

Certificates for properties of stability polynomials of graphs

Ranjie Mo Graham Farr Kerri Morgan

Clayton School of Information Technology
Monash University
Melbourne, Australia

{ranjie.mo,graham.farr,kerri.morgan}@monash.edu

Submitted: Aug 30, 2013; Accepted: Mar 14, 2014; Published: Mar 24, 2014

Mathematics Subject Classifications: 05C31, 05C69, 05E99, 68Q17

Abstract

A *stable* (or *independent*) set is a set of vertices where no two of the vertices in the set are adjacent. The *stability polynomial* $A(G; p)$ of a graph G is the probability that a set of randomly chosen vertices is stable where the probability of each vertex being chosen is p , with choices independent. This polynomial is analogous to the chromatic polynomial in a precise sense. This paper considers factorisation of stability polynomials, following work by Morgan and Farr on factorisation of the chromatic polynomial. The stability polynomial $A(G; p)$ is said to have an *s-factorisation* with *s-factors* H_1 and H_2 if $A(G; p) = A(H_1; p)A(H_2; p)$. This clearly occurs when G is a disjoint union of H_1 and H_2 . We find many other cases where such factorisation occurs even when G is connected. We find 152 different s-factorisations of connected graphs of order at most 9, and two infinite families. We introduce *certificates of s-factorisation* to explain s-factorisations in terms of the structure of G . Short certificates for s-factorisations of connected graphs of order at most 6 are found. Upper bounds for the lengths of the certificates of s-factorisations are given. We also use certificates to explain *stability equivalence*, when two graphs have the same stability polynomial. We give certifications of stability equivalence for two infinite families of graphs.

Keywords: stability polynomial; chromatic polynomial; certificate; stability equivalence; s-factorisation

1 Introduction

The *chromatic polynomial* $P(G; \lambda)$ of a graph G gives the number of different ways colours can be assigned to the vertices of a graph such that no pair of adjacent vertices get the same colour [1].

In this paper we will consider one of its relatives, the stability polynomial, and study its factorisation properties. Let G be a graph and $p \in [0, 1]$ be the probability of each vertex being chosen. The vertex choices are made independently. The *stability polynomial* $A(G; p)$ of a graph G is the probability that a set of randomly chosen vertices is stable (i.e., no two adjacent) [4, 8]. This polynomial plays the same role for graphic 2-polymatroids as the chromatic polynomial plays for graphs [8].

Let H_1 and H_2 be graphs. The stability polynomial of graph G is said to have an *s-factorisation* with *s-factors* H_1 and H_2 if

$$A(G; p) = A(H_1; p)A(H_2; p). \quad (1)$$

The graph G is said to be *s-factorised* or G is *s-factorisable*. In this research we will focus on the s-factorisations of connected graphs because it is known that any disconnected graph always has an s-factorisation [3]. In principle, s-factorisations of graph G could be used to reduce the time taken to calculate $A(G; p)$ by calculating the product of the stability polynomial of two smaller graphs H_1 and H_2 .

As a polynomial is an algebraic object, the study of its algebraic properties, including factorisation, is important. But there has been little research done on the factorisation of graph polynomials. Morgan and Farr [12] pioneered the use of certificates to study the algebraic properties of chromatic polynomials. In this paper, we introduce certificates for stability polynomials, and establish some of their fundamental properties.

One objective of this research is to search for s-factorisations of connected graphs and find short certificates to explain these cases. We searched all the stability polynomials of connected graphs of order at most 9 and found 17,461,965 s-factorisations corresponding to 273,192 graphs. These correspond to 152 different stability polynomials that have s-factorisation. We use *certificates* to explain these s-factorisations. A *certificate* is a sequence of expressions S_1, S_2, \dots, S_k satisfying the following properties. The first expression S_1 is the graph G . Each expression S_i is obtained from S_{i-1} by applying a property of the stability polynomial or an algebraic property. If the last expression is the graph H then we have a *certificate of equivalence*. If the last expression S_k is the product of the graphs H_1 and H_2 then we have a *certificate of factorisation*. A *certificate step* is a rule that uses some properties of the graph polynomial or algebraic properties to transform one expression to another [9]. A single certificate step is usually not enough to explain an s-factorisation.

We found s-factorisable connected graphs have order ≥ 6 . Short certificates of s-factorisations for all stability polynomials of graphs of order at most 8 are given. Short certificates for two infinite families of s-factorisations were found.

Another objective of this research is to find bounds for the lengths of certificates of s-factorisations and stability equivalence. We find upper bounds on the lengths of certificates of s-factorisations and stability equivalence, which are both exponential. The lengths of the short certificates we found for two infinite families are all constant values, which are significantly lower than the theoretical upper bounds. But we also give evidence that the lengths of certificates of equivalence cannot always be much shorter than quadratic.

The length of certificates may have implications for the computational complexity of stability equivalence and factorisation, which we discuss.

We also find certificate schemas for some certificates of s-factorisations. If there is a set of graphs that have s-factorisations which can be explained by certificates that use the same sequence of steps, these certificates will be generalised to a *certificate schema*. We found two certificate schemas by analysing the patterns in short certificates of s-factorisations.

This paper contains ten sections. Section 1 gives an overview of this research and introduces some definitions and notation. Section 2 introduces some properties of the stability polynomial, shows the relationship between the stability polynomial and some other graph polynomials and studies the previous research on graph polynomial factorisation. Section 3 presents computational results. Section 4 gives the certificate steps of the stability polynomial. Section 5 finds upper bounds for the length of certificates of stability equivalence. Section 6 examines the relationship between certificate length and computational complexity. It also gives a provisional lower bound on the lengths of some certificates of equivalence. Short certificates of stability equivalence and s-factorisations are presented in Section 7 and 8. Section 9 generalises some short certificates of s-factorisations and gives two certificate schemas. Section 10 summarises our work and points to some directions of future research.

1.1 Definitions and notation

Graphs defined in this paper are listed in Table 1. Some notation used in this paper is listed as follows. \overline{K}_n : the graph of order $n \geq 0$ with no edges; C_n : the cycle of order n ; K_n : the complete graph of order n ; P_n : path graph of order n with $n - 1$ edges; $G - v$: a graph obtained from G by deleting vertex v ; $G \cup G'$: disjoint union of graphs G and G' ; $G + e$: a graph obtained from G by adding the edge e ; $G + G'$: any graph obtained from G and G' by choosing one vertex from each graph and adding an edge between this pair of vertices. A *null graph* is a graph with no edges. The *independence number* $\alpha(G)$ is the size of a maximum stable set in graph G [15].

2 Background

2.1 Relationships between the stability polynomial and other graph polynomials

The stability polynomial is closely related to the independence polynomial, the chromatic polynomial and other polynomials.

The independence polynomial

The *independence polynomial* $I(G; x)$ is a generating polynomial for the number of stable

Notation	Explanation	Location in paper
Q_n	The graph of order n with $E(Q_n) = \{v_0v_2\} \cup \{v_iv_{i+1} : 0 \leq i \leq n-2\}$, $n \geq 3$	Thm 11, Cor. 13 Cor. 20, Fig. 6
Y_n	The graph of order n with $E(Y_n) = \{v_0v_2\} \cup \{v_iv_{i+1} : 1 \leq i \leq n-3\} \cup \{v_{n-1}v_1\}$ where $n \geq 4$	Cor. 13, Fig. 8
K_n^*	The graph of order $2n$ with $E(K_n^*) = \{v_iv_{n+i} : 0 \leq i \leq n-1\}$ $\cup \{v_jv_k : 0 \leq j, k \leq n-1, j \neq k\}$, $n \geq 1$	Cor. 14, Fig. 9
$\Upsilon_{m,n}$	The graph of order n with $E(\Upsilon_{m,n}) = \{v_iv_{i+1} : 0 \leq i \leq n-2\}$ $\cup \{v_{m-1}v_j : 0 \leq j \leq m-3\}$ where $3 \leq m \leq n$	Thm 15, Fig. 10
$K_{n \sim n}$	$K_{n \sim n} = K_n + K_n$	Cor. 19, Fig. 13

Table 1: Graphs defined in this paper

sets s_i of cardinality i [6]. It can be expressed as

$$I(G; x) = \sum_{X \text{ stable}} x^{|X|} = \sum_{i=0}^{\alpha(G)} s_i x^i \quad (2)$$

where $\alpha(G)$ is the size of a maximum stable set in G .

