_{\mathrm{DNA}}$. Equivalently, $\mathrm{Hom}(\varepsilon, w)$ is the set of generalized elements (closed terms) of type w, and thus catalogs the secondary structures of w.

4.5 Identity and composition

The identity morphism $\mathrm{id}_w\colon w \to w$ is the straight-through matching: position i on the source connects to position i on the target, with no internal arcs.

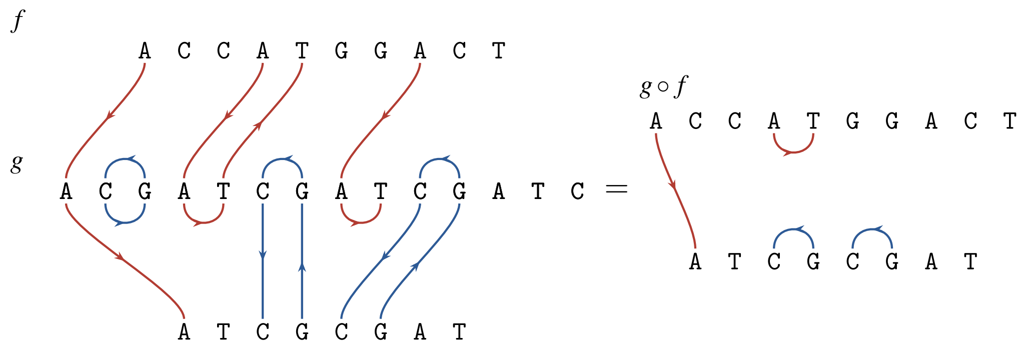


id_w

Given $f\colon x \to y$ and $g\colon y \to z$, the composite $g \circ f\colon x \to z$ is formed by vertical stacking: place the rectangle for f on top and the rectangle for g below, gluing along the shared y-boundary. After gluing, simplify by planar isotopy and cap-cup cancellations (the snake relations). Associativity of composition follows from the topological nature of stacking.

[3]From the internal type-theoretic viewpoint of categorical logic, objects are types (here: DNA words), morphisms are terms/proofs/programs (here: DNA diagrams), and maps $\varepsilon \to w$ are closed terms of type w, i.e., generalized elements of w.

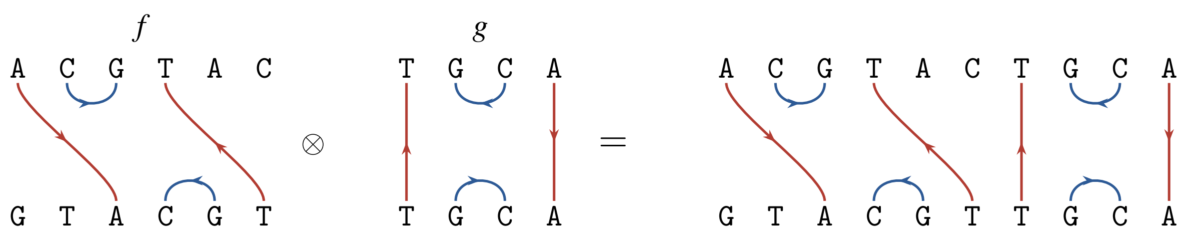
[4]All diagrams in this paper are schematic and intended for illustration only: the relative lengths of backbone bonds and Watson–Crick pair bonds are not to scale, and real hairpins have a minimum loop size imposed by geometric and physical constraints.



Under composition, closed loops—connected components with no boundary endpoints—may arise. We adopt the *normalized* convention throughout: every closed loop equals 1_ε and is erased. Other conventions (e.g. loops weighted by scalars δ_{AT}, δ_{CG}) yield valid alternative categories; see Section 6.

4.6 Tensor product on morphisms

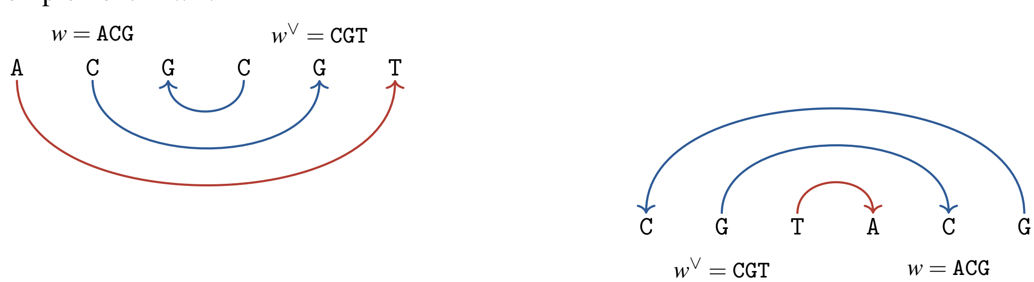
The tensor product $f \otimes g$ of morphisms $f\colon x_1 \to y_1$ and $g\colon x_2 \to y_2$ is their horizontal juxtaposition: the diagram for f is placed to the left of the diagram for g, giving a morphism $x_1x_2 \to y_1y_2$.



