Transition Polynomials, Double Covers, and Biomolecular Computing

Joanna A. Ellis-Monaghan[†]
Department of Mathematics
Saint Michael's College
One Winooski Park
Colchester, VT 05439

jellis-monagan@smcvt.edu

Abstract

Recent advances in biomolecular computing, assembling graphs from strands of DNA, require restricted oriented double coverings of the edges of a graph. We show that the generalized transition polynomial, a highly adaptable graph polynomial that encodes information about families of cycles in Eulerian graphs, provides a generating function formulation for this application. Although the requirements of the biomolecular construction are slightly different from a cycle double cover (an edge may be covered twice by the same "cycle" for example), results from variations of cycle double covers inform the original problem from biomolecular computing. In particular, we use a theorem of C. Thomassen to specify precisely when a graph may be constructed from a single strand of DNA, and theorems of Hongbing and Zhu to characterize graphs that require at least *m* strands of DNA in their construction.

Key words: graph polynomials, transition systems, Eulerian graphs, cycles, cycle double cover conjecture, bidirectional double tracing, oriented walks, biomolecular computing, DNA strands.

[†]Supported by the Vermont Genetics Network through the NIH Grant Number 1 P20 RR16462 from the BRIN program of the NCRR.

Introduction

Recently biomolecular computing has been a highly active area of research, particularly since Adleman's demonstration of encoding a graph in DNA and then using the tools of molecular biology to determine a directed Hamiltonian path ([Adl94]). Because of the great promise of biomolecular computing, recent research has focused on DNA self-assembly of nanoscale mathematical constructs, in particular graphs. Several different graphs have

been constructed from self-assembling DNA strands, including a cube ([CS91]), truncated octahedra ([ZS94]), and most recently a rigid octahedron ([SQJ04]). One of the techniques for constructing graphs essentially involves designing strands of DNA that double cover the desired graph by attaching to themselves along complementary base-pairs and thus forming stable double-helixes for the edges of the graph. The questions of when and how this technique can be applied fall naturally into the realm of graph theory, and form the substance of [JS02].

Questions raised in [JS02] about constructing graphs from DNA in this manner include the following:

1. Are there graph invariants (e.g. polynomials) that can give estimates on *ds* and *DS*, respectively the minimum and maximum numbers of DNA strands required to construct the graph?

2. For a fixed graph G, determine $ds(G)$ and $DS(G)$.

3. In general characterize triples (*k*, *m*, *n*) that can appear as *ds*, *DS*, and $\beta_1 + 1$, where β_1 is the first Betti number, for some graph *G*, and for those that can appear, characterize the graphs that realize them.

4. How are the double strand numbers of a cover of a graph related to the double strand numbers of the graph?

We reformulate the topological model of [JS02] in terms of bidirected double coverings of graphs. This allows us to address the first question using the generalized transition polynomial, $q(G,W;x)$, introduced in [E-MS02]. This polynomial is a tool for analyzing the cycle structures in Eulerian graphs. We show that if the weight system *W* is chosen correctly, $q(G,W;x)$ provides a generating function formulation for enumerating DNA strands in graph construction. In particular, we show that the exponents of the highest and lowest degree terms of an appropriately weighted generalized transition polynomial give exactly *ds* and *DS*, and the coefficients enumerate the ways to construct the graph from a given number of strands.

Graphical DNA constructs involve covering each edge of a graph with two complementary strands of DNA, which led us to model the process in terms of bidirected double coverings. This then naturally suggested considering results driven by the cycle double cover conjecture (CDC) to address characterizing graphs with specific minimum strand numbers. The "pure" form of the cycle double cover conjecture states that every graph can be covered by a set of cycles such that each edge appears in exactly two of the cycles. DNA constructs have both stronger and weaker requirements. In the CDC, a cycle may not repeat an edge, while in DNA constructs this is allowed. Also, orientation is not

considered in the CDC, but a DNA strand has a natural orientation, and the construction requires that each edge be covered once in each direction. Nonetheless, results from the CDC can provide valuable bounds for DNA constructions and among the many variations of the CDC are those that provide specific structural information for graphs constructed from DNA strands. In the last section, we apply a theorem of Thomassen [Tho90] that specifies precisely when $ds(G)=1$, i.e. when a graph might be constructed from a single strand of DNA. Then we use results from [HZ98] that give *ds* in terms of the minimum number of retractions in a bidirectional double tracing of the graph and properties of spanning trees.