$A(G; p)$ and $I(G; x)$ are related to each other by the following transformation. It is well known that

$$A(G; p) = (1-p)^n I\left(G; \frac{p}{1-p}\right) \quad (3)$$

and

$$I(G; x) = (1+x)^{-n} A\left(G; \frac{x}{1+x}\right) \quad (4)$$

When $|G| = |H_1| + |H_2|$, any results on s-factorisation $A(G; p) = A(H_1; p)A(H_2; p)$ can be translated to analogous factorisations for the independence polynomial, the clique polynomial and the positive matching polynomial.

The chromatic polynomial

The chromatic polynomial of a graph $G = (V, E)$ is

$$P_G(\lambda) = \sum_{X \subseteq E} (-1)^{|X|} \lambda^{n-\rho(X)} \quad (5)$$

where n is the number of vertices of G , and $\rho(X)$ is the rank of X .

It follows from (5) that

$$\frac{P_G(\lambda)}{\lambda^n} = \sum_{X \subseteq E} (-1)^{|X|} \lambda^{-\rho(X)}. \quad (6)$$

The stability polynomial can be written [8]

$$A(G; p) = \sum_{X \subseteq E} (-1)^{|X|} p^{\nu(X)} \quad (7)$$

where $\nu(X)$ is the number of vertices that are incident with edges in X . We note that ν is the rank function of a 2-polymatroid.

When $p = \frac{1}{\lambda}$, (7) becomes

$$A\left(G; \frac{1}{\lambda}\right) = \sum_{X \subseteq E} (-1)^{|X|} \lambda^{-\nu(X)}. \quad (8)$$

This was first shown in [8]. When we compare (5) and (8), we note that the right hand sides only differ in the rank function: ρ is the rank of the cycle matroid of G while $\nu(X)$ is the rank function of a 2-polymatroid. So the stability polynomial is a cousin of the chromatic polynomial: it plays the same role for graphic 2-polymatroid that the chromatic polynomial plays for matroids. This is what motivates us to investigate factorisation and equivalence for stability polynomials, having already done so for chromatic polynomials [9, 11, 12]

2.2 Properties of the stability polynomial

In this section we give five interesting properties of the stability polynomial. They enable us to compute the stability polynomial of a graph recursively. They may be found, in different forms, in one or more of [3, 6, 14]. We will use these properties in our certificate steps in Section 4.

Theorem 1. ([6], in terms of $I(G; x)$) For any vertex $v \in V(G)$ which is not incident to a loop, there is

$$A(G; p) = (1 - p)A(G - v; p) + p(1 - p)^d A(G - v - N(v); p) \quad (9)$$

where d is the degree of vertex v and $N(v)$ is the neighbour set of v .

Theorem 2. [14] For any edge $uv \in E(G)$,

$$A(G; p) = (1 - p)A(G - u; p) + (1 - p)A(G - v; p) - (1 - p)^2 A(G - u - v; p).$$

Proposition 3. [6] Let H_1 and H_2 be two disjoint graphs. If $G = H_1 \cup H_2$, then

$$A(G; p) = A(H_1; p)A(H_2; p). \quad (10)$$

Proposition 4. *Let G be a graph with loops, L is the set of vertices incident to loops. Then*

$$A(G; p) = (1 - p)^{|L|} A(G - L; p).$$

Theorem 5. [3] *For any edge $uv \in E(G)$,*

$$A(G; p) = A(G \setminus uv; p) - p^2(1 - p)^{d_{uv}} A(G - u - v - N(u) - N(v); p)$$

where $d_{uv} = |(N(u) \setminus \{v\}) \cup (N(v) \setminus \{u\})|$.

Proposition 3 is a basic case of stability polynomial factorisation. It gives a case of factorisation when the graphs H_1 and H_2 are disjoint. In this project we will find s-factorisations of graphs that do not have disjoint components but still satisfy (10). Studying s-factorisations may result in new rules for stability polynomials. A new rule is a proposition which can be explained by a certificate schema.

2.3 Graph polynomial factorisation

We review research on factorisation of chromatic polynomials and consider the relationship to the work of this paper.

2.3.1 Chromatic factorisations

The chromatic polynomial $P_G(\lambda)$ is said to have *chromatic factorisation* [9] with *chromatic factors* H_1 and H_2 if

$$P_G(\lambda) = \frac{P_{H_1}(\lambda)P_{H_2}(\lambda)}{P_{K_r}(\lambda)} \quad (11)$$

where H_1 and H_2 are not isomorphic to K_r and have $\chi(H_i) \geq r, i = 1, 2$. A graph G has a chromatic factorisation if $P_G(\lambda)$ does. Any *clique-separable* graph, that is, a graph that can be obtained by identifying r -clique in H_1 with an r -clique in H_2 , has a chromatic factorisation. It follows that any graph that is *chromatically equivalent* to (i.e., has the same chromatic polynomial as) a clique-separable graph also has a chromatic factorisation. Morgan and Farr [12, 11] defined a graph to be *strong non-clique-separable* if it is not chromatically equivalent to any clique-separable graph. Perhaps surprisingly, they found graphs that are strongly non-clique-separable, yet have a chromatic factorisation, identifying all such cases with at most 9 vertices, as well as an infinite family.

They introduced the concept of a *certificate* in order to explain these chromatic factorisations [12]. Certificates have also been used to explain chromatic equivalence [10].

A *certificate of chromatic factorisation* is a sequence of expressions P_0, P_1, \dots, P_i where $P_0 = P_G(\lambda)$ and $P_i = P_{H_1}(\lambda)P_{H_2}(\lambda)/P_{K_r}(\lambda)$ such that each $P_j, 1 \leq j \leq i$, is obtained from P_{j-1} by applying an algebraic property or a property of the chromatic polynomial. This certificate is used to explain why a given graph G has a chromatic factorisation with chromatic factors H_1 and H_2 .

Similarly a *certificate of chromatic equivalence* is a sequence of expressions P_0, P_1, \dots, P_i where $P_0 = P_G(\lambda)$ and $P_i = P_H(\lambda)$ such that each P_j , $1 \leq j \leq i$, is obtained from P_{j-1} by applying an algebraic property or a property of the chromatic polynomial. This certificate is used to explain why graphs G and H have the same chromatic polynomial.

Certificates may give a means of verifying these properties of chromatic polynomials without computing these polynomials from scratch (which is #P-hard). They give explanations of chromatic factorisation and chromatic equivalence that are purely graph theoretic, making no explicit use of polynomial algebra.

Certificates have been used to explain the chromatic factorisations of all strongly non-clique-separable graphs of order at most 9 [12], to explain chromatic factorisation of an infinite family of strongly non-clique-separable graphs [11], to show that every non-bipartite graph can be the chromatic factor of a chromatic factorisation $P_{H_1}(\lambda)P_{H_2}(\lambda)/P_{K_3}(\lambda)$ [13] and to find pairs of chromatically equivalent graphs [10]. Interestingly, although theoretically these certificates could be exponentially long [12], in practice they are remarkably short.

The length of these certificates have implications for the computational complexity of testing for chromatic equivalence, and of testing for the existence of a chromatic factorisation. In each case, if certificate length is bounded by a polynomial in graph size, then the problem belongs to NP [2].

2.3.2 Connections with current research

As certificates have proved to be a powerful tool in explaining algebraic properties such as factorisation and equivalence for the chromatic polynomial and we have seen that the stability polynomial is closely related to the the chromatic polynomial (see (6) and (7)), we use certificates to explain algebraic properties of the stability polynomial, namely s -factorisation and stability equivalence.

While the certificates used for the chromatic polynomial used certificate steps based on properties of the chromatic polynomial, the certificates used in our research are based on properties of the stability polynomial. In Section 4 we define a set of certificate steps based on the properties for the stability polynomial given in Section 2.2. We then use these steps to give certificates of s -equivalence for some infinite families of graphs in Section 7 and certificates of s -factorisation for some infinite families of graphs in Section 8.

Theorem 1, 2 and 5 in Section 2.2 can be used to construct recursive algorithms for the computation of the stability polynomial. We implemented an algorithm using (9) to calculate the stability polynomial. According to Farr [5], the time complexity of this algorithm is $O(2^n \text{ poly}(n))$, which is more efficient than the usual recursive deletion-contraction algorithm for computing the chromatic polynomial. This means that the stability polynomial can be computed for graphs of larger order than the chromatic polynomial.

3 Computational results on s-factorisations

By Proposition 3, any disconnected graph G has an s-factorisation $A(G; p) = A(H_1; p) \cdot A(H_2; p)$ where H_1 and H_2 are two disjoint components of G . However, it was not known if any connected graph has an s-factorisation. In this paper we focus on s-factorisations of connected graphs.