4.7 Evaluation, coevaluation, and rigidity

For each word $w = b_1 \cdots b_n$, we define:

- The *coevaluation* (cap):
$$\mathrm{coev}_w\colon \varepsilon \to w^\vee \otimes w.$$
This is the noncrossing diagram that pairs each base b_i in w with the corresponding complementary base $\overline{b_i}$ in $w^\vee$, forming n nested arcs. It represents the canonical duplex pairing.
- The *evaluation* (cup):
$$\mathrm{ev}_w\colon w \otimes w^\vee \to \varepsilon.$$
This is the vertically reflected version: arcs on the source side pair each position of w with its complement in $w^\vee$.



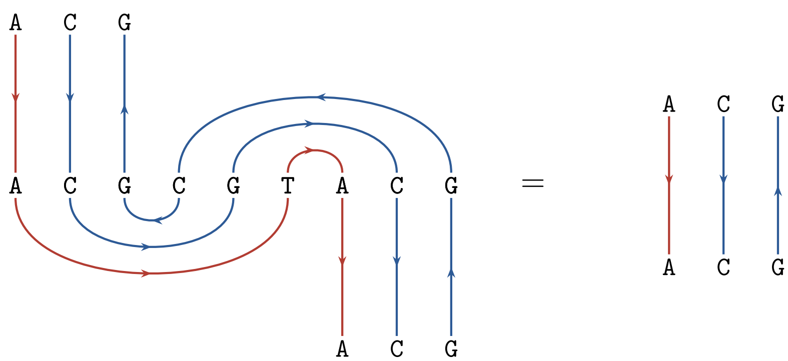
$\mathrm{ev}_w\colon w \otimes w^\vee \to \varepsilon$ $\qquad$ $\mathrm{coev}_w\colon \varepsilon \to w^\vee \otimes w$

Proposition 4.1. *The snake identities hold in* $\mathscr{D}_{\mathrm{DNA}}$*:*

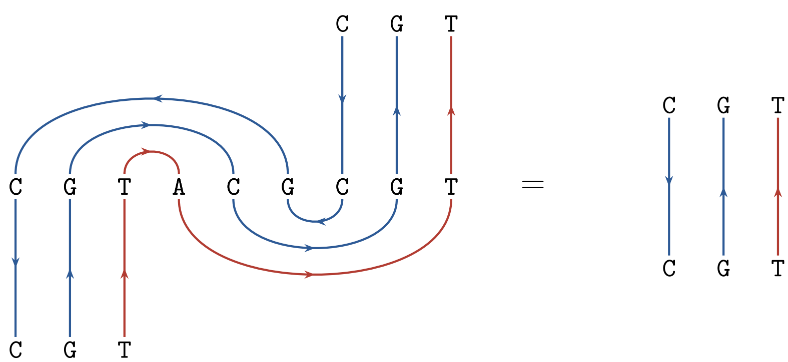
$$(\mathrm{ev}_w \otimes \mathrm{id}_w) \circ (\mathrm{id}_w \otimes \mathrm{coev}_w) = \mathrm{id}_w,$$

$$(\mathrm{id}_{w^\vee} \otimes \mathrm{ev}_w) \circ (\mathrm{coev}_w \otimes \mathrm{id}_{w^\vee}) = \mathrm{id}_{w^\vee}.$$

Proof. Each identity asserts that a zig-zag of arcs—a cup followed by a cap on the same strand—can be straightened to a single through-strand. In the diagrammatic picture, the left-hand side of the first identity is a wire for w that bends up to create a cup (introducing $w^\vee$ and w), then bends back down via a cap (annihilating w and $w^\vee$). By planar isotopy, this zig-zag straightens to the identity wire for w. The second identity is analogous. □



$(\mathrm{ev}_w \otimes \mathrm{id}_w) \circ (\mathrm{id}_w \otimes \mathrm{coev}_w) = \mathrm{id}_w$



