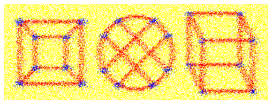
Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 23, no. 2, pp. 227–269 (2019)
DOI: 10.7155/jgaa.00490

Short certificates for chromatic equivalence

Zoe Bukovac[1] *Graham Farr*[1] *Kerri Morgan*[2]

[1]Faculty of Information Technology, Monash University,
Clayton, Victoria 3800, Australia
[2]Deakin University, Geelong, Australia
School of I.T., Faculty of Science, Engineering and Built Environment

Abstract

The chromatic polynomial gives the number of proper colourings of a graph in terms of the number of available colours. In general, calculating chromatic polynomials is #P-hard. Two graphs are *chromatically equivalent* if they have the same chromatic polynomial. At present, determining if two graphs are chromatically equivalent involves computation and comparison of their chromatic polynomials, or similar computational effort. In this paper we investigate a new approach, *certificates of chromatic equivalence*, first proposed by Morgan and Farr. These give proofs of chromatic equivalence, without directly computing the polynomials. The lengths of these proofs may provide insight into the computational complexity of chromatic equivalence and related problems including chromatic factorisation and chromatic uniqueness. For example, if the lengths of shortest certificates of chromatic equivalence are bounded above by a polynomial in the size of the graphs, then chromatic equivalence belongs to NP. After establishing some links of this type between certificate length and computational complexity, we give some theoretical and computational results on certificate length. We prove that, if the number of different chromatic polynomials falls well short of the number of different graphs, then for all sufficiently large n there are pairs of chromatically equivalent graphs on n vertices with certificate of chromatic equivalence of length $\Omega(n^2/\log n)$. We give a linear upper bound on shortest certificate length for trees. We designed and implemented a program for finding short certificates of equivalence using a minimal set of certificate steps. This program was used to find the shortest certificates of equivalence for all pairs of chromatically equivalent graphs of order $n \leq 7$.

Submitted:	Reviewed:	Revised:	Accepted:	Final:
November 2016	January 2019	January 2019	February 2019	March 2019
		Published: April 2019		
	Article type: Regular paper		Communicated by: M. Kaufmann	

Research supported in part by Australian Research Council Discovery Project DP110100957.
E-mail addresses: Zoe.Bukovac@monash.edu (Zoe Bukovac) Graham.Farr@monash.edu (Graham Farr) Kerri.Morgan@deakin.edu.au (Kerri Morgan)

1 Introduction

The *chromatic polynomial* $P(G;\lambda)$ of a graph G gives the number of λ-colourings of G. It was first introduced by Birkhoff as a possible algebraic approach to a proof for the Four Colour Theorem [1]. Calculating the chromatic polynomial is #P-hard [11, 13], even when restricted to the family of subgraphs of square lattices [9]. In general, for all $\lambda > 2$, determining if a graph is λ-colourable is NP-complete [12].

Two graphs G and G' are *chromatically equivalent*, written $G \sim G'$, if they have the same chromatic polynomial. It is possible to have chromatically equivalent graphs that are not isomorphic. No good characterisation of chromatically equivalent graphs is known, but there is a wealth of research on chromatic equivalence, much of which is summarised in [5] and [7]. Research in this area focuses on either small sets of graphs that have been found to be chromatically equivalent or infinite families of chromatically equivalent graphs. A graph G is *chromatically unique* if the only graphs which have the same chromatic polynomial are also isomorphic to G. The idea of chromatically unique graphs was introduced by Chao and Whitehead [4].

Certificates to verify instances of chromatic equivalence and other algebraic properties of the chromatic polynomial were first introduced by Morgan and Farr [20] in 2009. A certificate of this type is a sequence of algebraic transformations based on identities for the chromatic polynomial and algebraic properties. As calculating the chromatic polynomial is #P-hard in general, any method that can verify information about the chromatic polynomial of a graph without needing to calculate it is of interest.

In this article we consider certificates that can be used to help verify that two graphs are chromatically equivalent. We describe the relationship between certificate length and computational complexity of chromatic equivalence, chromatic uniqueness and chromatic factorisation. We then show that, if the number of different chromatic polynomials of degree n falls well short of the number of non-isomorphic graphs on n vertices (a plausible hypothesis, given the data), then for all sufficiently large n there are pairs of chromatically equivalent graphs on n vertices with certificate of chromatic equivalence of length $\Omega(n^2/\log n)$.

A program for finding short certificates of equivalence was designed and implemented [3]. In order to produce computationally feasible software, it uses a minimal set of certificate steps. This program was used to find the shortest certificates for all chromatically equivalent graphs of order $n \leq 7$. These certificates can be grouped into 15 classes which we call *schemas*. Although the best known upper bound on length of certificates is exponential [20], the certificates we found were all remarkably short. We give a linear bound on the lengths of certificates of equivalence for trees.

2 Definitions and fundamentals

Let $G = (V, E)$ be an undirected graph with vertex set $V = V(G)$ and edge set $E = E(G)$. In general we will use n and m for the size of V and E respectively. The set of unordered pairs of elements of V is denoted $V^{(2)}$. The *chromatic number* of a graph G, denoted $\chi(G)$, is the minimum number of colours required to colour G so that no adjacent vertices are given the same colour. We refer the reader to [6] for more information regarding common graph theory definitions.

We denote the disjoint union of two graphs G and H by $G \cup H$. If G and H share exactly one vertex, then the combined graph is denoted by $G \cup_1 H$.

Let u and v be vertices in G. If uv is an edge of G, then $G\backslash uv$ is the graph obtained by deleting the edge uv from G. We call this process *edge deletion*. If uv is not an edge of G, then $G + uv$ is the graph obtained by adding the edge uv to G. We call this process *edge addition*. For any pair of vertices u and v the graph G/uv is the graph obtained by identifying vertices u and v in G and discarding any multiple edges or loops obtained in the identification. If u and v are not adjacent in G, we call this process *vertex identification*. If u and v are adjacent in G, we call it *edge contraction*.

If two disjoint graphs H_1 and H_2 both contain a clique of at least size r then the graph G formed by identifying an r-clique in H_1 with an r-clique in H_2 is an *r-gluing*. The 0-gluing operation is just the disjoint union of the graphs. A graph that can be obtained by an r-gluing of two graphs is said to be a *clique-separable* graph.

The following two relations can be used to calculate the chromatic polynomial recursively. For most graphs this will take exponential time.

The deletion-contraction relation states that for any edge $e \in E$

$$P(G; \lambda) = P(G\backslash e; \lambda) - P(G/e; \lambda). \tag{1}$$

The addition-identification relation states that for any vertices $u, v \in V, uv \notin E$,

$$P(G; \lambda) = P(G + uv; \lambda) + P(G/uv; \lambda). \tag{2}$$

Whitney [22, §14] gives another method of evaluating the chromatic polynomial of a clique-separable graph. If G is an r-gluing of some graphs H_1 and H_2, then

$$P(G; \lambda) = \frac{P(H_1; \lambda)P(H_2; \lambda)}{P(K_r; \lambda)}. \tag{3}$$

Note that $P(K_0; \lambda) = 1$.

This result only helps evaluate $P(G; \lambda)$ when G is clique-separable, which does restrict it, but its divide-and-conquer nature means that under such circumstances it can offer a significant reduction in the complexity of calculating a graph's chromatic polynomial. It also gives a partial, initial link between factorisation of the chromatic polynomial — an algebraic property of the polynomial — and the structure of the corresponding graph. These considerations led us, in previous work [20, 19], to identify other situations where chromatic polynomials factorise in a similar way.

The chromatic polynomial is said to have a *chromatic factorisation* if there exist graphs H_1 and H_2 such that

$$P(G;\lambda) = \frac{P(H_1;\lambda)P(H_2;\lambda)}{P(K_r;\lambda)}$$

where $\chi(H_i) \geq r \geq 0$ and $H_i \ncong K_r$ for $i = 1, 2$ [20]. A graph G is said to have a chromatic factorisation if $P(G;\lambda)$ has a chromatic factorisation. It is clear from (3) that any clique-separable graph has a chromatic factorisation. Similarly any graph that is chromatically equivalent to a clique-separable graph has a chromatic factorisation. A *strongly non-clique-separable* graph is a graph that is not chromatically equivalent to any clique-separable graph. Morgan and Farr [20] showed that there exist chromatic factorisations for some strongly non-clique-separable graphs. They introduced the notion of a certificate to explain these factorisations and other properties of chromatic polynomials.

3 Certificates

Certificates to verify instances of chromatic equivalence and chromatic factorisation were first introduced by Morgan and Farr [20] in 2009. A certificate of this type is a sequence of transformations based on identities for the chromatic polynomial and algebraic properties. Each of the individual transformations in a certificate is called a *certificate step*.

The following is a description of each of the certificate steps used in [20]. Each new expression in a certificate is obtained by applying one of the following certificate steps to the previous expression in the certificate:

CS1 $G \longrightarrow (G\backslash e) - (G/e)$ for some edge $e \in E(G)$.

CS2 $(G\backslash e) - (G/e) \longrightarrow G$ for some edge $e \in E(G)$.

CS3 $G \longrightarrow (G + uv) + (G/uv)$ where the vertices $u, v \in V(G)$ and u, v are not adjacent in G.

CS4 $(G + uv) + (G/uv) \longrightarrow G$ where the vertices $u, v \in V(G)$.

CS5 $(G\backslash e) - G \longrightarrow (G/e)$ for some edge $e \in E(G)$.

CS6 $G \longrightarrow G_1G_2/K_r$ where G is isomorphic to the graph obtained by an r-gluing of G_1 and G_2.

CS7 $G_1G_2/K_r \longrightarrow G$ where G is isomorphic to the graph obtained by an r-gluing of G_1 and G_2.

CS8 Applying field operations to an expression a finite number of times to produce a different expression.

Certificate steps **(CS1)**, **(CS2)** and **(CS5)** are based on (1), certificate steps **(CS3)** and **(CS4)** are based on (2) and certificate steps **(CS6)** and **(CS7)** are based on (3).