We calculated the stability polynomials of all connected non-isomorphic graphs of order at most 9 using a recursive algorithm based on (9). The base case of this recursive algorithm is $A(\overline{K}_n; p) = 1$. If G has edges, then we compute $A(G; p)$ as the sum of two recursive calculations, $A(G - v; p)$ and $A(G - v - N(v); p)$. We then exhaustively searched for s-factorisations.

The computational results show that there are 17,461,965 s-factorisations corresponding to 28,576 s-factorisable graphs. These s-factorisable graphs have 152 different stability polynomials. Detailed results are given in Table 2 and Table 3.

Order	#Graphs	#Different polys	#s-factorisable graphs	#Different s-factorised polys
2	1	1	0	0
3	2	2	0	0
4	6	5	0	0
5	21	13	0	0
6	112	38	4	3
7	853	116	50	9
8	11117	391	955	42
9	261080	1438	27567	143

Table 2: Graphs and the stability polynomials

For each order, Table 2 gives the number of non-isomorphic graphs, the number of different stability polynomials, the number of graphs that have s-factorisation and the number of different stability polynomials that can be s-factorised. Table 3 gives the accumulated data about s-factorisations of graphs, including the number of connected graphs that can be s-factorised (Column C), the number of different stability polynomials that can be s-factorised (Column D) and the number of different s-factorisations (Column F).

Table 2 shows that connected graphs which have s-factorisations have order ≥ 6 . By analysing the ratio between the number of graphs of order n and the number of different stability polynomials of these graphs, we can see that when the order of graphs increases, the probability that different graphs have the same stability polynomials gets higher as n increase.

Table 3 shows that 28,576 of the 273,192 connected graphs of order 2 to 9 can be s-factorised. These s-factorisable graphs correspond to 152 different stability polynomials, and have 152 different s-factorisations. Another interesting fact given by Table 3 is that

Order	A	B	C	D	E	F
2	1	1	0	0	0	0
2-3	3	3	0	0	0	0
2-4	9	7	0	0	0	0
2-5	30	17	0	0	0	0
2-6	142	45	4	3	18	3
2-7	995	130	54	10	741	10
2-8	12112	420	1009	44	73617	44
2-9	273192	1503	28576	152	17461965	152

Table 3: s-factorisations

(A) Number of non-isomorphic graphs, (B) Number of different stability polynomials, (C) Number of s-factorisable graphs, (D) Number of different s-factorisable stability polynomials, (E) Number of s-factorisations, (F) Number of different s-factorisations.

the values in Column D are the same as the values in Column F, which means the number of different s-factorisable stability polynomials is equal to the number of different s-factorisations. This finding may suggest that a stability polynomial can be s-factorised in at most one way.

Our computational results confirmed that there are many cases of s-factorisation. A closer examination of the graph data summarised here also gave insights into possible families of graphs that have s-factorisations. By analysing the structure of these graphs, we identified several infinite families of graphs that have s-factorisations and found certificates of s-factorisation to explain the factorisations.

4 Certificate steps

Just as for the chromatic polynomial, we introduce certificates for the stability polynomial in order to give graph theoretic explanations of properties of these polynomials.

In this section we give certificate steps based on the properties of the stability polynomial which are presented in Section 2.2. We then define certificates of s-factorisation and certificates of s-equivalence.

CS1 $A(\overline{K}_n; p)$ becomes 1.

CS2 1 becomes $A(\overline{K}_n; p)$.

Figure 1 illustrates CS1 and CS2. In this paper, we use the convention of representing the stability polynomial of a graph by a picture of the graph itself.

CS3 $A(G; p)$ becomes $(1 - p)A(G - v; p) + p(1 - p)^d A(G - v - N(v); p)$ for some vertex $v \in V(G)$ where d is the degree of vertex v . (Theorem 1)

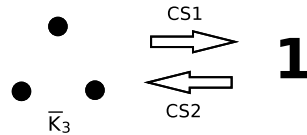


Figure 1: $A(\overline{K}_n; p) \leftrightarrow 1, n = 3$

CS4 Let $U = \{u_1, u_2, u_3, \dots, u_d\}$ be d distinct vertices in $V(G_1)$. Then $(1-p)A(G_1; p) + p(1-p)^d A(G_2; p)$ becomes $A(G; p)$ where $G = (V, E)$ is isomorphic to the graph G' with vertex set $V' = V(G_1) \cup \{v\}$, and edge set $E' = E(G_1) \cup \{vu_i : \text{where } u_i \in U\}$, and where $G_2 \cong G_1 - U - \{v\}$. (Theorem 1)

Figure 2 illustrates CS3 and CS4.

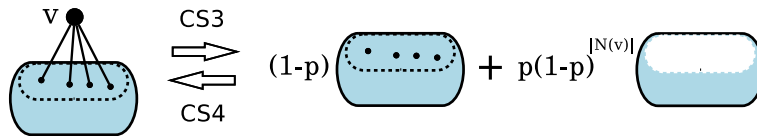


Figure 2: $A(G; p) \leftrightarrow (1-p)A(G-v; p) + p(1-p)^d A(G-v-N(v); p)$

CS5 $A(G; p)$ becomes $(1-p)A(G-u; p) + (1-p)A(G-v; p) - (1-p)^2 A(G-u-v; p)$ for some $uv \in E(G)$. (Theorem 2)

CS6 Let $N_1 \subseteq V(G_3)$ and $N_2 \subseteq V(G_3)$. Then $(1-p)A(G_1; p) + (1-p)A(G_2; p) - (1-p)^2 A(G_3; p)$ becomes $A(G; p)$ for some $uv \in E(G)$ where the following conditions are satisfied:

- (1) $G_1(V_1, E_1)$ is isomorphic to the graph with $V'_1 = V(G_3) \cup \{u\}$ and $E'_1 = E(G_3) \cup \{ua : a \in N_1\}$,
- (2) $G_2(V_2, E_2)$ is isomorphic to the graph with $V'_2 = V(G_3) \cup \{v\}$ and $E'_2 = E(G_3) \cup \{vb : b \in N_2\}$,
- (3) $G(V, E)$ is isomorphic to the graph with $V' = V(G_3) \cup \{u\} \cup \{v\}$ and $E' = E(G_3) \cup \{ua : a \in N_1\} \cup \{vb : b \in N_2\}$.

(Theorem 2)

Figure 3 illustrates CS5 and CS6.

CS7 $A(G; p)$ becomes $A(G \setminus e; p) - p^2(1-p)^d A(G-u-v-N(u)-N(v); p)$ for some $uv \in E(G)$ and $d = |(N(u) \setminus \{v\}) \cup (N(v) \setminus \{u\})|$. (Theorem 5)

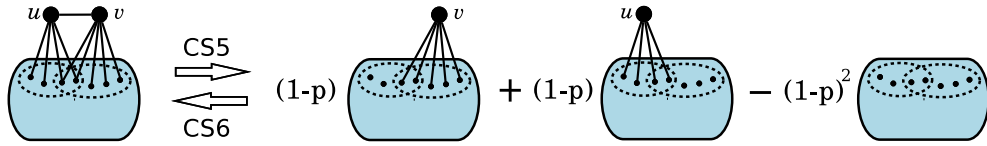


Figure 3: $A(G; p) \leftrightarrow (1 - p)A(G - u; p) + (1 - p)A(G - v; p) - (1 - p)^2 A(G - u - v; p)$

CS8 $A(G_1; p) - p^2(1 - p)^d A(G_2; p)$ becomes $A(G; p)$ where $G(V, E)$ is isomorphic to the graph with $V = V(G_2) \cup \{u\} \cup \{v\}$ and $E = E(G_2) \cup \{ua : a \in N_1\} \cup \{vb : b \in N_2\}$ when $N_1 \subseteq V(G_2), N_2 \subseteq V(G_2)$ and $d = |N_1 \cup N_2|$. (Theorem 5)

Figure 4 illustrates CS7 and CS8.

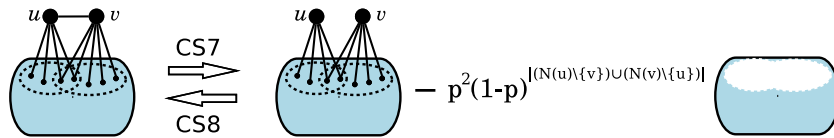


Figure 4: $A(G; p) \leftrightarrow A(G \setminus e; p) - p^2(1 - p)^d A(G - u - v - N(u) - N(v); p)$

CS9 $A(G; p)$ becomes $A(G_1; p)A(G_2; p)$ where G_1, G_2 are two disjoint graphs and $G_1 \cup G_2$ is isomorphic to G . (Proposition 3)

CS10 $A(G_1; p)A(G_2; p)$ becomes $A(G; p)$ where G_1, G_2 are two disjoint graphs and G is isomorphic to $G_1 \cup G_2$. (Proposition 3)

Figure 5 illustrates CS9 and CS10.