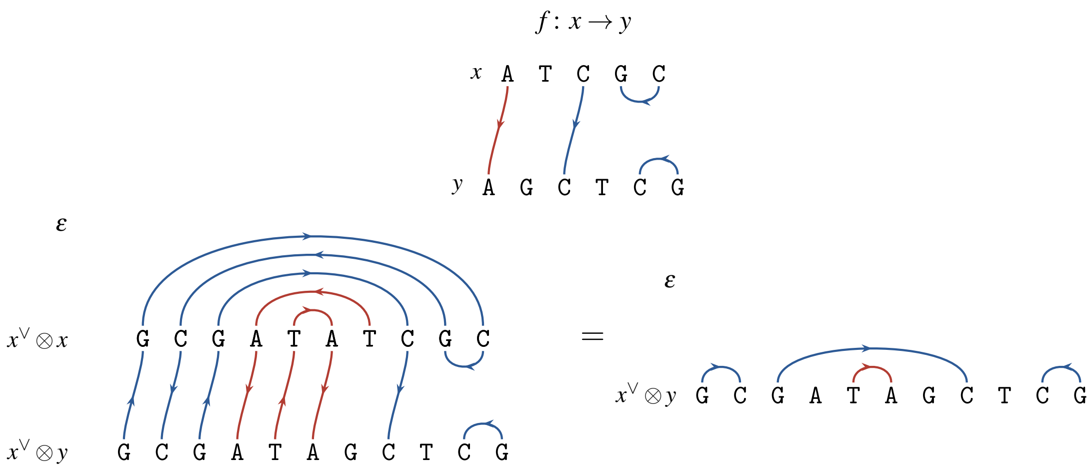
$(\mathrm{id}_{w^\vee} \otimes \mathrm{ev}_w) \circ (\mathrm{coev}_w \otimes \mathrm{id}_{w^\vee}) = \mathrm{id}_{w^\vee}$

Theorem 4.2. $\mathscr{D}_{\mathrm{DNA}}$ *is a strict pivotal monoidal category.*

Proof. The monoidal structure (concatenation, empty word, horizontal juxtaposition) is strict by construction. Proposition 4.1 establishes that every object w has a right dual $w^\vee$ with the required evaluation and coevaluation morphisms. The same data, read with the opposite convention, give left duals, so $\mathscr{D}_{\mathrm{DNA}}$ is rigid. Since $(w^\vee)^\vee = w$ for every word w, the double-dual functor is the identity. This is a (trivial) monoidal natural isomorphism $A \to A^{**}$, so $\mathscr{D}_{\mathrm{DNA}}$ is strictly pivotal. □

4.8 The bending correspondence

A standard consequence of rigidity is the bijection $\mathrm{Hom}(x,y) \cong \mathrm{Hom}(\varepsilon, x^\vee \otimes y)$, given by $\widehat{f} := (\mathrm{id}_{x^\vee} \otimes f) \circ \mathrm{coev}_x \colon \varepsilon \to x^\vee \otimes y$. In the DNA setting, any interaction diagram from x to y is equivalent to a secondary-structure-type diagram on the combined boundary word $x^\vee \otimes y$. For example, taking $x = \texttt{ATCGC}$, $y = \texttt{AGCTCG}$, and a morphism $f \colon x \to y$ with both internal arcs and through-strands:



4.9 The straightened picture and zip-and-transfer

The bending correspondence gives the operational punchline of the construction: composition in $\mathscr{D}_{\mathrm{DNA}}$ corresponds to an interface-level zip-and-transfer mechanism that can be implemented in DNA molecular computation under standard design constraints.[5] Given morphisms $f \colon x \to y$ and $g \colon y \to z$, let

$$\widehat{f} \colon \varepsilon \to x^\vee \otimes y, \qquad \widehat{g} \colon \varepsilon \to y^\vee \otimes z$$

be their straightened forms. These are secondary-structure states—morphisms from the empty word. The composite $g \circ f \colon x \to z$ is computed entirely at the secondary-structure level in three steps:

1. **Juxtapose.** Place $\widehat{f}$ and $\widehat{g}$ side by side (tensor), obtaining a state
$$\widehat{f} \otimes \widehat{g} \colon \varepsilon \to x^\vee \otimes y \otimes y^\vee \otimes z.$$
2. **Contract.** The adjacent segments y and $y^\vee$ are Watson–Crick complements. Contract them by applying the canonical evaluation $\mathrm{ev}_y \colon y \otimes y^\vee \to \varepsilon$. This pairs each base in y with its complement in $y^\vee$.
3. **Simplify.** Straighten the surviving connectivity and erase closed components that arise entirely within the contracted region (normalized loop convention).

