AI-Assisted Discovery of Convex Relaxations via Dual Agents

Sungyoon Kim [1] **Mert Pilanci** [1]

Abstract

Recent work shows that LLM agents can improve sharp-constant inequalities by searching for extremal constructions, which yield *upper* bounds. We address the complementary side: a *lower* bound holds for every admissible function and follows from a convex relaxation of the nonconvex problem, with tighter relaxations giving stronger bounds. We instantiate the autoresearch paradigm to discover such relaxations: a coding agent proposes valid tightening constraints, a theory agent verifies each one and searches for counterexamples, and every reported bound is certified by an explicit dual-feasible point checked in rigorous interval arithmetic. On two optimization constants studied by Georgiev et al. (2025) - the first autocorrelation inequality ($C_{6.2}$) and the Erdős minimum-overlap constant ($C_{6.5}$) - we improve the certified lower bounds from 1.28 to 1.2937 and from 0.379005 to 0.37912, respectively.

1. Introduction

Large language model (LLM) agents have begun to contribute to mathematical research in ways that were implausible only a short time ago. On competition mathematics, verification-and-refinement pipelines now reach gold-medal performance at the International Mathematical Olympiad (Huang & Yang, 2025), and frontier models have moved beyond contests toward genuine research: GPT-5 has produced new, human-verified results across several areas of mathematics, including settling previously open Erdős problems (Bubeck et al., 2025). Larger-scale efforts point the same way - Tsoukalas et al. (2026) report solving 9 of 353 open Erdős problems with a frontier agent under Lean - based formal verification, while contamination - free benchmarks built from unpublished problems, most prominently *First Proof* (Abouzaid et al., 2026), show models solving research-level mathematics rather than retrieving memorized patterns. Taken together, these episodes indicate that agentic AI is no longer only a tool for routine assistance, but a participant in mathematical discovery.

Part of what makes these systems interesting is that their strengths differ from those of human mathematicians. An agent does not tire, can pursue a search through the night, and can ingest a body of literature far larger than any individual could hold in mind at once. These differences suggest that AI can contribute to mathematics in a different way from humans - not by imitating human workflows, but by playing to strengths humans lack. A clear example arises in functional analysis, where evolutionary and LLM-guided search has been used to improve bounds on autocorrelation inequalities and the related Erdős minimum overlap problem (Georgiev et al., 2025; Yuksekgonul et al., 2026; Together AI, 2026; Novikov et al., 2025). There, the goal is to *construct* an extremal function: an explicit example that witnesses a bound. Because any single admissible function yields a valid bound, the task reduces to optimizing a score over a vast, highly nonconvex space of candidate functions-precisely the regime in which an agent can act as a tireless and creative *discoverer*, finding constructions that human search would reach only slowly, if at all.

We pursue a different way of leveraging these tools. Our work is motivated by two observations: AI-assisted coding has matured to the point where agents write competent, nontrivial programs (Jimenez et al., 2024), and agents are increasingly able to solve hard algorithmic problems (Jain et al., 2025; Zheng et al., 2026). We use these capabilities to improve the *other* side of these inequalities-the *lower* bounds. Unlike an upper bound, which follows from exhibiting one function, a lower bound must hold against *every* admissible f; one obtains such a bound by solving a convex relaxation of the original nonconvex problem, and a tighter relaxation yields a stronger bound (we make this precise in Section 2). Finding a good relaxation is itself an iterative, open-ended task: an agent repeatedly edits a program, benchmarks it, and tries to beat its own best result. This is a natural fit for the *autoresearch* paradigm (Karpathy, 2026), as seen in self-improving coding agents (Robeyns et al., 2025) and LLM-driven program search (Romera-Paredes et al., 2024). We instantiate this paradigm for the discovery of convex relaxations that certify lower bounds. A diffi-

[1]Department of Electrical Engineering, Stanford University, CA, USA. Correspondence to: Sungyoon Kim <sykim777@stanford.edu>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

culty particular to our setting is that an agent may simply be *wrong* - it may propose a relaxation whose constraints are not actually valid for all f, or commit prematurely to a single line of attack. The correctness of the convex problem is the central risk: an invalid constraint yields a bound that is not a bound at all. We therefore pair the coding agent with a *theory agent* whose role is to read the proposed relaxation, rigorously establish that each constraint is valid, and verify the accompanying argument before any improvement is accepted.

Using this dual-agent loop, we improve the certified lower bounds on two autocorrelation inequalities studied by Georgiev et al. (2025) - $C_{6.2}$, arising in additive combinatorics and $C_{6.5}$, controlling the asymptotics of the Erdős minimum overlap problem (see Section 2 for definitions). For the first, we raise the lower bound from 1.28 to 1.2937; for the second, we improve the best known lower bound of 0.379005 to 0.37912. To our knowledge, this is the first use of an autoresearch-style agentic loop to improve *certified lower bounds* on these inequalities: rather than searching for a better extremal example, the agents search for a better *program* - a convex relaxation-whose validity is established by mathematical proof and whose bound is certified by duality.

2. Preliminaries

2.1. Autocorrelation Inequalities

For absolutely integrable $f, g : \mathbb{R} \to \mathbb{R}$, the convolution is $(f * g)(t) = \int_{\mathbb{R}} f(x)g(t-x)\,dx$; when g equals f or a reflection of f, we refer to such expressions as *autocorrelations* Georgiev et al. (2025). Sharp constants in functional inequalities involving autocorrelations have been studied extensively; we refer to Georgiev et al. (2025) for a survey and for the problem numbering we adopt below, and Davis et al. (2026) for the history of bounds for these problems. We consider two such constants.

The first autocorrelation inequality (Problem 6.2 of (Georgiev et al., 2025)). Let $C_{6.2}$ denote the largest constant for which

$$\max_{-1/2\le t\le 1/2} \int_{\mathbb{R}} f(t-x)f(x)\,dx \;\ge\; C_{6.2}\left(\int_{-1/4}^{1/4} f(x)\,dx\right)^2 \tag{1}$$

holds for all non-negative $f : \mathbb{R} \to \mathbb{R}$. This constant arises in additive combinatorics, in connection with the size of Sidon sets (Cloninger & Steinerberger, 2017; Martin & O'Bryant, 2009). The previously known bounds are

$$1.28 \;\le\; C_{6.2} \;\le\; 1.502862, \tag{2}$$

where the lower bound is obtained by analytic/duality arguments (Cloninger & Steinerberger, 2017) and the upper bound by an explicit counterexample (Together AI, 2026).

The Erdős minimum overlap problem (Problem 6.5). Let $C_{6.5}$ denote the largest constant for which

$$\sup_{x\in[-2,2]} \int_{-1}^{1} f(t)\,g(x+t)\,dt \;\ge\; C_{6.5} \tag{3}$$

for all non-negative $f, g : [-1,1] \to [0,1]$ with $f + g = 1$ on $[-1,1]$ and $\int_{-1}^{1} f = 1$. Outside this interval $[-1,1]$, $f(x) = g(x) = 0$. This constant governs the asymptotics of the minimum overlap problem of Erdős (Georgiev et al., 2025). The known bounds are

$$0.379005 \;\le\; C_{6.5} \;\le\; 0.38087131058, \tag{4}$$

where the lower bound was obtained via a family of convex optimization problems (White, 2022) and the upper bound by a step-function construction (Together AI, 2026).

In both problems the quantity of interest is an infimum over an infinite-dimensional set of admissible functions, of a score that is nonconvex in f. A lower bound on the constant is therefore a statement that *no* admissible function drives the score below a given value-a universally quantified claim, in contrast to the existential claim established by a single construction.

2.2. Convex Relaxation and Certified Bounds

We briefly fix the vocabulary of relaxation and duality used throughout. Consider a nonconvex problem

$$p^\star \;=\; \inf_{f\in\mathcal{F}} \; J(f), \tag{5}$$

where $\mathcal{F}$ is the (infinite-dimensional) set of admissible functions and J is the score. A *relaxation* replaces (5) by a convex problem

$$r^\star \;=\; \inf_{z\in\mathcal{C}} \; \tilde{J}(z), \tag{6}$$

constructed so that every admissible f induces a feasible $z \in \mathcal{C}$ with $\tilde{J}(z) \le J(f)$. Any such relaxation satisfies $r^\star \le p^\star$, so $r^\star$ is a valid lower bound on the constant of interest. The relaxation is *valid* precisely when this inclusion holds-that is, when every constraint defining $\mathcal{C}$ is satisfied by the encoding of every admissible f. A *tighter* relaxation is one with a larger $r^\star$ (equivalently, a smaller feasible set that still contains all admissible encodings); tightening the relaxation improves the bound, up to the limit $r^\star = p^\star$.

Certified bounds via duality. Solving (6) numerically returns a value, but for a *certified* bound we appeal to weak duality. For any dual-feasible point λ, the dual objective $d(\lambda)$ satisfies

$$d(\lambda) \;\le\; r^\star \;\le\; p^\star \tag{7}$$

Exhibiting a single dual-feasible λ with $d(\lambda) = \beta$ therefore proves $p^\star \ge \beta$, and this certificate is checkable independently of the solver that produced it: one need only verify

that λ satisfies the dual constraints and evaluate $d(\lambda)$. This is the sense in which our bounds are certified, and it is what distinguishes a proven lower bound from a numerically reported optimal value. We return to how we find a rigorous dual certificate and a lower bound in Section 3.

3. Method

3.1. Problem Formulation

Each constant in Section 2.1 is the value of a min-max problem over an infinite-dimensional class of admissible functions. We describe the convex problem formulation through the minimum overlap problem, following White (2022); the autocorrelation inequality $C_{6.2}$ of (1) is treated analogously.

For the minimum overlap problem, the constant is the infimum over admissible f of a maximum over translates,

$$C_{6.5} \;=\; \inf_{f} \max_{t\in[-2,2]} \int_{-1}^{1} f(x)g(x+t)\,dx, \qquad (8)$$

taken over measurable $f : [-1,1] \to [0,1]$ and normalization $\int_{-1}^{1} f = 1$. White's reduction turns this nonconvex min–max into a convex program of a finite list of variables. Write $M(t) = \int_{-1}^{1} f(x)g(x+t)\,dx$ for the overlap at t, where $g(x) = 1 - f(x)$ for $x \in [-1,1]$ and 0 otherwise, and partition $[0,2]$ into grid cells $I_1, \ldots, I_N$ of width L. White's reduction then introduces the following variables: a scalar Ω bounding $\max_k M(k)$; the cell integrals of the overlap on the two shift directions,

$$w_j \;=\; \frac{1}{L}\int_{I_j} M(k)\,dk, \quad v_j \;=\; \frac{1}{L}\int_{I_j} M(-k)\,dk, \quad (9)$$

the cosine and sine Fourier coefficients c_k, d_k of f and a_m, b_m of M; and truncation slacks ε_m, δ_m that absorb the Fourier modes beyond the truncation order so that finitely many coefficients suffice.

Admissibility of f imposes constraints on these variables, and the essential point is that each one must be a genuine consequence of $0 \le f \le 1$ and $\int_{-1}^{1} f = 1$, so that no admissible f is excluded. Two are immediate. The property $\int_{-2}^{2} M(x)dx = 1$ gives the linear identity $L\sum_j (w_j + v_j) = 1$, and because Ω bounds the overlap at every translate, each cell integral obeys $w_j, v_j \le \Omega$. The remaining conditions - mean and second-moment bounds, the Fourier identities linking a_m, b_m to c_k, d_k, and the bounds on the truncation slack ϵ_m, δ_m, are derived in the same spirit; we refer the reader to White (2022) or Algorithm 1 for the full list. We retain only those conditions that are convex in the variables to make the problem convex. **Let $\mathcal{K}$ be the feasible set they cut out.**

The lower bound now follows from a single observation. Fix any admissible f and read off its variables: take $\Omega = \max_{k\in[-2,2]} M(k)$, let w_j, v_j be the cell integrals in (9), and let the remaining entries be the Fourier data of f. By construction this assignment satisfies every constraint of $\mathcal{K}$, so it is a feasible point of the convex program, and its objective value is exactly $\max_{k\in[-2,2]} M(k)$, the overlap score of f. The convex program minimizes that same objective over all of $\mathcal{K}$, returning a value $\Omega^\star$. Since every admissible f supplies a feasible point whose objective equals its own score, the minimum can only be smaller: $\Omega^\star \le \max_{k\in[-2,2]} M(k)$ for every admissible f, and therefore $\Omega^\star \le C_{6.5}$.

3.2. Successive Convex Relaxation

The quality of the convex problem is governed entirely by the constraint set $\mathcal{K}$. Tightening $\mathcal{K}$-adding a valid condition, or strengthening an existing one-shrinks the feasible set toward the true set of admissible coefficient vectors, and the minimum can only increase:

$$\mathcal{K}' \subseteq \mathcal{K} \quad \Longrightarrow \quad \min_{c\in\mathcal{K}'} \Phi(c) \;\ge\; \min_{c\in\mathcal{K}} \Phi(c). \qquad (10)$$

So long as every condition is valid, $\mathcal{K}'$ still contains all admissible encodings, and the larger value remains a lower bound on Ω. Iterating yields a monotone sequence of valid lower bounds increasing toward Ω.

We stress that "tightening" here means *adding valid constraints or improving existing ones*, not naively increasing resolution. One could enlarge the model with more Fourier modes or finer step functions; this refines the discretization but does not, in itself, tighten the relaxation in the sense of (10). The improvements we seek are mathematical: properties of admissible coefficient sequences that a coarser formulation failed to impose. The task at each iteration is therefore to *identify a valid constraint that previous relaxations overlooked*, or to sharpen one imposed too weakly. The concrete constraints discovered for each problem, and their effect on the bound are reported in Section 4.

3.3. The Dual-Agent Loop

We search for such constraints with a human-in-the-loop framework: a loop of two agents drives the discovery, and a human conducts a final review and compactification. The loop instantiates the autoresearch paradigm (Karpathy, 2026) for the discovery of certified relaxations. We used Claude Opus 4.6 for our agent experiments, and detailed prompts can be found in Section D.