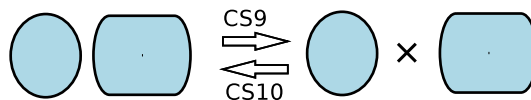


Figure 5: $A(G; p) \leftrightarrow A(G_1; p)A(G_2; p)$

CS11 Algebraic step.

There is some room for variety in the choice of our certificate steps. Some of the given steps could be dropped, since they could be proved (at some cost, in terms of extra steps) using others. The steps we have chosen correspond to the simplest recurrence relations that can be given for the stability polynomial.

These certificate steps can be used to explain s-factorisations and stability equivalences. A *certificate* is a sequence of expressions P_0, P_1, \dots, P_i , such that, for every $j, 1 \leq j \leq i$, P_j in the sequence is obtained from P_{j-1} by applying one of the certificate steps of stability polynomials. A *certificate of stability equivalence* for graph G and H has $P_0 = A(G; p)$ and $P_i = A(H; p)$. A *certificate of stability factorisation* has $P_0 = A(G; p)$ and $P_i = A(G_1; p)A(G_2; p)$. The certificate steps which are obtained from the properties of the stability polynomial are given in CS1, \dots , CS11.

We consider how the length of the certificates depends on the order of the corresponding graphs.

5 Upper bound for certificate length

In this section we will discuss the lengths of certificates for stability equivalences and s-factorisations. We use a naive approach to find an upper bound on length of certificates. In practice we found much shorter certificates. To show two graphs are stability equivalent we must transform expression $A(G; p)$ to expression $A(H; p)$ using certificate steps. We do this in two phases. First we transform the stability polynomial of G to an expression in null graphs. Then we transform this expression into the stability polynomial of H .

To show $A(G; p)$ has factorisation $A(H_1; p)A(H_2; p)$ we must transform expression $A(G; p)$ to expression $A(H_1; p)A(H_2; p)$. We can also first transform the stability polynomial of G into expression in null graphs and then transform this expression into product of the stability polynomial of H_1 and the stability polynomial of H_2 .

First of all, we use a naive approach to express the stability polynomial of graph G in terms of null graphs and find an upper bound for the number of certificate steps required to do so. For any graph G , if there exists a vertex $v \in V(G)$ with degree ≥ 1 , we apply certificate step CS3 on v and transform $A(G; p)$ to $(1-p)A(G-v; p) + p(1-p)^{\deg v} A(G-v-N(v); p)$. If graphs $G-v$ and $G-v-N(v)$ are not null graphs, we recursively apply CS3 on these graphs until all graphs in the expression are null graphs.

Let t_n be the maximum number of certificate steps required to express the stability polynomial of any graph on n vertices in terms of null graphs. This is the same as the number of applications of Theorem 1 required to compute the polynomial recursively. Observe that $t_1 = 0$ and $t_2 = 1$.

To get a simple upper bound for t_n , observe that a single application of CS3 produces a graph of $n-1$ vertices, which requires t_{n-1} further steps, and a graph of $n-1-\deg v \leq n-2$ vertices (where v is the vertex at which CS3 is used on G , which we choose so that $\deg v \geq 1$), which requires $\leq t_{n-2}$ further steps (unless G is a null graph, but in that case we are done already). So $t_n \leq 1 + t_{n-1} + t_{n-2}$. Putting $s_n := t_n + 1$, we have $s_n \leq s_{n-1} + s_{n-2}$. Hence s_n is bounded above by a Fibonacci sequence. It follows that $t_n = O(\varphi^n)$, where $\varphi = (1 + \sqrt{5})/2 \simeq 1.618$ is the golden ratio.

A little more care gives a stronger upper bound.

Lemma 6. $t_n = O(\psi^n)$ where $\psi \simeq 1.380$ is the positive root of $x^4 - x^3 - 1 = 0$.

Proof. (sketch)

We suppose that, throughout the computation, we always use CS9 whenever we have a disconnected graph, and CS3 whenever the graph is connected.

If we encounter a graph of maximum degree ≤ 2 , then it is easily dealt with. It is routine to show that the computation for such a graph requires only a linear number of certificate steps.

If the maximum degree is ≥ 3 , we choose a vertex v of degree ≥ 3 and apply CS3 there. This gives $t_n \leq 1 + t_{n-1} + t_{n-4}$. With $s_n = t_n + 1$ as before, we have $s_n \leq s_{n-1} + s_{n-4}$. Using standard theory of recurrence relations, we obtain $s_n = O(\psi^n)$, where ψ is the positive root of the characteristic equation of the recurrence, $x^4 - x^3 - 1 = 0$. \square

The first few values of t_n are given in Table 4.

We use t_n to give an upper bound on the lengths of certificates of stability equivalence.

Theorem 7. *The maximum length e_n of a certificate of stability equivalence for graphs of order $\leq n$ satisfies $e_n \leq 2t_n + 2n - 1$ and $e_n = O(\psi^n)$.*

Proof. Suppose $A(G; p) = A(H; p)$, we can transform G to a sum of null graphs in $\leq t_n$ applications of CS3, then use $\leq n - 1$ applications of CS1 to transform to $E_G(p)$, a polynomial in p . By applying an algebraic step CS11, $E_G(p)$ can be transformed to a polynomial $E_H(p)$ in p , which can be transformed back to H within $t_n + n - 1$ steps. Total certificate length is $\leq 2t_n + 2n - 1$. From Lemma 6 it follows that $e_n = O(\psi^n)$. \square

Theorem 8. *Let n, n_1, n_2 be the orders of graph G, H_1, H_2 , respectively. The maximum length f_N of a certificate of s -factorisation $A(G; p) = A(H_1; p)A(H_2; p)$ satisfies $f_N \leq 3t_N + 3N - 2$ where $N = \max\{n, n_1, n_2\}$ and $f_N = O(\psi^N)$.*

Proof. Similarly to the previous proof, we can transform G to null graphs, use $\leq n - 1$ applications of CS3, do algebra (this time, the algebra will produce an expression which is a product of two distinct parts), then transform one part of the resulting expression to H_1 and the other to H_2 . This latter transformation is the reverse of the steps needed to compute each of $A(H_1; p)$ and $A(H_2; p)$. Putting it all together, we have

$$f_N \leq t_n + n - 1 + 1 + t_{n_1} + n_1 - 1 + t_{n_2} + n_2 - 1 \leq 3t_N + 3N - 2. \text{ The result follows. } \square$$

We use N here, rather than any one of n, n_1, n_2 , because in principle the factors H_i may have more vertices than G (even though their stability polynomials have lower degrees than that of G). From (7) we can get $m = m_1 + m_2$, where m, m_1, m_2 is the number of edges of graph G, H_1, H_2 , respectively. Thus $m - 1 \geq m_i \geq n_i - 1$. So $N \leq \binom{n}{2}$.

We give the values of our upper bounds for $n \leq 9$ (since this is the range covered by our computations) in Table 4. The values of t_n for $n \leq 4$ are exact. We also give exact values of e_n and f_N where we have determined them. The first three values of e_n are zero because there are no non-isomorphic stability-equivalent graphs for $n \leq 3$. The entry $e_4 = 2$ comes from a certificate we give later, for $A(C_4; p) = A(Q_4; p)$ (Theorem 7). The first five values of f_N are zero because there are no s -factorisations for $N \leq 5$.

$n =$	1	2	3	4	5	6	7	8	9
$t_n \leq$	0	1	1	2	3	5	7	10	14
$e_n =$	0	0	0	2					
$e_n \leq$				11	15	21	27	35	45
$f_N =$	0	0	0	0	0				
$f_N \leq$						31	40	52	67

Table 4: Maximum certificate length for graphs of order n

6 Certificate length and complexity

The problem of computing the stability polynomial of a graph is $\#P$ -hard, since counting independent sets in a graph is $\#P$ -complete. Consider now the following problems.

STABILITY EQUIVALENCE

INPUT: Graphs G and H .

QUESTION: Is $A(G; p) = A(H; p)$?

STABILITY FACTORISATION

INPUT: Graph G .

QUESTION: Do there exist graphs H_1 and H_2 , neither of which is a null graph or isomorphic to G , such that $A(G; p) = A(H_1; p)A(H_2; p)$?

