MAXIMUM NUMBER OF SPANNING TREES OF A MULTIGRAPH WITH m EDGES: EXACT VALUES FOR $m \leq 22$

CARLES MARÍN
Independent researcher

Abstract. Let $\mathrm{T}(m)$ be the largest number of spanning trees of a connected loopless multigraph with m edges, the number of vertices being free; this is the first question of Problem 2 of Kenyon (1996). We give $\mathrm{T}(m)$ and the complete list of extremal multigraphs, up to isomorphism, for $m \leq 22$, together with the maxima $S(m)$ over simple graphs and $M(m)$ over multigraphs with a multiple edge. For $m \leq 18$ every value and every list of extremizers is backed by a certificate that a stand-alone checker, using only the Python standard library, accepts; the completeness of every enumeration in these certificates is checked by the checker's own counting, without nauty. For $19 \leq m \leq 22$ the values come from an exhaustive search with the same proved pruning rules, whose skeleton lists come from nauty. The sequence begins $1, 2, 3, 5, 8, 16, 24, 45, 81, 135, 231, 432, 720, 1280, \ldots$, ends with $\mathrm{T}(22) = 130691$, attained only by the circulant $C_{11}(1, 3)$, and for $8 \leq m \leq 22$ every extremizer is simple with degrees 3 and 4.

1. The problem

All multigraphs are finite and loopless; parallel edges are distinct edges. For a multigraph G let $\tau(G)$ be its number of spanning trees, so $\tau(G) = 0$ if G is disconnected.

Definition 1. For $m \geq 1$ let $\mathrm{T}(m) = \max \tau(G)$ over connected loopless multigraphs with m edges and any number of vertices; $S(m)$ is the same maximum over simple graphs and $M(m)$ over multigraphs with at least one edge of multiplicity ≥ 2 ($M(1)$ is undefined). An *extremizer* of $\mathrm{T}(m)$ is a multigraph attaining it.

Loops would not change $\mathrm{T}(m)$: a loop lies in no spanning tree, and T is increasing (Section 3). Since a connected multigraph with m edges has $2 \leq n \leq m+1$ vertices, $\mathrm{T}(m)$ is a finite maximum.

Kenyon [4], in his work on tilings of a rectangle by squares, asked which graph with m edges has the largest number of spanning trees, what the supremum of $\tau(G)^{1/m}$ is over all graphs, and what it is over planar graphs (Problem 2; we paraphrase it, see the quotation in [5]). In a recent preprint with a Lean formalisation, Majchrzak [5] proves $\tau(G)^2\, 8^m < 27^m$ for every multigraph with $m \geq 1$ edges and that the base $\sqrt{27/8} = 1.8371\ldots$ is optimal, which answers the second question; he notes that the first and the third are not answered there. The same base appears for regular graphs: McKay [6] showed that $27/8$ is the best possible base for $\tau(G)^{1/n}$ over 4-regular graphs on n vertices, which have $m = 2n$ edges. This note gives exact data for the first question for $m \leq 22$.

The classical extremal problem fixes both the number of vertices n and the number of edges m (Kelmans–Chelnokov [3], Shier [9], Cheng [1], Petingi–Boesch–Suffel [8]); multigraphs are treated in [2, Ch. 3]. Here n is free, so $\mathrm{T}(m)$ is the maximum over n of the multigraph values.

Date: November 2026.

m	$T(m)$	#ext.	n	$S(m)$	$M(m)$	status
1	1	1	2	1	–	C
2	**2**	1	2	1	2	C
3	**3**	2	2, 3	3	3	C
4	**5**	1	3	4	5	C
5	**8**	2	3, 4	8	8	C
6	16	1	4	16	13	C
7	**24**	2	4, 5	24	24	C
8	45	1	5	45	40	C
9	81	1	6	81	69	C
10	135	1	6	135	126	C
11	231	1	7	231	210	C
12	432	1	7	432	366	C
13	720	2	7, 8	720	648	C
14	1280	1	8	1280	1152	C
15	2304	1	8	2304	1992	C
16	4096	1	8	4096	3495	C
17	7168	1	9	7168	6272	C
18	12584	1	10	12584	11264	C
19	22720	1	10	22720	19712	**E**
20	40960	1	10	40960	34944	**E**
21	72657	1	11	72657	62296	**E**
22	130691	1	11	130691	112640	**E**

Table 1. $\mathrm{T}(m)$, the number of extremizers up to isomorphism, their numbers of vertices, the simple maximum $S(m)$ and the non-simple maximum $M(m)$. Status: C = certified (Theorem 2); E = exhaustive search only (Result 3). Colours: certified rows, search-only rows (below the rule; the two shades alternate with m); $\mathrm{T}(m)$ in bold: some extremizer has a multiple edge.