Each application of a certificate step links a single expression to the next one in the certificate. In a certificate that starts with a graph G, each expression can be evaluated to a polynomial which is equal to the chromatic polynomial of G. The certificate can be verified by checking, for each pair of consecutive graph expressions in the certificate along with the nominated certificate step linking them, that the nominated certificate step correctly transforms the first expression of the pair to the second. Importantly, the actual chromatic polynomial of G is not calculated or required at any point in the process of verifying a certificate for G.

3.1 Certificates of equivalence

A *certificate of equivalence* for $G \sim G'$ consists of

- a sequence of expressions $E_0, E_1, \ldots, E_l$ where E_0 is the graph G and E_l is the graph G';
- for each $i \in \{1, \ldots, l\}$, a specification of a certificate step from {CS1,. . . ,CS8} along with which graphs in E_{i-1} and E_i it is applied to, such that the specified step applied to these graphs does indeed transform E_{i-1} to E_i.

We say the certificate is a *certificate from G to G'*. The *length* of the certificate is number of certificate steps, l, applied to transform G into G' (as distinct from the number of expressions, $l + 1$, in the certificate).

An *algebraic certificate of equivalence* is obtained from a certificate of equivalence by expanding each algebraic step $E_{i-1} \longrightarrow E_i$ (CS8) into a sequence of expressions $E_{i-1} = E_i^{(0)}, \ldots, E_i^{(k_i)} = E_i$ where each $E_i^{(j-1)} \longrightarrow E_i^{(j)}$ is obtained by a single application of a field axiom.

The *algebraic length* of a certificate step is defined to be 1 for all steps except the algebraic step (CS8), and the algebraic length of an application of CS8 is defined to be the number of applications of field axioms needed to carry it out. So an algebraic step $E_{i-1} \longrightarrow E_i$ (CS8) expanded into $E_{i-1} = E_i^{(0)}, \ldots, E_i^{(k_i)} = E_i$ contributes k_i to the algebraic length. For example, if H_1 and H_2 are graphs, then $H_1 \longrightarrow H_1+H_2-H_2$ is a valid application of (CS8), and has algebraic length 2, since it has two applications of field axioms: $x \to x + 0$ and $0 \to y - y$. In an algebraic certificate of equivalence, it would be expanded into $H_1 \longrightarrow H_1 + 0 \longrightarrow H_1 + H_2 - H_2$. We will see another example in the next subsection. The *algebraic length* of a certificate is the sum of the algebraic lengths of all its certificate steps, which is one less than the total number of expressions in the algebraic certificate. Clearly the length of a certificate is at most its algebraic length.

We mostly work just with certificates of equivalence and their (non-algebraic) lengths. But it is sometimes important to do the more detailed accounting required for algebraic lengths.

Figure 1 gives a certificate of equivalence of length 2 steps. The certificate steps performed in the certificate in Figure 1 are as follows:

$$\begin{aligned} G &\longrightarrow (G\backslash e) - (G/e) && \textbf{(CS1)} \\ &\longrightarrow (G\backslash e + f). && \textbf{(CS2)} \end{aligned}$$

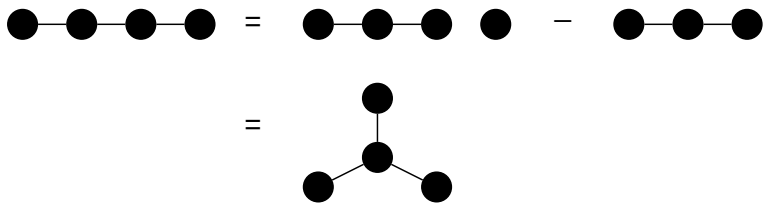

Figure 1: A certificate of equivalence of length 2.

3.2 Certificates of factorisation

A *certificate of factorisation* for $P(G;\lambda) = P(H_1;\lambda)P(H_2;\lambda)/P(K_r;\lambda)$ is defined as for a certificate of equivalence, except that the final expression E_l is H_1H_2/K_r.

Figure 2 gives a certificate of factorisation. The certificate steps performed in the certificate in Figure 2 are as follows:

$$\begin{aligned} G &\longrightarrow H_3 - H_4 && \textbf{(CS1)} \\ &\longrightarrow \frac{H_1H_5}{K_2} - \frac{H_1H_6}{K_2} && 2\times\textbf{(CS6)} \\ &\longrightarrow \frac{H_1}{K_3}\left(\frac{K_3H_5}{K_2} - \frac{K_3H_6}{K_2}\right) && \textbf{(CS8)} \\ &\longrightarrow \frac{H_1}{K_3}\left(H_7 - H_8\right) && 2\times\textbf{(CS7)} \\ &\longrightarrow \frac{H_1H_2}{K_3}. && \textbf{(CS2)} \end{aligned}$$

This certificate has length 7. Its algebraic length is 9, since the sole algebraic step CS8 involves three applications of field axioms: $1 \longrightarrow x/x$ twice and $xy + xz \longrightarrow x(y+z)$ once.

4 Certificate length

Certificates have been a powerful tool in proving results on chromatic factorisations [20, 19, 21] and chromatic equivalence [18]. Although the best known

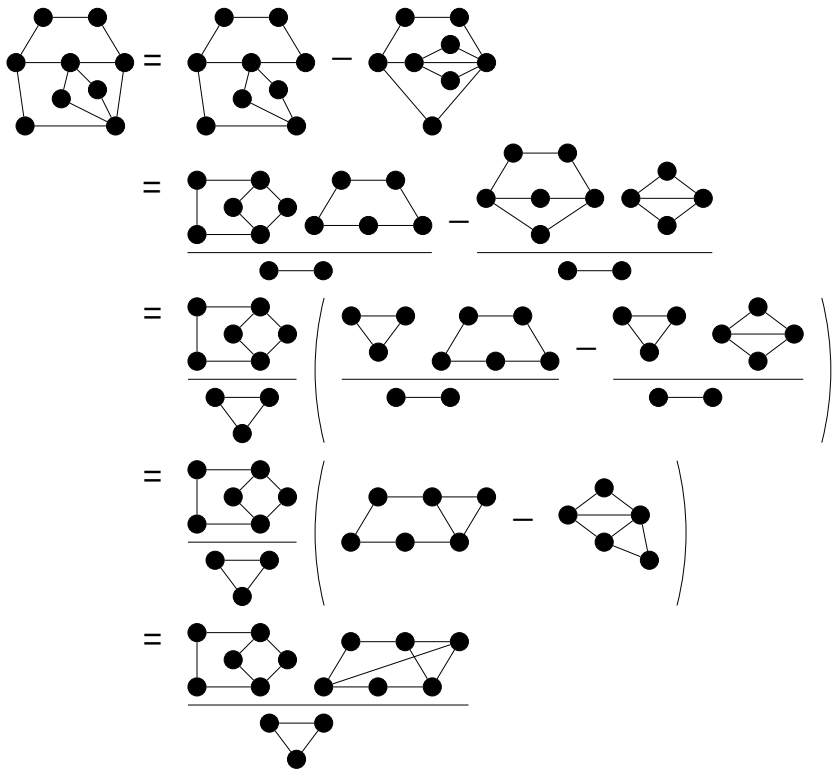

Figure 2: A certificate of factorisation for a strongly non-clique-separable graph from [17].

upper bound on certificate length is exponential, in practice certificates seem to be very short. When they exist, short certificates enable results on chromatic equivalence and chromatic factorisation to be easily verified while bypassing the cost of computing the chromatic polynomial. As certificates are useful and there was no existing software to find certificates in general, we hope that our program to find certificates of equivalence may be a valuable tool in the study of the chromatic polynomial, particularly for chromatic equivalence.

4.1 Certificate length and complexity

The lengths of certificates have potential implications for the computational complexity of determining when two graphs are chromatically equivalent, of determining when a graph is chromatically unique, and of determining when a graph has a chromatic factorisation.

Consider the following problems.

CHROMATIC EQUIVALENCE
INPUT: Graphs G and H.
QUESTION: Is $P(G;\lambda) = P(H;\lambda)$?

CHROMATIC UNIQUENESS
INPUT: Graph G.
QUESTION: Is there no other graph $H \ncong G$ such that $P(G;\lambda) = P(H;\lambda)$?

CHROMATIC FACTORISATION
INPUT: Graph G.
QUESTION: Do there exist graphs H_1 and H_2, each having fewer vertices than G, such that

$$P(G;\lambda) = \frac{P(H_1;\lambda)P(H_2;\lambda)}{P(K_r;\lambda)},$$

for some nonnegative integer r such that $H_1 \ncong K_r$ and $H_2 \ncong K_r$?

CHROMATIC EQUIVALENCE can be solved quickly once the chromatic polynomials are known. It follows that it belongs to $\mathrm{P}^{\#\mathrm{P}}$, the class of sets recognisable in polynomial time with the aid of a #P-oracle. Computation of chromatic polynomials does not solve CHROMATIC UNIQUENESS, but can be used to certify chromatic *non*-uniqueness, with the help of a *guessed* graph H not isomorphic to the input graph G: use a #P-oracle to compute the chromatic polynomials of G and H, check that they are equal, and use the #P-oracle again to check that $H \ncong G$ (possible since GRAPH ISOMORPHISM $\in$ NP $\subseteq \mathrm{P}^{\#\mathrm{P}}$). It follows that CHROMATIC UNIQUENESS belongs to co-$\mathrm{NP}^{\#\mathrm{P}}$.

In similar vein, if we guess a factorisation — by specifying H_1, H_2 and r — for an input to CHROMATIC FACTORISATION, then the guess is easy to check if the chromatic polynomials of G, H_1 and H_2 are known. This means that CHROMATIC FACTORISATION belongs to $\mathrm{NP}^{\#\mathrm{P}}$.

These complexity classes — relativisations of P, NP, and co-NP with respect to a #P oracle — are very large, in the sense that they contain all the power of #P and hence, by Toda's Theorem, the entire Polynomial Hierarchy. It is natural to ask whether CHROMATIC EQUIVALENCE, CHROMATIC UNIQUENESS and CHROMATIC FACTORISATION belong to complexity classes within the Polynomial Hierarchy, and especially whether they belong to NP.

Certificates of chromatic equivalence, or chromatic factorisation, may provide a tool for attacking this question, because of the following.