STABILITY EQUIVALENCE can be solved with the aid of an oracle for $\#P$, since the oracle can be used to compute $A(G; p)$ and $A(H; p)$ and then compare them. Hence STABILITY EQUIVALENCE belongs to $P^{\#P}$, which is the class of decision problems solvable in polynomial time with the aid of a $\#P$ oracle.

For STABILITY FACTORISATION, a $\#P$ -oracle yields $A(G; p)$. On the face of it, we are then stuck unless we know H_1 and H_2 . If these are given to us, then we can use the $\#P$ -oracle to compute $A(H_1; p)$ and $A(H_2; p)$ and hence verify that $A(G; p) = A(H_1; p)A(H_2; p)$. So STABILITY FACTORISATION belongs to $NP^{\#P}$, the class of decision problems which can be solved nondeterministically in polynomial time with the aid of a $\#P$ -oracle.

The notion of certificates of s -equivalence and s -factorisation opens up a new line of enquiry into the complexity of these problems.

In principle, certificates allow us to verify s -equivalence, or a claimed s -factorisation, without computing the stability polynomials themselves. Verification only requires the verification of each certificate step, which can be done in polynomial time for each step.

However, if the certificates are very long, then the amount of effort involved in verifying a certificate may be comparable with the effort required to compute the stability polynomials from scratch. We saw this in the previous section, when the certificates we used for our upper bounds on certificate length mimicked the computations of the polynomials.

Nonetheless, it may be that our upper bounds are unnecessarily loose, and that much shorter certificates can be found. This may have implications for the complexity of our two problems.

Theorem 9. (a) *If the length of the shortest certificate of equivalence between two stability equivalent graphs of at most n vertices is bounded by a polynomial in n , then STABILITY EQUIVALENCE is in NP.*

(b) *If the length of the shortest certificate of factorisation for a stability factorisation of a graph of n vertices is bounded by a polynomial in n , then STABILITY FACTORISATION is in NP.*

Proof. In each case, if certificate length is polynomially bounded, then the entire certificate can be verified in polynomial time (using the above observation that each step can be verified in polynomial time). \square

It is encouraging that the certificates we have found empirically (see later sections) are short, although short certificates will tend to be easier to find!

However, it seems unlikely that the shortest certificates of stability equivalence are $o(n^2/(\log n)^2)$, in general. This is because of the following result, whose proof is based on a suggestion of Ian Wanless.

Theorem 10. *If the number of different stability polynomials of graphs of order n is $\leq 2^{bn^2}$, for some constant $b < 1/2$ and for sufficiently large n , then there exist pairs of stability equivalent graphs on n vertices whose shortest certificate of stability equivalence has length $\Omega(n^2/(\log n)^2)$.*

Proof. Suppose the number of different stability polynomials is $\leq 2^{bn^2}$, for sufficiently large n .

Suppose $0 < \varepsilon < \frac{1}{2} - b$. The number of graphs on n vertices, up to isomorphism, is at least $2^{(\frac{1}{2}-\varepsilon)n^2}$, for sufficiently large n (see, e.g., [7]). From now on, suppose n is large enough for this to hold. So there must exist a set of $\geq 2^{(\frac{1}{2}-\varepsilon-b)n^2}$ stability equivalent graphs on n vertices. Let G be any one of these graphs.

If every certificate of stability equivalence for graphs of order n has length $\leq L$, then the size of the stability equivalence class of G is bounded above by the number of valid sequences of L certificate steps starting with G .

Now, if a certificate starts with a single graph and has length $\leq L$, then each of its L expressions (after G) can contain at most L graphs. This is because each certificate step can increase the number of graphs by at most 1.

Consider the number of possible applications of a certificate step to an expression of at most L graphs. Firstly, there is the choice of the graph at which the step is applied (from $\leq L$ possibilities), then possibly the choice of a vertex or two within the graph (from $\leq n^2$ possibilities), then possibly the choice of the second graph needed for certain certificate steps (from $\leq L-1$ possibilities), and even a third graph for some steps ($\leq L-2$ possibilities). So there are $\leq c_0 n^2 L^3$ choices altogether, for some constant c_0 .

The number of possible certificates of stability equivalence, starting with G and of length $\leq L$, is therefore $\leq (c_0 n^2 L^3)^L = 2^{c_1 \cdot \log n \cdot \log L \cdot L}$ (with $c_1 = 6 \log_2 c_0$).

This must be at least as great as the number of graphs in the stability equivalence class of G . Therefore

$$2^{(\frac{1}{2} - \varepsilon - b)n^2} \leq 2^{c_1 \cdot \log n \cdot \log L \cdot L}.$$

Hence

$$L \log L \geq c_2 \cdot \frac{n^2}{\log n}, \tag{12}$$

where $c_2 = (\frac{1}{2} - \varepsilon - b)/c_1$. Suppose $L \leq c_3 n^2 / ((\log n)^2 f(n))$, for sufficiently large n , constant c_3 and appropriate choice of function f . Then (12) gives

$$\frac{n^2}{(\log n)^2 f(n)} \cdot (2 \log n - 2 \log \log n - \log f(n) + \log c_3) \geq c_2 \cdot \frac{n^2}{\log n},$$

from which we conclude that $f(n)$ is bounded above by a suitable constant. (Here we use $c_2 > 0$, which requires $b < \frac{1}{2}$.) It follows that any upper bound on certificate length must be $\Omega(n^2 / (\log n)^2)$. \square

The hypothesis of Theorem 10 seems likely to be true. The data in Table 2 suggests that $b < 0.15$ and decreases as n increases beyond 5.

7 Stability equivalence

The computational results given in Section 3 show that many non-isomorphic graphs have the same stability polynomial. In this section, short certificates for two infinite families of stability equivalent graphs are presented.

Theorem 11. *The stability polynomial $A(C_n; p) = A(Q_n; p)$ for $n \geq 3$.*

Proof. The following is a certificate of equivalence to show $A(C_n; p) = A(Q_n; p)$:

$$A(C_n; p) = (1 - p)A(P_{n-1}; p) + p(1 - p)^2 A(P_{n-3}; p) \tag{CS3}$$

$$= A(Q_n; p). \tag{CS4}$$

This certificate is illustrated in Figure 6. \square

Theorem 11 shows that for any $n \geq 3$, $A(C_n; p) = A(Q_n; p)$ has a certificate of length 2. The length of certificates for this infinite family of stability equivalences is much lower than the exponential upper bound given in Section 5.

Theorem 12. *For any $uv \in E(G)$, if $(N(v) \setminus \{u\}) \subseteq (N(u) \setminus \{v\})$. Then*

$$A(G; p) = A(G'; p)$$

where G' is a graph with $V(G') = V(G) \cup \{w\}$, $E(G') = (E(G) \setminus uv) \cup \{uw\}$

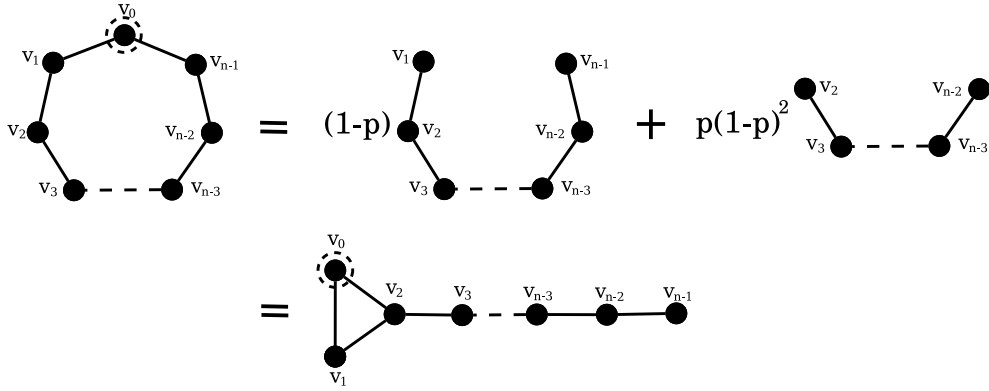


Figure 6: $A(C_n; p) = A(Q_n; p)$

Proof. The following is a certificate of the stability equivalence:

$$A(G; p) = (1 - p)A(G - u; p) + p(1 - p)^{|N(u)|}A(G - N(u); p) \quad (\text{CS3})$$

$$= (1 - p)A(G - u; p)A(K_1; p) + p(1 - p)^{|N(u)|}A(G - N(u); p)A(K_1; p) \quad (\text{CS2})$$

$$= (1 - p)A(\{G - u\} \cup K_1; p) + p(1 - p)^{|(N(u) \setminus \{v\}) \cup \{w\}|}A(\{G - N(u)\} \cup K_1; p) \quad (\text{CS10})$$

$$= A(G'; p). \quad (\text{CS4})$$

Figure 7 illustrates this certificate. □

Theorem 12 shows that an infinite family of graphs have stability equivalences that can be explained by certificates of length 4. The length of certificates for these stability equivalences is much lower than the exponential upper bound given in Section 5 and is independent of the order of the graph. Theorem 12 can be used to show the following two infinite families of graphs are stability equivalent.