2. Results

Theorem 2 (computer-assisted). *For $1 \le m \le 18$ the values $\mathrm{T}(m)$, $S(m)$ and, for $m \ge 2$, $M(m)$ in Table 1 are exact, and the extremizers of $\mathrm{T}(m)$ are exactly the multigraphs of Table 2, up to isomorphism. The extremizers of $S(m)$ and $M(m)$ are listed in the data files.*

The proof is the acceptance of 53 certificates (three classes for each m, none for $M(1)$) by the checker of Section 5; its trust base is the checker's code and the rules of Section 3, each proved below.

Computational result 3. For $19 \le m \le 22$ the exhaustive search of Section 4 gives the values in Table 1 and the unique extremizers of Table 2. These rows rest on the same proved rules, on the completeness of the lists produced by nauty [7], and on the correctness of the search code; there is no certificate.

Remark 4 (read off the tables, checked by script).
(1) A multiple edge is needed to attain $\mathrm{T}(m)$ only for $m = 2$ and $m = 4$; for $m = 3, 5, 7$ a simple and a non-simple extremizer tie. Hence $S(m) = \mathrm{T}(m)$ for every $m \le 22$ except $m = 2, 4$.
(2) For $8 \le m \le 22$ every extremizer is simple with all degrees 3 or 4, and it is unique except at $m = 13$ (two). Identified up to isomorphism: K_4 ($m = 6$), $K_{3,3}$ (9), $K_{3,4}$ (12), $K_{4,4}$

m	τ	n	degrees	graph6 (skeleton)	multiplicities ≥ 2
1	1	2	1^2	`A_`	–
2	2	2	2^2	`A_`	**0–1×2**
3	3	3	2^3	`Bw`	–
3	3	2	3^2	`A_`	**0–1×3**
4	5	3	$3^2\,2$	`Bw`	**0–1×2**
5	8	4	$3^2\,2^2$	`C^`	–
5	8	3	$4\,3^2$	`Bw`	**0–1×2, 0–2×2**
6	16	4	3^4	`C~`	–
7	24	5	$3^4\,2$	`D]w`	–
7	24	4	$4^2\,3^2$	`C~`	**0–1×2**
8	45	5	$4\,3^4$	`D]{`	–
9	81	6	3^6	`EFz_`	–
10	135	6	$4^2\,3^4$	`EFzo`	–
11	231	7	$4\,3^6$	`FCxv_`	–
12	432	7	$4^3\,3^4$	`F?~v_`	–
13	720	7	$4^5\,3^2$	`FFzf_`	–
13	720	8	$4^2\,3^6$	`G?o~f_`	–
14	1280	8	$4^4\,3^4$	`G?zVf_`	–
15	2304	8	$4^6\,3^2$	`G?zvf_`	–
16	4096	8	4^8	`G?~vf_`	–
17	7168	9	$4^7\,3^2$	`H?o~fbo`	–
18	12584	10	$4^6\,3^4$	`I?qbF_]{?`	–
19	22720	10	$4^8\,3^2$	`I?bF`xw{?`	–
20	40960	10	4^{10}	`I?rFf_{N?`	–
21	72657	11	$4^9\,3^2$	`J?`F?|wlF_?`	–
22	130691	11	4^{11}	`J?bFF`wN?{?`	–

Table 2. All extremizers of $\mathrm{T}(m)$. The graph6 string [7] encodes the underlying simple graph (skeleton) on vertices $0,\dots,n-1$; the last column lists the edges u–v of multiplicity $k \geq 2$ in that labelling. Degrees count multiplicities; $4^2\,3^4$ means two vertices of degree 4 and four of degree 3. Colours as in Table 1: certified, search only (below the rule), rows of the same m share a shade; multiplicities in bold: the extremizer is not simple.

minus an edge (15), $K_{4,4} \cong C_8(1,3)$ (16), the circulants $C_{10}(1,4)$ (20) and $C_{11}(1,3)$ (22). The extremizer for $m = 14$ is bipartite; those for $m = 13, 17, 18, 19, 21$ are not.