Coding agent. The coding agent proposes an improvement to the current relaxation-typically a new constraint on the coefficient vector c, or a strengthening of an existing one-together with a rigorous proof that the constraint is valid, i.e. that it holds for the coefficients of every admissible f. It then implements the modified convex program and solves it, producing a candidate bound.

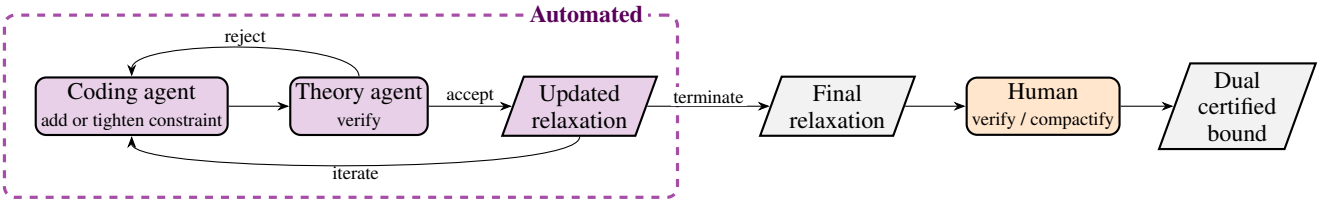

Figure 1. The framework. The *automated* block (purple, dashed) alternates between a coding agent that adds or tightens a constraint and a theory agent that verifies its validity and searches for counterexamples; accepted constraints update the relaxation, and the loop repeats. On termination it yields the final relaxation, which a human verifies and compactifies; the resulting bound is certified by an explicit dual-feasible point.

Theory agent. The theory agent reads the proposed constraint and its proof and checks that the constraint is a sound consequence of the admissibility of f. Critically, it does not only attempt to confirm the argument: it also *actively searches for a counterexample*-an admissible f whose coefficients violate the proposed constraint. A constraint survives only if its justification holds and no counterexample is found. This guards the invariant on which the entire bound rests: a constraint that some admissible f violates would shrink $\mathcal{K}$ below the admissible set and silently invalidate the lower bound.

The prompts used for the coding agent and the theory agent, as well as the problem descriptions, are in Section D.

Human review and compactification. The agent loop tends to accumulate constraints, not all of which are necessary: many are redundant at the optimum and contribute nothing to the bound. Once the loop has produced a candidate relaxation, a human performs a final pass-reading the program and the validity arguments, and removing redundant constraints to obtain a minimal program. Removing a constraint enlarges $\mathcal{K}$ and so, by (10), can only lower the bound; a constraint is therefore dropped only when the certified value was not meaningful in the history of objective values.

Certifying Correctness Two conditions must hold for a reported bound to be valid, and they are established by different means. First, every constraint in the final program must be a true property of all admissible f. This is established by a mathematical argument, produced by the agents and confirmed by human review; we do *not* formally verify these arguments (e.g. in a proof assistant), so this is the step at which correctness ultimately rests on human judgment. Second, the reported number must actually be a lower bound for the resulting convex program. This does *not* rest on human judgment: given the program, its dual is formed by the standard duality recipe, and any dual-feasible point yields a valid bound by weak duality (Section 2.2). We certify each bound by exhibiting such a point, rigorously checking whether the point is dual-feasible, and evaluating the dual objective in interval arithmetic.

We work in the canonical conic form used by CVXPY (Diamond & Boyd, 2016). Every program in this paper can be written as

$$\min_x \; c^\top x \quad \text{subject to} \quad Ax + s = b, \;\; s \in \mathcal{K}, \tag{11}$$

where $\mathcal{K}$ is a product of the nonnegative orthant, second-order cones, and positive-semidefinite cones; CVXPY's canonicalization produces exactly this form. The corresponding dual is

$$\max_y \; -b^\top y \quad \text{subject to} \quad A^\top y + c = 0, \;\; y \in \mathcal{K}^*, \tag{12}$$

with $\mathcal{K}^*$ the dual cone, and weak duality gives, for any dual-feasible y,

$$-b^\top y \;\le\; \min_x c^\top x \;=\; \Omega^\star. \tag{13}$$

A single dual-feasible $\tilde{y}$ therefore proves $\Omega^\star \ge -b^\top \tilde{y}$, independently of the solver that produced it. We obtain a candidate witness $\tilde{y}$ from CVXPY and verify it rigorously, which raises two issues: $\tilde{y}$ is only numerically dual-feasible, and controlling the rounding introduced by irrational problem data.

Cone membership is the first requirement. For the orthant and second-order blocks it reduces to a finite set of sign and norm inequalities, checked directly. The positive-semidefinite blocks are more delicate: certifying that a matrix is PSD in the presence of rounding cannot be done by computing eigenvalues naively, since a matrix that is PSD in exact arithmetic may have a small negative computed eigenvalue. We instead certify PSD-ness by a verified factorization carried out in interval arithmetic; the details are deferred to Appendix E.

The second issue is rounding. The coefficients A, b, c contain irrational entries (trigonometric moments and Fejér weights), so the matrices actually stored, $\bar{A}, \bar{b}, \bar{c}$, are rounded; writing $\varepsilon_A, \varepsilon_b, \varepsilon_c$ for elementwise bounds on $\|A - \bar{A}\|_\infty$, $\|b - \bar{b}\|_\infty$, and $\|c - \bar{c}\|_\infty$, each true coefficient is enclosed in an interval of the corresponding width. Both the stored problem and the final objective therefore carry rounding, and a rigorous bound must absorb both. Let

r bound the dual residual $\|A^\top \tilde{y} + c\|_\infty$, the extent to which $\tilde{y}$ fails exact dual feasibility, and let X be an a priori bound on $\|x\|$ over primal-optimal x^*. Weak duality together with these error terms yields the certified bound

$$\Omega^\star \;=\; c^\top x^\star \;\geq\; -b^\top \tilde{y} - \big(r + \varepsilon_A \|\tilde{y}\|_1 + \varepsilon_c\big)\, X - \varepsilon_b \|\tilde{y}\|_1. \tag{14}$$

Each quantity on the right is a guaranteed bound, and using the bound leads to a rigorous lower bound of Ω^*. We evaluate the entire right-hand side in interval arithmetic with outward rounding, so the reported value is the lower endpoint of a verified enclosure. See Section C for a detailed derivation.

4. Results

We apply the dual-agent approach described in Section 3 to two constants $C_{6.2}$ and $C_{6.5}$ For each we report the convex program discovered by the loop, and the certified lower bound it yields. All bounds are certified by an explicit dual-feasible point evaluated in exact rational arithmetic (Section 3); no floating-point margin enters the final certificate. At last, an analysis of the loop's behavior for the first autocorrelation inequality is discussed in Section 4.3.

4.1. Discovered convex programs

For $C_{6.5}$ the loop took the convex program of White (2022) as its starting point and searched for valid constraints to tighten it. The program (Algorithm 1) shows the resulting convex program, with the boxed pair of constraints are the ones added by the loop. The discovered constraints are a pair of *Bochner conditions*: $T_f \succeq 0$ enforces that the moment sequence of f is realizable by a nonnegative measure, and $I - T_f \succeq 0$ enforces the same for its complement $g = 1 - f$. White's formulation does not explicitly use the constraint $0 \leq f \leq 1$ pointwise, where our advantage is introduced. The loop identified both conditions automatically, and the theory agent verified their validity (Bochner's theorem, applied to f and to $g = 1 - f$) before they were accepted. For a rigorous proof of the added constraint see Section A.

The convex program depends on six parameters $h_1, h_2, p_1, p_2, q_1, q_2$ that bracket three quantities of an admissible f: the mean $E(M)$ of the overlap, with $h_1 \leq E(M) \leq h_2$; the first cosine coefficient c_1 of f, with $p_1 \leq c_1 \leq p_2$; and the first sine coefficient d_1, with $q_1 \leq d_1 \leq q_2$. Over all admissible f these quantities range within $E(M) \in [0, 2]$, $c_1 \in [0, 1]$, and $d_1 \in [-1, 1]$ (for a proof see (White, 2022)). Write $\Omega(h_1, h_2, p_1, p_2, q_1, q_2)$ for the certified optimum of the program with the parameters constrained to the box $[h_1, h_2] \times [p_1, p_2] \times [q_1, q_2]$. For any such box the program is a valid relaxation, so its optimum lower-bounds the score of every admissible f whose $(E(M), c_1, d_1)$ falls in that box.

Since every admissible f has $(E(M), c_1, d_1)$ somewhere in the full box $\mathcal{B} = [0, 2] \times [0, 1] \times [-1, 1]$, solving the program over all of $\mathcal{B}$ at once already gives a valid lower bound on Ω. That bound is weak, however: a relaxation over a large parameter box is loose, so its optimum comes out well below Ω. The bound sharpens as the box shrinks - constraining the parameters to a small region tightens the relaxation and raises its certified optimum. We therefore split $\mathcal{B}$ into sub-boxes and bound each one separately, which is the branch-and-bound step. Cover $\mathcal{B}$ by sub-boxes $B' = [h_1, h_2] \times [p_1, p_2] \times [q_1, q_2]$; a single certified dual point for $\Omega(h_1, h_2, p_1, p_2, q_1, q_2)$ lower-bounds the objective *simultaneously for every* $(E(M), c_1, d_1) \in B'$, and the smaller B' is, the tighter this local bound. A sub-box whose certificate already meets the target is cleared; otherwise it is split further and the procedure recurses, until all of $\mathcal{B}$ is covered. The reported bound is the smallest certified optimum over the cover - a valid lower bound on Ω, and a sharper one than the single whole-box solve.

For $C_{6.2}$ no prior convex program was available; the loop constructed one from scratch (Algorithm 2). The agentic system introduced variables as the following: a discretized nonnegative measure p on $[-\frac{1}{4}, \frac{1}{4}]$ (the support reduction of Lemma B.1), with trigonometric moments a_k, b_k. The program combines: a probability simplex and a positive-semidefinite moment matrix; per-cell Fourier envelopes bracketing a_k, b_k, computed in exact integer arithmetic so envelope validity carries no floating-point tolerance; a Fejér–Riesz Gram matrix Q certifying $\Omega - (F_K * g)(t) \geq 0$ for all t (the sum-of-squares encoding of the pointwise lower bound on the autocorrelation peak, $g = f * f$); a quadratic energy bound following from $\int g^2 = \sum_m |\hat{f}(m)|^4$; and plain and *localized* Bochner conditions, the latter valid because $\cos(2\pi x) \geq 0$ exactly on $[-\frac{1}{4}, \frac{1}{4}]$. The derivation of the novel convex program is in Section B.

4.2. Improved bounds

Both constants have long histories of incremental improvement, and in both cases the recent gains on the *upper* (construction) side came from search-based methods, while the *lower* (duality) side had not moved since 2022. Our framework improves the lower side of both.

Minimum overlap ($C_{6.5}$). The minimum-overlap constant $C_{6.5} = \lim_n M(n)/n$ has an even longer record. Early lower bounds came from averaging and rearrangement arguments - Erdős' $\frac{1}{4}$ (Erdős, 1955), and Moser's $\sqrt{4 - \sqrt{15}} \approx 0.35639$ (Moser & Murdeshwar, 1966) - until White (2022) reformulated the lower bound as a convex program and obtained 0.379005. The upper side has been refined by step-function constructions: 0.382002 (Haugland,

Algorithm 1 Discovered convex program for the Erdős minimum-overlap constant $C_{6.5}$.

Require: grid size N ($L = 2/N$), truncation T, modes R; box h_1, h_2, p_1, p_2 and fixed q_1, q_2

minimize Ω
over $\Omega \le 1$, $\{w_j, v_j\}$, $\{c_k, d_k\}_{k=1}^{T}$, $\{a_m, b_m\}_{m=1}^{2R}$, $\{\varepsilon_m, \delta_m\}_{m=1}^{R}$
subject to
$0 \le w_j,\ v_j \le \Omega \le 1$
$L \sum_j (w_j + v_j) = 1$
$h_1 \le L^2 \sum_j (j w_j - (j-1) v_j)$
$L^2 \sum_j ((j-1) w_j - j v_j) \le h_2$
$\frac{2}{3} + \frac{h_1^2}{2} \le L^3 \sum_j j^2 (w_j + v_j)$
$L^3 \sum_j (j-1)^2 (w_j + v_j) \le \frac{2}{3} + \frac{h_2^2}{2}$
$\frac{L}{2} \sum_j \alpha_{j,m}^{-} (w_j + v_j) \le \frac{4 \sin(\pi m/2)}{\pi m} a_m - 2(a_m^2 + b_m^2)$
$\frac{L}{2} \sum_j (\beta_{j,m}^{-} w_j - \beta_{j,m}^{+} v_j) \le -\frac{4 \sin(\pi m/2)}{\pi m} b_m$
$-\frac{4 \sin(\pi m/2)}{\pi m} b_m \le \frac{L}{2} \sum_j (\beta_{j,m}^{+} w_j - \beta_{j,m}^{-} v_j)$
$a_m = \frac{1}{2} c_{m/2},\ b_m = \frac{1}{2} d_{m/2}$ (m even)
$a_m = \varepsilon_m + \frac{2m \sin(\pi m/2)}{\pi} \left(\frac{1}{2m^2} + \sum_{k=1}^{T} \frac{(-1)^k}{m^2 - 4k^2} c_k \right)$
$b_m = \delta_m + \frac{4}{\pi} \sum_{k=1}^{T} \frac{k(-1)^k \sin(\pi m/2)}{m^2 - 4k^2} d_k$ (m odd)
$|\varepsilon_m|, |\delta_m| \le$ tail bound ($2m - 1 < 2T$)
$|c_k|, |d_k| \le \frac{2}{\pi},\ \sum_k (c_k^2 + d_k^2) \le \frac{1}{2}$
$p_1 \le c_1 \le p_2,\ q_1 \le d_1 \le q_2$
$\frac{L}{2} \sum_j \alpha_{j,2}^{+} (w_j + v_j) \ge -\frac{1}{2} (p_2^2 + \max\{q_1^2, q_2^2\})$
$T_f \succeq 0, \qquad I - T_f \succeq 0$
$(T_f)_{kl} = \frac{1}{4} \delta_{kl} + \frac{1}{2} (a_{|k-l|} - \mathbf{i}\, \mathrm{sgn}(k-l) b_{|k-l|})$
Branch and bound: Fix (q_1, q_2) and sweep the (h, p) rectangle. Certify each rectangle by an exact-arithmetic dual point; if certified, clear it, otherwise split and recurse. return smallest certified value over cleared rectangles

1996) and 0.380926 (Haugland, 2016), then 0.380924 AlphaEvolve (Georgiev et al., 2025), 0.380876 (Yuksekgonul et al., 2026) and 0.380871 in (Together AI, 2026). Our augmented program certifies

$$0.37912 \le C_{6.5} \ ,$$

improving White's 0.379005. The bound is the minimum certified value over the branch-and-bound sweep of the (h, p) box; the subdivision tree is shown in Figure 4 (Appendix E). Table 1 summarizes the record.