Corollary 13. For any $n \geq 3$,

$$A(Q_n; p) = A(Y_{n+1}; p) \quad \square$$

Corollary 14. For any $n \geq 2$,

$$A(K_{n+1}; p) = A(K_n^*; p)$$

Proof. Let $\{v_i : 0 \leq i \leq n - 1\} \cup \{w\}$ be the vertex set of K_{n+1} . Applying Theorem 12 on all edges in $\{v_i w : 0 \leq i \leq n - 1\}$ will provide certificates for this infinite family of stability equivalences. Theorem 12 is used n times. Thus the length of certificates provided by this proof is $4n$. This length is linear, which is much shorter than the exponential upper bound given in Section 5. This proof is illustrated in Figure 9. □

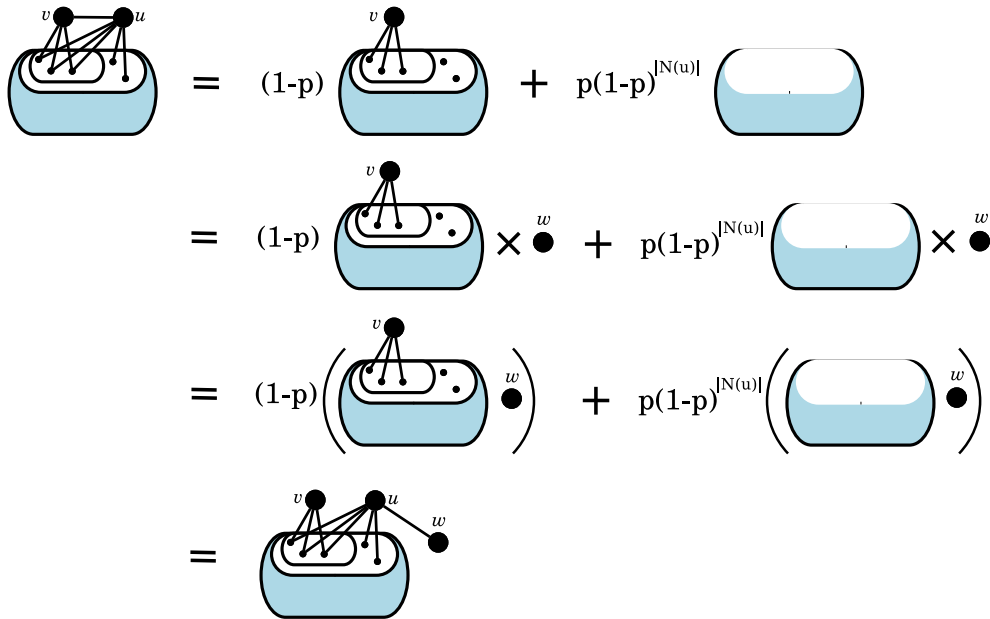


Figure 7: Edge breaking theorem

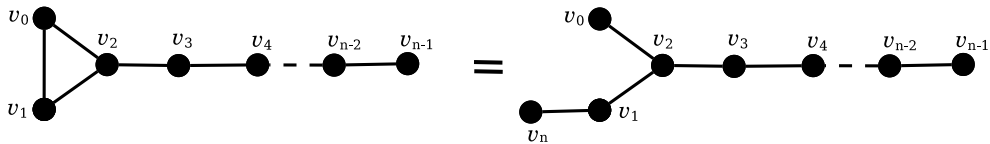


Figure 8: $A(Q_n; p) = A(Y_{n+1}; p)$

8 S-factorisations

In this section, we studied the s-factorisations provided by the computational results and found short certificates for two infinite families of s-factorisations.

Theorem 15. *For any $n \geq 3$, we have*

$$A(\Upsilon_{n,2n}; p) = A(P_{n-1}; p)A(\Upsilon_{n+1,n+1}; p)$$

Proof. The following is a certificate of this s-factorisation:

$$A(\Upsilon_{n,2n}; p) = (1 - p)A(\Upsilon_{n,2n} \cup P_{n-1}; p) + p(1 - p)^2A(P_{n-1} \cup P_{n-2}; p) \quad (\text{CS3})$$

$$= (1 - p)A(\Upsilon_{n,2n}; p)A(P_{n-1}; p) + p(1 - p)^2A(P_{n-1}; p)A(P_{n-2}; p) \quad (\text{CS9}) \times 2$$

$$= A(P_{n-1}; p)[(1 - p)A(\Upsilon_{n,n}; p) + p(1 - p)^2A(P_{n-2}; p)] \quad (\text{CS11})$$

$$= A(P_{n-1}; p)A(\Upsilon_{n+1,n+1}; p). \quad (\text{CS4})$$

Figure 10 illustrates this certificate. The length of these certificates is 5, which is much lower than the exponential upper bound given in Section 5. \square

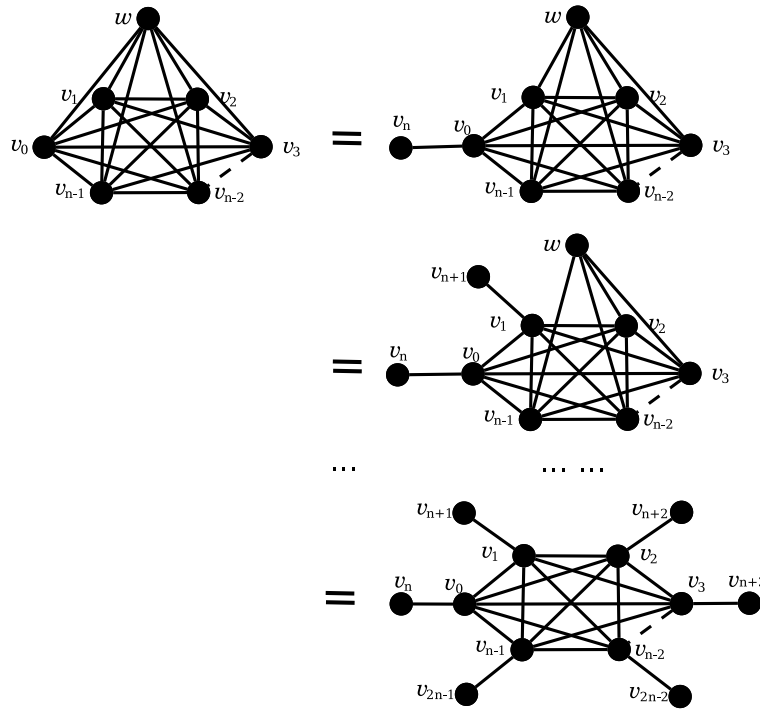


Figure 9: $A(K_{n+1}; p) = A(K_n^*; p)$

Theorem 16. (General symmetry theorem) Suppose $G \cong G'$, $u \in V(G)$, u' is the image of u in G' under the isomorphism, $N(u)$ is the neighbour set of u in G . Let $N_2 \subseteq V(G_2)$, G_3 is a graph with vertex set $V(G_3) = V(G) \cup V(G') \cup V(G_2)$ and edge set $E(G_3) = E(G) \cup E(G') \cup E(G_2) \cup \{uv, u'v : v \in N_2\} \cup \{uu'\}$. Then

$$A(G_3; p) = A(G - u; p)A(G_4; p)$$

where G_4 is a graph with vertex set $V(G_4) = V(G_2) \cup V(G') \cup \{w\}$ and edge set $E(G_4) = E(G_2) \cup E(G') \cup \{wv, u'v : v \in N(u)\}$

Proof. The following is a certificate of this s-factorisation:

$$\begin{aligned}
 A(G_3; p) &= p(1-p)^{|N(u) \cup N_2 \cup \{u'\}|} A((G - u - N(u)) \cup (G' - u') \cup (G_2 - N_2); p) \\
 &\quad + (1-p)A((G - u) \cup (G_3 - G); p) \tag{CS3} \\
 &= p(1-p)^{|N(u) \cup N_2 \cup \{u'\}|} A(G - u - N(u); p)A(G' - u'; p)A(G_2 - N_2; p) \\
 &\quad + (1-p)A(G - u; p)A(G_3 - G; p) \tag{CS9} \times 2 \\
 &= p(1-p)^{|N(u) \cup N_2 \cup \{u'\}|} A(G - u - N(u); p)A(G_2 - N_2; p) \\
 &\quad + A(G - u; p)[(1-p)A(G_3 - G; p)] \tag{CS11} \\
 &= A(G - u; p)A(G_4; p). \tag{CS6}
 \end{aligned}$$