The following conventions are used throughout this paper. Graphs are finite and may have loops and multiple edges. A graph is said to be Eulerian if all its vertices have even degrees, but connectedness is not required. A cycle is a graph isomorphic to a polygon. We permit *free loops,* which consist of an edge in the shape of a circle, but with no vertex.

DNA constructs in biomolecular computing

We focus our attention on techniques used in [ZS94] and [CS91] where single strands of DNA trace out each edge of a graph, once in each direction, binding to themselves along segments of Watson-Crick complements to form double-helical edges and stable branched junctions at the vertices. The sequences of nucleotides are carefully specified so that only the two 'sides' of an edge are complements of, and hence bind to, each other. [ZS94] and [CS91] provide tables of the single strand DNA sequences used to construct a truncated octahedron and a cube, respectively. We begin with the topological formalism for this process provided in [JS02].

Definition 1: A *thickened graph of G,* $F(G)$, is a compact orientable surface such that *G* is topologically embedded in $F(G)$ as a deformation retract.

Definition 2: The number of boundary components of a thickened graph $F(G)$ is denoted by $\#\partial F(G)$. The *maximal double strand number,* $DS(G)$, is the maximum number of boundary components over all possible thickened graphs of *G*, and the *minimum double strand number,* $ds(G)$, is the minimum possible number of boundary components.

DNA strands along the boundaries, as in Figure 1, bind together to form a double stranded version of the graph. Thus, the minimum and maximum double strand numbers are the minimum and maximum numbers of DNA strands needed to construct the graph. [ZS94] and [CS91] both use a maximal number of

DNA strands, essentially by embedding the graph on a sphere and using the face boundaries as the thickened graph. In Figure 1, the double strand number of the thickened graph on the left is 4, and of the thickened graph on the right is 2. We will show later that $ds(K_4) = 2$.

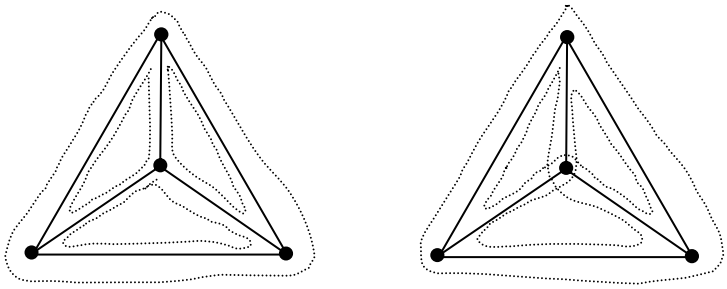


Fig. 1. Two thickened graphs of K_4.

The orientability requirement forced by the DNA strands being paired by their Watson-Crick complementarities allows us to reformulate this topological model in terms of a bidirectional double covering of the graph *G*. The boundaries of thickened graphs will then correspond to certain families of Eulerian subgraphs. This reformulation will also allow us to apply results from specialized cycle double coverings in the final section of this paper.

Definition 3. The *bidirectional double graph of G,* $B(G)$, is the graph that results from replacing each edge of *G* by two oriented edges, one in each direction, as in Figure 2.

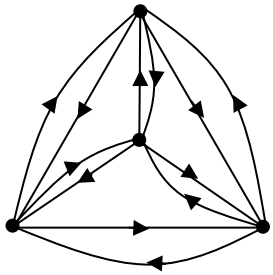


Fig. 2. $B(K_4)$

Note that $B(G)$ is an Eulerian digraph, that is, a directed Eulerian graph such that the in-degree is equal to the out-degree at each vertex.

We will show in the next section that, with this formulation, boundaries of thickened graphs will correspond to certain restricted families of consistently

oriented cycles in $B(G)$ that cover each edge of $B(G)$ exactly once. The number of boundary components is the number of components in the family.