Autocorrelation ($C_{6.2}$). $C_{6.2}$ has been bounded below through a sequence of Fourier-analytic arguments: 1.262 (Martin & O'Bryant, 2009) and 1.2749 (Matolcsi & Vinuesa, 2010) - who also conjectured a ceiling of 1.276 for their method - followed by 1.28 via a finite-case relaxation

Algorithm 2 Discovered semidefinite program for the Sidon-set autocorrelation constant $C_{6.2}$.

Require: cells $2N$ on $[-\frac{1}{4}, \frac{1}{4}]$, modes K

minimize Ω
over $\Omega \ge 0$, $p \in \mathbb{R}_{\ge 0}^{2N}$, moments a_k, b_k, $M \succeq 0$, Hermitian $Q \succeq 0$, slacks $v_k \ge 0$
subject to
$\sum_j p_j = 1$
$M \succeq 0,\ M_{00} = 1,\ M_{0k} = a_k,\ M_{0,K+k} = b_k$
$\gamma_k^{-} \cdot p \le a_k \le \gamma_k^{+} \cdot p,\ \sigma_k^{-} \cdot p \le b_k \le \sigma_k^{+} \cdot p$
$\Omega \ge 1 + 2 \sum_k w_k v_k,\ v_k \ge (M_{kk} + M_{K+k,K+k})^2$
$Q \succeq 0,\ \mathrm{tr}(Q) = \Omega - 1$
$\sum_i Q_{i,i+k} = \mathbf{i}\, 2 w_k M_{k,K+k} - w_k (M_{kk} - M_{K+k,K+k})$
$\begin{bmatrix} T_R & -T_I \\ T_I & T_R \end{bmatrix} \succeq 0,\ T_{ij} = a_{|i-j|} - \mathbf{i}\, \mathrm{sgn}(i-j) b_{|i-j|}$
$\begin{bmatrix} T_R^{\nu} & -T_I^{\nu} \\ T_I^{\nu} & T_R^{\nu} \end{bmatrix} \succeq 0,$
$(T^{\nu})_{ij} = \nu_{i-j},\ \nu_k = \frac{1}{2} (\hat{\mu}_{k-1} + \hat{\mu}_{k+1})$

Table 1. State of the art for the Erdős minimum-overlap constant $C_{6.5}$ (Eq. (3)). Lower bounds are the duality side; upper bounds come from step-function constructions and recent search-based methods. Records as in Davis et al. (2026).

Lower bounds	Upper bounds
0.25 (Erdős, 1955)	0.380926 (Haugland, 2016)
0.35639 (Moser & Murdeshwar, 1966)	0.380924 (Georgiev et al., 2025)
0.379005 (White, 2022)	0.380876 (Yuksekgonul et al., 2026)
0.37912 (Ours)	0.380871 (Together AI, 2026)

(Cloninger & Steinerberger, 2017). The upper side, by explicit constructions, currently stands at 1.502862 (Together AI, 2026). Our discovered SDP certifies

$$1.2937 \le C_{6.2}$$

improving the previous lower record of 1.28. The certificate is an explicit dual-feasible point of the SDP, verified positive semidefinite and feasible in exact interval arithmetic, so the bound holds without reliance on the numerical solver or floating-point error analysis. Table 2 summarizes the record.

4.3. Loop convergence and verification

Figure 2 plots the certified primal objective against the loop iteration. The objective increases as the coding agent adds or tightens valid constraints; points shown in red are proposals the verifier rejected as *invalid* (a constraint failing to hold for some admissible f, established by an explicit counterexample), as opposed to merely non-binding. The loop makes progress on the certified bound while the verifier

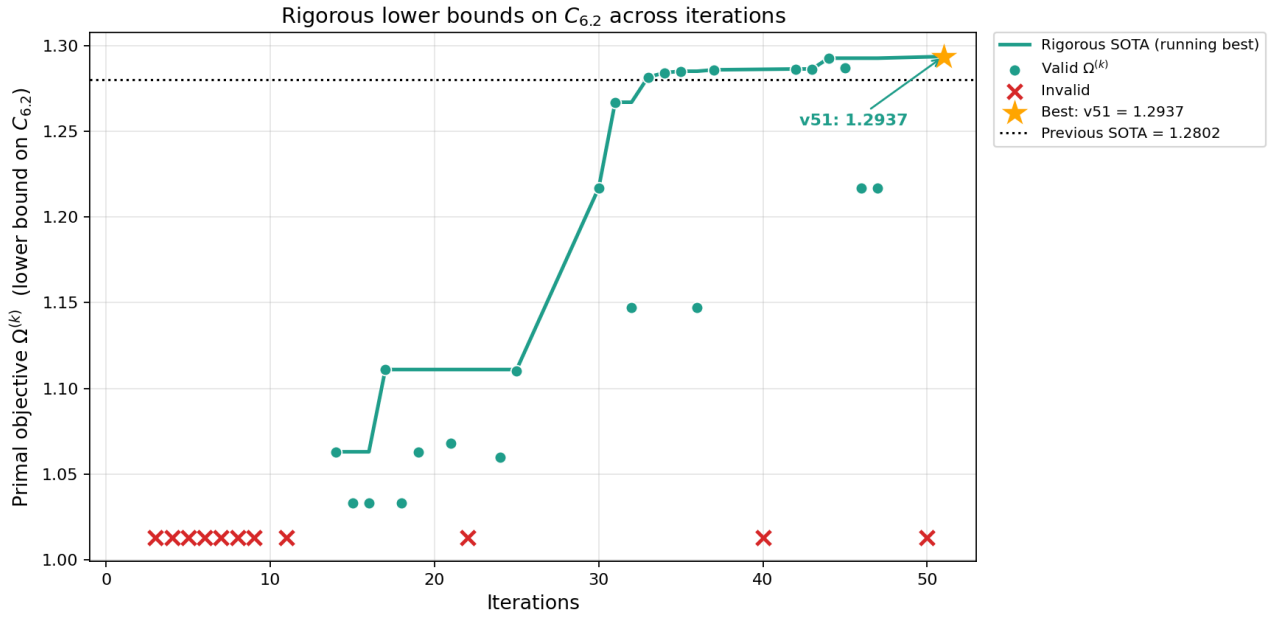

Figure 2. Certified primal objective of the convex program as a function of the loop iteration (version). The objective increases as the coding agent adds or tightens valid constraints; points in red are proposals the theory agent rejected as invalid (failing for some admissible f), as opposed to merely non-binding.

Table 2. State of the art for the Sidon-set autocorrelation constant $C_{6.2}$(Eq. (1)). Records as in Davis et al. (2026).

Lower bounds	Upper bounds
1.262 (Martin & O'Bryant, 2009)	1.503133 (Wang et al., 2025)
1.2749 (Matolcsi & Vinuesa, 2010)	1.503871 (Ye et al., 2026)
1.28 (Cloninger & Steinerberger, 2017)	1.502870 (Yuksekgonul et al., 2026)
1.2937 (Ours)	1.502862 (Together AI, 2026)

filters out invalid proposals that would otherwise corrupt it.

Beyond catching invalid *constraints*, the theory agent also corrects the search *direction*. In one episode the coding agent restricted its candidate functions to even functions, obtaining an invalid bound; the theory agent flagged that the even restriction is a genuine loss of generality - it discards admissible asymmetric configurations, so the resulting bound applies only to the even-restricted constant and not to $C_{6.2}$ itself - and directed the next iterations to drop it. After a few iterations the coding agent got rid of the even assumption: following the verifier's recommendation, the coding agent got rid of the even function assumption and started looking for a genuine lower bound. For the raw interaction snippets see Section F.

5. Conclusion

We presented a human-in-the-loop framework that applies the autoresearch paradigm to a target it has not previously been used for: the discovery of convex relaxations that certify *lower* bounds on sharp-constant inequalities. Where prior agent-based work improves such constants by constructing extremal examples, which bound the constant from above, our agents search for valid constraints that tighten a convex relaxation and thereby raise a certified lower bound. A coding agent proposes and implements constraints, a theory agent checks their validity and searches for counterexamples, and a human reviews and compactifies the final program; the reported bound is then certified by an explicit dual-feasible point verified in exact arithmetic. On two autocorrelation inequalities of (Georgiev et al., 2025), this yielded improved certified lower bounds.

Although we demonstrate the framework on autocorrelation inequalities, nothing in it is specific to these problems, or even to sharp-constant inequalities. The same loop applies wherever a lower bound on a nonconvex optimum can be expressed as a convex relaxation that is tightened by adjoining valid constraints and certified by duality-a setting that recurs across optimization, combinatorics, and control. The agents need only the ability to propose candidate constraints, argue their validity, and implement the resulting convex program; the certification step is problem-agnostic. Mapping out the class of nonconvex problems on which this is effective is, in our view, the most promising direction the method opens.

CODING AGENT

Iteration v7 assumes f is even. The WLOG reductions and the windowed-integral constraints are proved; the bound rises monotonically to 1.344 at N=20, strictly above 1 for the first time.

THEORY AGENT

The proof is valid - but it bounds the *even-restricted* constant only. Since $C_{6.2} \leq C_{y.2}^{\text{even}}$, this is not a bound on $C_{6.2}$: the even restriction discards admissible asymmetric f.

Concretely, 1.344 sits below the unrestricted target $C_{6.2} \geq 1.28$. Drop the even restriction next - e.g. a test functional on the autocorrelation $F * \check{F}$, whose Fourier data is nonnegative for *all* admissible f.

CODING AGENT

Iteration v14 removes the even restriction, giving the first rigorous bound on unrestricted $C_{6.2}$.

Figure 3. A representative exchange from the loop (condensed). The coding agent restricts to even f to make progress; the theory agent verifies the proof but flags that the even restriction is a genuine loss of generality, so the bound does not apply to $C_{6.2}$, and redirects the search. Iteration v14 follows this recommendation and obtains the first rigorous bound on the unrestricted constant.

Two limitations also indicate where the framework can be strengthened. The validity of each constraint currently rests on a mathematical argument confirmed by human review rather than formally verified. Recent progress on automated proof generation in Lean-both agentic prover-repair pipelines (Ospanov et al., 2026) and autoformalization of abstract results in convex optimization (Li et al., 2026)-suggests that this step could be machine-checked end to end, removing the residual reliance on human judgment and making the loop fully autonomous; coding agents have already been used to autoformalize substantial bodies of mathematics in Lean (Urban, 2026). Folding the final review and compactification into the loop is the corresponding step on the discovery side. Together, these would turn the present human-in-the-loop framework into an end-to-end automated and formally verified one.

Acknowledgements

This work was supported in part by the National Science Foundation (NSF) CAREER Award under Grant CCF-2236829, in part by the National Institutes of Health under Grant 1R01AG08950901A1, in part by the Office of Naval Research under Grant N00014-24-1-2164, and in part by the Defense Advanced Research Projects Agency under Grant HR00112490441.

References

Abouzaid, M., Blumberg, A. J., Hairer, M., Kileel, J., Kolda, T. G., Nelson, P. D., Spielman, D., Srivastava, N., Ward, R., Weinberger, S., et al. First proof. *arXiv preprint arXiv:2602.05192*, 2026.

Bubeck, S., Coester, C., Eldan, R., Gowers, T., Lee, Y. T., Lupsasca, A., Sawhney, M., Scherrer, R., Sellke, M., Spears, B. K., et al. Early science acceleration experiments with gpt-5. *arXiv preprint arXiv:2511.16072*, 2025.

Cloninger, A. and Steinerberger, S. On suprema of autoconvolutions with an application to sidon sets. *Proceedings of the American Mathematical Society*, 145(8):3191–3200, 2017.

Davis, D., Ivanisvili, P., Tao, T., and contributors. Optimization constants in mathematics. GitHub repository, 2026. URL https://github.com/teorth/optimizationproblems.

Diamond, S. and Boyd, S. Cvxpy: A python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Erdős, P. Some remarks on number theory. *Riveon Lematematika*, 9:45–48, 1955.

Georgiev, B., Gómez-Serrano, J., Tao, T., and Wagner, A. Z. Mathematical exploration and discovery at scale. *arXiv preprint arXiv:2511.02864*, 2025.

Haugland, J. K. Advances in the minimum overlap problem. *journal of number theory*, 58(1):71–78, 1996.

Haugland, J. K. The minimum overlap problem revisited. *arXiv preprint arXiv:1609.08000*, 2016.

Huang, Y. and Yang, L. F. Winning gold at imo 2025 with a model-agnostic verification-and-refinement pipeline. *arXiv preprint arXiv:2507.15855*, 2025.