Theorem 1 *(a) If every pair of chromatically equivalent graphs has a certificate of equivalence of algebraic length bounded by a polynomial in n, then CHROMATIC EQUIVALENCE is in NP.*
(b) If every pair of chromatically equivalent graphs has a certificate of equivalence of algebraic length bounded by a polynomial in n, then CHROMATIC UNIQUENESS is in the class co-NP^{GI} of problems whose complements can be solved nondeterministically in polynomial time with the aid of an oracle for the

GRAPH ISOMORPHISM problem.
(c) If every graph with a chromatic factorisation has a certificate of factorisation of algebraic length bounded by a polynomial in n, then CHROMATIC FACTORISATION is in NP.

Proof:
(a)

Every instance of chromatic equivalence can be explained by a certificate of equivalence [20]. If there always exists such a certificate that is polynomially bounded in algebraic length, then it can be used as part of the guess, in a nondeterministic polynomial-time algorithm for chromatic equivalence. In order to enable efficient verification of the guess, it needs more than just a sequence of expressions. The information required includes statements of which certificate steps are used, nominations of which graphs in the expressions are "active" in each certificate step, some mappings between vertex sets of various pairs of graphs, and nominations of vertex pairs to be joined or unjoined or identified. We now give details.

Suppose every pair of chromatically equivalent graphs has an algebraic certificate of equivalence of algebraic length $\leq cn^k$, where c and k are constants. Given two chromatically equivalent graphs G and H, let $\mathbf{C}$ be such an algebraic certificate of equivalence for them. Let the sequence of expressions in $\mathbf{C}$ be $E_0, E_1, \ldots, E_l$, where $E_0 = G$, $E_l = H$ and $l \leq cn^k$ where $n = |V(G)| = |V(H)|$. For each $i \in \{0, \ldots, l-1\}$, let s_i be the number of the certificate step used to transform E_i to E_{i+1}, where $1 \leq s_i \leq 8$ and s_i indicates that certificate step CSs_i is used. If an algebraic step is used ($s_i = 8$), then we also need to specify which specific field axiom application is currently being used for this particular step in the algebraic certificate; we denote this by A_i. This means specifying the axiom together with the direction in which it is used. (For example, the additive inverse axiom could be used either as $0 \longrightarrow \Gamma - \Gamma$ or as $\Gamma - \Gamma \rightarrow 0$, where Γ is a graph. These are two separate field axiom applications.)

For each i such that step i is not part of an algebraic step (i.e., $s_i \neq 8$), we give the following information, which will enable the step to be verified. We specify which graphs in E_i and E_{i+1} are to play the roles of the graphs on each side of the arrow "$\longrightarrow$" in CSs_i. Suppose CSs_i has a_i graphs on the left of its arrow and b_i graphs on the right (where $\{a_i, b_i\} = \{1, 2\}$ when $1 \leq i \leq 5$ and $\{a_i, b_i\} = \{1, 3\}$ when $i \in \{6, 7\}$). Let $L_1^{(i)}, \ldots, L_{a_i}^{(i)}$ be the actual graphs in E_i that are used as the graphs on the left side of the arrow in CSs_i, and let $R_1^{(i)}, \ldots, R_{b_i}^{(i)}$ be the actual graphs in E_{i+1} that are used as the graphs on the right side of the arrow in CSs_i. The *root graph* of step i is defined to be L_1 if $1 \leq s_i \leq 6$ and R_1 if $s_i = 7$. The intention is simply that the vertex set of every graph appearing in CSs_i is a subset of the vertex set of the root graph, possibly with relabelling.

For each $i \in \{0, \ldots, l-1\}$, we need two functions $\rho_1^{(i)}, \rho_2^{(i)}$ that specify the correspondences between the vertices in the graphs used in step i. For each $j = 1, 2$, the function $\rho_j^{(i)}$ maps the vertex set of the root graph for step i to the vertex set of the j-th of the two or three non-root graphs used in CSs_i for step

i. These functions must respect adjacency in precisely the right way; they are not isomorphisms but, informally speaking, they are as close to isomorphisms as they can be or need to be under the circumstances. For example, consider CS1 (if $s_i = 1$), with root graph $L_1^{(i)}$. The relation $\rho_1^{(i)}$ is a bijection from $V(L_1^{(i)})$ to $V(L_1^{(i)} \backslash e)$ that preserves adjacency except that the endpoints of e in $L_1^{(i)}$ are not adjacent in $L_1^{(i)} \backslash e$. The relation $\rho_2^{(i)}$ is a surjection from $V(L_1^{(i)})$ to $V(L_1^{(i)}/e)$ that preserves adjacency except that the endpoints of e in $L_1^{(i)}$ are mapped to the same vertex in $L_1^{(i)}/e$ (and this vertex is not adjacent to itself in $L_1^{(i)}/e$, since loops are discarded in the version of contraction used when working with chromatic polynomials). We omit the details of the precise adjacency-respecting requirements for the other certificate steps; it is routine to work them out, based on the operations used.

If $1 \leq s_i \leq 5$, let $u^{(i)}v^{(i)}$ be the vertex pair used in CSs_i. These two vertices are to be nominated in the vertex set of the root graph $L_1^{(i)}$. They are adjacent if $s_i \in \{1, 4\}$ and nonadjacent if $s_i \in \{2, 3, 5\}$. Once nominated there, the functions $\rho_1^{(i)}, \rho_2^{(i)}$ enable the corresponding vertices in the other graphs used in CSs_i to be worked out.

If $s_i \in \{6, 7\}$, let U_i be the vertex set of the separating clique and let $r_i := |U_i|$ be its size. This set is to be nominated as a subset of $L_1^{(i)}$ if $s_i = 6$ and $R_1^{(i)}$ if $s_i = 7$. Once nominated there, the functions $\rho_1^{(i)}, \rho_2^{(i)}$ enable the corresponding subsets of vertices in the other graphs used in CSs_i to be worked out.

This completes the description of the information required for nonalgebraic steps.

For each i such that step i is part of an algebraic step ($s_i = 8$), involving application of a field axiom A_i, the information we give is slightly different. The field axiom applications may be represented as a list of rules of the form: left-side $\longrightarrow$ right-side. So we still need to nominate graphs $L_1^{(i)}, \ldots, L_{a_i}^{(i)}$ in E_i and graphs $R_1^{(i)}, \ldots, R_{b_i}^{(i)}$ in E_{i+1} that are used in this field axiom application A_i. But it is no longer the case that $\{a_i, b_i\}$ is always either $\{1, 2\}$ or $\{1, 3\}$. For example, if we are applying the distributive law, $A{\cdot}(B{+}C) \longrightarrow A{\cdot}B{+}A{\cdot}C$, then $(a_i, b_i) = (3, 4)$. We sometimes require an isomorphism between two graphs, which we denote by $\rho_1^{(i)}$, in order to enable verification that they are isomorphic and can be cancelled (in an application of either the additive inverse or multiplicative inverse axiom). But such mappings are mostly not needed. Furthermore, there is no need to specify vertex pairs or subsets. Applications of field axioms do not change any graphs, although they may introduce or remove graphs.

When giving all this information, some data representation decisions must arise, although (within reason) these decisions make no difference to whether or not the certificates can be verified in polynomial time. For example, each occurrence of a graph in an expression could be represented anew, by its own data structure spelling out all its vertices and edges, regardless of how many times graphs isomorphic to it have appeared previously (either earlier in the same expression, or in a previous expression in this certificate). Then the certificate would also have to specify many isomorphisms so that graphs which were the

same could indeed be verified to be so. Alternatively, we could give a detailed representation of a graph only the first time we use it, and thereafter just give an appropriate reference back to that graph. This is simpler and more economical, and we assume it is done this way, but our argument is easily adapted to the former approach.

We are now in a position to show that, under the given hypothesis, CHROMATIC EQUIVALENCE is in NP.

Given two chromatically equivalent graphs G and H, we use as our guess, or *certificate* (using this term now in its broader complexity-theoretic sense, which inspired but is not equivalent to our specific usage in "certificate of equivalence" etc.), the following information:

$$\forall i \in \{0, \ldots, l-1\} : \\ \left(s_i, E_i, (L_j^{(i)} : 1 \leq j \leq a_i), (R_j^{(i)} : 1 \leq j \leq b_i), \rho_1^{(i)}, \rho_2^{(i)}, u^{(i)}v^{(i)}, U_i, A_i\right). \tag{4}$$

There will always be some missing items in this list, and some special symbol can be used to represent them. For nonalgebraic steps ($s_i \leq 7$), A_i is absent. The vertex pair $u^{(i)}v^{(i)}$ is only needed if $s_i \leq 5$, and U_i is only needed if $s_i \in \{6, 7\}$. For algebraic steps ($s_i = 8$), the function $\rho_2^{(i)}$, vertex pairs $u^{(i)}v^{(i)}$ and vertex subsets U_i are not required, and the function $\rho_1^{(i)}$ may not be required.

For each i, the verification that $E_i \longrightarrow E_{i+1}$ requires us to verify that all the rules we have laid down in the construction of (4) are satisfied. This can be done in polynomial time. It includes: checking that the lists of the $L_j^{(i)}$ and $R_j^{(i)}$ are consistent with the nominated certificate step CSs_i (and field axiom A_i, where applicable); checking that the portions of the expressions that are not designated in the $L_j^{(i)}$ and $R_j^{(i)}$ for use by CSs_i are just copied across; and checking the appropriate adjacency-respecting properties of the maps $\rho_1^{(i)}, \rho_2^{(i)}$, taking into account the $u^{(i)}v^{(i)}$ and U_i. The fact that this all takes polynomial time depends on the facts that the number of graphs in each expression E_i is bounded by a linear function of certificate length l, and that the numbers of vertices of the graphs in the expressions may be restricted to some linear bound (using steps CS1–CS7; some applications of field axioms in CS8 could introduce much larger graphs in cancelling pairs, but for these to have any effect they must interact, via one of CS1–CS7, with a graph that ultimately derives from G or H using nonalgebraic steps, which constrains their size).

We must do this verification for each i, but this requires at most polynomially many iterations by our initial assumption on certificate length. So the entire verification procedure takes polynomial time. Therefore, if the lengths of certificates of equivalence are polynomially bounded, then CHROMATIC EQUIVALENCE is in NP.