(3) T is strictly increasing, $\mathrm{T}(m)^{1/m}$ decreases only from $m = 6$ to $m = 7$ ($1.5874 \to 1.5746$), and $\mathrm{T}(22)^{1/22} = 1.7083$, below $\sqrt{27/8} = 1.8371$.

(4) $M(m)/S(m) \in [0.81, 0.94]$ for $6 \leq m \leq 22$, $m \neq 7$ (at $m = 7$ they tie at 24), and every extremizer of $M(m)$ with $4 \leq m \leq 22$ has maximal multiplicity exactly 2; this follows from the table by Corollary 12.

Majchrzak [5, §9] reports, from an exhaustive computation over multigraphs with at most 14 edges, that the maximum of $\tau^{1/m}$ is 1.667031, at $\tau = 1280$, $m = 14$; this agrees with $\mathrm{T}(14) = 1280$ and with Table 1, where $\mathrm{T}(m)^{1/m}$ is largest at $m = 14$ among $m \leq 14$.

The sequence $(\mathrm{T}(m))$ was not in the OEIS when this note was written (searched on 25 November 2026 for the whole initial segment and for the segments $45,\dots,432$ and $16, 45,\dots,432$).

3. Rules, with proofs

Notation: $R_{uv}(G)$ is the effective resistance between u and v with unit resistance on every copy. We use three classical facts (Kirchhoff): (K1) the matrix-tree theorem, valid for multigraphs; (K2)

the fraction of spanning trees using a given copy $f = uv$ is $R_{uv}(G)$, so $\tau(G-f) = \tau(G)(1-R_{uv})$ and $\tau(G+uv) = \tau(G)(1+R_{uv})$; (K3) Rayleigh monotonicity: adding edges does not increase effective resistances. The full statements, with the domain of each rule in each class, are in `PRUNING_RULES.md` of the data.

Lemma 5 (monotonicity). $\mathrm{T}(m+1) > \mathrm{T}(m)$.

Proof. Add a parallel copy of an edge uv of an extremizer; by (K2) τ is multiplied by $1+R_{uv} > 1$. □

Lemma 6 (cut vertex). *If G has a cut vertex, $G = G_1 \cup_v G_2$ with $m_1 + m_2 = m$, $m_i \geq 1$, and $\tau(G) = \tau(G_1)\tau(G_2) \leq \mathrm{T}(m_1)\mathrm{T}(m_2)$. Moreover, for $m \geq 2$ (multigraphs), $m \geq 3$ (simple graphs) and in the class of non-simple multigraphs such a G is never extremal: there is G' in the same class with m edges and $\tau(G') > \tau(G)$.*

Proof. A spanning tree of G is the union of one of G_1 and one of G_2. For the second claim: if G has a bridge b, contract it (τ unchanged, $m-1$ edges, no loop) and either duplicate an edge (multigraphs) or subdivide an edge of a cycle, which adds $\tau(G/b-f) \geq 1$ trees (simple graphs; if there is no cycle, G is a tree and C_m is better). If G has no bridge and v is a cut vertex, let u be a neighbour of v on one side, e a copy of uv, $w \neq v$ a vertex on the other side and $H = G-e$, which is connected. Every u–w path in H passes through v, so $R_{uw}(H) = R_{uv}(H)+R_{vw}(H) > R_{uv}(H)$, and by (K2) $\tau(H+uw) > \tau(H+uv) = \tau(G)$; $H+uw$ is simple if G is. Non-simple class: contracting a bridge keeps every multiple edge and the added copy is one more; without bridges, every edge lies inside one side of v, and if G has a single pair of multiplicity ≥ 2, of multiplicity exactly 2, we take u on the side that does not contain that pair, so that e is a simple edge and $H+uw$ keeps the pair (otherwise removing one copy leaves a multiple edge anyway). □