Jain, N., Gu, A., Li, W.-D., Yan, F., Zhang, T., Wang, S., Solar-Lezama, A., Sen, K., and Stoica, I. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pp. 58791–58831, 2025.

Jimenez, C. E., Yang, J., Wettig, A., Yao, S., Pei, K., Press, O., and Narasimhan, K. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, volume 2024, pp. 54107–54157, 2024.

Karpathy, A. autoresearch: AI agents running research on single-GPU nanochat training automatically. https://github.com/karpathy/autoresearch, 2026. GitHub repository, released March 2026; accessed 2026-05-21.

Li, C., Ma, W., Wang, Z., and Wen, Z. Sita: A framework for structure-to-instance theorem autoformalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19224–19232, 2026.

Martin, G. and O'Bryant, K. The supremum of autoconvolutions, with applications to additive number theory. *Illinois Journal of Mathematics*, 53(1):219–235, 2009.

Matolcsi, M. and Vinuesa, C. Improved bounds on the supremum of autoconvolutions. *Journal of mathematical analysis and applications*, 372(2):439–447, 2010.

Moser, L. and Murdeshwar, M. On the overlap of a function with the translation of its complement. In *Colloquium Mathematicum*, volume 15, pp. 93–97. Institute of Mathematics Polish Academy of Sciences, 1966.

Novikov, A., Vũ, N., Eisenberger, M., Dupont, E., Huang, P.-S., Wagner, A. Z., Shirobokov, S., Kozlovskii, B., Ruiz, F. J., Mehrabian, A., et al. Alphaevolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

Ospanov, A., Farnia, F., and Mohit, R. Apollo: Automated llm and lean collaboration for advanced formal reasoning. *Advances in Neural Information Processing Systems*, 38: 41599–41633, 2026.

Robeyns, M., Szummer, M., and Aitchison, L. A self-improving coding agent. *arXiv preprint arXiv:2504.15228*, 2025.

Romera-Paredes, B., Barekatain, M., Novikov, A., Balog, M., Kumar, M. P., Dupont, E., Ruiz, F. J., Ellenberg, J. S., Wang, P., Fawzi, O., et al. Mathematical discoveries from program search with large language models. *Nature*, 625 (7995):468–475, 2024.

Together AI. EinsteinArena: New state-of-the-art bounds for open problems. https://github.com/togethercomputer/EinsteinArena-new-SOTA, 2026. GitHub repository; new SOTA bounds for open problems in combinatorics and harmonic analysis obtained by AI agents. Accessed 2026-05-25.

Tsoukalas, G., Kovsharov, A., Shirobokov, S., Surina, A., Firsching, M., Bérczi, G., Ruiz, F. J., Suggala, A., Wagner, A. Z., Wieser, E., et al. Advancing mathematics research with ai-driven formal proof search. *arXiv preprint arXiv:2605.22763*, 2026.

Urban, J. 130k lines of formal topology in two weeks: Simple and cheap autoformalization for everyone? *arXiv preprint arXiv:2601.03298*, 2026.

Wang, Y., Su, S.-R., Zeng, Z., Xu, E., Ren, L., Yang, X., Huang, Z., He, X., Ma, L., Peng, B., et al. Thetaevolve: Test-time learning on open problems. *arXiv preprint arXiv:2511.23473*, 2025.

White, E. P. Erdos' minimum overlap problem. *arXiv preprint arXiv:2201.05704*, 2022.

Ye, H., Lin, H., Tang, J., Luo, Y., Yang, C., Su, C., Thapa, R., Yang, R., Liu, R., Li, Z., et al. Evaluation-driven scaling for scientific discovery. *arXiv preprint arXiv:2604.19341*, 2026.

Yuksekgonul, M., Koceja, D., Li, X., Bianchi, F., McCaleb, J., Wang, X., Kautz, J., Choi, Y., Zou, J., Guestrin, C., et al. Learning to discover at test time. *arXiv preprint arXiv:2601.16175*, 2026.

Zheng, Z., Cheng, Z., Shen, Z., Zhou, S., Liu, K., He, H., Li, D., Wei, S., Hao, H., Yao, J., et al. Livecodebench pro: How do olympiad medalists judge llms in competitive programming? *Advances in Neural Information Processing Systems*, 38, 2026.

A. Additional constraint of the convex problem of Erdős minimum overlap problem

Our program for the Erdős minimum overlap constant $C_{6.5}$ takes White's convex program (White, 2022) as its base and adds two positive-semidefinite constraints discovered by the loop. We treat White's program as given and refer the reader to White (2022) for its derivation; this appendix justifies only the added constraints.

Let T_f be the Hermitian Toeplitz matrix built from the trigonometric moments of f,

$$(T_f)_{kl} \;=\; \tfrac{1}{4}\,\delta_{kl} \;+\; \tfrac{1}{2}\big(a_{|k-l|} - \mathrm{i}\,\mathrm{sgn}(k-l)\,b_{|k-l|}\big), \tag{15}$$

where a_m, b_m are the cosine and sine Fourier coefficients of f (so that the (k,l) entry depends only on the lag $k-l$). The loop added the pair

$$T_f \succeq 0 \qquad \text{and} \qquad I - T_f \succeq 0. \tag{16}$$

We show both are valid necessary conditions, i.e. that every admissible f produces moments satisfying (16); adding them therefore tightens the relaxation without excluding any admissible f.

Proposition A.1. *For every admissible f, the matrix T_f of (15) satisfies $T_f \succeq 0$ and $I - T_f \succeq 0$.*

Proof. By construction T_f is the moment (Toeplitz) matrix of f: its entries are the Fourier coefficients of f arranged by lag, so for any complex vector z,

$$z^* T_f z \;=\; \int_{-1}^{1} f(x) \Big|\sum_k z_k e^{\mathrm{i}k\pi x}\Big|^2 dx.$$

Since $f \geq 0$, the integrand is nonnegative, hence $z^* T_f z \geq 0$ for all z and $T_f \succeq 0$. This is Bochner's theorem: a sequence is the moment sequence of a nonnegative measure iff its Toeplitz form is positive semidefinite.

For the second condition, apply the same argument to the complement $g = 1 - f$. Because $f \leq 1$ we have $g \geq 0$, so g is also a nonnegative function and its moment matrix T_g satisfies $T_g \succeq 0$. The constant function 1 has moment matrix equal to the identity I (its only nonzero Fourier coefficient is the constant term, which contributes the diagonal), and the moment map is linear, so $T_g = T_1 - T_f = I - T_f$. Hence $I - T_f = T_g \succeq 0$. □

B. Derivation of the convex problem of the First Autocorrelation Inequality

This section derives the convex relaxation for the sharp constant $C_{6.2}$ of `Problem_6.2.md` and proves rigorously that its optimum Ω^* is a valid lower bound, $C_{6.2} \geq \Omega^*$. The development follows the same template as White's program for the Erdős overlap problem (Section 1): identify the Fourier/moment data of an admissible f, prove a list of *necessary* linear and semidefinite conditions on that data, and observe that the relaxed feasible set therefore contains every admissible f.

B.1. Reduction and normalisation

We seek a lower bound for

$$C_{6.2} = \inf_{f \geq 0} \frac{\max_{-1/2 \leq t \leq 1/2} \int_{\mathbb{R}} f(t-x) f(x)\, dx}{\big(\int_{-1/4}^{1/4} f(x)\, dx\big)^2}.$$

Lemma B.1 (Support reduction)**.** *It suffices to bound the infimum over $f \geq 0$ with $\operatorname{supp} f \subseteq [-1/4, 1/4]$. Concretely, if for all $g \geq 0$ with $\operatorname{supp} g \subseteq [-1/4, 1/4]$ we have $\max_{|t| \leq 1/2} (g * g)(t) \geq C\big(\int_{-1/4}^{1/4} g\big)^2$, then the same inequality holds for all $f \geq 0$.*

Proof. Given $f \geq 0$, set $g = f \cdot \mathbf{1}_{[-1/4,1/4]}$, so $0 \leq g \leq f$ and $\operatorname{supp} g \subseteq [-1/4, 1/4]$. The denominators agree: $\int_{-1/4}^{1/4} g = \int_{-1/4}^{1/4} f$. For the numerator, $g \leq f$ and both are nonnegative, so $(g*g)(t) = \int g(t-x) g(x)\, dx \leq \int f(t-x) f(x)\, dx = (f * f)(t)$ pointwise; taking $\max_{|t| \leq 1/2}$ preserves the inequality. Hence $\max(f*f) \geq \max(g*g) \geq C(\int g)^2 = C(\int f)^2$. □

By homogeneity we normalise $\int_{-1/4}^{1/4} f = 1$. Writing $g = f * f$ and $\Omega = \max_{|t| \leq 1/2}(g)(t) = \|f * f\|_\infty$, the reduced problem is

$$S \;=\; \inf_f \Big\{ \|f * f\|_\infty \;:\; f \geq 0,\; \int_{-1/4}^{1/4} f = 1,\; \operatorname{supp} f \subseteq [-\tfrac{1}{4}, \tfrac{1}{4}] \Big\}, \qquad C_{6.2} = S. \tag{17}$$

B.2. Fourier and moment data

For an admissible f define the real Fourier coefficients

$$a_k = \int_{-1/4}^{1/4} f(x)\cos(2\pi kx)\,dx, \qquad b_k = \int_{-1/4}^{1/4} f(x)\sin(2\pi kx)\,dx, \qquad k = 0, 1, \ldots, K,$$

so that $\hat{f}(k) = a_k - ib_k$ and $\hat{f}(-k) = a_k + ib_k$, with $a_0 = \int f = 1$ and $b_0 = 0$. Because $g = f * f$ has $\hat{g}(k) = \hat{f}(k)^2$, the autocorrelation's spectrum is determined by (a_k, b_k).

We also discretise the localisation window into $2N$ equal cells $I_j = \big[-\frac{1}{4} + \frac{j}{4N}, -\frac{1}{4} + \frac{j+1}{4N}\big]$, $j = 0, 1, \ldots, 2N-1$, and set the local masses

$$p_j = \int_{I_j} f(x)\,dx \;\geq\; 0.$$

B.3. Necessary conditions

(1) Mass. Since the cells partition $[-1/4, 1/4]$ and $\int_{-1/4}^{1/4} f = 1$,

$$\sum_{j=0}^{2N-1} p_j = 1, \qquad p_j \geq 0. \tag{18}$$

(2) Fourier coefficients from local masses. For each cell define the trigonometric extrema

$$C_{kj}^{-} = \min_{x\in I_j}\cos(2\pi kx), \quad C_{kj}^{+} = \max_{x\in I_j}\cos(2\pi kx), \quad S_{kj}^{-} = \min_{x\in I_j}\sin(2\pi kx), \quad S_{kj}^{+} = \max_{x\in I_j}\sin(2\pi kx).$$

Lemma B.2 (Coefficient envelopes)**.** *For every admissible f and every k,*

$$\sum_{j=0}^{2N-1} C_{kj}^{-} p_j \;\leq\; a_k \;\leq\; \sum_{j=0}^{2N-1} C_{kj}^{+} p_j, \qquad \sum_{j=0}^{2N-1} S_{kj}^{-} p_j \;\leq\; b_k \;\leq\; \sum_{j=0}^{2N-1} S_{kj}^{+} p_j. \tag{19}$$

Proof. Since $f \geq 0$ and $\operatorname{supp} f$ is partitioned by the I_j,

$$a_k = \int_{-1/4}^{1/4} f(x)\cos(2\pi kx)\,dx = \sum_j \int_{I_j} f(x)\cos(2\pi kx)\,dx.$$

On each cell, $C_{kj}^{-} \leq \cos(2\pi kx) \leq C_{kj}^{+}$ for all $x \in I_j$, and $\int_{I_j} f = p_j \geq 0$. Multiplying the pointwise bounds by the nonnegative density f and integrating gives $C_{kj}^{-} p_j \leq \int_{I_j} f\cos(2\pi kx)\,dx \leq C_{kj}^{+} p_j$. Summing over j yields the bound on a_k; the argument for b_k is identical with $\sin$. □

(3) Moment matrix. Let $y = (1, a_1, \ldots, a_K, b_1, \ldots, b_K)^\top \in \mathbb{R}^{2K+1}$ and $M = y\,y^\top \in \mathbb{R}^{(2K+1)\times(2K+1)}$, indexed $0, 1, \ldots, 2K$.

Lemma B.3 (Moment matrix is feasible and rank one)**.** *The matrix $M = yy^\top$ satisfies $M \succeq 0$ and*

$$M_{0,0} = 1, \quad M_{0,k} = a_k, \quad M_{0,K+k} = b_k, \quad M_{k,k} = a_k^2, \quad M_{K+k,K+k} = b_k^2 \quad (1 \leq k \leq K).$$

Proof. $M = yy^\top$ is a Gram matrix of a single vector, hence PSD: $u^\top M u = (y^\top u)^2 \geq 0$ for all u. The listed entries are the corresponding products $y_i y_j$: $y_0 = 1$, $y_k = a_k$, $y_{K+k} = b_k$, so $M_{0,0} = 1$, $M_{0,k} = a_k$, $M_{0,K+k} = b_k$, $M_{k,k} = a_k^2$, $M_{K+k,K+k} = b_k^2$. □

In the relaxation we keep only the convex consequences: $M \succeq 0$, the affine entry constraints of Lemma B.3 linking the first row/diagonal to (a, b), and the envelopes (19). We do *not* impose $\operatorname{rank} M = 1$ (nonconvex). Every admissible f produces a feasible M, so no admissible point is lost.

(4) The objective bound via Parseval.