We do not spell out the proofs of (b) and (c) in such detail, as they are similar in essence. We just outline them and comment on the key points of difference.

(b)

Suppose again that every pair of chromatically equivalent graphs has an algebraic certificate of equivalence of algebraic length $\leq cn^k$, where c and k are constants. Under this hypothesis, we prove that the class of graphs that are *not* chromatically unique belongs to $\mathrm{NP}^{\mathrm{GI}}$.

Let G be a graph that is not chromatically unique. Let H be a graph that is chromatically equivalent to G but not isomorphic to it. Our guess is now H together with an algebraic certificate of chromatic equivalence for G and H of algebraic length $\leq cn^k$, with all the associated information described in part (a). In other words, the guess is H together with the information in (4). To verify this guess, we first use the GRAPH ISOMORPHISM oracle to verify that $G \ncong H$, then we verify the information in (4) exactly as we did in part (a). The verification takes polynomial time, because of the oracle use and the reasoning given in (a). This completes the proof of (b).

(c)

Suppose now that every graph with a chromatic factorisation has an algebraic certificate of factorisation of algebraic length $\leq cn^k$, where c and k are constants. Let G be a graph with chromatic factorisation $P(G;\lambda) = P(H_1;\lambda)\, P(H_2;\lambda)/P(K_r;\lambda)$ using graphs H_1 and H_2 and clique K_r where $r \geq 0$.

Given a graph G which has a chromatic factorisation, our guess consists of an algebraic certificate of factorisation of algebraic length $\leq cn^k$, together with all the required associated information. This information is as in (4), except that the final expression E_l is not a single graph but rather the expression $H_1 \cdot H_2/K_r$. The verification is as in (a) above. □

4.2 Certificate length and the number of chromatic polynomials

The numbers $\#\mathrm{CP}(n)$ of different chromatic polynomials of connected graphs on n vertices for $n \leq 10$ is given by [20, Table 1] (for $8 \leq n \leq 10$) and [10]. (Our Proposition 3, below, justifies the focus on connected graphs.)

n	1	2	3	4	5	6	7	8	9	10
$\#\mathrm{CP}(n)$	1	1	2	5	14	50	231	1650	21121	584432
$n^{-2}\log_2 \#\mathrm{CP}(n)$	0.000	0.000	0.111	0.145	0.152	0.157	0.160	0.167	0.177	0.192

The ultimate trend of $\#\mathrm{CP}(n)$ is not clear from this data. We know that it can be no more than the number of different connected unlabelled graphs on n vertices, which is asymptotic to the number $2^{n(n-1)/2}$ of labelled graphs on n vertices (since, asymptotically, almost all labelled graphs are connected and have identity automorphism group). Does there exist $b < \frac{1}{2}$ such that $\#\mathrm{CP}(n) \leq 2^{bn^2}$?

Although $b = 0.2$ would suffice for $n \leq 10$, the numbers $n^{-2}\log_2 \#\mathrm{CP}(n)$ grow at an increasing rate over the range $6 \leq n \leq 10$, suggesting strongly that the true value of any such b is significantly greater. This growth cannot continue forever, due to the upper bound $\frac{1}{2}$ mentioned above. Does it flatten

out strictly below this upper bound? It is impossible to tell; there is too little data for any tentative extrapolation, let alone a persuasive one. By contrast, for stability polynomials, the answer to the analogous question appears to be affirmative [16]. Stability polynomials are essentially chromatic polynomials for graphic 2-polymatroids, which are cousins of graphic matroids.

Such a $b < \frac{1}{2}$ exists if and only if Bollobás, Pebody and Riordan's second conjecture, that asymptotically almost all graphs are chromatically unique [2], is false. That conjecture still seems to be wide open. The authors at the time wrote that they "do not have much evidence" for it, although "the simplest approach to disproving [it] fails" [2, p. 344]. There has been little progress since, at least for general graphs.

As we have just seen, the data gives no reason for taking a position either way on the conjecture, and the analogous conjecture for graphic 2-polymatroids actually seems likely to be false. We argue that, in exploring connections between the conjecture and certificate length, both possibilities (true/false) for the conjecture deserve consideration. There seems to be little optimism in the community that it will be resolved in the near future.

This question about b has implications for lengths of certificates of chromatic equivalence.

Theorem 2 *If $\#CP(n) \leq 2^{bn^2}$ for some fixed $b < \frac{1}{2}$, then for sufficiently large n, there exists a pair of chromatically equivalent graphs for which every certificate of equivalence has algebraic length $\Omega(n^2/\log n)$.*

Proof: Assume $\#\mathrm{CP}(n) \leq 2^{bn^2}$ for some fixed $b < \frac{1}{2}$.

The number of connected unlabelled graphs on n vertices is at least $2^{(\frac{1}{2}-\varepsilon)n^2}$, for some fixed $\varepsilon > 0$. So the average size of a chromatic equivalence class is $\geq 2^{(\frac{1}{2}-\varepsilon)n^2}/2^{bn^2} = 2^{(\frac{1}{2}-\varepsilon-b)n^2}$. Therefore there exists a chromatic equivalence class consisting of at least this many graphs. Let G be one of these graphs.

Suppose that, for sufficiently large n, every pair of chromatically equivalent graphs on n vertices has a certificate of chromatic equivalence of algebraic length $\leq L$.

Each expression in the algebraic certificate must have $\leq 2L$ terms, since the greatest possible increase in expression size, due to application of a single certificate step CS1–CS7 or a single application of a field axiom as part of CS8, is two. (The length might be unchanged, or decrease, instead.) It also has $\leq 2L$ instances of arithmetic operations from the standard set $\{+, -, \times, /\}$, for similar reasons. For each expression in the certificate, we apply one of CS1–CS7 or we apply a field axiom (and each field axiom might be applied in one of two directions). The total number of options here is constant, but we must then choose where in the expression to apply them. This involves choosing one or two graphs in the expression (for a non-algebraic certificate step, or for some field axiom applications) or one of the instances of an arithmetic operation (for other field axiom applications), so the number of choices of these is $\leq 4L^2 + 2L$. The upshot of this is that, for each expression, there are $\leq c_0 L^2$ expressions that may be obtained from it in a single non-algebraic certificate step or application

of a field axiom, where c_0 is a constant. Since there are $\leq 2L$ expressions in the certificate, the total number of certificates is $\leq L^{c_1 L}$, for some constant c_1. This implies that the size of the chromatic equivalence class of G has this same upper bound. So we have

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

giving

$$n^2 \leq c_2 L \log L$$

for some constant c_2. If for every constant $\varepsilon_1 > 0$ we have $L \leq \varepsilon_1 n^2/\log n$, then

$$n^2 \leq c_2 \frac{\varepsilon_1 n^2}{\log n}(2\log n - \log\log n + \log \varepsilon_1),$$

giving $1 \leq c_2 \varepsilon_1$ for all $\varepsilon_1 > 0$, a contradiction. So, for some $\varepsilon_1 > 0$, we must have $L > \varepsilon_1 n^2/\log n$. It follows that $L = \Omega(n^2/\log n)$. □

Our focus in this section on connected graphs is justified by the following remark, which tells us that disconnected graphs give us nothing really new.

Proposition 3 *If G_1 and G_2 are disconnected and chromatically equivalent, then there exist two connected chromatically equivalent graphs G_1^- and G_2^- whose chromatic polynomial is the same as that of G_1 and G_2 except for a factor λ^l.*

Proof: Suppose two disconnected graphs G_1 and G_2 have the same chromatic polynomial. They must have the same number of components, since this number is given by the multiplicity of zero as a chromatic root. Call this number k. Suppose we do the following to each graph: mark one vertex in each nonempty component, identify all these marked vertices (so combining all the components into a single component), and add new isolated vertices if necessary to ensure that the total number of isolated vertices is $k-1$. Let G_1' and G_2' be the graphs formed from G_1 and G_2 by this construction, and let G_1^- and G_2^- be the connected graphs formed from G_1' and G_2' by deleting all isolated vertices.

It is routine to show that the above construction $G_i \mapsto G_i'$ leaves the chromatic polynomial unchanged, using the fact that $P(H_1 \cup H_2; \lambda) = P((H_1 \cup_1 H_2) \cup K_1; \lambda)$ for any disjoint graphs H_1 and H_2 (see, e.g., [8, p. 56]). We therefore have

$$\lambda^{k-1} P(G_1^-; \lambda) = P(G_1'; \lambda) = P(G_1; \lambda) = P(G_2; \lambda) = P(G_2'; \lambda) = \lambda^{k-1} P(G_2^-; \lambda).$$

□

4.3 Certificate length and trees

All trees of a given order are chromatically equivalent. In investigating the relationship between chromatic equivalence and certificate length, it is natural to take a look at trees.

Theorem 4 *For each pair of trees of order $n \geq 3$, there is a certificate of chromatic equivalence for the pair, using only steps CS1 and CS2, of length $\leq 2(n-3)$.*

Proof: We use induction on n.

If $n = 3$, then there is only one tree on n vertices (up to isomorphism), which has a trivial certificate of chromatic equivalence with itself, consisting just of itself, which has length 0.

Now suppose $n > 3$. Suppose T_1 and T_2 are any two trees on n vertices. Now, any tree can be obtained from some tree with one fewer vertices by adding a leaf at an appropriate vertex. So, for $i = 1, 2$, we suppose T_i is obtained by adding, to a tree T_i^- on $n-1$ vertices, a leaf $v_i w_i$ to some vertex v_i of T_i^-, with w_i being a new vertex of degree 1 not in T_i^-. Since T_1^- and T_2^- have $< n$ vertices, the inductive hypothesis applies, and there is a certificate of chromatic equivalence between them which only uses CS1 and CS2 and has length at most $2((n-1)-3) = 2n-8$. Modify this certificate as follows. Every graph in it, starting with T_1^- at the beginning, has a copy of v_1 (possibly identified with other vertices as well). Attach a new leaf $v_1 w_1$ at every such copy of v_1, throughout the certificate. This gives a new certificate of equivalence, since all the certification steps remain valid after addition of the leaves. This new certificate starts with T_1 and demonstrates its equivalence to $T_2^- + v_1 w_1$, which is obtained by adding the leaf $v_1 w_1$ to T_2^-. We append two more certificate steps, certifying the chromatic equivalence of $T_2^- + v_1 w_1$ and T_2:

$$\begin{aligned}
T_2^- + v_1 w_1 &\longrightarrow (T_2^- + v_1 w_1) \backslash v_1 w_1 - (T_2^- + v_1 w_1)/v_1 w_1 \qquad (CS1) \\
&= (T_2^- \cup K_1) - T_2^- \\
&= (T_2^- + v_2 w_2) \backslash v_2 w_2 - (T_2^- + v_2 w_2)/v_2 w_2 \\
&\longrightarrow T_2^- + v_2 w_2 \qquad (CS2) \\
&= T_2.
\end{aligned}$$

Altogether, this gives a certificate of chromatic equivalence for T_1 and T_2 of total length $\leq 2(n-3)$. □

This upper bound is attained by the certificate of equivalence between the path P_n and the star S_n, each on n vertices.

