

A Bipartite Graph with a Nonunimodal Acyclic Polynomial

Yuran Xiu
IIIS, Tsinghua University
xiuyr23@mails.tsinghua.edu.cn

14 October 2026

Abstract

We construct a bipartite graph on 230 vertices whose acyclic polynomial, which enumerates vertex subsets inducing forests, is not unimodal. This answers Question 3.3 of Barton, Brown, and Pike negatively. The graph has a common set of 50 vertices and 15 branches, each with two sets of six vertices. A partition according to the number of selected common vertices gives a three-term formula for the complete induced-forest polynomial. Its coefficients have a strict descent from degree 62 to degree 63 and a later strict ascent from degree 66 to degree 67. The actual graph, the unrestricted subset-counting formula, and the exact inequalities have been checked in Lean.

1 The question and the construction

For a finite simple graph G , let

$$A(G, x) = \sum_{\substack{S \subseteq V(G) \\ G[S] \text{ is a forest}}} x^{|S|} = \sum_{j=0}^{|V(G)|} a_j(G) x^j. \quad (1)$$

The polynomial counts vertex subsets, rather than edge subsets or spanning forests. Its coefficients are *unimodal* if some index m satisfies $a_0 \leq \dots \leq a_m \geq a_{m+1} \geq \dots \geq a_{|V(G)|}$. The convention is weak unimodality, allowing repeated coefficients.

Barton, Brown, and Pike ask in Question 3.3 of the published version of [1] whether $A(G, x)$ is unimodal for every bipartite graph. We give an explicit negative answer.

For positive integers a, b, c, t , define $G(a, b, c, t)$ as follows. Take mutually disjoint sets

$$R, M_1, \dots, M_t, L_1, \dots, L_t, \quad |R| = c, |M_i| = a, |L_i| = b.$$

Join every vertex of M_i to every vertex of $R \cup L_i$, and include no other edges. Thus

$$R \cup \bigcup_{i=1}^t L_i \quad \text{and} \quad \bigcup_{i=1}^t M_i \quad (2)$$

are bipartition classes. The graph has $c + t(a + b)$ vertices and $ta(c + b)$ edges.

Theorem 1.1. *The bipartite graph $G(6, 6, 50, 15)$ has 230 vertices and 5040 edges, with bipartition classes of sizes 140 and 90. Its acyclic polynomial is not unimodal.*

We will derive a formula for the whole family and then give exact integer inequalities for this witness. No minimum-order assertion is made, and no assertion about root locations is needed.

2 An induced-forest criterion

Fix a selected vertex subset S . Write

$$q = |S \cap R|, \quad m_i = |S \cap M_i|, \quad \ell_i = |S \cap L_i|.$$

A complete bipartite graph $K_{u,v}$ is a forest if and only if $u \leq 1$ or $v \leq 1$. If $u, v \geq 2$, it contains a four-cycle; otherwise it is a star or an edgeless graph.

Lemma 2.1. *The graph induced by S is a forest precisely in the following cases:*

1. $q = 0$, and for every i , $m_i \leq 1$ or $\ell_i \leq 1$;
2. $q = 1$, and for every i , $m_i \leq 1$ or $\ell_i = 0$;
3. $q \geq 2$, and $\sum_i m_i \leq 1$.

Proof. For $q = 0$, the selected graph is the disjoint union of K_{m_i, ℓ_i} , so the complete-bipartite criterion applies to each branch independently.

For $q = 1$, the branches meet at the single selected vertex of R . Any simple cycle must lie in one branch: a cycle passing between branches would need to revisit that common vertex. The branch containing M_i, L_i , and the common vertex is K_{m_i, ℓ_i+1} . It is a forest exactly when $m_i \leq 1$ or $\ell_i = 0$, giving the second condition.

For $q \geq 2$, any two selected vertices in $\bigcup_i M_i$, whether or not in the same branch, form a four-cycle with two selected vertices of R . Hence at most one vertex from all the M_i together can be selected. Conversely, with at most one such vertex, the selected graph consists of a star and isolated vertices. It is therefore a forest. This proves both necessity and sufficiency in all cases. $\square$

3 The exact polynomial

Define

$$\begin{aligned} B(x) &= (1 + ax)(1 + x)^b + ((1 + x)^a - 1 - ax)(1 + bx), \\ C(x) &= (1 + ax)(1 + x)^b + (1 + x)^a - 1 - ax, \\ Q(x) &= (1 + x)^c - 1 - cx. \end{aligned} \tag{3}$$

Proposition 3.1. *For every positive a, b, c, t ,*

$$A(G(a, b, c, t), x) = B(x)^t + cx C(x)^t + Q(x)(1 + atx)(1 + x)^{bt}. \tag{4}$$

Proof. Use the three cases of Lemma 2.1. If $q = 0$, the generating polynomial for one branch is B : selecting zero or one vertex of M_i permits an arbitrary subset of L_i , giving $(1 + ax)(1 + x)^b$; selecting at least two vertices of M_i permits zero or one vertex of L_i , giving the other term. These options are disjoint. Branches are disjoint, so their contribution is B^t .