Hence the search runs over $n = 2$ (a bundle of m parallel edges, $\tau = m$) and over biconnected multigraphs with $n \geq 3$, in each of the three classes; a multigraph is biconnected if and only if its skeleton is. The certificates of $M(m)$ do not use the domination in the non-simple class: they bound such G by the product, $\tau(G) \leq \max_{2\leq a<m} M(a)\mathrm{T}(m-a)$, which is below $M(m)$ for every $3 \leq m \leq 22$ (checked on the table).

Lemma 7 (degree two). *If G is biconnected with $n \geq 3$ and has a vertex of degree 2, then $\tau(G) \leq \mathrm{T}(m-1) + \mathrm{T}(m-2)$, with $\mathrm{T}(0) = 1$.*

Proof. The vertex x has two distinct neighbours u, w (otherwise its neighbour is a cut vertex), and $H = G - x$ is connected with $m-2$ edges. Either x is a leaf ($2\tau(H)$ trees) or it uses both edges ($\tau(H/uw)$ trees), so $\tau(G) = \tau(H) + \tau(H+uw)$ by deletion–contraction. □

The degree of x counts multiplicities. Since $\mathrm{T}(m-1) + \mathrm{T}(m-2) < \mathrm{T}(m)$ for $m = 6$ and $8 \leq m \leq 22$ (checked on the computed values, never assumed), the extremizers of $\mathrm{T}(m)$ there have minimum degree ≥ 3; for the other m the search runs without this rule. The search applies it only to simple graphs (skeletons with $e = m$ edges, where the degree in the skeleton is the degree); skeletons of non-simple multigraphs are enumerated with no degree restriction, since a vertex of degree 2 in the skeleton may have degree ≥ 3 in the multigraph.

Lemma 8 (counting bounds). *For connected G on n vertices, $\tau(G) \leq \binom{m}{n-1}$ and $\tau(G) \leq \frac{1}{n}\left(\frac{2m}{n-1}\right)^{n-1}$.*

Proof. A tree is a set of $n-1$ edges; and $\tau = \frac{1}{n}\prod_{i\geq 2}\lambda_i$ with $\sum \lambda_i = 2m$ by (K1), then AM–GM. □

Lemma 9 (bounds on a fixed skeleton). *Let S be a simple graph with e edges, G a multigraph on it with multiplicities $m_f = 1 + x_f$ and $r = m - e = \sum x_f$, and N_f the number of spanning trees of S through f. Then*

$$\tau(G) \leq \tau(S)\, P(n-1, r) \quad \text{and} \quad \tau(G) \leq \tau(S) \prod_f \Big(1 + x_f \frac{N_f}{\tau(S)}\Big),$$

where $P(k,r)$ is the largest product of k integers ≥ 1 with sum $k+r$: writing $k+r = qk+s$ with $0 \leq s < k$, $P(k,r) = q^{k-s}(q+1)^s$.

Proof. First: $\tau(G) = \sum_T \prod_{f \in T} m_f$ over trees T of S, and each product has $n-1$ factors whose excesses add up to at most r. Second: add the extra copies of f one by one. By (K2) the j-th one multiplies τ by $1+R$, where R is the current resistance of f's endpoints; the network contains S (conductance $\tau(S)/N_f$ between them) in parallel with $j-1$ extra copies, so by (K3) $R \leq 1/(\tau(S)/N_f + j - 1)$, and the product telescopes. The maximum of the right-hand side over $\sum x_f = r$ is attained by the greedy choice (each copy goes to an edge maximising $N_f/(\tau(S)+x_f N_f)$, the marginal gain), since each factor is log-concave in x_f; with a cap c on multiplicities, edges with $x_f = c-1$ are skipped. □