The generalized transition polynomial

The generalized transition polynomial $q(G,W;x)$, a graph polynomial introduced in [E-MS02], is a natural tool for considering families of cycles in Eulerian digraphs. It generalizes the transition polynomials for 4-regular graphs, $Q(G,A,\tau)$, given in [Jae90], that unifies common characteristics of the Martin, Penrose and (via the medial graph of a planar graph) Tutte polynomial. We review only the necessary definitions here, referring the reader to [E-MS02] for details of the algebraic properties and applications of this polynomial. Despite the unavoidable formalism, the basic idea is simply to pair, at each vertex, only those edges which form parts of desirable cycles. A generating function results from pairing edges in all allowable ways and counting the number of cycles formed by each choice of pairings.

Two sets of terminologies appear in the literature for the concepts we discuss here, one using *transition systems* (see [Fle90] for example), and one using *state models* (see [Kau87] for example). We try to indicate corresponding terms in the definitions.

For an edge *e*, we denote the half edge of *e* that is incident to a vertex *v* by e_v. We distinguish the two half edges of a loop. A *pairing* at a vertex *v* is a choice of two half edges incident to *v*.

A *vertex state*, or *transition*, is a choice of local reconfiguration of a graph at a vertex by pairing the half edges incident with that vertex. Figure 3 shows the three possible vertex states of a vertex of degree 4.

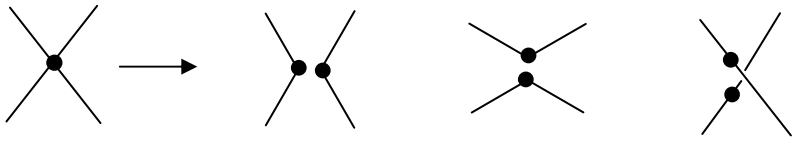


Fig. 3. All possible vertex states for a vertex of degree 4.

A *graph state*, or *transition system*, $S(G)$, is the result of choosing a vertex state at each vertex of degree greater than 2. We will write $St(G)$ for the

set of graph states of *G*, and when the graph is clear from context write *S* for $S(G)$.

A *skein-relation* for graphs is a formal sum of weighted vertex states, together with an evaluation of the terminal forms (the graph states, which here are a disjoint union of 2-regular graphs). See [E-M98] for a detailed discussion of these concepts, which are appropriated from knot theory, in their most general form, and [Yet90] for a general theory of invariants given by linear recursion relations.

A *skein-type*, (or *state model*, or *transition*) *polynomial* is one which is computed by repeated applications of skein relations. See [Jae90] for a comprehensive treatment of these in the case of 4-regular graphs.

Definition 4: A *pair weight* is an association of a value $p(e_v, e'_v)$ to a pair of half edges incident with a vertex *v* in *G*.

Definition 5: A *weight system*, $W(G)$, of an Eulerian graph *G* is an assignment of a pair weight to every possible pair of adjacent half edges of *G*. We will simply write *W* for $W(G)$ when the graph is clear from context.

Definition 6: The *vertex state weight* of a vertex state is $\prod p(e_v, e'_v)$ where the product is over the pairs of half edges comprising the vertex state. Thus, associated to each vertex of degree 2*n* in *G*, there are $\prod_{i=0}^{n-1}(2n-(2i+1))$ not necessarily distinct values, each corresponding to a possible vertex state.

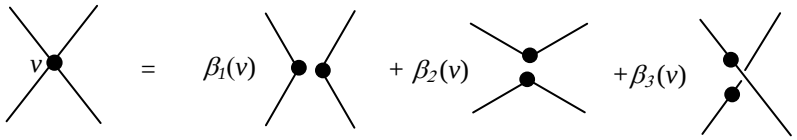
Definition 7: The *state weight* of a graph state *S* of a graph *G* with weight system *W* is $\omega(S) = \prod \omega(v, S)$, where $\omega(v, S)$ is the vertex state weight of the vertex state at *v* in the graph state *S*, and where the product is over all vertices of *G*.

Definition 8 (Recursive Definition): $q(G, W; x)$ may be defined recursively by repeatedly applying the skein relation $q(G, W) = \sum \beta_i q(G_i, W(G_i))$ at any vertex *v* of degree greater than 2. Here the G_i's are the graphs that result from locally replacing a vertex *v* of degree 2*n* in *G* by one of its $\prod_{i=0}^{n-1}(2n-(2i+1))$ vertex states. The β_i's are the vertex state weights. Furthermore, each G_i has a weight system, $W(G_i)$, based on the

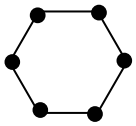
canonical identification between the half edges of G_i and G as follows. For any pair of adjacent half edges in G_i which are not both part of the new vertex state (i.e. which are not both incident to v in G), the pair weight in G_i is the same as it was in G. All the pairs of half edges adjacent to the newly formed vertices of degree 2 in G_i have pair weight equal to 1.