Lemma B.4 (Spectral lower bound on Ω)**.** *Let $w_k = 1 - \frac{k}{K+1} \in [0,1)$ be the Fejér weights. For admissible f with $\int_{-1/4}^{1/4} f = 1$, the sharp identity is*

$$\Omega \;=\; \|f * f\|_\infty \;\geq\; 1 + 2\sum_{k=1}^{\infty} \left(a_k^2 + b_k^2\right)^2, \tag{20}$$

and consequently the weighted, truncated relaxation used by the program holds:

$$\Omega \;\geq\; 1 + 2\sum_{k=1}^{K} w_k\, v_k, \qquad \textit{where} \quad v_k \geq \left(M_{k,k} + M_{K+k,K+k}\right)^2 = (a_k^2 + b_k^2)^2. \tag{21}$$

Proof. Let $g = f * f \geq 0$ (nonnegative as a self-convolution of $f \geq 0$). Its total mass is $\int_{\mathbb{R}} g = \left(\int f\right)^2 = 1$. Therefore $\Omega = \|g\|_\infty = \|g\|_\infty \int g \geq \int g^2$, since $\int g^2 \leq \|g\|_\infty \int g$ for $g \geq 0$. By Parseval and $\hat{g}(m) = \hat{f}(m)^2$,

$$\int g^2 = \sum_{m\in\mathbb{Z}} |\hat{g}(m)|^2 = \sum_{m\in\mathbb{Z}} |\hat{f}(m)|^4 = 1 + 2\sum_{k=1}^{\infty} (a_k^2 + b_k^2)^2,$$

using $\hat{f}(0) = 1$ and $|\hat{f}(\pm k)|^2 = a_k^2 + b_k^2$. This proves (20). For (21): every term $(a_k^2 + b_k^2)^2 \geq 0$ and $0 \leq w_k \leq 1$, so

$$1 + 2\sum_{k=1}^{K} w_k (a_k^2 + b_k^2)^2 \;\leq\; 1 + 2\sum_{k=1}^{\infty} (a_k^2 + b_k^2)^2 \;\leq\; \Omega.$$

With $a_k^2 = M_{k,k}$, $b_k^2 = M_{K+k,K+k}$ (Lemma B.3) and the epigraph variable $v_k \geq (M_{k,k} + M_{K+k,K+k})^2$, the left side is exactly the program's constraint. The weights and truncation only *weaken* the bound, so it remains a valid lower bound on Ω. □

B.4. The three positive-semidefinite blocks

The program contains three Hermitian PSD blocks, each of size $K+1$ and each the real embedding of a Hermitian Toeplitz matrix. We verify all three from the quadratic-form/Bochner principle: a Hermitian Toeplitz matrix $T = (\tau_{i-j})$ is positive semidefinite whenever its symbol $\tau_k = \int w(x) e^{-2\pi i k x}\, dx$ arises from a *nonnegative* weight w, because then for all $z \in \mathbb{C}^{K+1}$

$$z^* T z = \sum_{i,j} \bar{z}_i\, \tau_{i-j}\, z_j = \int w(x) \left| \sum_{j=0}^{K} z_j e^{2\pi i j x} \right|^2 dx \geq 0. \tag{22}$$

The three blocks differ only in the choice of nonnegative weight w.

(5a) Plain Bochner–Toeplitz on f. Recall $\hat{f}(k) = a_k - i b_k$ (with $\hat{f}(-k) = \overline{\hat{f}(k)}$), and form the Hermitian Toeplitz T_f with $(T_f)_{ij} = \hat{f}(i-j)$, i.e. real part $a_{|i-j|}$ and imaginary part $-b_{|i-j|}$ for $i > j$, $+b_{|i-j|}$ for $i < j$, diagonal $a_0 = 1$. In the program this is the `big` block.

Lemma B.5 (Plain Bochner block)**.** $T_f \succeq 0$.

Proof. Apply (22) with weight $w = f \geq 0$: the symbol is $\tau_k = \int_{-1/4}^{1/4} f(x) e^{-2\pi i k x}\, dx = \hat{f}(k) = a_k - i b_k$, which are precisely the entries of T_f. Hence $z^* T_f z = \int_{-1/4}^{1/4} f(x) \left| \sum_j z_j e^{2\pi i j x} \right|^2 dx \geq 0$ since $f \geq 0$. □

(5b) Localised Bochner with $h(x) = \cos(2\pi x)$. Form the Hermitian Toeplitz T_ν with symbol ν_k, where, writing sgn for the sign function,

$$\operatorname{Re} \nu_k = \tfrac{1}{2}\left(a_{|k-1|} + a_{|k+1|}\right), \qquad \operatorname{Im} \nu_k = -\tfrac{1}{2}\left(\operatorname{sgn}(k-1)\, b_{|k-1|} + \operatorname{sgn}(k+1)\, b_{|k+1|}\right).$$

In the program this is the `big_nu` block, assembled from `nu_re` and `nu_im`.

Lemma B.6 (Localised Bochner block)**.** *Define ν_k by the affine maps $\operatorname{Re}\nu_k = \frac{1}{2}(a_{|k-1|} + a_{|k+1|})$, $\operatorname{Im}\nu_k = -\frac{1}{2}(\operatorname{sgn}(k-1)\, b_{|k-1|} + \operatorname{sgn}(k+1)\, b_{|k+1|})$, and let $T_\nu = (\nu_{i-j})_{i,j=0}^{K}$. Then:*

(i) *ν_k equals the k-th Fourier coefficient of $x \mapsto \cos(2\pi x) f(x)$;*

(ii) *$\nu_{-k} = \overline{\nu_k}$, so T_ν is Hermitian Toeplitz; equivalently $\operatorname{Re} T_\nu$ is symmetric and $\operatorname{Im} T_\nu$ is skew-symmetric, which is exactly the structure required for the real embedding $\left[\begin{smallmatrix} \operatorname{Re} T_\nu & -\operatorname{Im} T_\nu \\ \operatorname{Im} T_\nu & \operatorname{Re} T_\nu \end{smallmatrix}\right]$ to represent T_ν;*

(iii) *the program imposes the single semidefinite constraint $T_\nu \succeq 0$ (no trace or band-sum equalities are attached to this block), and $T_\nu \succeq 0$ holds for every admissible f.*

Proof. (i) Since $\cos(2\pi x) = \frac{1}{2}(e^{2\pi i x} + e^{-2\pi i x})$, multiplication by $\cos(2\pi x)$ shifts Fourier coefficients by ± 1: $\widehat{\cos(2\pi\cdot)f}(k) = \frac{1}{2}(\hat{f}(k-1) + \hat{f}(k+1))$. Using $\hat{f}(m) = a_{|m|} - i\,\operatorname{sgn}(m)\, b_{|m|}$ for all $m \in \mathbb{Z}$ (with $\hat{f}(0) = a_0$ real), the real part of $\frac{1}{2}(\hat{f}(k-1) + \hat{f}(k+1))$ is $\frac{1}{2}(a_{|k-1|} + a_{|k+1|})$ and the imaginary part is $-\frac{1}{2}(\operatorname{sgn}(k-1) b_{|k-1|} + \operatorname{sgn}(k+1) b_{|k+1|})$, matching ν_k exactly.

(ii) From (i), $\nu_{-k} = \widehat{\cos(2\pi\cdot)f}(-k) = \overline{\widehat{\cos(2\pi\cdot)f}(k)} = \overline{\nu_k}$ because $\cos(2\pi x) f(x)$ is real-valued. Hence $(T_\nu)_{ji} = \nu_{j-i} = \nu_{-(i-j)} = \overline{\nu_{i-j}} = \overline{(T_\nu)_{ij}}$, so T_ν is Hermitian; writing $T_\nu = \operatorname{Re} T_\nu + i \operatorname{Im} T_\nu$, $\operatorname{Re} T_\nu$ is symmetric and $\operatorname{Im} T_\nu$ is skew-symmetric, which is precisely what makes the real embedding a faithful representation (its spectrum is that of T_ν doubled).

(iii) Apply (22) with weight $w(x) = \cos(2\pi x) f(x)$:

$$z^* T_\nu z = \int_{-1/4}^{1/4} \cos(2\pi x)\, f(x) \Big| \sum_{j=0}^{K} z_j e^{2\pi i j x} \Big|^2 dx.$$

On $\operatorname{supp} f = [-1/4, 1/4]$ we have $2\pi|x| \le \pi/2$, so $\cos(2\pi x) \ge 0$ throughout the support of f; the integrand is therefore nonnegative and $z^* T_\nu z \ge 0$. □

Remark B.7. The localiser $h(x) = \cos(2\pi x)$ is chosen precisely because its first nonnegativity interval $[-1/4, 1/4]$ coincides with $\operatorname{supp} f$. Any $h \ge 0$ on $[-1/4, 1/4]$ whose Fourier support is bounded would furnish a valid localised block; $\cos(2\pi x)$ is the simplest single-frequency choice and gives the ± 1 coefficient shift above.

(5c) Fejér–Riesz block Q for the autocorrelation. Let $g = f * f$, so $\hat{g}(k) = \hat{f}(k)^2 = (a_k - i b_k)^2 = (a_k^2 - b_k^2) - i\,(2 a_k b_k)$, and let F_K be the K-th Fejér kernel with weights $w_k = 1 - \frac{k}{K+1}$, $F_K \ge 0$, $\int F_K = 1$. Define Q as the Hermitian Toeplitz matrix with symbol

$$q_0 = \Omega - 1, \qquad q_k = -\, w_k\, \hat{g}(k) \;\; (1 \le k \le K),$$

in the program this is the Q block (the constraints $\operatorname{Re}\operatorname{tr} Q = \Omega - 1$ and the banded real/imaginary identities).

Lemma B.8 (Fejér–Riesz block)**.** *Let $z(t) = (1, e^{it}, \dots, e^{iKt})^\top$ and let Q be the Hermitian $(K+1) \times (K+1)$ Gram matrix of the localised symbol $h(t) = \Omega - (F_K * g)(t)$, i.e. $h(t) = z(t)^* Q\, z(t)$. With M the rank-one moment matrix of Lemma B.3 (so $M_{k,k} = a_k^2$, $M_{K+k,K+k} = b_k^2$, $M_{k,K+k} = a_k b_k$), the symbol coefficients $q_k := \sum_{l-j=k} Q_{jl}$ satisfy*

$$\operatorname{Re} q_k = -w_k\,(M_{k,k} - M_{K+k,K+k}), \qquad \operatorname{Im} q_k = +\,2\, w_k\, M_{k,K+k} \qquad (1 \le k \le K),$$

the k-th diagonal sum of Q reproducing the program's banded constraints. The zeroth constraint is the trace, $\operatorname{tr} Q = q_0 = \Omega - 1$. Moreover $Q \succeq 0$.

Proof. Substituting $\hat{g}(k) = (a_k^2 - b_k^2) - i\,(2 a_k b_k)$ into $q_k = -w_k \hat{g}(k)$ gives

$$\operatorname{Re} q_k = -w_k(a_k^2 - b_k^2) = -w_k(M_{k,k} - M_{K+k,K+k}), \qquad \operatorname{Im} q_k = +w_k(2 a_k b_k) = 2 w_k M_{k,K+k}.$$

Expanding $h(t) = z(t)^* Q z(t) = \sum_{j,l} Q_{jl} e^{i(l-j)t}$ and matching the coefficient of e^{ikt} gives $\hat{h}(k) = \sum_{l-j=k} Q_{jl} = q_k$, so these are exactly the banded (diagonal-sum) constraints of the program. For $k = 0$, $\hat{g}(0) = (\int f)^2 = 1$ and $w_0 = 1$, whence $q_0 = \hat{h}(0) = \Omega - \widehat{F_K * g}(0) = \Omega - 1$; since the 0-th diagonal is the main diagonal, $\operatorname{tr} Q = q_0 = \Omega - 1$.

The symbol h is a nonnegative trigonometric polynomial of degree K: indeed $g = f * f \geq 0$ and $F_K \geq 0$ with $\int F_K = 1$, so by Jensen/averaging $(F_K * g)(t) \leq \|g\|_\infty = \Omega$, giving $h = \Omega - F_K * g \geq 0$, while $\hat{h}(k) = w_k \hat{g}(k)$ vanishes for $|k| > K$ since $w_k = 0$ there. By the Fejér–Riesz theorem there is an analytic polynomial $p(t) = \sum_{j=0}^{K} c_j e^{ijt}$ with

$$h(t) = |p(t)|^2 = z(t)^* \, (cc^*) \, z(t), \qquad c = (c_0, \ldots, c_K)^\top.$$

Hence $Q = cc^* \succeq 0$ realises the symbol h with the prescribed diagonal sums: it is the Fejér–Riesz certificate, and $Q \succeq 0$ as claimed. □

Real embedding. As in Section 1, each Hermitian block $T = A + iB$ (A symmetric, B skew-symmetric) is realised in CVXPY through the real embedding $\left[\begin{smallmatrix} A & -B \\ B & A \end{smallmatrix}\right] \succeq 0$, equivalent to $T \succeq 0$ because the embedding's spectrum is that of T with each eigenvalue doubled. In the program this is the `cp.bmat([[T_R,-T_I],[T_I,T_R]])` construction, used for all three Hermitian blocks.