The result is $\widehat{g \circ f} \colon \varepsilon \to x^\vee \otimes z$.

As an elementary discrete thermodynamic model, assign one unit to each base-pair bond and treat reachable structures with maximal bond count as stable; bond transfer itself is a rewiring step that preserves

[5] This correspondence is operational when each evaluation step is thermodynamically downhill and kinetically accessible. In strand-displacement systems, toeholds and branch-migration domains are designed so intended duplex products are favored over competitors [12, 21, 10].

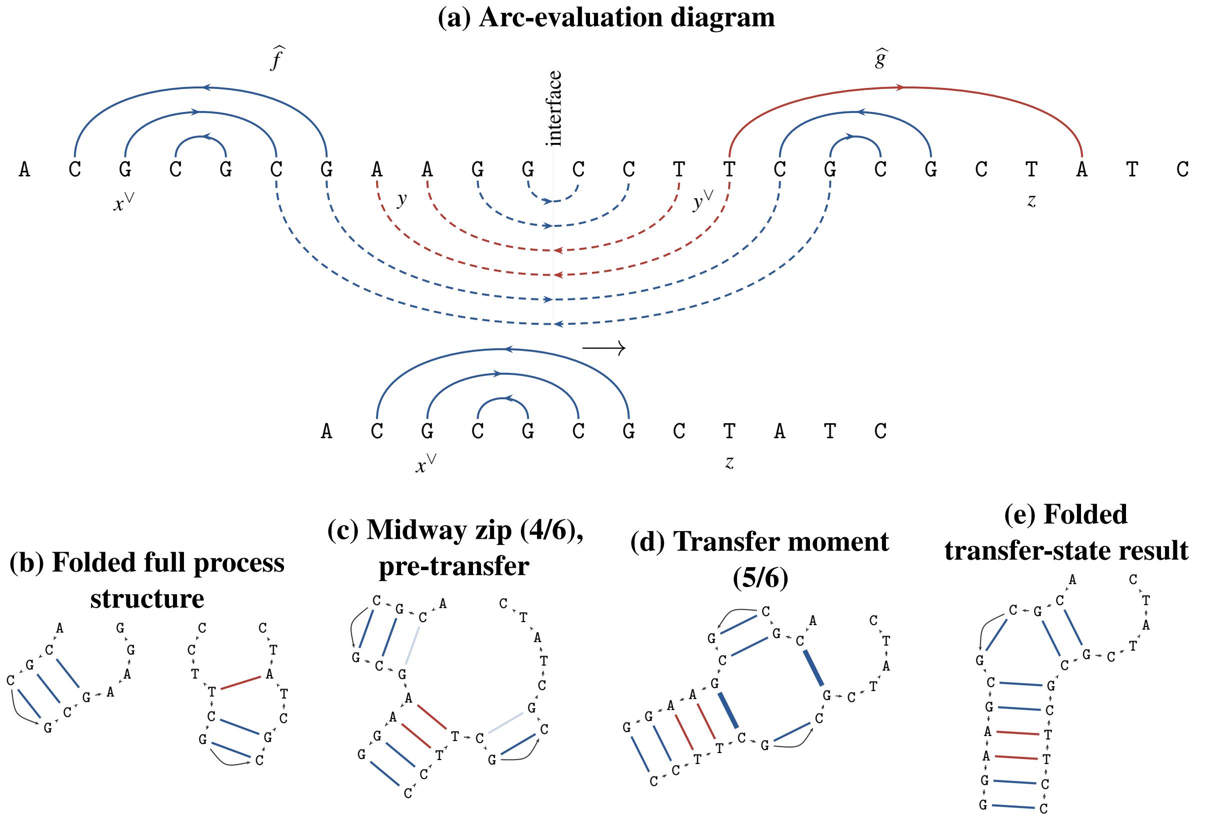
bond count, while the full zip-and-transfer trajectory is favorable when additional interface bonds form and the final state has a larger total bond count than the isolated initial states.

What happens combinatorially in step (2) is a *zip-and-transfer* operation. The complementary segments y and $y^{\vee}$ zip together—each base in y pairs with its complement in $y^{\vee}$. But some of these positions already participate in arcs from $\widehat{f}$ or $\widehat{g}$. When a position in y that is paired (via $\widehat{f}$) to some position in $x^{\vee}$ zips together with its complement in $y^{\vee}$ that is paired (via $\widehat{g}$) to some position in z, the base-pair connectivity *transfers through* the interface: the $x^{\vee}$ position ends up paired to the z position. Arcs that do not bridge the interface close into loops and are erased.