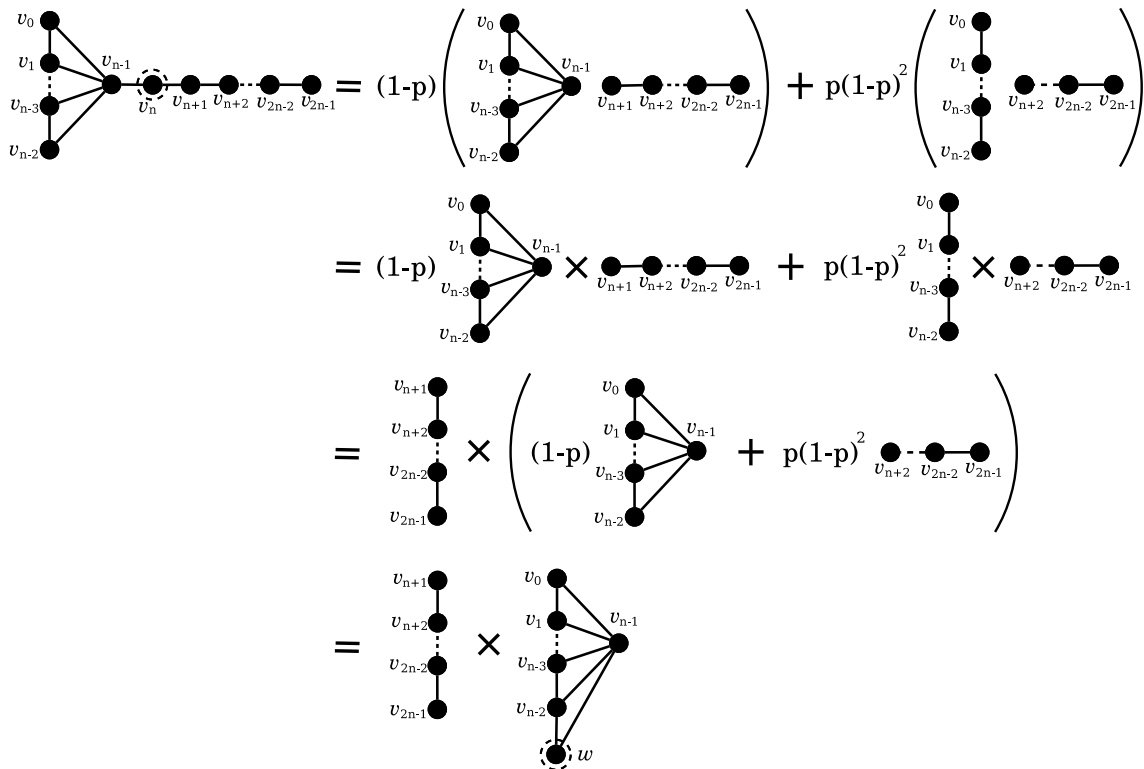


Figure 10: $A(\Upsilon_{n,2n}; p) = A(P_{n-1}; p)A(\Upsilon_{n+1,n+1}; p)$

The length of certificates provided by this proof is 5. This independent of the order of the graph and is much lower than the exponential upper bound given in Section 5. Figure 11 illustrates this certificate. \square

Theorem 16 is a property of stability polynomial which can be used to decompose certain graphs with isomorphic subgraphs. We applied Theorem 12 on some more specialised cases and found the following interesting corollaries.

Corollary 17. (*Symmetry corollary*) *If $G \cong G'$, $u \in V(G)$ and u' is the image of u in G' under the isomorphism, $N(u)$ is the neighbour set of u in G . Let G_2 be a graph with vertex set $V(G_2) = V(G) \cup V(G')$ and edge set $E(G_2) = E(G) \cup E(G') \cup \{uu'\}$. Then*

$$A(G_2; p) = A(G - u; p)A(G_{u+}; p)$$

where G_{u+} is a graph with vertex set $V(G_{u+}) = V(G) \cup \{w\}$ and edge set $E(G_{u+}) = E(G) \cup \{wu\} \cup \{wv : v \in N(u)\}$.

Figure 12 illustrates this corollary.

The length of certificates provided by Corollary 17 is 5 steps.

Corollary 18. *For any graph G , there exists a graph H such that H has an s -factorisation with G as a factor.*

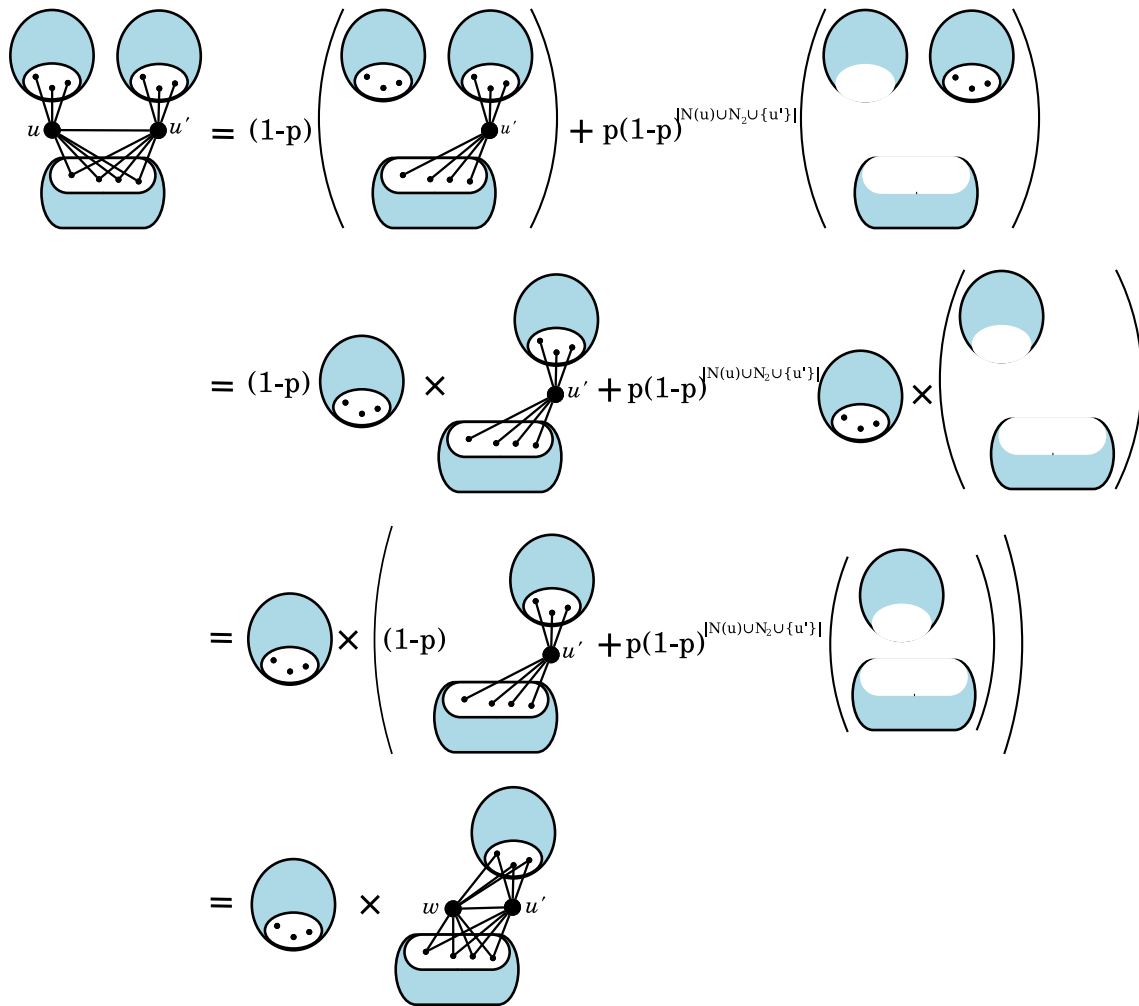


Figure 11: General symmetry theorem

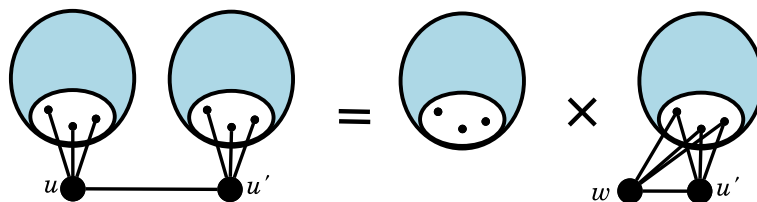


Figure 12: Symmetry corollary

This result is analogous to one due to Morgan and Farr for chromatic factorisation [12].