Lemma 10 (multiplicity). *If $m \geq 4$ and G is biconnected with $n \geq 3$ and an edge uv of multiplicity $k \geq 3$, then removing one copy of uv and subdividing another gives G' with m edges and $\tau(G') > \tau(G)$. So the extremizers of $\mathrm{T}(m)$, $m \geq 4$, have multiplicities ≤ 2 (the bundle with $n = 2$ has $\tau = m$, while the triangle with multiplicities $m-2, 1, 1$ has $\tau = 2m-3 > m$). Among non-simple multigraphs the same holds for $k \geq 4$.*

Proof. With $H = G$ minus one copy and $R = R_{uv}(H)$, $\tau(G) = \tau(H)(1+R)$ and $\tau(G') = \tau(H)(2-R)$, so G' is better if and only if $R < 1/2$. In H the $k-1 \geq 2$ copies of uv are in parallel with the rest, which has positive conductance between u and v when $k = 3$ (else u or v is a cut vertex); hence $R < 1/2$. For $k \geq 4$, $R \leq 1/3$ and G' keeps a multiple edge. □

Lemma 11 (a multiple edge). *If G has an edge uv of multiplicity $k \geq 2$, then $(k-1)\,\tau(G) \leq k\,\mathrm{T}(m-1)$, strictly if G is biconnected with $n \geq 3$.*

Proof. Let H be G minus one copy of uv. In H the $k-1$ remaining copies join u and v, so by (K3) $R_{uv}(H) \leq 1/(k-1)$, and by (K2) $\tau(G) = \tau(H)(1+R_{uv}(H)) \leq \frac{k}{k-1}\mathrm{T}(m-1)$. If G is biconnected with $n \geq 3$, the rest of H has positive conductance between u and v and the inequality for R is strict. □

Corollary 12. *For $4 \leq m \leq 22$, every extremizer of $M(m)$ has multiplicities ≤ 2.*

Proof. The bundle has $\tau = m < 2m-3 \leq M(m)$; a non-simple G with a cut vertex has $\tau(G) \leq \max_{2 \leq a < m} M(a)\mathrm{T}(m-a) < M(m)$ (Lemma 6 and the table); a biconnected one with an edge of multiplicity $k \geq 3$ has $\tau(G) < \frac{3}{2}\mathrm{T}(m-1) \leq M(m)$ by Lemma 11 and the table (equality in the last step at $m = 7$ and $m = 13$). For $m \leq 18$ the values used are certified; for $19 \leq m \leq 22$ they are those of Result 3. □

Remark 13 (what is not proved). For the class of non-simple multigraphs the step $k = 3$ of Lemma 10 fails when uv is the only multiple edge (G' is simple, outside the class); so the certificates of $M(m)$ use the cap 3 of Lemma 10, and "the best non-simple multigraph has multiplicities ≤ 2" is proved here only for $4 \leq m \leq 22$, through the table (Corollary 12), not for all m. Two further bounds that were considered (a reduction to a degree-≥ 3 core and a degree-product bound) were not proved and are not used.

4. Method

Search. For each m the search has two parts, each with its own threshold. The non-simple part (skeletons with $n \leq e < m$ edges, no degree restriction) uses a non-simple lower bound L_M: the bundle, or an extremizer of $m-1$ plus a parallel copy of one of its edges. The simple part ($e = m$) uses $\max(L_0, \mathrm{T}(m-1) + \mathrm{T}(m-2) + 1)$, where $L_0 \geq L_M$ also uses the extremizer of $m-1$ plus any edge, products at a cut vertex and $K_{a,b}$ with $ab = m$, and it is enumerated by `geng -C -d3` with $n \leq 2m/3$ (Lemma 7); if nothing reaches that threshold, the simple part is rerun without

the degree rule (this happened only for $m = 3, 4, 5, 7$). For $19 \le m \le 22$ the extremizer is simple, so $S(m) = \mathrm{T}(m)$. For each admissible n the biconnected skeletons are enumerated by `geng -C`; n and skeletons are discarded by Lemmas 8 and 9, comparing integers exactly, and the surviving skeletons are expanded into all multiplicity vectors up to isomorphism (`multig`). τ is computed by elimination modulo $p = 2^{31} - 1$; since $\tau \le \binom{22}{11} < p$, the residue is τ. Every extremizer is recomputed by an independent exact determinant. A guard compares, batch by batch, the number of graphs read with nauty's own count (`-u`) and repeats a batch that does not match; it was added after truncated output streams were observed, and all rows of Table 1 come from runs with the guard. A floating-point pre-filter on a GPU exists in the code but the values do not depend on it (it was off in the runs of Table 1). The run for $m = 22$ took 2639 s of wall-clock time on a desktop computer (AMD Ryzen 9 7900, 12 cores, 32 GB; nauty 2.8.8 in Docker, Python 3.14 with NumPy). Running the checker needs only Python; reproducing the search needs nauty as well.