For a concrete generic instance with nontrivial structure on all three boundaries, take

$$x = \texttt{CGCGT}, \qquad y = \texttt{CGAAGG}, \qquad z = \texttt{CGCTATC},$$

and morphisms $f: x \to y$, $g: y \to z$ chosen so that the straightened composition has two transfer paths and one surviving u-turn:



In panels (c)→(d), the highlighted bonds are the same pairings viewed before and after transfer: the bold interface bonds in (c) are rewired into the bold $x^{\vee}$–z bonds in (d).

Strand displacement [19, 21, 16] is the best-studied kinetically controlled instance of this mechanism: the interface has the specific structure of a toehold-initiated branch migration, and kinetic control ensures directionality. More broadly, branch migration—the stepwise transfer of base-pair partners along a homologous duplex [11]—is the biological process that directly instantiates the transfer operation of our categorical composition. A fully worked specialization with explicit morphisms and diagrams is given in Appendix A (Example A.1); our claim is therefore a mechanistic correspondence at the level of connectivity and design constraints, not a full quantitative model of reaction kinetics or thermodynamic landscapes.

5 From grammar to DNA

The rigid structure of $\mathscr{D}_{\text{DNA}}$ matches the categorical backbone of compositional models of natural language.

5.1 Pregroup grammars and compositional semantics

A *pregroup* [8] is a partially ordered monoid in which every element a has adjoints a^l, a^r satisfying $a^l a \leq 1 \leq aa^l$ and $aa^r \leq 1 \leq a^r a$. A pregroup grammar assigns to each word one or more types from a free pregroup over basic types such as n (noun) and s (sentence); a string is grammatical if the product of its types reduces to s. Categorically, a pregroup is a rigid monoidal category that happens to be a poset, with contractions as evaluations and expansions as coevaluations.

Coecke, Sadrzadeh, and Clark [1] observed that this shared rigid structure can be exploited compositionally: a strong monoidal functor $F : \mathscr{G} \to \mathscr{S}$ from a grammatical category $\mathscr{G}$ to a semantic target $\mathscr{S}$ transports grammatical reductions into semantic operations. In the original DisCoCat model, $\mathscr{S} = \mathbf{FdVect}$. We propose $\mathscr{D}_{\text{DNA}}$ as an alternative target.

5.2 The functor $F : \mathscr{G} \to \mathscr{D}_{\text{DNA}}$

A strong monoidal functor $F : \mathscr{G} \to \mathscr{D}_{\text{DNA}}$ is determined by the following data:

- **On basic types.** Each basic grammatical type is assigned a DNA sequence. For instance, one might set $F(n) = N$ and $F(s) = S$ for chosen words $N, S \in \Sigma^*$. The choice of sequences is part of the model design and is not determined by the category.
- **On duals.** Since F must respect duality, we have $F(n^r) = F(n)^\vee = N^\vee$. This is automatic from the monoidal functor requirement.
- **On cups and caps.** The grammatical contraction $n \otimes n^r \leq 1$ maps to the DNA evaluation $\text{ev}_N : N \otimes N^\vee \to \varepsilon$, which is the canonical Watson–Crick duplex formation. This is the key structural match: *grammatical reduction is base pairing*.
- **On lexical items.** A word of grammatical type t is a state $I \to t$ in the grammatical category (or a lexical extension of it). Under F, this becomes a morphism $\varepsilon \to F(t)$—a secondary structure on the DNA sequence $F(t)$. Different words of the same grammatical type correspond to different secondary structures on the same sequence.

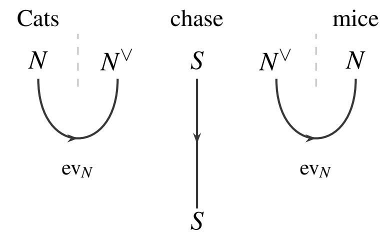
Example 5.1. Consider the sentence “Cats chase mice” with the type assignment

$$\text{Cats} : n, \qquad \text{chase} : n^r \otimes s \otimes n^l, \qquad \text{mice} : n.$$

The grammatical reduction contracts $n \otimes n^r$ on the left and $n^l \otimes n$ on the right, yielding the sentence type s. Under F, this becomes:

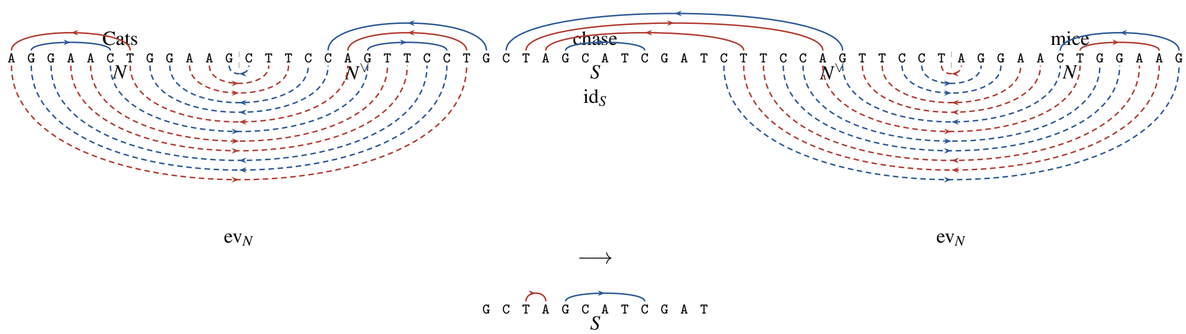
$$\varepsilon \xrightarrow{F(\text{Cats}) \otimes F(\text{chase}) \otimes F(\text{mice})} N \otimes N^\vee \otimes S \otimes N^\vee \otimes N \xrightarrow{\text{ev}_N \otimes \text{id}_S \otimes \text{ev}_N} S.$$

The intermediate step—the application of the evaluation maps—performs Watson–Crick pairing between the adjacent complementary segments N and $N^\vee$ on each side. In this concrete instance, one transferred arc from the left interface and one from the right interface land fully on S. The resulting secondary structure on S is the “meaning” of the sentence in the DNA semantics.

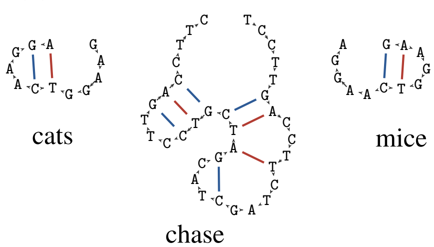


Concretely, let N and S be fixed DNA words of length 12. In this example, the last three positions of *cats* and *chase* are free, as are the first three positions of *chase* and *mice*; each lexical structure is non-trivial, maximally bound, and has no hairpins of size less than 3. The functor images of the lexical items are drawn as solid arcs; the dashed cups perform Watson–Crick evaluation on each N–$N^\vee$ interface. The transfer result on S contains at least one fully transferred arc from the right side, encoding the sentence meaning as a secondary structure on S.

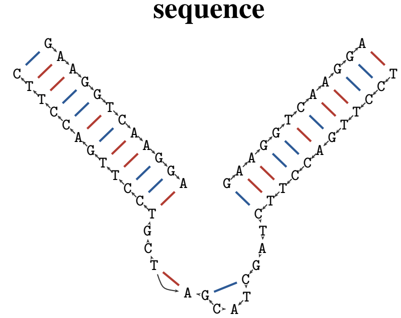
(a) Arc-evaluation diagram



(b) Folded full sentence structure



(c) Full folded transfer state on the sentence sequence



cats chase mice

Remark 5.2. In the straightened view, grammatical reduction—the contraction of adjacent type-antitype pairs—is precisely the zip-and-transfer operation of Section 4.9 applied to complementary DNA interfaces. Parsing a sentence thus corresponds to a cascade of zip-and-transfer steps, structurally analogous to the strand-displacement cascades used in DNA computing [10, 14].

6 Discussion

Several natural variants of $\mathscr{D}_{\mathrm{DNA}}$ suggest themselves. Rather than erasing closed loops, one can assign scalar weights δ_{AT}, δ_{CG} to produce a category enriched over a scalar ring, enabling thermodynamic refinements via nearest-neighbor energy models [12]. Dropping the noncrossing constraint would model pseudoknotted structures; categorically, this amounts to adding a braiding, moving from a pivotal to a braided or tortile monoidal category [7]. One could also replace literal sequences by equivalence classes (e.g. families satisfying a design relation), so that a type image $F(n)$ denotes a domain rather than a single fixed word.

The short sequences used in the grammar examples of Section 5 are purely illustrative. In practice, physically stable type assignments would require sequences tens of nucleotides long, with meaning stored in stems exceeding the thermal stability threshold [12]. Longer complementary interfaces open a richer design space: one can program selective binding profiles by tuning sequences, as exploited by DNA molecular computation [14, 21, 10, 16] and systematic sequence design tools such as NUPACK [20].