Corollary 19. For any $n \geq 2$,

$$A(K_{n \sim n}; p) = A(K_{n-1}; p)A(K_{n+1}; p)$$

Figure 13 illustrates this corollary.

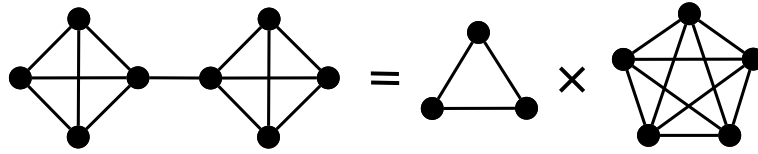


Figure 13: $A(K_{n\sim n}; p) = A(K_{n-1}; p)A(K_{n+1}; p)$ when $n = 4$

The length of certificates provided by Corollary 19 is 5.

Corollary 20. For any $n \geq 2$,

$$A(P_{2n}; p) = A(P_{n-1}; p)A(C_{n+1}; p).$$

Proof. Let G be a P_n and $G' \cong G$, $V(G) = \{v_0, v_1, \dots, v_n\}$, $E(G) = \{v_0v_2\} \cup \{v_iv_{i+1} : 0 \leq i \leq n-2\}$. $G' \cong G$ and v'_i is the image of v_i in G' under the isomorphism. $G_2 = (G \cup G') + \{v_0v'_0\}$, G_2 is a P_{2n} . G_4 is a Q_n with vertex set $V(G_4) = V(G') \cup \{w\}$ and edge set $E(G_4) = E(G') \cup \{wv, u'v : v \in N_G(v'_0)\}$. By Theorem 16, $A(G_2; p) = A(G - v_0; p)A(G_4; p)$. Thus

$$A(P_{2n}; p) = A(P_{n-1}; p)A(Q_{n+1}; p) \tag{13}$$

By Corollary 13 and Theorem 11 $A(Q_{n+1}; p) = A(Y_{n+2}; p) = A(C_{n+1}; p)$. Applying this to (13) gives

$$A(P_{2n}; p) = A(P_{n-1}; p)A(C_{n+1}; p).$$

□

The length of these certificates is $5 + 2 = 7$. Figure 14 illustrates this corollary.

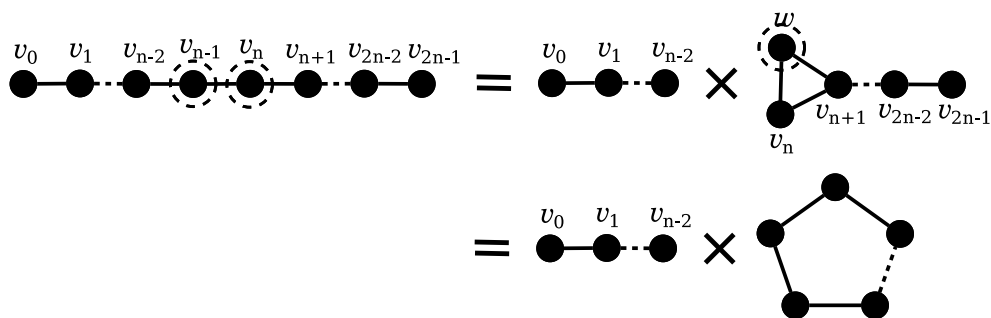


Figure 14: $A(P_{2n}; p) = A(P_{n-1}; p)A(C_{n+1}; p)$

9 Certificate schemas of s-factorisations

The certificates for s-factorisations presented in Section 8 show that two different families of s-factorisations can be explained by the same sequence of certificate steps. We group these certificates of s-factorisations into two certificate schemas as follows:

$$\begin{aligned}
 A(G; p) &= (1-p)A(H_1; p) + p(1-p)^d A(H_2; p) && \text{(CS3)} \\
 &= (1-p)A(H_3; p)A(H_4; p) + p(1-p)^d A(H_3; p)A(H_5; p) && \text{(CS9)} \times 2 \\
 &= A(H_3; p)((1-p)A(H_4; p) + p(1-p)^d A(H_5; p)) && \text{(CS11)} \\
 &= A(H_3; p)A(H_6; p) && \text{(CS4)}
 \end{aligned}$$

Schema 1

The length of certificates constructed by Schema 1 is 5 steps. This schema can be used to build the certificates in Theorems 15 and 16 and Corollaries 17 and 19. This Schema gives short certificates for s-factorisations for stability polynomials of all graphs of order ≤ 8 .

$$\begin{aligned}
 A(G; p) &= (1-p)A(H_1; p) + (1-p)A(H_2; p) - (1-p)^2 A(H_3; p) && \text{(CS5)} \\
 &= (1-p)A(H_4; p)A(H_5; p) + (1-p)A(H_4; p)A(H_6; p) \\
 &\quad - (1-p)^2 A(H_4; p)A(H_7; p) && \text{(CS9)} \times 3 \\
 &= A(H_4; p)((1-p)A(H_5; p) + (1-p)A(H_6; p) \\
 &\quad - (1-p)^2 A(H_7; p)) && \text{(CS11)} \\
 &= A(H_4; p)A(H_8; p) && \text{(CS6)}
 \end{aligned}$$

Schema 2

The length of certificates constructed by Schema 2 is 6. This schema can be used to build the certificates of Theorems 15 and 16 and Corollaries 17 and 19.

10 Conclusions and further work

This paper initiates an algebraic study for the stability polynomial, focusing on the fundamental algebraic properties of equality and factorisation. We gave upper bounds for lengths of certificates for stability equivalence and s-factorisations. There remains the challenge of determining better upper bounds, especially ones that do better than just, in effect, computing the polynomials. If a polynomial upper bound on certificate length can be found, then both those computational problems belong to NP. Determining whether this is the case remains an open problem.

In this research we found short certificates for all s-factorisations of graphs of order 6 and one s-factorisation of graphs of order 7. This suggests the task of finding short

certificates for s-factorisations of graphs of some higher orders. An open question that arises from this research is: what are the shortest certificates for infinite families of s-factorisations and stability equivalence? We found some short certificates but it is not known if these certificates are the shortest. Finding shortest certificates for infinite families of s-factorisations and stability equivalence could be challenging.

Acknowledgements

We thank Ian Wanless for a suggestion that led to the proof of Theorem 10.

This research was supported in part by the Australian Research Council's *Discovery Projects* funding scheme (project number DP110100957).

References

- [1] G.H. Birkhoff. A determinant formula for the number of ways of coloring a map. *Ann. of Math.*, 14:42–46, 1912-1913.
- [2] Z. Bukovac, G. Farr, and K. Morgan. Short certificates for chromatic equivalence. Submitted, 2013.
- [3] G. Farr. The stability polynomial of a graph. (Unpublished manuscript), 1991.
- [4] G. Farr. A correlation inequality involving stable set and chromatic polynomials. *J. Combin. Theory Ser. B*, 58:14–21, 1993.
- [5] G. Farr. The chromatic polynomial and its relatives. (Unpublished manuscript), 2011.
- [6] I. Gutman and F. Harary. Generalizations of the matching polynomial. *Utilitas Mathematica*, 24:97–106, 1983.
- [7] F. Harary and E.M. Palmer. *Graphical Enumeration*. Academic Press, New York, 1973.
- [8] T. Helgason. *Aspects of the theory of hypermatroids*, pages 191–213. Lecture Notes in Mathematics 411. Springer, Berlin, 1974.
- [9] K. Morgan. *Algebraic Aspects of the Chromatic Polynomial*. PhD thesis, Monash University, 2010.
- [10] K. Morgan. Pairs of chromatically equivalent graphs. *Graphs Combin.*, 27:547–556, 2010.
- [11] K. Morgan and G. Farr. Certificates of factorisation for a class of triangle-free graphs. *Electron. J. Combin.*, 16:R75, 2009.
- [12] K. Morgan and G. Farr. Certificates of factorisation for chromatic polynomials. *Electron. J. Combin.*, 16:R74, 2009.
- [13] K. Morgan and G. Farr. Non-bipartite chromatic factors. *Discrete Mathematics*, 312:1166–1170, 2012.

- [14] J. Reicher. *Computation of Stability Polynomials*. BCSE Hons dissertation, Monash University, 1997.
- [15] V.R. Rosenfeld. The independence polynomial of rooted products of graphs. *Discrete Appl. Math.*, 158:551–558, 2010.