For computation of certificate lengths for trees of order ≤ 7, see §6.1.2.

5 Software information

This section provides some information about the `certsearch` software produced in this project. For more detailed information about the finer points of the program implementation, we refer the reader to the source files available at `http://users.monash.edu/~kmorgan/Zoe/`.

5.1 Building the software

The `certsearch` software was written in the C programming language and was compiled with gcc. A makefile is included with the source code at `http://users.monash.edu/~kmorgan/Zoe/`.

The program `certsearch` uses `nauty` version 2.4, developed by Brendan McKay [14, 15] and available at `http://cs.anu.edu.au/~bdm/nauty`. In the software, `nauty` is used for the graph isomorphism checks performed during the search. The program also uses a function from some work by Kerri Morgan [17], which is used as an interface to `nauty`. This function was modified during this research to make it compatible with the graph data structures used by `certsearch`. The source code files for `nauty` and the modified code from Morgan are included along with the other source files required to build the `certsearch` program.

Also provided are the `n_polys` files, which contain lists of all of the chromatic equivalence classes for all non-chromatically unique graphs of order 4, 5, 6 and 7. These files were provided by Kerri Morgan. The `graphs*` files are also included. They give the adjacency matrices of all graphs of orders 4, 5, 6 and 7. These files are provided by Brendan McKay and are made available at `http://cs.anu.edu.au/~bdm/data/graphs.html`. Both the `n_polys` and the `graphs*` files are required by the automated exhaustive search functions.

5.2 Using the software

When running `certsearch` the user is presented with a number of options. Option 2 runs the batch experiments for all of the pairs of graphs of order $4 \leq n \leq 7$ and was used to find all of the computational results in this thesis. The certificates found during this search option are written out to the `order_*_certificates` files in the `graphs` directory.

5.3 Interpreting output certificates

This section contains information about interpreting the data output to file by `certsearch`. Please note that in the certificates in the `order_*_certificates` files and Appendix B, the graphs G and $G0$ are the same graph.

In the software, order n graphs are always defined over the set of vertices $\{0, \ldots, n-1\}$. The only graph operations that the software performs are edge deletion, edge addition and vertex identification. Edge contraction is implemented by removing the edge and then performing a vertex identification. Vertex identification alters the order of the resulting graph, and thus the labelling of the vertex set and the edge set. It is important to understand how this change is implemented in order to interpret the output of the program correctly. We use the following map to relabel our vertices and edges when identifying vertices v_i and v_j, $i < j$.

$$\phi(v_k, j) = \begin{cases} v_k & \text{if } k < j \\ v_{k-1} & \text{if } k \geq j. \end{cases}$$

The graph obtained by identifying vertices v_i and v_j, $i < j$, in G has vertex set $\{0, 1, \ldots, n-2\}$ and edge set $\{\phi(v_k, j)\phi(v_l, j) : v_k v_l \in E(G)\}$.

With this information, together with the adjacency matrices in the `graphs*` files, it is possible to interpret the certificates in Appendix B. Note that the certificates listed in the `order_*_certificates` files, which are available at `http://users.monash.edu/~kmorgan/Zoe/`, are preceded by the edge lists of the two graphs involved for which the certificate shows chromatic equivalence. The first of the two graphs listed is G in the certificate. The second is the graph found in the final expression of the certificate.

6 Experiments

The experiments were carried out as follows. For each chromatic equivalence class, a list of all unordered pairs of these graphs was created. Each of these pairs were given as an input to the program which used a bounded depth first search algorithm. Each node in the search tree represents an expression $E = \sum_{i=0}^{l} \text{sign}(i)G_i$, $l \geq 0$, where the G_i are graphs and $\text{sign}(i) \in \{\pm 1\}$. At each node we branch on all possible steps. First we branch on steps of type (CS1) and (CS3), that is deletion/contraction on edges in G_i or addition/identification for non-adjacent pairs of vertices in G_i, and then we branch on steps of type (CS2) and (CS4), that is, where the inverse of either an addition/identification or deletion/contraction operation can be applied to some pair of graphs in E. In order to reduce the search time, a second input to the program gave an upper bound M on the length of certificate to be found. If no certificate of length at most M was found, the bound was increased so a certificate could be found. If a certificate of length M is found during the search we record the certificate and decrement M, backtrack to the previous level of the search tree and continue the search for a shorter certificate. This algorithm found a shortest certificate for each pair and wrote this certificate out to the file of results for the corresponding graph order. This procedure was performed for graph orders 4, 5, 6, and 7.

In order to reduce the search space, our program only uses certificate steps **(CS1)**–**(CS4)**. This was a natural choice as these steps are based on the fundamental operations used to compute the chromatic polynomial (see (1) and (2)). The program finds the shortest certificates that use only these types of certificate step. The lengths of these certificates give an upper bound on the shortest lengths of all certificates of equivalence for graphs of these orders.

A list of the chromatic equivalence classes for graphs of order $4 \leq n \leq 7$ was provided by Kerri Morgan. These lists contain lists of the graphs, indexed by certain integers and arranged by equivalence class. The indices correspond to graph data provided by Brendan McKay, which is available at `http://cs.anu.edu.au/~bdm/data/graphs.html`. The program also uses `nauty` version 2.4, available at `http://cs.anu.edu.au/~bdm/nauty`, also developed by McKay [14, 15] to perform isomorphism checking during the running of the search algorithm. There were a total of 3821 pairs of chromatically equivalent graphs from 157 equivalence classes.

All of the experimental runs using `certsearch` were completed on a computer with the following specifications.

Computer:	Lenovo ThinkPad X1
Processor:	Intel(R) Core(TM) i5–2520M CPU @ 2.50GHz
Speed:	800.00 MHz
Memory (RAM):	3.8 GB
Operating System:	Linux openSUSE 12.2

6.1 Results

One of the main reasons for conducting the experiments was to find information about the length of shortest certificates of chromatic equivalence. Table 1 lists the certificate length data from the experiments. For each graph order, it lists the number of shortest certificates found of each length. Although the experiments considered only graphs of order at most 7, the certificate lengths that they found are, relative to corresponding graph order, very short. All but seven of the certificates found have length bounded by the order of the graphs. The remaining seven certificates have length 8 and were for graphs of order 7. All certificate lengths are $\leq 2n-6$. While these experiments only consider graphs of very small order, it is encouraging that so far the shortest certificates produced have been very short indeed, especially since the best known upper bound on the length of certificates is $< 2^{n^2/2}$, which is exponential in the order of the pair of graphs.

Figure 3 is an example of a certificate found using `certsearch`. All certificates found during the experiments for orders $4 \leq n \leq 6$ can be found in Appendix B. These certificates, as well as those for the graphs of order 7, are available at `http://users.monash.edu/~kmorgan/Zoe/`.

6.1.1 Schemas

A *schema* is a template for a certificate [20]. It represents a class of certificates that all share certain common subsequences of steps. A certificate which follows the pattern of certificate steps given in a schema is said to *belong* to the schema. Appendix A lists the schemas to which the certificates of equivalence found for graphs of order $4 \leq n \leq 6$ belong. These schemas were obtained by analysing the certificate data produced from the experiments.

Table 2 details the lengths of these schemas and the number of shortest certificates found by our program belonging to these schemas. Schemas given in Appendix A are not all of the *possible* schemas for certificates up to length 6, they are only those to which at least one certificate in the results belongs.

The program finds just one of potentially many shortest certificates for each input pair of graphs. The set of schemas to which the resulting certificates belong to are in part artefacts of how the graphs are labelled, as the labelling of edges affects the order of edge selection by the program. The order in which possible certificate steps are attempted will also affect which of the shortest

Graph Order	Length 2	Length 4	Length 6	Length 8	Total
4	1				1
5	8	1			9
6	113	48	2		163
7	1610	1759	272	7	3648

Table 1: The lengths of shortest certificates found for chromatically equivalent pairs of graphs of order ≤ 7.

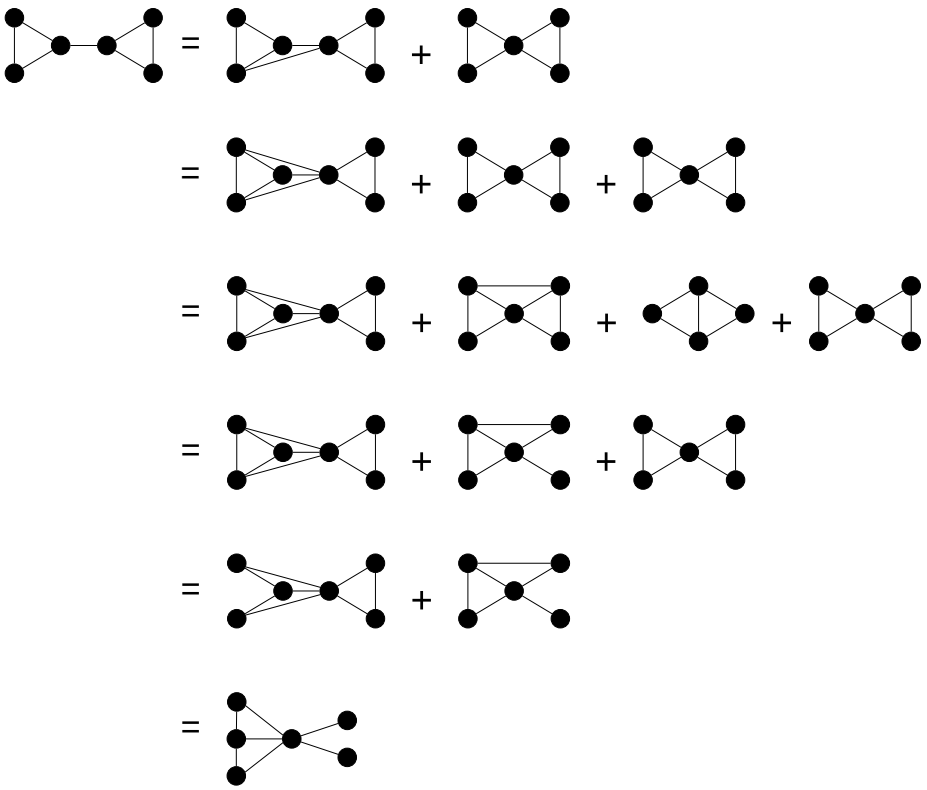

Figure 3: A certificate of equivalence for two graphs of order 6, belonging to Schema 14.