B.5. The relaxed convex program and its validity

Definition B.9 (Relaxation)**.** With $g = f * f$, $\hat{f}(k) = a_k - \mathbf{i} b_k$, Fejér weights $w_k = 1 - \frac{k}{K+1}$, and $\gamma_k^{\pm}, \sigma_k^{\pm}$ the per-cell lower/upper envelopes of $\cos(2\pi kx), \sin(2\pi kx)$, the value Ω^* is the optimum of

$$\begin{aligned}
\Omega^* = \min_{\Omega,\, p,\, a,\, b,\, M,\, Q,\, v} \quad & \Omega \\
\text{s.t.} \quad & \textstyle\sum_j p_j = 1,\ p_j \geq 0 && \text{(mass)} \\
& \gamma_k^- \cdot p \leq a_k \leq \gamma_k^+ \cdot p, \quad \sigma_k^- \cdot p \leq b_k \leq \sigma_k^+ \cdot p && \text{(Lemma B.2)} \\
& M \succeq 0,\ M_{0,0} = 1,\ M_{0,k} = a_k,\ M_{0,K+k} = b_k && \text{(Lemma B.3)} \\
& v_k \geq (M_{k,k} + M_{K+k,K+k})^2, \quad \Omega \geq 1 + 2\textstyle\sum_{k=1}^{K} w_k \, v_k && \text{(Lemma B.4)} \\
& Q \succeq 0,\ \operatorname{tr} Q = \Omega - 1 && \text{(Lemma B.8)} \\
& \textstyle\sum_i Q_{i,i+k} = \mathbf{i}\, 2 w_k M_{k,K+k} - w_k \big(M_{k,k} - M_{K+k,K+k}\big) && \text{(Fejér band, Lemma B.8)} \\
& T_f \succeq 0, \quad (T_f)_{ij} = a_{|i-j|} - \mathbf{i}\, \operatorname{sgn}(i-j)\, b_{|i-j|} && \text{(Lemma B.5)} \\
& T_\nu \succeq 0, \quad (T_\nu)_{ij} = \nu_{i-j}, \quad \nu_k = \tfrac{1}{2}(\hat{\mu}_{k-1} + \hat{\mu}_{k+1}) && \text{(Lemma B.6).}
\end{aligned} \tag{23}$$

The envelope coefficients $\gamma_k^{\pm}, \sigma_k^{\pm}$ are computed by the *analytical* cell extrema of $\cos(2\pi kx), \sin(2\pi kx)$ on each cell (the `cos_extrema_on_cell`/`sin_extrema_on_cell` routines), not a grid sample. This matters for rigor: a grid minimum over-estimates the true minimum, which would make $a_k \geq (\text{grid min}) \cdot p$ *stricter* than correct and could exclude an admissible f. The analytical extrema (endpoints together with the interior critical points $x = m/(2k)$ for cos and $x = (2m+1)/(4k)$ for sin) give the true cell min/max, so Lemma B.2 holds exactly.

Theorem B.10 (Validity of the relaxation)**.** $C_{6.2} = S \geq \Omega^*$.

Proof. Let f be any admissible function for the reduced problem (17): $f \geq 0$, $\int_{-1/4}^{1/4} f = 1$, $\operatorname{supp} f \subseteq [-1/4, 1/4]$. We construct a feasible point of (23) with objective value $\|f * f\|_\infty$.

Set $p_j = \int_{I_j} f$, a_k, b_k the Fourier coefficients of f, $y = (1, a, b)$, $M = yy^\top$, $\Omega = \|f * f\|_\infty$, $v_k = (a_k^2 + b_k^2)^2$, and T_f, T_ν, Q the three Hermitian Toeplitz blocks built from f's data. Each constraint of (23) is then satisfied:

- the mass constraint by (18);
- the coefficient envelopes by Lemma B.2 (analytical cell extrema);
- $M \succeq 0$ and its entry identities by Lemma B.3;
- the objective inequality $\Omega \geq 1 + 2\sum_k w_k v_k$ by Lemma B.4;
- $T_f \succeq 0$ by Lemma B.5 (weight $f \geq 0$);
- $T_\nu \succeq 0$ by Lemma B.6 (weight $\cos(2\pi x) f \geq 0$ on $\operatorname{supp} f$);

- $Q \succeq 0$ and the Fejér band identities by Lemma B.8 (weight $\Omega - F_K * g \geq 0$).

Every one of these is a *necessary* condition - proved above for an arbitrary admissible f, not merely for an optimiser - so the constructed point is feasible. Its objective is $\Omega = \|f * f\|_\infty$. Therefore

$$\Omega^* \leq \|f * f\|_\infty.$$

Taking the infimum over all admissible f gives $\Omega^* \leq S = C_{6.2}$, i.e. Ω^* is a valid lower bound. □

C. Floating point precision details

A numerical conic solver returns an approximate primal–dual pair that is in general neither exactly feasible nor exactly optimal, and is computed in floating-point arithmetic on data that is itself a floating-point rounding of the intended (rational/real) problem. To turn the solver output into a *mathematically rigorous* lower bound we use a certified weak-duality argument: we verify in directed-rounding interval arithmetic that a candidate dual point lies in the dual cone, and charge every rounding and data error to a penalty controlled by an a-priori bound on the primal solution norm.

C.1. Canonical primal and dual

Our certified-duality method relies on the canonicalisation produced by `CVXPY` for the SCS standard form. After canonicalisation the primal is

$$\min_x \; c^\top x \quad \text{s.t.} \quad Ax + s = b, \;\; s \in K, \tag{24}$$

with the product cone $K = K_{\text{zero}} \times K_{\geq 0} \times K_{\text{SOC}} \times K_{\text{PSD}}$ in exactly that order. The Lagrangian dual is

$$\max_y \; -b^\top y \quad \text{s.t.} \quad A^\top y + c = 0, \;\; y \in K^*, \tag{25}$$

where K^* is the dual cone (here $K^* = K$ blockwise, except the zero cone dualises to the free cone). For any dual-feasible y, weak duality gives $-b^\top y \leq c^\top x^* = \Omega^*$, so a rigorously dual-feasible y certifies the lower bound $-b^\top y$.

C.2. Error model and the master certificate

The stored data (A, b, c) are floating-point; the intended data are $(\hat{A}, \hat{b}, \hat{c})$, with rigorous outward-rounded elementwise error bounds

$$\max_{ij} |A_{ij} - \hat{A}_{ij}| \leq \varepsilon_A, \qquad \max_i |b_i - \hat{b}_i| \leq \varepsilon_b, \qquad \max_j |c_j - \hat{c}_j| \leq \varepsilon_c.$$

We estimate $\varepsilon_A, \varepsilon_b$ and ε_c by using mpmath arbitrary-precision library and verify that for our problem sizes, $\epsilon_A, \epsilon_b, \epsilon_c \leq 1e-12$.

Now we show how we can get a certified lower bound.

Theorem C.1 (Certified lower bound)**.** *Let y be verified to lie in K^* (Algorithm 3). Let $r \geq \|A^\top y + c\|_\infty$ be an outward-rounded upper bound on the floating dual-feasibility residual and $X \geq \|x^*\|_1$ an a-priori bound (Section C.3). With*

$$\varepsilon_{\text{tot}} = r + \varepsilon_A \|y\|_1 + \varepsilon_c, \qquad \Pi = \varepsilon_{\text{tot}}\, X + \varepsilon_b\, \|y\|_1,$$

one has

$$\Omega^* \;\geq\; L \;:=\; (-b^\top y)_\downarrow - \Pi_\uparrow,$$

computed with directed rounding ($\downarrow$ down, $\uparrow$ up).

Proof. Let x^* be any optimiser of the intended primal, so $\hat{A}x^* + \hat{s} = \hat{b}$ with $\hat{s} \in K$, and $\Omega^* = \hat{c}^\top x^*$. Since $y \in K^*$ and $\hat{s} \in K$, $\langle y, \hat{s} \rangle \geq 0$, hence

$$\hat{c}^\top x^* \geq \hat{c}^\top x^* - \langle y, \hat{s} \rangle = \hat{c}^\top x^* - \left\langle y, \hat{b} - \hat{A}x^* \right\rangle = -\hat{b}^\top y + (\hat{c} + \hat{A}^\top y)^\top x^*.$$

Now replace intended data by floating data plus a bounded perturbation. Writing $\hat{A} = A + \Delta_A$, $\hat{b} = b + \Delta_b$, $\hat{c} = c + \Delta_c$ with the stated elementwise bounds,

$$-\hat{b}^\top y = -b^\top y - \Delta_b^\top y \geq -b^\top y - \varepsilon_b \|y\|_1,$$

and

$$(\hat{c} + \hat{A}^\top y)^\top x^* = (c + A^\top y)^\top x^* + (\Delta_c + \Delta_A^\top y)^\top x^* \geq -\big(\|c + A^\top y\|_\infty + \varepsilon_A \|y\|_1 + \varepsilon_c\big)\|x^*\|_1,$$

using Hölder's inequality $|w^\top x^*| \leq \|w\|_\infty \|x^*\|_1$, the residual bound $\|c + A^\top y\|_\infty \leq r$, and the elementwise bounds $\|\Delta_A^\top y\|_\infty \leq \varepsilon_A \|y\|_1$, $\|\Delta_c\|_\infty \leq \varepsilon_c$. Combining and using $\|x^*\|_1 \leq X$,

$$\Omega^* = \hat{c}^\top x^* \geq -b^\top y - \big(r + \varepsilon_A \|y\|_1 + \varepsilon_c\big)X - \varepsilon_b \|y\|_1 = -b^\top y - \Pi.$$

Evaluating $-b^\top y$ with downward rounding and Π with upward rounding only decreases the right-hand side, so $\Omega^* \geq L$. □

C.3. A-priori bound on $\|x^*\|_1$

The constant X is computed in the *CVXPY-canonicalised* coordinates, block by block, with all arithmetic outward-rounded in interval form; the routines return the upper endpoint.

Theorem C.2 (Primal bound, $C_{6.5}$)**.** *Let $L = 2/N$ and let ε be the equality tolerance. Every primal-feasible point x of the overlap program satisfies $\|x\|_1 \leq X_{6.5}$, where $X_{6.5}$ is the outward-rounded sum of the block bounds established in the proof.*

Proof. We bound each block from its governing constraint.

Objective. The objective variable contributes $\Omega \leq 1$, since $\Omega \leq 1$ is imposed directly (an average overlap cannot exceed 1).

Mass. From the normalization $L \sum_j (w_j + v_j) \leq 1 + \varepsilon$ and $w_j, v_j \geq 0$,

$$\|w\|_1 + \|v\|_1 = \sum_j (w_j + v_j) \leq \frac{1+\varepsilon}{L} = \frac{N(1+\varepsilon)}{2}.$$

Fourier coefficients. The Parseval constraint gives $|c_k|, |d_k| \leq 2/\pi$, and there are T of each, so $\|c\|_1 + \|d\|_1 \leq \frac{2}{\pi} \cdot 2T = \frac{4T}{\pi}$.

Derived moments. The coefficients a_m, b_m are entries of the positive-semidefinite moment matrix associated with f: a_m and b_m are the real and imaginary parts of an off-diagonal entry $(T^f)_{ij}$ at lag $|i-j| = m$. For a PSD matrix the off-diagonal entries are controlled by the diagonal,

$$|(T^f)_{ij}| \;\leq\; \sqrt{(T^f)_{ii}\,(T^f)_{jj}} \;\leq\; \tfrac{1}{2}\big((T^f)_{ii} + (T^f)_{jj}\big),$$

and each diagonal entry is a zeroth moment of a measure of total mass at most 2, so $(T^f)_{ii} \leq 2$ and therefore $|a_m|, |b_m| \leq 2$. With at most $2R$ indices for each of a and b,

$$\|a\|_1 \leq 2 \cdot 2R = 4R, \qquad \|b\|_1 \leq 4R, \qquad \text{hence} \qquad \|a\|_1 + \|b\|_1 \leq 8R.$$

Truncation slacks. The slacks are bounded in closed form by

$$|\varepsilon_{2m-1}| \leq \frac{2(2m-1)}{\pi\big(4 - ((2m-1)/T)^2\big)\sqrt{6T^3}}, \qquad |\delta_{2m-1}| \leq \frac{4}{\pi\big(4 - ((2m-1)/T)^2\big)\sqrt{2T}},$$

for $1 \leq m \leq R$ (the hypothesis $R \leq T$ keeps each denominator positive); their ℓ_1 contribution is the sum of these closed forms, evaluated in interval arithmetic.

Canonicalization auxiliaries. Conic canonicalization introduces $4R$ square epigraph variables, each bounded by 4 since $a_m, b_m \in [-2, 2]$; two sum-of-squares epigraphs, each bounded by $4T/\pi^2$ from the Parseval budget; and $4R + 1$ absolute-value slacks, each bounded by the tolerance ε.

Summing all contributions and rounding outward gives $X_{6.5} \geq \|x^\star\|_1$. □

Theorem C.3 (Primal bound, $C_{6.2}$)**.** *Let $\omega = \Omega_{\text{ub}} - 1$ with $\Omega_{\text{ub}} = 2$. Every primal-feasible point x of the autocorrelation program satisfies $\|x\|_1 \leq X_{6.2}$, where $X_{6.2}$ is the outward-rounded sum of the block bounds established in the proof.*

Proof. The cap $\Omega \le \Omega_{\mathrm{ub}} = 2$ holds because the program's optimum lies below the best known upper bound on $C_{6.2}$, so $\omega = \Omega_{\mathrm{ub}} - 1 \ge \Omega - 1 \ge 0$. We bound each variable block in turn.

Objective and mass. The objective variable satisfies $\Omega \le \Omega_{\mathrm{ub}} = 2$. The simplex constraint $\sum_j p_j = 1$ with $p_j \ge 0$ gives $\|p\|_1 = 1$.

Moments. The envelope constraints $\gamma_k^- \cdot p \le a_k \le \gamma_k^+ \cdot p$ and the bound $|\gamma_k^\pm| \le 1$ (cosine enclosures) give $|a_k| \le \sum_j p_j = 1$ for each of the $K+2$ indices $k = 0, \dots, K+1$, so $\|a\|_1 \le K+2$; likewise $\|b\|_1 \le K+1$.

