Structure-Exploiting Interior-Point Methods for Factor-Risk and Scenario-Based Portfolio Optimisation

QUARCC
Quantitative Research Note
November 2026

Abstract. Portfolio optimisation problems have structure that general-purpose conic modelling discards. We study two interior-point kernels that exploit it. For factor-risk quadratic programs, slack elimination and the Woodbury identity reduce per-iteration linear algebra from $O(n^3)$ to $O(nr^2 + r^3)$. For scenario-based linear and quadratic programs, including CVaR, worst-realisation, mean-absolute-deviation, and drawdown models, per-scenario auxiliaries are eliminated exactly, leaving a small dense Schur system. Constraint generation solves CVaR and worst-realisation models on a subset of scenarios and returns the optimal value of the full problem. Screening exactness and a suboptimality bound valid at any iterate are machine-checked in Lean 4. In single-thread benchmarks on synthetic data, a 1,000-security factor-risk program solves in 23.6 ms, against 1,165.3 ms with CVXPY and Clarabel and 87.9 ms with a hand-built sparse Clarabel model. At 360,000 scenarios, CVaR models solve 12–17× faster than Riskfolio-Lib with weights agreeing to 1.1×10^{-8}. Speedups depend on data and model: they fall to 1.2–5.2× on drawdown measures. Along a 20-point CVaR frontier at 360,000 scenarios, 2 of 19 risk-capped problems stall and return no solution.

Factor-Risk Quadratic Programs

Cross-sectional equity strategies solve the same problem many times, across gross targets, support sizes, and formation dates. The kernel is a convex QP in weights $\mathbf{x} \in \mathbb{R}^n$:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2}\mathbf{x}^\top \mathbf{P}\mathbf{x} + \mathbf{q}^\top \mathbf{x} + c_0 \\ \text{s.t.} \quad & \boldsymbol{\ell} \le \mathbf{x} \le \mathbf{u}, \\ & \mathbf{g}_\ell \le \mathbf{G}\mathbf{x} \le \mathbf{g}_u, \quad \mathbf{A}\mathbf{x} = \mathbf{b}. \end{aligned} \tag{1}$$

Box bounds encode position caps, liquidity limits, and fixed signs, $\mathbf{G}$ holds exposure bands, and $\mathbf{A}$ holds equalities such as gross leverage for a fixed sign vector. Bands with $\mathbf{g}_\ell = \mathbf{g}_u$ have no strict interior, so they are moved to the equality block, and linearly dependent rows are removed. For exposures $\mathbf{B} \in \mathbb{R}^{n \times k}$, factor covariance $\mathbf{F} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$, and specific variances $\mathbf{d}$, risk is $\boldsymbol{\Sigma} = \mathbf{B}\mathbf{F}\mathbf{B}^\top + \operatorname{diag}(\mathbf{d})$. With risk aversion γ and quadratic trading-cost weights $\boldsymbol{\tau}_2$, the Hessian stays factored:

$$\mathbf{P} = \mathbf{U}\operatorname{diag}(\mathbf{c})\,\mathbf{U}^\top + \operatorname{diag}(\mathbf{p}_d), \tag{2}$$

where $\mathbf{U} = \mathbf{B}\mathbf{V}\boldsymbol{\Lambda}^{1/2}$, $\mathbf{c} = \gamma\mathbf{1}_k$, and $\mathbf{p}_d = \gamma\mathbf{d} + \boldsymbol{\tau}_2$. Penalties on linear aggregates preserve this form: $\frac{1}{2}\lambda_j(\mathbf{v}_j^\top \mathbf{x} - \beta_{0,j})^2$ appends $\mathbf{v}_j$ to $\mathbf{U}$ and λ_j to $\mathbf{c}$, and shifts $\mathbf{q}$ and c_0.