Length		2		4											6	
Schema		S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15
Order	4	1														
	5	4	4	1												
	6	62	51	3	6	11	1	2	2	9	5	6	1	2	1	1

Table 2: The distribution of encountered shortest certificates amongst the schemas for graphs of order ≤ 6.

certificates is found for a given graph. Consequently, the schemas to which the certificates from our results belong are not necessarily the only ones for which certificates of the same length for each pair could be produced, although they are as short as any. There may exist other certificates of the same length for a given pair of graphs that belong to some other schema; either one of the others listed in Appendix A, or another schema altogether.

Nevertheless, we are still able to draw some important conclusions from the information we do have. Since all the shortest certificates that were found belong to a small set of only 15 schemas, and there certainly exist other possible schemas of these lengths, we can say that the entire set of possible schemas may not need to be considered when searching for shortest certificates.

The vast majority of the certificates found belong to Schemas 1 and 2. This is not unexpected, as these two schemas describe the only two sequences of certificate steps (when restricted to certificate steps of type **(CS1)**–**(CS4)**) that can produce a certificate of length 2. Schemas 14 and 15 both describe length 6 certificates, and the remaining schemas describe certificates with length 4.

The *edge difference* of graphs G and G' is the smallest $d \in \mathbb{N}$ such that there exists $A \subseteq E(G)$ and $B \subseteq V^{(2)} \setminus E(G)$, $|A| + |B| = d$ where $G - A + B$ is isomorphic to G'. If $G \sim G'$, then $|A| = |B|$ and d is even. In each of the schemas for $G \sim G'$, the final expression gives a graph, isomorphic to G', obtained by deleting and adding some edge. For example, the final expression in Schema 3 is the graph $(G+e\backslash f+g\backslash h)$. The edge difference of pairs of graphs with certificates that belong to this schema is 4. Schemas 1 to 12 and 15 give certificates with length equal to the edge difference of the pair of graphs.

However, in Schemas 13 and 14, the edge difference for them both is two less than their respective certificate lengths. Our program only uses certificate steps **(CS1**–**CS4)**. It is possible that the use of all the types of certificate steps listed in Section 3 may produce certificates for these pairs of graphs that have a length less than or equal to their edge difference.

6.1.2 Certificates for trees

Table 3 gives the lengths of the shortest certificates of equivalence found for pairs of trees of order n, for the range $4 \leq n \leq 7$.

The bound on certificate length suggested by the table aligns with Theorem 4.

Tree Order	Length 2	Length 4	Length 6	Length 8
4	1			
5	2	1		
6	9	5	1	
7	27	20	7	1

Table 3: The lengths of shortest certificates found for chromatically equivalent trees of order ≤ 7.

7 Conclusions and future work

The certificates found using the `certsearch` software tool are all very short. They also belong to only a small number of schemas. We give an upper bound of $2(n-3)$ on the lengths of certificates of equivalence for trees. This class of graph includes the star and path graphs, which were subsequently shown for orders $4 \leq n \leq 7$ to have the longest certificates amongst all graphs of the same order in the experimental results.

In general, the certificates that have been found so far are significantly shorter than the upper bounds on their length known at this time, so it is possible that further research could uncover tighter upper bounds. Although the chromatic polynomial has been investigated in considerable depth, there has been little research into its algebraic theory. Chromatic equivalence has been the topic of much research, but knowledge about the characterisation of chromatically equivalent graphs in general is far from complete. The certificates of equivalence that have been found so far provide some tantalising hints as to how they may behave generally, but there remain a great number of things about them that are unknown. Consequently, there is a wealth of potential directions for further research into certificates for properties of the chromatic polynomial. Some of these avenues are outlined below.

The schemas found in this research could be used to reduce the time taken to find certificates. By first searching for certificates between pairs of graphs using the schemas found in this research, it may be possible to find certificates for larger orders of graph. Attempting to find certificates that belong to the more common schemas may improve the time taken to find certificates.

Our program finds a shortest certificate for a given pair of graphs. However, there may be other certificates for such a pair that have the same length. A search for *all* of the certificates of shortest length for a pair of chromatically equivalent graphs could be implemented. This may give a wider range of possible schemas which could be used in our search for short certificates. The search algorithm could be expanded to include the complete set of certificate steps studied by Morgan and Farr [20]. It is quite possible that shorter certificates of equivalence could be found for some of the certificates found in this project. It is also possible that such a method would find shorter certificates, in general.

Certificates of factorisation use the same certificate steps as certificates of equivalence. Extending the search capabilities of our algorithms to include

searching for certificates of factorisation is an avenue for further work.

Acknowledgements

Zoe Bukovac thanks Chris Monteith for assistance with `nauty`. We thank the referees for their helpful comments.

References

[1] G. Birkhoff. A determinant formula for the number of ways of coloring a map. *Ann. of Math.*, 14:42–46, 1912–1913.

[2] B. Bollobás, L. Pebody, and O. Riordan. Contraction-deletion invariants for graphs. *J. Combin. Theory Ser. B*, 80:320–345, 2000. doi:10.1006/jctb.2000.1988.

[3] Z. Bukovac. Certificates for properties of chromatic polynomials of graphs, 2012. BCompSc Honours Thesis, Monash University.

[4] C. Chao and E. Whitehead Jr. On chromatic equivalence of graphs. In Y. Alavi and D. Lick, editors, *Theory and Applications of Graphs: Proceedings, Michigan, May 11-15, 1976*, volume 642 of *Lecture Notes in Mathematics*, pages 121–131, Berlin, 1978. Springer-Verlag. doi:10.1007/BFb0070369.

[5] G. Chia. A bibliography on chromatic polynomials. *Discrete Math.*, 172:175–191, 1997. doi:10.1016/S0012-365X(97)90031-5.

[6] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, fourth edition, 2010.

[7] F. Dong, K. Koh, and C. Soh. Divisibility of certain coefficients of the chromatic polynomials. *Discrete Math.*, 275:311–317, 2004. doi:10.1016/j.disc.2003.05.007.

[8] F. Dong, K. Koh, and K. Teo. *Chromatic Polynomials and Chromaticity of Graphs*. World Scientific, Singapore, 2005.

[9] G. Farr. The complexity of counting colourings of subgraphs of the grid. *Combin. Probab. Comput.*, 15:377–383, 2006. doi:10.1017/S0963548305007364.

[10] T. Hoppe and A. Petrone. A245883: Number of distinct chromatic polynomials among all connected graphs on n nodes, 2014. 5 August 2014. URL: http://oeis.org/A245883.

[11] F. Jaeger, D. Vertigan, and D. Welsh. On the computational complexity of the Jones and Tutte polynomials. *Math. Proc. Cambridge Philos. Soc.*, 108:35–53, 1990. doi:10.1017/S0305004100068936.

[12] R. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum, Boston, 1972. doi:10.1007/978-1-4684-2001-2_9.

[13] N. Linial. Hard enumeration problems in geometry and combinatorics. *SIAM J. Algebraic Discrete Methods*, 7:331–335, 1986. doi:10.1137/0607036.

[14] B. McKay. Practical graph isomorphism. *Congressus Numerantium*, 30:45–87, 1981.

[15] B. McKay and A. Piperno. Practical graph isomorphism, II. *J. Symbolic Comput.*, 60:94–112, 2014. doi:10.1016/j.jsc.2013.09.003.

[16] R. Mo, G. Farr, and K. Morgan. Certificates for properties of stability polynomials of graphs. *Electron. J. Combin.*, 21:#P1.66 (25pp), 2014.

[17] K. Morgan. *Algebraic Aspects of the Chromatic Polynomial*. PhD thesis, Monash University, 2010. Available from http://arrow.monash.edu.au/hdl/1959.1/470667.

[18] K. Morgan. Pairs of chromatically equivalent graphs. *Graphs Combin.*, 27:547–556, 2011. doi:10.1007/s00373-010-0984-z.

[19] K. Morgan and G. Farr. Certificates of factorisation for a class of triangle-free graphs. *Electron. J. Combin.*, 16:Research Paper R75, 2009.

[20] K. Morgan and G. Farr. Certificates of factorisation for chromatic polynomials. *Electron. J. Combin.*, 16:Research Paper R74, 2009.

[21] K. Morgan and G. Farr. Non-bipartite chromatic factors. *Discrete Math.*, 312:1166–1170, 2012. doi:10.1016/j.disc.2011.12.002.

[22] H. Whitney. *The coloring of graphs*. PhD thesis, Harvard University, 1932.

Appendices

A Schemas

All of the certificates found for pairs of chromatically equivalent graphs of order $4 \leq n \leq 6$ belong to one of the following schemas.