If $q = 1$, the selected common vertex contributes cx . In each branch, selecting at most one middle vertex still permits any subset of L_i , whereas selecting two or more middle vertices forces L_i to be empty. This gives C per branch and $cx C^t$ overall.

For $q \geq 2$, the common-vertex choices give Q . At most one of the at middle vertices is selected, giving $1 + atx$, and all bt vertices in the L_i can be chosen freely, giving $(1 + x)^{bt}$. Multiplication counts the choices by total cardinality. Adding the three disjoint cases proves the formula for every coefficient, including the empty subset and all zero coefficients. $\square$

For $a = b = 6$, the two branch polynomials are

$$B(x) = 1 + 12x + 66x^2 + 220x^3 + 270x^4 + 192x^5 + 74x^6 + 12x^7, \quad (5)$$

$$C(x) = 1 + 12x + 66x^2 + 130x^3 + 150x^4 + 102x^5 + 38x^6 + 6x^7. \quad (6)$$

Taking $c = 50$ and $t = 15$ in (4) gives the fully specified polynomial

$$A(G, x) = B(x)^{15} + 50xC(x)^{15} + ((1+x)^{50} - 1 - 50x)(1+90x)(1+x)^{90}. \quad (7)$$

The three summands have degrees 105, 106, and 141 respectively, and their leading coefficients are positive. Thus $A(G, x)$ has degree 141, with coefficients 142 through 230 equal to zero.

4 The coefficient obstruction

Exact expansion of (7) gives the following four coefficients; they are displayed in full to make the obstruction independently reproducible.

j	$a_j(G)$
62	9123216381063631302887063705400567361247760
63	9073337824390668447443149884464189449688960
66	8868418977379140826474515610686710306988855
67	8903285435871416312212086814708156560116100

In particular,

$$a_{62} - a_{63} = 49878556672962855443913820936377911558800 > 0,$$

$$a_{67} - a_{66} = 34866458492275485737571204021446253127245 > 0.$$

If the coefficients were unimodal, the first inequality would force the mode to be at most 62, whereas the second would force it to be at least 67. This is impossible and proves Theorem 1.1.

For an explicit coefficient-reproduction procedure, let $b_j = [x^j]B$ and $c_j = [x^j]C$ be given by (5)–(6), and define

$$u_{0,j} = v_{0,j} = \begin{cases} 1 & j = 0, \\ 0 & j \neq 0, \end{cases} \quad u_{r+1,j} = \sum_{i=0}^7 b_i u_{r,j-i}, \quad v_{r+1,j} = \sum_{i=0}^7 c_i v_{r,j-i}.$$

Terms with indices outside the polynomial support are zero. Using $\binom{90}{s} = 0$ for $s \notin [0, 90]$, formula (7) is equivalently the integer recurrence

$$a_j = u_{15,j} + 50v_{15,j-1} + \sum_{r=2}^{50} \binom{50}{r} \left(\binom{90}{j-r} + 90 \binom{90}{j-r-1} \right). \quad (8)$$

It computes the displayed coefficients using only addition and multiplication of integers. No numerical approximation or extrapolation from small graphs is involved.

5 Formal verification and reproducibility

The project `bipartite-forest-counterexample` contains the actual graph on 230 vertices. The file `WITNESS.json` also lists its two shores, all 5040 edges, and all 231 coefficients including trailing zeros. The Lean declaration `Target21.counting_bridge` proves the unconditional equivalence between the actual induced-forest subset count and the formula for the witness. Together with the exact coefficient identities, it gives `Target21.question33_counterexample` and `Target21.question33_negative`. The finite sequence on indices $0, \dots, 230$ and its zero-padded extension are both covered. The ordinary family formula in Proposition 3.1 is proved above; the formal completion claim is for the concrete graph and the resulting negative answer to the universal source question.

The 12-module build uses Lean 4.27.0 and mathlib revision `a3a10db0e9d66acbebf76c5e6a135066525ac900`, with all dependency revisions pinned in `lake-manifest.json`. After provisioning those dependencies, run `python3 scripts/verify.py -packages /path/to/packages`. Saved successful build, statement-match, and transitive axiom-audit records are identified in `verification/result.json`. The final endpoints use only `propext`, `Classical.choice`, and `Quot.sound`. They use no proof placeholders, additional mathematical axioms, or native-computation certificates. The pinned compiler and cached library proof objects are part of the trust boundary. Formal verification checks these mathematical declarations, not the prose or bibliographic priority of the manuscript.

References

- [1] C. Barton, J. I. Brown, and D. A. Pike, *Acyclic polynomials of graphs*, Australas. J. Combin. **82**(2) (2022), 146–181. https://ajc.maths.uq.edu.au/pdf/82/ajc_v82_p146.pdf.