5. Certificates and the independent checker

A certificate is a JSON Lines tree whose root is the class (all, simple or non-simple multigraphs with m edges). Internal nodes split the class (cut vertex or not; $n = 2, \dots, m + 1$; a vertex of degree 2 or not; the number e of skeleton edges), exclude a subclass by a bound of Section 3 or by domination (Lemmas 6, 10), or list skeletons; each listed skeleton is bounded or fully expanded. In mode `ALL_EXTREMIZERS` every exclusion must be *strictly* below the claimed value and every leaf attaining it must be isomorphic to a listed witness, so acceptance proves the value and the completeness of the list of extremizers. Every witness must have m edges, no loop, be connected and lie in the class. An exclusion by domination is justified by the finiteness of the class: a maximiser exists and is not strictly dominated, so every extremizer lies in the subclasses that are not excluded; the improved graph must lie in the root class, and the checker accepts each domination rule only in the classes where it is proved (Lemma 6 in the classes of all and of simple multigraphs; Lemma 10 with $k \ge 3$ for all multigraphs, $k \ge 4$ for non-simple ones). Each listed labelled skeleton is either bounded or expanded into all its labelled multiplicity vectors, with no symmetry reduction, so the multigraphs are covered once the skeletons are.

The checker `comprobar.py` imports nothing from the search. It recomputes every bound in integers, the coverage of every split, τ of every witness and leaf (Bareiss), and the completeness of every list of skeletons, by *method A*: each listed graph is in the class, their canonical forms (own code) are pairwise distinct, and $\sum n!/|\mathrm{Aut}(S)|$, with $|\mathrm{Aut}(S)|$ also from its own code, equals the number of labelled connected or biconnected graphs with n vertices and e edges. These numbers come from generating functions, exponential in the vertices and ordinary in the edges: $C = \log \sum_n (1 + y)^{\binom{n}{2}} x^n/n!$ for connected graphs and the block decomposition $C^\bullet = x \exp(B'(C^\bullet))$, $C^\bullet = xC'$, for biconnected ones, solved by Lagrange inversion over $\mathbb{Q}[y]$. Their totals reproduce OEIS A001187 and A013922 for $n \le 10$, and a labelled enumeration agrees for every (n, e) with $n \le 6$. For $n \le 6$ the lists are also checked by *method C*, a labelled enumeration. Bounds that use T, S, M of smaller m take them only from certificates already accepted. No node of the certificates for $m \le 18$ relies on nauty.

All 53 certificates were accepted when produced ($m = 18$: 659, 462 and 235 s for the three classes) and again from the copies in the published package (log in the data). Their extremizers coincide, up to isomorphism, with those of the search for every $m \le 18$. The checker rejects deliberately corrupted certificates (a branch removed, a skeleton duplicated in place of another, a multiplicity or a determinant changed, "$\le$" written for "$<$", an inadmissible witness, a rule outside its domain, a dependency not yet checked); these tests belong to our tool and are not in the package.