Repeated application of this relation reduces G to a weighted sum of disjoint unions of cycles, (the graph states). These terminal forms are evaluated by identifying each cycle (including any free loops) with the variable x, weighted by the product of the pair weights over all pairs in the cycle.

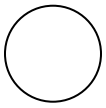
For example, if v is a vertex of degree 4, this definition can be represented pictorially as:



where each of the new vertices of degree two have pair weight 1 for the unique pair of edges incident to them, and where the terminal evaluation is then:



= Cx, where C is the product of all the pair weights for each pair of adjacent edges, and



= x for free loops.

Definition 8 (State Model Definition):

$$q(G,W;x)=\sum_{St(G)}\left(\left(\prod\omega(v,S)\right)x^{k(S)}\right)=\sum_{St(G)}\omega(S)\,x^{k(S)}\,.$$

Note that since this state model definition is clearly equivalent to the recursive definition, and is independent of the order of the vertices to which the recursion is applied, $q(G,W;x)$ is well defined.

Definition 9 (Generating Function Definition): Collecting like terms in the preceding state model definition leads to the generating function form, $q(G,W;x)=\sum f_n(G)\,x^n$. Here, $f_n(G)=\sum\omega(S)$, where the sum is over all states of G with n components.

THEOREM 10. *Let G be a graph with bidirectional doubled graph* $B(G)$. *Define a weight system* $W(B(G))$ *by assigning a pair weight of* 1 *to pairs of half edges at a vertex where one half edge is directed inward, the other outward, both are not associated to the same underlying edge of G. Assign a pair weight of* 0 *otherwise. Then* $q(B(G),W;x)=\sum f_n(B(G))x^n$ *is a generating function for the number of thickened graphs with n boundary components, i.e.* $f_n(B(G))$ *is exactly the number of thickened graphs* $F(G)$ *with* $\#\partial F(G)=n$.

Proof: Observe that with the weight system $W(B(G))$ as defined, each graph state with non-zero coefficient corresponds to a thickened graph $F(G)$ since the cycles in the state form the boundary components of a compact orientable surface for which *G* is a deformation retract. This follows from the fact that only incoming edges are paired with outgoing edges, resulting in consistently oriented cycles, with each edge covered exactly once in each direction. Furthermore, any pair of edges "doubling back", or retracting, receives a weight of 0, and hence any state with such a configuration does not appear in the sum. For example, compare Figure 4a, which shows the possible boundary component configurations in the neighborhood of a vertex of degree 3 as in [JS02, Figure 1], and Figure 4b, which illustrates the weight system $W(B(G))$ at a 6-valent vertex of $B(G)$ resulting from a 3-valent vertex of *G* (dashed lines). All but two of the vertex states have coefficient 0, and the remaining two, with coefficient 1, correspond exactly to possible boundary segments of a thickened graph.

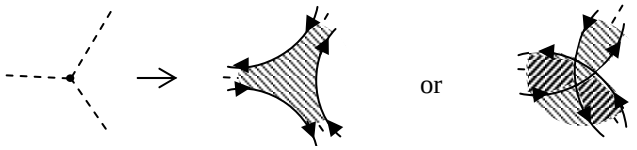


Fig. 4a. Possible local boundary configureations.

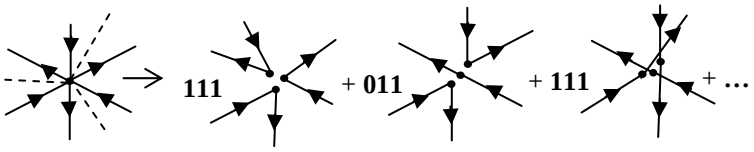


Fig. 4b. Weight system applied to a 6-valent vertex.