Woodbury normal equations. We use a primal–dual predictor–corrector method (Mehrotra, 1992). With barrier weights $\mathbf{w}_\bullet = \mathbf{z}_\bullet \oslash \mathbf{s}_\bullet$, pivoting out slacks and inequality multipliers reduces each Newton step to

$$\mathbf{M}\,\Delta\mathbf{x} + \mathbf{A}^\top \Delta\mathbf{y} = \mathbf{r}_{\text{rhs}}, \qquad \mathbf{A}\,\Delta\mathbf{x} = -\mathbf{r}_e, \tag{3}$$

where

$$\mathbf{M} = \mathbf{H} + \mathbf{U}_{\text{aug}}\operatorname{diag}(\mathbf{c}_{\text{aug}})\,\mathbf{U}_{\text{aug}}^\top, \tag{4}$$

$\mathbf{H}$ is diagonal, $\mathbf{U}_{\text{aug}} = [\mathbf{U}, \mathbf{G}^\top]$ has rank at most $r = k + m$ for m bands, and $\mathbf{c}_{\text{aug}}$ collects $\mathbf{c}$ and the band barrier weights. By the Sherman–Morrison–Woodbury identity (Woodbury, 1950), applying $\mathbf{M}^{-1}$ needs only an $r \times r$ capacitance system:

$$\mathbf{K} = \operatorname{diag}(\mathbf{c}_{\text{aug}})^{-1} + \mathbf{U}_{\text{aug}}^\top \mathbf{H}^{-1}\mathbf{U}_{\text{aug}}, \tag{5}$$

$$\mathbf{M}^{-1}\mathbf{V} = \mathbf{H}^{-1}\mathbf{V} - \mathbf{H}^{-1}\mathbf{U}_{\text{aug}}\mathbf{K}^{-1}\mathbf{U}_{\text{aug}}^\top \mathbf{H}^{-1}\mathbf{V}. \tag{6}$$

Forming $\mathbf{K}$ costs $O(nr^2)$ and factorising it costs $O(r^3)$. Equalities add a $p \times p$ Schur complement $\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^\top$.

Outer loops. Across a rebalance ladder, $\mathbf{U}$, $\mathbf{G}$, $\mathbf{A}$, and the active-bound masks are invariant, so they are prepared once. A hard risk budget is enforced by bracketing and bisecting a shift in γ with warm starts. An L1 turnover penalty is handled without splitting variables by iteratively reweighted least squares, with diagonal weights $\tau_{1,i}/\sqrt{(q_i^{(t-1)})^2 + \epsilon^2}$ for previous trades $\mathbf{q}^{(t-1)}$, so each pass is a QP of unchanged dimension and rank.

Scenario-Based Linear and Quadratic Programs

Scenario-based risk measures share a different structure: a small dense block $\mathbf{x}$ of size m (weights and a few scalars) and a large sparse block $\mathbf{v}$ of per-scenario auxiliaries:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{v}} \quad & \frac{1}{2}\mathbf{x}^\top \mathbf{P}\mathbf{x} + \mathbf{c}_x^\top \mathbf{x} + \frac{1}{2}\mathbf{v}^\top \operatorname{diag}(\mathbf{q}_v)\mathbf{v} + \mathbf{c}_v^\top \mathbf{v} \\ \text{s.t.} \quad & \mathbf{G}_x\mathbf{x} + \mathbf{G}_v\mathbf{v} \le \mathbf{h}, \quad \mathbf{E}\mathbf{x} = \mathbf{e}. \end{aligned} \tag{7}$$

This form covers CVaR (Rockafellar and Uryasev, 2000), worst realisation (Young, 1998), mean absolute deviation (Konno and Yamazaki, 1991), semi-deviation, lower partial moments, and drawdown measures (Chekhlov, Uryasev, and Zabarankin, 2005), each with minimum-risk, utility, and maximum-return objectives. Maximum Sharpe ratio follows from the homogenisation of Charnes and Cooper (1962).

Exact elimination. With barrier weights $\mathbf{W}$ on the inequality rows, define

$$\mathbf{S}_{vv} = \operatorname{diag}(\mathbf{q}_v) + \mathbf{G}_v^\top \mathbf{W}\mathbf{G}_v, \quad \mathbf{B} = \mathbf{G}_v^\top \mathbf{W}\mathbf{G}_x. \tag{8}$$

Eliminating $\Delta\mathbf{v}$ leaves a dense system in $\Delta\mathbf{x}$ with matrix

$$\mathbf{P} + \mathbf{G}_x^\top \mathbf{W} \mathbf{G}_x - \mathbf{B}^\top \mathbf{S}_{vv}^{-1} \mathbf{B}. \tag{9}$$

Nothing of size $T \times T$ is formed, and an iteration costs $O(Tm^2 + m^3)$ when $\mathbf{S}_{vv}$ is diagonal. When each row touches one auxiliary, as in CVaR, the Schur complement collapses to a weighted Gram matrix with weights $\omega_i = w_i - (w_i g_i)^2 / (\mathbf{S}_{vv})_{jj} \geq 0$.