The categorical framework complements several existing lines of work. Winfree [18] connects the physical structure of DNA assemblies to the Chomsky hierarchy, showing that tree-like self-assembly generates exactly the context-free languages; our noncrossing diagrams inhabit the same structural regime, though a full formal equivalence remains to be established. Phillips and Cardelli [9] describe a programming language for composable strand-displacement circuits under an explicit no-secondary-structure assumption for strands; $\mathscr{D}_{\mathrm{DNA}}$ removes that restriction, so it is strictly more general while retaining an algebraic account of the same compositional operations, with zip-and-transfer as the primitive. Fontana, Schuster, and collaborators [5, 13] studied RNA secondary structures as phenotypes under a genotype–phenotype map induced by maximum-bond (equivalently, energy-minimizing) folding; in our setting, that map can itself be read as a computation semantics on structures. The algorithmic-chemistry program and its constructive dynamical extension [2, 3, 4] then supply the complementary viewpoint: interacting typed entities compute by composition while constructing new entities. $\mathscr{D}_{\mathrm{DNA}}$ realizes this point through duality: terms are secondary structures and, at the same time, morphisms (functions), so zip-and-transfer is simultaneously chemical interaction and functional composition.

We close with three open questions. (1) Which functors $\mathscr{G} \to \mathscr{D}_{\mathrm{DNA}}$ produce linguistically useful semantic distinctions? (2) Do naturally occurring DNA secondary structures admit grammatical interpretations under some choice of functor? (3) Can weighted-loop conventions encode nearest-neighbor free energies in a way that makes the resulting semantics sensitive to binding stability?

Acknowledgements

This work was first inspired during the author's doctoral studies in the laboratory of Erik Winfree at Caltech and conceived while the author was an Omidyar Postdoctoral Fellow at the Santa Fe Institute. The author gratefully acknowledges the financial support of the Fontana Lab at the Department of Systems Biology, Harvard Medical School, during the development of this work, and thanks Walter Fontana personally for his encouragement and intellectual generosity. The author also thanks Martha Lewis and Emily Riehl for useful conversations during the early stages of this project.

References

[1] B. Coecke, M. Sadrzadeh, and S. Clark. Mathematical foundations for a compositional distributional model of meaning. *Linguistic Analysis*, 36:345–384, 2010. arXiv:1003.4394.

[2] W. Fontana. Algorithmic chemistry: A model for functional self-organization. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*. Addison-Wesley, 1992.

[3] W. Fontana and L. W. Buss. The arrival of the fittest: Toward a theory of biological organization. *Bulletin of Mathematical Biology*, 56:1–64, 1994.

[4] W. Fontana and L. W. Buss. The barrier of objects: From dynamical systems to bounded organizations. In J. Casti and A. Karlqvist, editors, *Boundaries and Barriers*, pages 56–116. Addison-Wesley, 1996.

[5] W. Fontana, P. F. Stadler, E. G. Bornberg-Bauer, T. Griesmacher, I. L. Hofacker, M. Tacker, P. Tarazona, E. D. Weinberger, and P. Schuster. RNA folding and combinatory landscapes. *Physical Review E*, 47:2083–2099, 1993.

[6] A. Joyal and R. Street. The geometry of tensor calculus, I. *Advances in Mathematics*, 88:55–112, 1991.

[7] C. Kassel. *Quantum Groups*, volume 155 of *Graduate Texts in Mathematics*. Springer, 1995.

[8] J. Lambek. Type grammar revisited. In *Logical Aspects of Computational Linguistics*, volume 1582 of *Lecture Notes in Computer Science*, pages 1–27. Springer, 1999.

[9] A. Phillips and L. Cardelli. A programming language for composable DNA circuits. *Journal of the Royal Society Interface*, 6:S419–S436, 2009.

[10] L. Qian and E. Winfree. Scaling up digital circuit computation with DNA strand displacement cascades. *Science*, 332:1196–1201, 2011.

[11] C. M. Radding. Homologous pairing and strand exchange in genetic recombination. *Annual Review of Genetics*, 16:405–437, 1982.

[12] J. SantaLucia, Jr. A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proceedings of the National Academy of Sciences*, 95:1460–1465, 1998.

[13] P. Schuster, W. Fontana, P. F. Stadler, and I. L. Hofacker. From sequences to shapes and back: A case study in RNA secondary structures. *Proceedings of the Royal Society B*, 255:279–284, 1994.