On the other hand, the boundary components of a thickened graph, because of the orientability, give a family of consistently directed cycles that cover each edge of the graph once in each direction, without any "doubling back." Thus, each thickened graph corresponds to exactly one of the allowed graph states in $B(G)$. See Figure 5. Note that state weight of each state is 1, so $f_n\left(B(G)\right)$ simply enumerates those with n components. ///

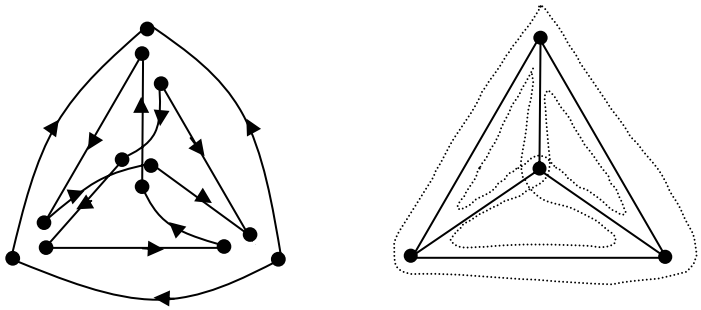


Fig. 5. A graph state of $B(K_4)$ *with weight* 1 *and corresponding thickened graph.*

COROLLARY 11. *Let G be a graph with M and m, respectively, the highest and lowest powers of x appearing in* $q\left(B(G),W;x\right)$. *Then* $DS(G)=M$ *and* $ds(G)=m$. *Furthermore,* $f_M\left(B(G)\right)$ *and* $f_m\left(B(G)\right)$ *give the number of ways that G may be constructed from either a maximal or minimal number of strands.*

Covering results applied to DNA constructions

The cycle double cover conjecture states that "every bridgeless graph has a family of cycles so that each edge appears in exactly two cycles in the family" (see [Sey79] and [Sze73]).

Cycle double covers do not quite model DNA constructions, since DNA strands, unlike cycles, may cover the same edge twice. Furthermore, the natural orientation of DNA requires that each edge be covered once in each direction, but the need to maintain adjacencies in the underlying graph prohibits traversing an edge in one direction immediately followed by traversing it in the reverse direction (a *retracting*).

However, in [Tho90], Thomassen characterized graphs with bidirectional, retracting-free, double tracings in terms of their spanning trees. A bidirectional

double tracing is a walk (so edges may be repeated) in a graph that traces each edge exactly once in each direction. Thus, such a retracting-free tracing provides exactly the thickened graph or graph state of the doubled graph required for a single DNA strand to construct the graph. Subsequently, Hongbing and Zhu [HZ98] extended this result by relating the minimum number of closed walks in a retracting-free oriented walk double covering of a graph to both the minimum number of retractions in any bidirectional double tracing of it and to properties of its spanning trees.

Since we have reformulated the thickened graph in terms of bidirectional double graphs, we are now able simply to apply these results directly to the questions raised in [JS02].

THEOREM 12. $ds(G)=1$ *if and only if G is connected, has no vertex of degree* 1, *and has a spanning tree T such that every connected component of* $G-E(T)$ *has an even number of edges or a vertex v with* $\deg_G(v)\geq 4$.

Proof: This follows immediately from [Tho90, Theorem 3.3], which states that under exactly these conditions does *G* have a retraction-free bidirectional double tracing. Since such a tracing gives a graph state of $B(G)$ with one component and coefficient 1, and vice versa, this is equivalent to $ds(G)=1$. ///

As an example, note that K_4 has, up to isomorphism, only two spanning trees. The components remaining when the edges of either tree are removed do not meet the criteria of Theorem 12, and hence $ds(K_4)\neq 1$. Since we exhibited a graph state of $B(G)$ with two components in Figure 5, it follows that $ds(K_4)=2$.

The results of [HZ98] further determine a precise characterization of graphs where $ds(G)=m$ for any positive integer *m*, but we first need to provide some definitions from that paper. Let $r(G)$ be the minimum number of retractions in any bidirectional double tracing of *G*. Let $c(G)$ be the minimum number of closed walks in a retracting-free oriented walk double covering of *G*, and note that $c(G)=ds(G)$. Let *T* be a spanning tree of *G*, and write $o(T)$ for the number of components of $G-E(T)$ with an odd number of edges and no vertices such that $\deg_G(v)>3$. These are referred to as the *odd components* of *T*. Let $o(G)$ be the minimum of $o(T)$ over all spanning trees of *G*.