Schema 1:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)} \\ &= (G+e\backslash f). && \textbf{(CS4)} \end{aligned}$$

Schema 2:

$$\begin{aligned} G &= (G\backslash e) \;-\; (G/e) && \textbf{(CS1)} \\ &= (G\backslash e+f). && \textbf{(CS2)} \end{aligned}$$

Schema 3:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)} \\ &= (G+e\backslash f) \;-\; (G+e/f) \;+\; (G/e) && \textbf{(CS1)} \\ &= (G+e\backslash f+g) \;+\; (G/e) && \textbf{(CS2)} \\ &= (G+e\backslash f+g\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 4:

$$\begin{aligned} G &= (G\backslash e) \;-\; (G/e) && \textbf{(CS1)} \\ &= (G\backslash e+f) && \textbf{(CS2)} \\ &= (G\backslash e+f\backslash g) \;-\; (G\backslash e+f/g) && \textbf{(CS1)} \\ &= (G\backslash e+f\backslash g+h). && \textbf{(CS2)} \end{aligned}$$

Schema 5:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f\backslash g) \;+\; (G/e) && \textbf{(CS4)}\\ &= (G+e+f\backslash g\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 6:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e) \;+\; (G/e\backslash f) \;+\; (G/e/f) && \textbf{(CS1)}\\ &= (G+e) \;+\; (G/e\backslash f\backslash g) && \textbf{(CS4)}\\ &= (G+e\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 7:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f\backslash g) \;+\; (G+e/f) && \textbf{(CS4)}\\ &= (G+e+f\backslash g\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 8:

$$\begin{aligned} G &= (G\backslash e) \;-\; (G/e) && \textbf{(CS1)}\\ &= (G\backslash e+f) && \textbf{(CS2)}\\ &= (G\backslash e+f+g) \;+\; (G\backslash e+f/g) && \textbf{(CS3)}\\ &= (G\backslash e+f+g\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 9:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e\backslash f) && \textbf{(CS4)}\\ &= (G+e\backslash f+g) \;+\; (G+e\backslash f/g) && \textbf{(CS3)}\\ &= (G+e\backslash f+g\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 10:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e\backslash f) \;-\; (G+e/f) \;+\; (G/e) && \textbf{(CS1)}\\ &= (G+e\backslash f\backslash g) \;-\; (G+e/f) && \textbf{(CS4)}\\ &= (G+e\backslash f\backslash g+h). && \textbf{(CS2)} \end{aligned}$$

Schema 11:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e\backslash f) && \textbf{(CS4)}\\ &= (G+e\backslash f\backslash g) \;-\; (G+e\backslash f/g) && \textbf{(CS1)}\\ &= (G+e\backslash f\backslash g+h). && \textbf{(CS2)} \end{aligned}$$

Schema 12:

$$\begin{aligned} G &= (G\backslash e) \;-\; (G/e) && \textbf{(CS1)}\\ &= (G\backslash e+f) \;+\; (G\backslash e/f) \;-\; (G/e) && \textbf{(CS3)}\\ &= (G\backslash e+f\backslash g) \;-\; (G/e) && \textbf{(CS4)}\\ &= (G\backslash e+f\backslash g+h). && \textbf{(CS2)} \end{aligned}$$

Schema 13:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e) \;+\; (G/e\backslash f) \;-\; (G/e/f) && \textbf{(CS1)}\\ &= (G+e) \;+\; (G/e\backslash f+g) && \textbf{(CS2)}\\ &= (G+e\backslash h). && \textbf{(CS4)} \end{aligned}$$

Schema 14:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f+g) \;+\; (G+e/f/g) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f+g\backslash h) \;+\; (G/e) && \textbf{(CS4)}\\ &= (G+e+f\backslash i) \;+\; (G+e/f+g\backslash h) && \textbf{(CS4)}\\ &= (G+e+f\backslash i\backslash j). && \textbf{(CS4)} \end{aligned}$$

Schema 15:

$$\begin{aligned} G &= (G+e) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f) \;+\; (G+e/f) \;+\; (G/e) && \textbf{(CS3)}\\ &= (G+e+f\backslash g) \;+\; (G/e) && \textbf{(CS4)}\\ &= (G+e+f\backslash g\backslash h) && \textbf{(CS4)}\\ &= (G+e+f\backslash g\backslash h+i) \;+\; (G+e+f\backslash g\backslash h/i) && \textbf{(CS3)}\\ &= (G+e+f\backslash g\backslash h+i\backslash j). && \textbf{(CS4)} \end{aligned}$$

B Certificates

The following are the certificates found for pairs of chromatically equivalent graphs of order $4 \leq n \leq 6$. A somewhat more verbose version of these certificates, along with all of the certificates for the graphs of order 7, can be found in the files labelled `order_*_certificates` available at `http://users.`

monash.edu/~kmorgan/Zoe/. The numbers given to denote which graphs each certificate corresponds to are those listed in the graphs* files, also found at http://users.monash.edu/~kmorgan/Zoe/. These files give the adjacency matrices of the graphs.

B.1 Order 4

Graph Pair	Certificate
2 & 1	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}

Table 4: Certificates belonging to Schema 1.

B.2 Order 5

Graph Pair	Certificate
10 & 4	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}
11 & 6	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}
12 & 6	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,2)}
12 & 11	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,2)}

Table 5: Certificates belonging to Schema 1.

Graph Pair	Certificate
2 & 1	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)}
9 & 2	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(2,1)}
4 & 3	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,1)}
10 & 3	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,3)}

Table 6: Certificates belonging to Schema 2.

Graph Pair	Certificate
9 & 1	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)} - G4{G1/(2,0)} + G2{G0/(1,0)} = G5{G3+(2,1)} + G2{G0/(1,0)} = G6{G5-(4,0)}

Table 7: Certificates belonging to Schema 3.

B.3 Order 6

59 & 11	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(5,2)}
31 & 7	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(5,2)}
31 & 9	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(5,2)}
5 & 4	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
19 & 5	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(4,2)}
19 & 15	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(4,1)}
107 & 74	G = G1{G0+(3,2)} + G2{G0/(3,2) = G3{G1-(5,3)}
51 & 25	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,3)}

Table 8: Certificates belonging to Schema 1.

51 & 33	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
51 & 46	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(3,1)}
60 & 36	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(5,2)}
65 & 36	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(4,2)}
65 & 60	G = G1{G0+(3,2)} + G2{G0/(3,2)} = G3{G1-(4,2)}
63 & 61	G = G1{G0+(3,2)} + G2{G0/(3,2)} = G3{G1-(5,2)}
15 & 5	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(5,2)}
16 & 6	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)}
20 & 18	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(4,2)}
21 & 18	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(3,0)}
21 & 20	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)}
30 & 6	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
30 & 16	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,4)}
30 & 20	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(4,1)}
54 & 39	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
54 & 49	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(3,1)}
67 & 57	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(4,2)}
85 & 57	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}

Table 9: Certificates belonging to Schema 1 (Continued).

100 & 69	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,1)}
87 & 71	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}
17 & 8	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)}
24 & 22	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
32 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
32 & 17	G = G1{G0+(5,3)} + G2{G0/(5,3)} = G3{G1-(4,0)}
32 & 22	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(4,1)}
32 & 24	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(4,0)}
32 & 27	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(5,2)}
50 & 22	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,3)}
50 & 24	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,4)}
50 & 32	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
76 & 22	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
77 & 24	G = G1{G0+(3,0)} + G2{G0/(3,0)} = G3{G1-(5,1)}
77 & 32	G = G1{G0+(3,0)} + G2{G0/(3,0)} = G3{G1-(3,1)}
34 & 12	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
34 & 26	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1-(4,0)}
41 & 40	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}

Table 10: Certificates belonging to Schema 1 (Continued).

52 & 26	G = G1{G0+(5,2)} + G2{G0/(5,2)} = G3{G1-(3,0)}
52 & 40	G = G1{G0+(2,0)} + G2{G0/(2,0)} = G3{G1-(3,1)}
52 & 41	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
79 & 40	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
79 & 52	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,0)}
55 & 43	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
98 & 92	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,2)}
78 & 35	G = G1{G0+(3,0)} + G2{G0/(3,0)} = G3{G1-(3,1)}
80 & 37	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
84 & 62	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}
91 & 58	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)}
91 & 86	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,2)}
89 & 64	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,1)}
88 & 70	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}
93 & 70	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,2)}
93 & 88	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,2)}
103 & 75	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)}
86 & 58	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)}

Table 11: Certificates belonging to Schema 1 (Continued).

9 & 7	`G = G1{G0-(4,2)} - G2{G0/(4,2)}` `= G3{G1+(5,2)}`
23 & 7	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(5,3)}`
23 & 9	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(4,3)}`
31 & 23	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(3,2)}`
2 & 1	`G = G1{G0-(4,0)} - G2{G0/(4,0)}` `= G3{G1+(5,4)}`
4 & 2	`G = G1{G0-(4,0)} - G2{G0/(4,0)}` `= G3{G1+(5,0)}`
5 & 2	`G = G1{G0-(4,1)} - G2{G0/(4,1)}` `= G3{G1+(5,1)}`
15 & 2	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(5,3)}`
15 & 4	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(3,1)}`
33 & 25	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(3,2)}`
46 & 33	`G = G1{G0-(3,1)} - G2{G0/(3,1)}` `= G3{G1+(4,1)}`
6 & 3	`G = G1{G0-(4,1)} - G2{G0/(4,1)}` `= G3{G1+(5,1)}`
16 & 3	`G = G1{G0-(4,1)} - G2{G0/(4,1)}` `= G3{G1+(5,4)}`
18 & 6	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(4,0)}`
18 & 16	`G = G1{G0-(4,1)} - G2{G0/(4,1)}` `= G3{G1+(5,1)}`
20 & 6	`G = G1{G0-(3,0)} - G2{G0/(3,0)}` `= G3{G1+(4,3)}`
20 & 16	`G = G1{G0-(4,2)} - G2{G0/(4,2)}` `= G3{G1+(5,2)}`
21 & 16	`G = G1{G0-(4,2)} - G2{G0/(4,2)}` `= G3{G1+(5,2)}`

Table 12: Certificates belonging to Schema 2.

49 & 39	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,1)}
82 & 49	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(2,1)}
10 & 8	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)}
22 & 10	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,0)}
22 & 17	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)}
24 & 8	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(5,3)}
24 & 10	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)}
27 & 8	G = G1{G0-(4,0)} - G2{G0/(4,0)} = G3{G1+(5,4)}
32 & 8	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(5,3)}
50 & 27	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,1)}
77 & 27	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(5,3)}
77 & 50	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,3)}
77 & 76	G = G1{G0-(5,0)} - G2{G0/(5,0)} = G3{G1+(5,3)}
26 & 12	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,0)}
40 & 12	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)}
40 & 34	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)}
41 & 12	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)}
41 & 26	G = G1{G0-(4,0)} - G2{G0/(4,0)} = G3{G1+(5,3)}

Table 13: Certificates belonging to Schema 2 (Continued).