[14] G. Seelig, D. Soloveichik, D. Y. Zhang, and E. Winfree. Enzyme-free nucleic acid logic circuits. *Science*, 314:1585–1588, 2006.

[15] P. Selinger. A survey of graphical languages for monoidal categories. In *New Structures for Physics*, volume 813 of *Lecture Notes in Physics*, pages 289–355. Springer, 2011.

[16] D. Soloveichik, G. Seelig, and E. Winfree. DNA as a universal substrate for chemical kinetics. *Proceedings of the National Academy of Sciences*, 107:5393–5398, 2010.

[17] M. S. Waterman. Secondary structure of single-stranded nucleic acids. *Advances in Mathematics, Supplementary Studies*, 1:167–212, 1978.

[18] E. Winfree. *Algorithmic Self-Assembly of DNA*. PhD thesis, California Institute of Technology, 1998.

[19] B. Yurke, A. J. Turberfield, A. P. Mills, Jr., F. C. Simmel, and J. L. Neumann. A DNA-fuelled molecular machine made of DNA. *Nature*, 406:605–608, 2000.

[20] J. N. Zadeh, C. D. Steenberg, J. S. Bois, B. R. Wolfe, M. B. Pierce, A. R. Khan, R. M. Dirks, and N. A. Pierce. NUPACK: Analysis and design of nucleic acid systems. *Journal of Computational Chemistry*, 32:170–173, 2011.

[21] D. Y. Zhang and E. Winfree. Control of DNA strand displacement kinetics using toehold exchange. *Journal of the American Chemical Society*, 131:17303–17314, 2009.

A Worked toehold-mediated strand displacement

Example A.1 (Toehold-mediated strand displacement)**.** Let t be a toehold and s a branch migration domain. Consider

$$f : s \to ts, \qquad g : ts \to \varepsilon,$$

where f has through-strands on the s positions (the target toehold positions are initially unmatched) and g is the empty morphism. As a concrete instance, set

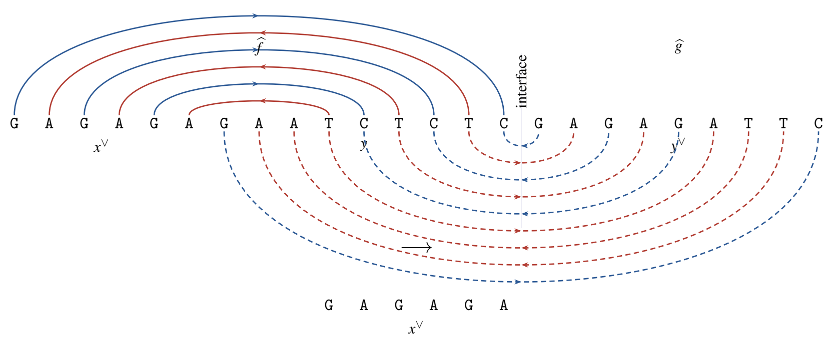
$$t = \texttt{GAA}, \qquad s = \texttt{TCTCTC}, \qquad ts = \texttt{GAATCTCTC}.$$

Then $\widehat{f}\colon \varepsilon \to s^\vee \otimes ts$ forms the substrate duplex on the s segment, while $\widehat{g}\colon \varepsilon \to (ts)^\vee = s^\vee t^\vee$ is a free invader strand. In folded form, the initial state has three strands

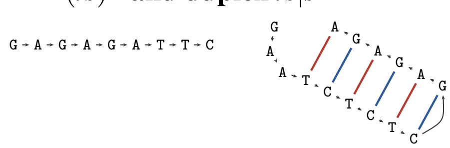
$$(ts)^\vee, \qquad ts, \qquad s^\vee,$$

with $(ts)^\vee$ free and ts paired with $s^\vee$ via $\mathrm{coev}_{s^\vee}$. Contracting $ts \otimes (ts)^\vee$ performs toehold binding first and then branch migration along s. At the folded-complex level, this transfer yields the duplex $(ts)^\vee \otimes ts$ together with released $s^\vee$.

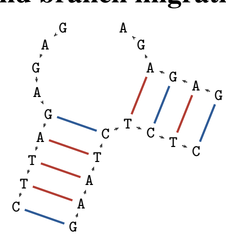
(a) Arc-evaluation diagram



(b) Folded initial state: free $(ts)^\vee$ and duplex $ts|s^\vee$



(c) Midway toehold binding and branch migration



(d) Folded transfer result: duplex $(ts)^\vee|ts$ and free $s^\vee$