THEOREM 13. *If G is a connected, non-cycle graph with no vertices of degree* 1, *then for any positive integer m,* $ds(G) = m$ *and only if G has a spanning tree with* $m-1$ *odd components, and no spanning tree with fewer than* $m-1$ *odd components.*

Proof: This follows from combining [HZ98, Theorem 2] and [HZ98, Theorem 5]. [HZ98, Theorem 2] states that if *G* is a connected, non-cycle graph with no vertices of degree 1, then $r(G) = c(G) - 1$, and [HZ98, Theorem 5] states that $r(G) = o(G)$. Since $c(G) = ds(G)$, we have that $ds(G) = m$ if and only if $o(G) = m$ and the result follows from the definition of $o(G)$. ///

In closing we note that many other natural graph theoretical questions arise out of this application in biomolecular computing, beyond those briefly mentioned in the introduction. The authors of [HK98] provide an polynomial time algorithm for determining $r(G)$ (and hence $ds(G)$), but note that the running time is about $O(n^{40})$ and call for a more direct algorithm. To this call we add the impetus of this application in biomolecular computing. Also, the octahedron of [SQJ04] uses a 4-strand covering of the graph, with one double helix wrapped around another to form an edge. Thus, it might also be fruitful to consider generalizing the above results to retracting-free oriented walk 2*n* covering of a graph.

Bibliography

[Adl94] L. M. ADLEMAN, Molecular Computation of Solutions to Combinatorial Problems. *Science*, **266** (5187) Nov. 11 (1994) 1021-1024.

[CS91] J. CHEN, N. C. SEEMAN, Synthesis from DNA of a molecule with the connectivity of a cube, *Nature*, **350** (1991) 631-633.

[E-M98] J. A. ELLIS-MONAGHAN, New Results for the Martin Polynomial, *Journal of Combinatorial Theory*, Series B, **74** (1998) 326-352.

[E-MS02] J. A. ELLIS-MONAGHAN, I. SARMIENTO, Generalized Transition Polynomials, *Congressus Numerantium*, **155** (2002) 57-69.

[Fle90] H. FLEISCHNER, Eulerian Graphs and Related Topics, Part 1, Volume I, *Ann. Discrete Math.* **45** (1990).

[HZ98] F. HONGBING, X ZHU, Oriented Walk Double Covering and Bidirectional Double Tracing, *J. Graph Theory* **29** (1998) 89-102.

[Jae90] F. JAEGER, On Transition Polynomials of 4-regular Graphs, *Cycles and Rays (Montreal, PQ, 1987)* 123-150, NATO Adv. Sci. Inst. Ser. C: Math. Phys. Sci., 301, *Kluwer Acad. Publ., Dordrecht,* 1990.

[JS02] N. JONOSKA, M. SAITO, Boundary Components of Thickened Graphs, *Revised papers of the 7th International Meeting on DNA based computers* (eds. N. Jonoska, N.C. Seeman). Springer LNCS. **2340** (2002), 70-81.

[Kau87] L. KAUFFMAN, State Models and the Jones Polynomial, *Topology,* **26** No. 3 (1987) 395-407.

[Sey79] P. D. SEYMOUR, Sums of Circuits, in *Graph Theory and Related Topics* (J. A. Bondy and U. S. R. Murty, eds.), Academic Press, New York, (1979) 342-355.

[SQJ04] W. M. SHIH, J. D. QUISPE, G. F. JOYCE, A 1.7 Kilobase Single-stranded DNA that Folds into a Nanoscale Octahedron, *Nature*, **427** (2004) 618-621.

[Sze73] G. SZEKERES, Polyhedral Decompositions of Cubic Graphs, *Bull. Austral. Math. Soc.* **8** (1973) 367-387.

[Tho90] C. THOMASSEN, Bidirectional Retraction-Free Double Tracings and Upper Embeddability of Graphs, *J. Combin. Theory, Ser. B* **50** (1990) 198 -207.

[Yet90] D. YETTER, On Graph Invariants Given by Linear Recurrence Relations, *J. Combin. Theory, Ser. B* **48** (1990) 6-18.

[ZS94] Y. ZHANG, N. C. SEEMAN, Contruction of a DNA-Truncated Octahedron, *J. Am. Chem. Soc.* **116** (1994) 1661-1669.