Moment matrix. We first bound the trace. The energy constraint $\Omega \ge 1 + 2\sum_{k=1}^K w_k v_k$ together with the cap gives

$$\sum_{k=1}^K w_k v_k \;\le\; \tfrac{1}{2}(\Omega - 1) \;\le\; \tfrac{\omega}{2}.$$

The slack constraint $v_k \ge (M_{kk} + M_{K+k,K+k})^2$ and the nonnegativity of the diagonal of the PSD matrix M give $M_{kk} + M_{K+k,K+k} \le \sqrt{v_k}$, so by Cauchy–Schwarz,

$$\sum_{k=1}^K \big(M_{kk} + M_{K+k,K+k}\big) \;\le\; \sum_{k=1}^K \sqrt{v_k} \;\le\; \Big(\sum_{k=1}^K \tfrac{1}{w_k}\Big)^{1/2} \Big(\sum_{k=1}^K w_k v_k\Big)^{1/2} \;\le\; \Big(\sum_{k=1}^K \tfrac{1}{w_k}\Big)^{1/2} \sqrt{\tfrac{\omega}{2}}.$$

The Fejér weights are $w_k = \frac{K+1-k}{K+1}$, so $\frac{1}{w_k} = \frac{K+1}{K+1-k} \le K+1$ for $1 \le k \le K$, whence $\sum_{k=1}^K \frac{1}{w_k} \le K(K+1)$. With $M_{00} = 1$,

$$\operatorname{tr} M \;=\; 1 + \sum_{k=1}^K \big(M_{kk} + M_{K+k,K+k}\big) \;\le\; 1 + \sqrt{\tfrac{K(K+1)\,\omega}{2}}.$$

For the vectorized block, $M \succeq 0$ gives $|M_{ij}| \le \sqrt{M_{ii}M_{jj}} \le \frac{1}{2}(M_{ii} + M_{jj})$. The symmetric vectorization has entries M_{ii} on the diagonal and $\sqrt{2}\, M_{ij}$ off-diagonal, so each diagonal entry M_{ii} enters the off-diagonal sum with total weight $\sqrt{2} \cdot \frac{1}{2} \cdot$ (number of partners) $\le \frac{\sqrt{2}}{2}\,(2K)$. Hence

$$\|\operatorname{svec}(M)\|_1 = \sum_i M_{ii} + \sqrt{2}\sum_{i<j} |M_{ij}| \le \big(1 + \sqrt{2}\,K\big) \operatorname{tr} M.$$

Slacks and epigraph. From $\sum_k w_k v_k \le \omega/2$ and $w_k \le 1$ we obtain $\|v\|_1 = \sum_k v_k$; using $w_k \ge w_K = \frac{1}{K+1}$ this is bounded by $(K+1)\sum_k w_k v_k \le (K+1)\omega/2$. The epigraph auxiliary cannot exceed $\|v\|_1$ by its defining inequality.

Hermitian block. The constraint $\operatorname{tr} Q = \Omega - 1 = \omega$ together with $Q \succeq 0$ gives $\sum_i Q_{ii} = \omega$, and the same PSD/vectorization argument as for M yields $\|\operatorname{Herm}(Q)\|_1 \le \big(1 + \frac{\sqrt{2}\,K}{2}\big)\omega$, the $\sqrt{2}$ again being the off-diagonal scaling of the vectorization.

Summing the block bounds - Ω_{ub}, 1, $K+2$, $K+1$, $(1+\sqrt{2}K)\operatorname{tr} M$ with $\operatorname{tr} M \le 1 + \sqrt{K(K+1)\omega/2}$, $(K+1)\omega/2$ (twice, for v and the epigraph), and $(1 + \frac{\sqrt{2}K}{2})\omega$ - each rounded outward, gives $X_{6.2} \ge \|x^\star\|_1$. □

C.4. Rigorous certificate of dual-cone containment

It remains to verify $y \in K^*$. We walk the product cone in SCS order using directed-rounding interval arithmetic (the `mpmath` interval type `iv`). The nonnegative and second-order cones are self-dual; the PSD cone is self-dual and is certified by a residual-plus-spectral-bound test: a floating Cholesky factor L furnishes a rigorous lower bound on $\lambda_{\min}(LL^\top)$ through the exact inverse $B = L^{-1}$, and the discrepancy between $LL^\top$ and the block is absorbed by a rigorous residual bound, the two combined by Weyl's inequality.

D. Prompts used for the agent

The pipeline is driven by three role prompts: a problem specification (one per inequality), a proposal agent (the coding agent that proposes and implements constraints), and a verifier agent (the theory agent that adjudicates each proof). The texts below are reproduced verbatim.

Algorithm 3 Rigorous dual-cone containment

Require: candidate dual y; cone dimensions $(\dim_{\text{zero}}, \dim_{\geq 0}, \{q_i\}, \{n_j\})$; PSD shift $\sigma > 0$; precision `dps`
Ensure: OK (all blocks certified) and a rigorous margin per block

Zero cone: the dual is free; skip.
Nonnegative cone: verify $\min_i y_i \geq 0$ by floating comparison (the float value is itself the witness); margin $\leftarrow \min_i y_i$.
Second-order cones $y = (t, z)$: interval-enclose $\|z\|_2 = \sqrt{\sum_k z_k^2}$ and verify $(t - \|z\|_2)_\downarrow > 0$; margin $\leftarrow (t - \|z\|_2)_\downarrow$.
PSD cones $(n_j \times n_j)$: reshape the SCS svec block to a symmetric Y (off-diagonals divided by $\sqrt{2}$), then:
  (a) attempt a floating Cholesky of $A_{\text{mid}} = \frac{1}{2}(Y + Y^\top) - \sigma I$, retaining the computed lower factor L; return FAIL if it does not complete;
  (b) form the exact interval inverse $B = L^{-1}$ by triangular forward substitution and $C = B^\top B = (LL^\top)^{-1}$; bound $\lambda_{\max}(C) \leq \lambda_{\max}^+$ by Gershgorin, giving the rigorous bound $\lambda_{\min}(LL^\top) \geq 1/\lambda_{\max}^+$;
  (c) rigorously enclose the residual $E = (A - \sigma I) - LL^\top$ in Frobenius norm over the interval $A_{ij} \in [Y_{ij} - \rho, Y_{ij} + \rho]$, obtaining $\|E\|_F^+ \geq \|E\|_2$;
  (d) certify with margin $m = \sigma + 1/\lambda_{\max}^+ - \|E\|_F^+$, a rigorous lower bound on $\lambda_{\min}(A)$; the block passes iff $m > 0$.
return OK = (all blocks certified), with the per-block margins.

`proposal_6.2.md` (coding agent, inequality 6.2)

```
Hi Claude, our mission is to find rigorous advancements of a series of
    Autocorrelation inequalities that has been proposed. Imagine you are an expert
    in functional analysis, optimization, Fourier analysis, and combinatorics.

Your job is to propose a convex-optimization like formulation that may be motivated
    from the form proposed in White's paper (the paper directory is in
    2201.05704v1.pdf) and constantly improve it. Basically you are going to show a
    lower-bound or an upper-bound that is achieved by relaxing the complicated
    problem into a decomposed set of constrained convex problems, then optimize
    each problem. So your main job is to find novel constraints that were
    overlooked by previous versions of the program - and use it to optimize. See
    2201.05704v1.pdf for an insight: Lemma 3,4,5 is used to construct a convex
    optimiation problem in section 5.

The problem that you need to solve is specified in Problem.md. Also in Problem.md
    there are tips for the problem. Note that we are trying to tackle these
    problems in a similar approach - but we are not trying to solve only the erdos
    minimum overlap problem.

Your workflow will be the following:

0. This is the initial step. See if you have folder v0. If you do, then run the
    code as a baseline to check that it works. If there is no v0, you can come up
    with the most naive version of the convex optimization problem that can be
    meaningful.

After this you have an iterative loop.

1. Read the most recent version folder. Then, create {nextversion} folder and copy
    the contents from the previous folder.
2. Come up with a novel proposal that improves upon the previous implementation.
    The proposal should not be something like a hyperparameter tweak; we will
    enhance the hyperparameters after the convex program has converged.
3. After you come up with a proposal, you have to generate a "rigorous" proof that
    the constraint indeed holds. For an example see Lemma 3,4,5 in the convex
    optimization problem of White and how it is related to the convex optimization
    problem in section 5.
4. Store the proof in the current version folder. Name it as rigorousproof.md
```

```
5. Now this is the important: you should be idle until the theorist agent reads
    your proof and gives a verdict. It will be in the same folder, verdict.md. If
    the verdict is VALID, it means your proposed proof is rigorous and makes sense.
    If the verdict is INVALID, it means your proposed proof has a flaw.
6. IF the verdict was VALID, implement the constraint in the convex problem.
7. IF the verdict was INVALID, go to 2 and propose a different constraint that
    makes mathematical sense.

*** THINGS TO CONSIDER ***
- A validation should take less than 10 minutes. Don't heavily optimize the
    parameter space (split the original problem into too many different convex
    optimization problems) or increase the problem parameters too much. Our job is
    NOT to improve the bound for now, but to improve the PROGRAM that will
    eventually lead to better bounds.

- MATHEMATICAL RIGOR IS THE MOST IMPORTANT!!! Do not simply assume that you are
    doing an approximation that is good enough or do not propose things that may or
    may not be a mathematically valid lower or upper bound.

- NEVER STOP: Once the experiment loop has begun (after the initial setup), do NOT
    pause to ask the human if you should continue. Do NOT ask "should I keep
    going?" or "is this a good stopping point?". The human might be asleep, or gone
    from a computer and expects you to continue working indefinitely until you are
    manually stopped. You are autonomous. If you run out of ideas, think harder -
    read papers referenced in the code, re-read the in-scope files for new angles,
    try combining previous near-misses, try more radical architectural changes. The
    loop runs until the human interrupts you, period.

- Especially, I will not initiate a new interaction when the verdict.md is given.
    So when I say idle I mean don't stop the current session and wait.
```

`Problem_6.2.md` (problem description, inequality 6.2)

```
We want a lower bound on the sharp constant (C_{6.2}) in the inequality [
    \max_{-1/2\le t\le 1/2}\int_{\mathbb R} f(t-x)f(x),dx ;\ge;
    C_{6.2}\left(\int_{-1/4}^{1/4} f(x),dx\right)^2 ] for all nonnegative functions
    (f:\mathbb R\to\mathbb R).

Equivalently, (C_{6.2}) is the infimum of [ \frac{\displaystyle \max_{-1/2\le t\le
    1/2}\int_{\mathbb R} f(t-x)f(x),dx} {\displaystyle \left(\int_{-1/4}^{1/4}
    f(x),dx\right)^2} ] over all nonnegative (f) for which the denominator is
    nonzero.

The numerator is a localized self-convolution: [ (ff)(t)=\int_{\mathbb R}
    f(t-x)f(x),dx, ] so the problem asks for the smallest possible value of [
    \max_{|t|\le 1/2}(ff)(t) ] relative to the square of the mass of (f) on
    ([-1/4,1/4]).

To prove a valid lower bound (L) for (C_{6.2}), we need to show that every
    admissible nonnegative (f) satisfies [ \max_{|t|\le 1/2}(f*f)(t) ;\ge;
    L\left(\int_{-1/4}^{1/4} f(x),dx\right)^2. ]

A natural way to do this is to formulate a constrained optimization problem. The
    idea is to replace the original infinite-dimensional problem by a tractable
    problem whose feasible set contains all objects arising from admissible (f),
    and whose constraints encode necessary properties that every such (f) must
    satisfy.

This makes sense because of the relaxation principle:
```

```
every genuine nonnegative (f) gives a feasible point of the constrained problem,
the constrained problem is designed to be easier to analyze or solve,
and minimizing over a larger feasible set can only decrease the value.
Therefore, if the constrained optimization problem has optimum
    (L_{\mathrm{relax}}), then automatically [ L_{\mathrm{relax}} \le C_{6.2}. ] So
    (L_{\mathrm{relax}}) is a valid lower bound for the original sharp constant.

In other words, to obtain a rigorous lower bound, we should construct a tractable
    optimization problem whose feasible set contains every admissible (f), and
    which encodes necessary conditions such as

[ f\ge 0, ] together with whatever additional structure we can prove is relevant
    for the quantities [ \max_{|t|\le 1/2}(f*f)(t) \qquad\text{and}\qquad
    \int_{-1/4}^{1/4} f(x),dx. ]

For example, one may introduce auxiliary variables representing local mass,
    convolution values, Fourier coefficients, or moment data, and then impose
    constraints that every true (f) must satisfy. Solving the resulting constrained
    optimization problem then produces a mathematically valid lower bound for
    (C_{6.2}).

So the overall strategy is:

rewrite (C_{6.2}) as an infimum over nonnegative (f),
derive necessary constraints satisfied by every admissible (f),
build a constrained optimization problem from those necessary conditions,
minimize the target quantity over that relaxed feasible set,
use the resulting optimum as a provable lower bound for (C_{6.2}).
The current SOTA lower bound is 1.28.
```