52 & 34	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)}
79 & 26	G = G1{G0-(4,0)} - G2{G0/(4,0)} = G3{G1+(5,4)}
43 & 14	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)}
56 & 43	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(5,0)}
83 & 43	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(5,2)}
83 & 55	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)}
83 & 56	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(2,1)}
106 & 102	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(4,3)}
35 & 28	G = G1{G0-(4,1)} - G2{G0/(4,1)} = G3{G1+(5,1)}
78 & 28	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(5,3)}
62 & 37	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,2)}
84 & 37	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,3)}
84 & 80	G = G1{G0-(4,1)} - G2{G0/(4,1)} = G3{G1+(5,3)}
58 & 45	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,1)}
86 & 45	G = G1{G0-(3,1)} - G2{G0/(3,1)} = G3{G1+(4,3)}

Table 14: Certificates belonging to Schema 2 (Continued).

77 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} - G4{G1/(3,1)} + G2{G0/(1,0)} = G5{G3+(3,2)} + G2{G0/(1,0)} = G6{G5-(5,1)}
77 & 22	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} - G4{G1/(3,1)} + G2{G0/(1,0)} = G5{G3+(3,2)} + G2{G0/(1,0)} = G6{G5-(5,2)}
56 & 55	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)} - G4{G1/(3,0)} + G2{G0/(1,0)} = G5{G3+(5,0)} + G2{G0/(1,0)} = G6{G5-(4,0)}

Table 15: Certificates belonging to Schema 3.

4 & 1	G = G1{G0-(4,0)} - G2{G0/(4,0)} = G3{G1+(5,0)} = G4{G3-(4,1)} - G5{G3/(4,1)} = G6{G4+(5,1)}
5 & 1	G = G1{G0-(4,1)} - G2{G0/(4,1)} = G3{G1+(5,1)} = G4{G3-(4,0)} - G5{G3/(4,0)} = G6{G4+(5,4)}
15 & 1	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(5,3)} = G4{G3-(4,1)} - G5{G3/(4,1)} = G6{G4+(5,4)}
21 & 3	G = G1{G0-(4,2)} - G2{G0/(4,2)} = G3{G1+(5,2)} = G4{G3-(4,1)} - G5{G3/(4,1)} = G6{G4+(5,4)}
56 & 14	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(5,0)} = G4{G3-(3,1)} - G5{G3/(3,1)} = G6{G4+(4,1)}
83 & 14	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(5,2)} = G4{G3-(3,1)} - G5{G3/(3,1)} = G6{G4+(4,3)}

Table 16: Certificates belonging to Schema 4.

19 & 2	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,0)} + G4{G1/(2,0)} + G2{G0/(1,0)} = G5{G3-(4,1)} + G2{G0/(1,0)} = G6{G5-(5,1)}
19 & 4	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,0)} + G4{G1/(2,0)} + G2{G0/(1,0)} = G5{G3-(4,2)} + G2{G0/(1,0)} = G6{G5-(5,0)}
17 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,0)} + G4{G1/(2,0)} + G2{G0/(1,0)} = G5{G3-(5,2)} + G2{G0/(1,0)} = G6{G5-(4,1)}
27 & 22	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,1)} + G4{G1/(2,1)} + G2{G0/(1,0)} = G5{G3-(4,0)} + G2{G0/(1,0)} = G6{G5-(1,0)}
27 & 24	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,1)} + G4{G1/(2,1)} + G2{G0/(1,0)} = G5{G3-(4,0)} + G2{G0/(1,0)} = G6{G5-(3,0)}
76 & 17	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G4{G1/(3,0)} + G2{G0/(1,0)} = G5{G3-(3,1)} + G2{G0/(1,0)} = G6{G5-(5,1)}
77 & 8	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G4{G1/(3,0)} + G2{G0/(1,0)} = G5{G3-(3,1)} + G2{G0/(1,0)} = G6{G5-(4,2)}
77 & 17	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G4{G1/(3,0)} + G2{G0/(1,0)} = G5{G3-(3,1)} + G2{G0/(1,0)} = G6{G5-(5,2)}
79 & 34	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G4{G1/(3,0)} + G2{G0/(1,0)} = G5{G3-(3,1)} + G2{G0/(1,0)} = G6{G5-(5,1)}
91 & 45	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,1)} + G4{G1/(2,1)} + G2{G0/(1,0)} = G5{G3-(4,1)} + G2{G0/(1,0)} = G6{G5-(5,1)}
80 & 62	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,1)} + G4{G1/(2,1)} + G2{G0/(1,0)} = G5{G3-(4,0)} + G2{G0/(1,0)} = G6{G5-(5,3)}

Table 17: Certificates belonging to Schema 5.

27 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G1{G0+(1,0)} + G3{G2+(1,0)} + G4{G2/(1,0)} = G1{G0+(1,0)} + G5{G3-(3,2)} = G6{G1-(4,3)}

Table 18: Certificates belonging to Schema 6.

46 & 25	G = G1{G0+(2,1)} + G2{G0/(2,1)} = G3{G1+(5,0)} + G4{G1/(5,0)} + G2{G0/(2,1)} = G4{G1/(5,0)} + G5{G3-(3,1)} = G6{G5-(4,0)}
50 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(4,3)} + G4{G1/(4,3)} + G2{G0/(1,0)} = G4{G1/(4,3)} + G5{G3-(5,0)} = G6{G5-(1,0)}

Table 19: Certificates belonging to Schema 7.

18 & 3	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,0)} = G4{G3+(4,3)} + G5{G3/(4,3)} = G6{G4-(5,0)}
20 & 3	G = G1{G0-(3,0)} - G2{G0/(3,0)} = G3{G1+(4,3)} = G4{G3+(4,0)} + G5{G3/(4,0)} = G6{G4-(5,0)}

Table 20: Certificates belonging to Schema 8.

21 & 6	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)} = G4{G3+(2,1)} + G5{G3/(2,1)} = G6{G4-(4,2)}
30 & 3	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(5,2)}
85 & 67	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)} = G4{G3+(3,0)} + G5{G3/(3,0)} = G6{G4-(5,3)}
27 & 17	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(5,3)}
50 & 8	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(4,2)}
50 & 17	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(5,4)}
76 & 32	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3+(2,1)} + G5{G3/(2,1)} = G6{G4-(4,2)}
52 & 12	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(4,2)}
55 & 14	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3+(2,0)} + G5{G3/(2,0)} = G6{G4-(4,2)}

Table 21: Certificates belonging to Schema 9.

30 & 18	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,0)} - G4{G1/(4,0)} + G2{G0/(1,0)} = - G4{G1/(4,0)} + G5{G3-(4,1)} = G6{G5+(3,1)}
76 & 24	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,1)} - G4{G1/(5,1)} + G2{G0/(1,0)} = - G4{G1/(5,1)} + G5{G3-(3,1)} = G6{G5+(2,1)}
40 & 26	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,0)} - G4{G1/(4,0)} + G2{G0/(1,0)} = - G4{G1/(4,0)} + G5{G3-(4,1)} = G6{G5+(5,2)}
41 & 34	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,0)} - G4{G1/(4,0)} + G2{G0/(1,0)} = - G4{G1/(4,0)} + G5{G3-(4,2)} = G6{G5+(5,3)}
79 & 41	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(2,0)} - G4{G1/(2,0)} + G2{G0/(1,0)} = - G4{G1/(2,0)} + G5{G3-(4,2)} = G6{G5+(2,1)}

Table 22: Certificates belonging to Schema 10.

30 & 21	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(5,4)} = G4{G3-(3,0)} - G5{G3/(3,0)} = G6{G4+(3,2)}
22 & 8	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,0)} = G4{G3-(4,2)} - G5{G3/(4,2)} = G6{G4+(5,2)}
24 & 17	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(4,1)} = G4{G3-(3,0)} - G5{G3/(3,0)} = G6{G4+(5,3)}
76 & 10	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3-(4,2)} - G5{G3/(4,2)} = G6{G4+(5,2)}
79 & 12	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3-(4,2)} - G5{G3/(4,2)} = G6{G4+(3,0)}
76 & 50	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1-(3,1)} = G4{G3-(2,0)} - G5{G3/(2,0)} = G6{G4+(4,1)}

Table 23: Certificates belonging to Schema 11.

82 & 39	G = G1{G0-(2,0)} - G2{G0/(2,0)} = G3{G1+(2,1)} + G4{G1/(2,1)} - G2{G0/(2,0)} = G5{G3-(4,2)} - G2{G0/(2,0)} = G6{G5+(5,2)}

Table 24: Certificates belonging to Schema 12.

82 & 54	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G1{G0+(1,0)} + G3{G2-(1,0)} - G4{G2/(1,0)} = G1{G0+(1,0)} + G5{G3+(3,2)} = G6{G1-(4,0)}
76 & 27	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G1{G0+(1,0)} + G3{G2-(1,0)} - G4{G2/(1,0)} = G1{G0+(1,0)} + G5{G3+(3,2)} = G6{G1-(4,2)}

Table 25: Certificates belonging to Schema 13.

76 & 8	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G4{G1/(3,0)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G5{G4+(2,1)} + G6{G4/(2,1)} + G2{G0/(1,0)} = G3{G1+(3,0)} + G7{G5-(3,2)} + G2{G0/(1,0)} = G7{G5-(3,2)} + G8{G3-(3,1)} = G9{G8-(4,2)}

Table 26: Certificates belonging to Schema 14.

19 & 1	G = G1{G0+(1,0)} + G2{G0/(1,0)} = G3{G1+(2,0)} + G4{G1/(2,0)} + G2{G0/(1,0)} = G5{G3-(4,1)} + G2{G0/(1,0)} = G6{G5-(5,1)} = G7{G6+(4,0)} + G8{G6/(4,0)} = G9{G7-(4,2)}

Table 27: Certificates belonging to Schema 15.