6. Controls

(1) *Brute force, $m \le 12$.* All 41 207 connected simple skeletons with $n \le 13$ and at most 12 edges, with all multiplicities and no pruning: the same $\mathrm{T}(m)$ and the same extremizers, compared up to isomorphism with multiplicities.
(2) *Ablation, $m \le 14$.* With the pruning rules switched off (all, or in groups), T, M and the lists of extremizers output by the search (skeleton graph6 with multiplicities) are unchanged.
(3) *Negative control.* A false rule (discarding $n \ge m/2$ for $n \ge 3$; the bundle $n = 2$ is kept) changes $\mathrm{T}(4)$ from 5 to 4 and trips an assertion at $m = 5$.
(4) *Skeleton bound.* The second bound of Lemma 9 was compared with all biconnected non-simple multigraphs with $n \le 7$ for $m = 6, 8, 10$: never violated, and attained on 3, 11 and 46 skeletons.
(5) *Lemmas 6 and 10.* Checked on every multigraph with $m \le 12$, $n \le 7$: no counterexample.
(6) *Recomputation.* Every extremizer of T and of M for $m \le 22$ was recomputed in Sage (`spanning_trees_count` and an integer determinant): 47 of 47 equal (26 extremizers of T, 21 of M); and every entry of the three lists of T, S, M ($26 + 23 + 21 = 70$, a graph counted once per list) once more by a separate exact rational determinant, with a negative control (removing any copy must change τ).
(7) *Lemmas 6 and 11 by brute force.* On all labelled connected multigraphs with $n \le 6$ and $m \le 8$: the maxima of the three classes equal the table; $(k-1)\tau \le k\mathrm{T}(m-1)$ holds for all 384 650 multiple edges, strictly in the biconnected case (while $\tau \le \mathrm{T}(m-1)$ fails 10 107 times); and the construction of Lemma 6 in the non-simple class gives a non-simple G' with larger τ for all 229 089 non-simple multigraphs with a cut vertex.
(8) *One-edge relocation.* For no extremizer of T (26) or of M (21) is there an edge copy $f = uv$ and a pair ab, admissible in the class, with $R_{ab}(G - f) > R_{uv}(G - f)$, exact rational arithmetic; such a pair would give $\tau(G - f + ab) > \tau(G)$ by (K2). Control: K_4 with a tripled edge admits 5 such moves.
(9) *Determinant routine on losers.* The modular routine of the search agrees with an independent Bareiss determinant on 3300 random connected multigraphs with $3 \le n \le 13$, $m \le 22$, almost all non-extremal.

7. Data

The Zenodo record that holds this PDF contains: the values (`T_m_values.csv`, `.json`); all extremizers of T, S and M for $m \le 22$ with edge lists and multiplicities (`extremal_graphs.json`, `.txt`); the 53 certificates with SHA-256 sums; the checker, the re-check of the graph file and the search code (MIT licence), with the rules and their proofs (`PRUNING_RULES.md`) and the certificate format; the logs of every check above; and a README with the commands that reproduce the checks with the Python standard library only.

Acknowledgement

I worked with an AI assistant.

References

[1] C.-S. Cheng, *Maximizing the total number of spanning trees in a graph: two related problems in graph theory and optimum design theory*, J. Combin. Theory Ser. B **31** (1981), 240–248.

[2] D. J. Gross, J. T. Saccoman and C. L. Suffel, *Spanning Tree Results for Graphs and Multigraphs: A Matrix-Theoretic Approach*, World Scientific, 2014, doi:10.1142/8963.
[3] A. K. Kelmans and V. M. Chelnokov, *A certain polynomial of a graph and graphs with an extremal number of trees*, J. Combin. Theory Ser. B **16** (1974), 197–214.
[4] R. Kenyon, *Tiling a rectangle with the fewest squares*, J. Combin. Theory Ser. A **76** (1996), 272–291, doi:10.1006/jcta.1996.0104.
[5] F. Majchrzak, *Spanning trees per edge: a sharp bound and a question of Kenyon, with a Lean 4 formalisation of the bound and of its sharpness*, preprint, version 2.0.0, Zenodo (2026), doi:10.5281/zenodo.24523144.
[6] B. D. McKay, *Spanning trees in regular graphs*, European J. Combin. **4** (1983), 149–160.
[7] B. D. McKay and A. Piperno, *Practical graph isomorphism, II*, J. Symbolic Comput. **60** (2014), 94–112.
[8] L. Petingi, F. Boesch and C. Suffel, *On the characterization of graphs with maximum number of spanning trees*, Discrete Math. **179** (1998), 155–166.
[9] D. R. Shier, *Maximizing the number of spanning trees in a graph with n nodes and m edges*, J. Res. Nat. Bur. Standards Sect. B **78B** (1974), 193–196.