`proposal_erdos.md` (coding agent, Erdős; differs only in the hard-square tip)

```
Hi Claude, our mission is to find rigorous advancements of a series of
    Autocorrelation inequalities that has been proposed. Imagine you are an expert
    in functional analysis, optimization, Fourier analysis, and combinatorics.

Your job is to propose a convex-optimization like formulation that may be motivated
    from the form proposed in White's paper (the paper directory is in
    2201.05704v1.pdf) and constantly improve it. Basically you are going to show a
    lower-bound or an upper-bound that is achieved by relaxing the complicated
    problem into a decomposed set of constrained convex problems, then optimize
    each problem. So your main job is to find novel constraints that were
    overlooked by previous versions of the program - and use it to optimize. See
    2201.05704v1.pdf for an insight: Lemma 3,4,5 is used to construct a convex
    optimiation problem in section 5.

The problem that you need to solve is specified in Problem.md. Also in Problem.md
    there are tips for the problem. Note that we are trying to tackle these
    problems in a similar approach - but we are not trying to solve only the erdos
    minimum overlap problem.

Your workflow will be the following:

0. This is the initial step. See if you have folder v0. If you do, then run the
    code as a baseline to check that it works. If there is no v0, you can come up
    with the most naive version of the convex optimization problem that can be
    meaningful.
```

```
After this you have an iterative loop.

1. Read the most recent version folder. Then, create {nextversion} folder and copy
    the contents from the previous folder.
2. Come up with a novel proposal that improves upon the previous implementation.
    The proposal should not be something like a hyperparameter tweak; we will
    enhance the hyperparameters after the convex program has converged.
3. After you come up with a proposal, you have to generate a "rigorous" proof that
    the constraint indeed holds. For an example see Lemma 3,4,5 in the convex
    optimization problem of White and how it is related to the convex optimization
    problem in section 5.
4. Store the proof in the current version folder. Name it as rigorousproof.md
5. Now this is the important: you should be idle until the theorist agent reads
    your proof and gives a verdict. It will be in the same folder, verdict.md. If
    the verdict is VALID, it means your proposed proof is rigorous and makes sense.
    If the verdict is INVALID, it means your proposed proof has a flaw.
6. IF the verdict was VALID, implement the constraint in the convex problem.
7. IF the verdict was INVALID, go to 2 and propose a different constraint that
    makes mathematical sense.

*** THINGS TO CONSIDER ***
- A validation should take less than 10 minutes. Don't heavily optimize the
    parameter space (split the original problem into too many different convex
    optimization problems) or increase the problem parameters too much. Our job is
    NOT to improve the bound for now, but to improve the PROGRAM that will
    eventually lead to better bounds. For this problem specifically, choose a hard
    square region h_1 = h_2 = 0.015, p_1 = p_2 = 0.381, -q_1=q_2=0.02 to verify and
    see if the bound is improving.

- MATHEMATICAL RIGOR IS THE MOST IMPORTANT!!! Do not simply assume that you are
    doing an approximation that is good enough or do not propose things that may or
    may not be a mathematically valid lower or upper bound.

- NEVER STOP: Once the experiment loop has begun (after the initial setup), do NOT
    pause to ask the human if you should continue. Do NOT ask "should I keep
    going?" or "is this a good stopping point?". The human might be asleep, or gone
    from a computer and expects you to continue working indefinitely until you are
    manually stopped. You are autonomous. If you run out of ideas, think harder -
    read papers referenced in the code, re-read the in-scope files for new angles,
    try combining previous near-misses, try more radical architectural changes. The
    loop runs until the human interrupts you, period.

- Especially, I will not initiate a new interaction when the verdict.md is given.
    So when I say idle I mean don't stop the current session and wait.
```

`Problem_erdos.md` (problem description, Erdős minimum overlap problem)

```
We want a lower bound on [ \inf_f; \sup_{x\in[-2,2]} \int_{-1}^1 f(t), g(x+t),dt, ]
    under the assumptions [ f,g:[-1,1]\to[0,1],\qquad g=1-f \text{ on }[-1,1]. ]

Since (g) is determined by (f) on ([-1,1]), this is really an optimization over a
    single function (f). The quantity inside, [ \int_{-1}^1 f(t),g(x+t),dt, ]
    measures the overlap between (f) and a shifted copy of its complement (g). For
    each shift (x\in[-2,2]), it gives the overlap at that displacement, and then
    the supremum over (x) picks the worst-case overlap. The infimum over (f) asks
    for the smallest possible such worst-case overlap.

To prove a valid lower bound (L), we need to show that every admissible (f)
    satisfies [ \sup_{x\in[-2,2]} \int_{-1}^1 f(t),g(x+t),dt \ge L. ]
```

```
A natural way to do this is to replace the original infinite-dimensional problem by
    a constrained optimization problem. The idea is to impose a collection of
    constraints that every genuine admissible pair ((f,g)) must satisfy, and then
    optimize over all objects satisfying those constraints.

This is useful because of the relaxation principle:

every admissible (f) gives a feasible point of the constrained problem,
the constrained problem is designed to be easier to analyze or solve,
and minimizing over a larger feasible set can only decrease the value.
Therefore, if the constrained optimization problem has optimum
    (L_{\mathrm{relax}}), then automatically [ L_{\mathrm{relax}} \le \inf_f;
    \sup_{x\in[-2,2]} \int_{-1}^1 f(t), g(x+t),dt. ] So (L_{\mathrm{relax}}) is a
    valid lower bound for the original problem.

In other words, to obtain a rigorous lower bound, we should construct a tractable
    optimization problem whose feasible set contains all admissible (f) and encodes
    necessary conditions such as [ 0\le f\le 1,\qquad 0\le g\le 1,\qquad f+g=1
    \text{ on }[-1,1], ] together with any additional structural constraints we can
    prove. Solving that constrained problem then gives a mathematically valid lower
    bound for the original inf-sup quantity.

So the overall strategy is:

start from the exact inf-sup problem,
derive necessary constraints satisfied by every admissible (f),
build a constrained optimization problem from those necessary conditions,
use its optimum as a provable lower bound.
The current SOTA is 0.379005.
```

`verifier.md` (theory agent)

```
Hi Claude, our mission is to verify proposals of a series of Autocorrelation
    inequalities that has been proposed. Imagine you are an expert in functional
    analysis, optimization, Fourier analysis, and combinatorics. Also imagine you
    are extra critical of other people's proposals and you want them to be crystal
    clear and very rigorous.

At each problem folder, you will have a subfolder named {versionname}. At each most
    recent {versionname}, if the versionname is bigger than 0, you will see a
    rigorousproof.md file.

Your job is to verify that the rigorousproof.md file is indeed rigorous and you can
    trust the proof.

I want you to first review the proof, and write a brief summary of the proof.

Then see if there are any counterexamples for each line of the proof. If there is a
    counterexample, you should report the counterexample in the same {versionname}
    folder, in verdict.md file.

If there is no issue in the proof, in verdict.md after you summarize and noting
    some comments, give the final verdict as VALID.

If you found a logical fallacy or a counterexample, give the final verdict as
    INVALID.
```

```
Also this is optional: you can explain whether the approach was effective or not,
    and if you have a big picture you can guide the proposal agent as a feedback
    too.

*** THINGS TO CONSIDER ***

- MATHEMATICAL RIGOR IS THE MOST IMPORTANT!!! Do not simply assume that you are
    doing an approximation that is good enough or do not propose things that may or
    may not be a mathematically valid lower or upper bound.

- Monitor every 5 minutes

- NEVER STOP: Once the experiment loop has begun (after the initial setup), do NOT
    pause to ask the human if you should continue. Do NOT ask "should I keep
    going?" or "is this a good stopping point?". The human might be asleep, or gone
    from a computer and expects you to continue working indefinitely until you are
    manually stopped. You are autonomous. If you run out of ideas, think harder -
    read papers referenced in the code, re-read the in-scope files for new angles,
    try combining previous near-misses, try more radical architectural changes. The
    loop runs until the human interrupts you, period.
```

E. A comprehensive list of branch and bound for the Erdős minimum overlap problem

Table 3. Certified lower bounds on Ω for the twelve single-cell paper rows (A1–A5, A8, A10–A15), each verified at $(N, T, R) = (10000, 4000, 10)$ via dual certification.

h_1	h_2	p_1	p_2	q_1	q_2	Ω
0.75	2.00	0.00	1.00	-1.000	1.000	0.461948
0.40	0.75	0.00	1.00	-1.000	1.000	0.409417
0.20	0.40	0.00	1.00	-1.000	1.000	0.398605
0.10	0.20	0.00	1.00	-1.000	1.000	0.385800
0.08	0.10	0.00	1.00	-1.000	1.000	0.385068
0.00	0.08	0.00	1.00	0.050	1.000	0.381533
0.00	0.08	0.00	0.25	-0.025	0.025	0.398928
0.00	0.08	0.25	0.30	-0.025	0.025	0.389547
0.00	0.08	0.30	0.33	-0.025	0.025	0.383319
0.00	0.08	0.50	1.00	-0.025	0.025	0.397131
0.00	0.08	0.45	0.50	-0.025	0.025	0.386126
0.06	0.08	0.33	0.45	-0.025	0.025	0.390070

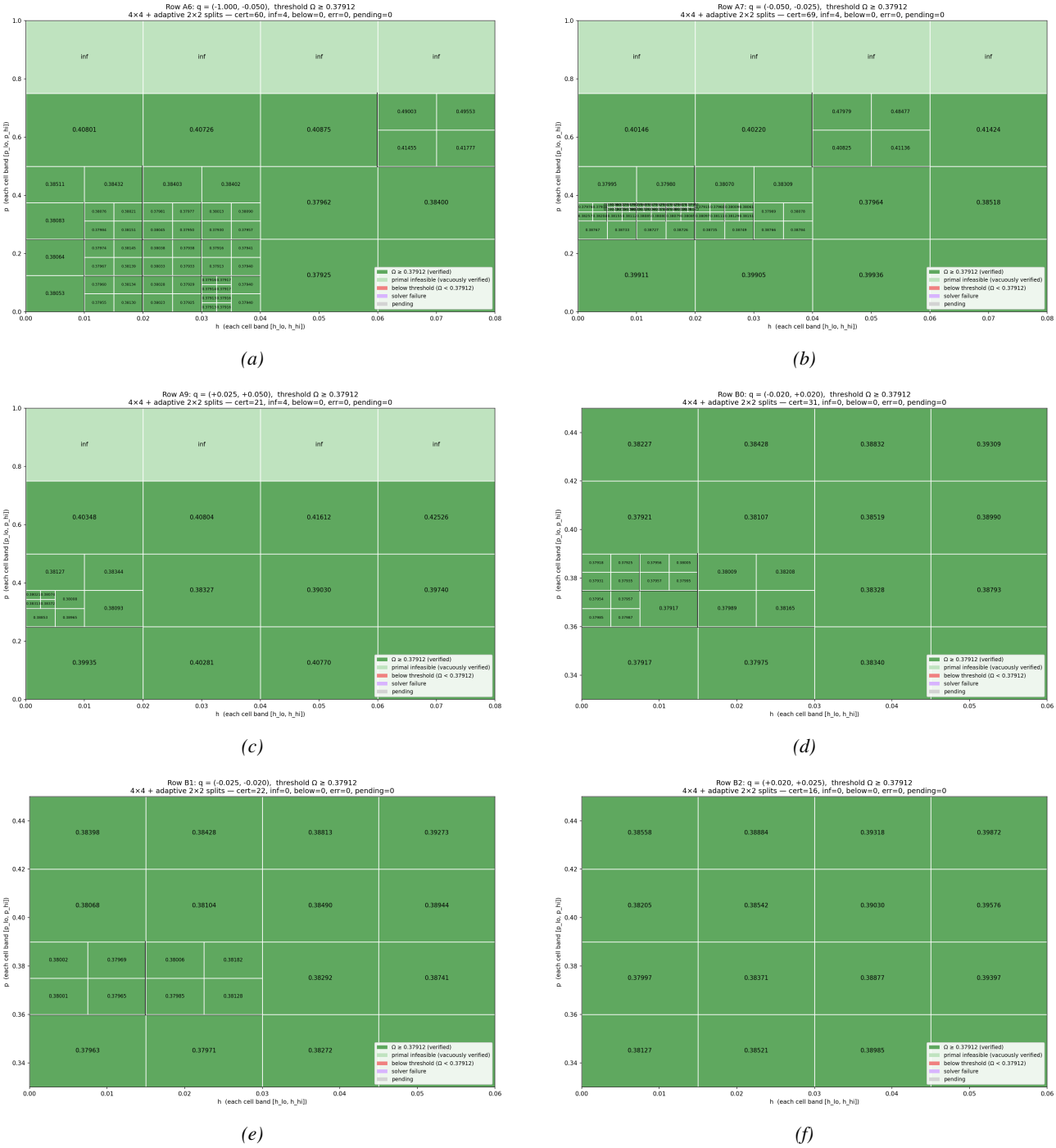

Figure 4. Splitting (h, p) for lower bound verification.

F. Interaction Between the Theory and Coding agent

`rigorousproof.md v7 [snippet selected]` (coding agent)

Goal of v7

The goal of v7 is to produce a **fully rigorous** convex relaxation whose optimum lower-bounds $C_{6.2}^{\text{even}}$, the even-restricted version of $C_{6.2}$:

$$C_{6.2}^{\text{even}} := \inf_{f \text{ admissible AND } f \text{ even}} \|f * f\|_\infty.$$

We address every load-bearing issue raised in the verdicts of v1, v2, v3, v4, v5: the false v0 band-integral identity is **dropped**, the WLOG reductions are **explicitly justified**, and the (W) and (G) constraints are derived without any "approximation" hand-wave.

The relaxation does **not** bound $C_{6.2}$ in general (we have $C_{6.2} \leq C_{6.2}^{\text{even}}$, so a lower bound on the even-restricted constant is automatically a lower bound on the even-restricted constant only). v8 will attempt the harder extension to non-even f.

Figure 5. v7 rigorous proof.md, evidence of even function limitation

`verdict.md v7 [snippet selected]` (theory agent)

Comment 2. v7 still bounds the *even* constant only.

The proposer is explicit that (C^{\text{even}}*{6.2} |ge C{6.2}), so a bound on the even constant is *not* automatically a bound on (C_{6.2}). The v7 bound of 1.344 at (N = 20) is well below the SOTA (C_{6.2}\ge 1.28) (the SOTA holds for the unrestricted constant, and a fortiori for the even constant), so v7 does **not** improve the project's main numerical target.

What v7 *does* establish is that the rigorous LP machinery (centered grid + (W) family + (G) symmetry) produces a non-trivial bound on the even-restricted problem. v8's job is to remove the even restriction.

Figure 6. v7 verdict.md sampled, theory agent correction

`rigorousproof.md v14 [snippet selected]` (coding agent)

Goal

v14 produces the **first rigorous lower bound on $C_{6.2}$ in this project that does not require an even-f restriction**. The construction follows the v9 verdict §5 recommendation: use the Fejér-kernel test functional applied to the **autocorrelation** $F * \tilde{F}$ where $\tilde{F}(t) = F(-t)$, instead of F itself. The autocorrelation has manifestly nonneg Fourier-side data, which side-steps the M[K+k, K+k] inflation degeneracy that defeated v10–v13.

Figure 7. v14 rigorous proof.md, evidence of correction