Drawdown auxiliaries form a chain, so $\mathbf{S}_{vv}$ is banded after reverse Cuthill–McKee ordering (Cuthill and McKee, 1969). With $\mathbf{S}_{vv} = \mathbf{L}\mathbf{L}^\top$ and $\mathbf{N} = \mathbf{L}^{-1}\mathbf{B}$, the last term of (9) is $\mathbf{N}^\top\mathbf{N}$, which needs one triangular sweep over m right-hand sides.

Risk caps. A risk cap touches many auxiliaries, and eliminating it directly would densify $\mathbf{S}_{vv}$. A low-rank update is numerically unreliable because the barrier weights grow without bound as the cap becomes active. These rows, with matrix $\mathbf{C}$ and weights $\mathbf{W}_c$, therefore enter through a bordered system:

$$\begin{bmatrix} \mathbf{H}_0 & \mathbf{C} \\ \mathbf{C}^\top & -\mathbf{W}_c^{-1} \end{bmatrix}. \tag{10}$$

Iteration. A Mehrotra (1992) starting point takes the affine step from a trivial point and shifts slacks and multipliers into the interior. Two multiple centrality correctors (Gondzio, 1996) reuse each factorisation, pushing complementarity products into $[0.1, 10]$ times the target, and are kept only while they lengthen the step. Primal and dual variables share one step length, scaled by 0.99.

Exact Scenario Screening

For CVaR with scenario set U, $|U| = T$, losses $\ell_i(\mathbf{x})$, and $c = 1/(\alpha T)$, the Rockafellar–Uryasev objective on $S \subseteq U$ is

$$F_S(\mathbf{x}, t) = t + c \sum_{i \in S} \max\big(0, \ell_i(\mathbf{x}) - t\big). \tag{11}$$

Proposition 1. Let $S \subseteq U$ and let $(\mathbf{x}, t)$ minimise F_S over a feasible set K. If $\ell_i(\mathbf{x}) \leq t$ for all $i \in U \setminus S$, then $(\mathbf{x}, t)$ minimises F_U over K.

Proof. The dropped terms are nonnegative, so $F_S \leq F_U$ on K. They vanish at $(\mathbf{x}, t)$ by hypothesis, so $F_U(\mathbf{x}, t) = F_S(\mathbf{x}, t) \leq F_S(\mathbf{y}, q) \leq F_U(\mathbf{y}, q)$ for every $(\mathbf{y}, q) \in K$. □

The same argument applies to worst realisation, where the restricted problem is a relaxation. The algorithm starts from the highest-loss scenarios of an equal-weight and an inverse-volatility portfolio, keeping $\max(\lceil 1.5\alpha T \rceil, 4n, 64)$ scenarios of each for CVaR and $\max(4n, 64)$ for worst realisation, solves the restricted problem, adds any violated scenarios with a margin of near-violations, and repeats. Violation is tested with a relative slack of 10^{-10}. Screening is applied for $T > 1500$. The optimal value is exact up to the solver tolerance and this slack, and weights are unique only on non-degenerate problems.

Termination and Suboptimality Certificate

A solve is labelled optimal only if the primal residual, the dual residual relative to the terms that cancel in it, and the relative gap are all at most 10^{-8}. If the primal residual and gap converge while the dual residual stalls between 10^{-8} and 10^{-5}, the status is optimal but inaccurate and weights are returned. Otherwise no weights are returned. In the factor-risk kernel, an iterate that is primal feasible but not converged is reported as not converged.

Proposition 2. Let f be convex with gradient $\mathbf{g}$ at any point $\mathbf{z}$, feasible or not, let $\boldsymbol{\lambda} \geq 0$ and $\mathbf{y}$ be multipliers for $\mathbf{G}\mathbf{z} \leq \mathbf{h}$ and $\mathbf{E}\mathbf{z} = \mathbf{e}$, and let $\mathbf{r}_d = \mathbf{g} + \mathbf{G}^\top\boldsymbol{\lambda} + \mathbf{E}^\top\mathbf{y}$. Then every feasible $\mathbf{z}^\star$ satisfies

$$\begin{aligned} f(\mathbf{z}) - f(\mathbf{z}^\star) \leq \boldsymbol{\lambda}^\top(\mathbf{h} - \mathbf{G}\mathbf{z}) + \mathbf{y}^\top(\mathbf{e} - \mathbf{E}\mathbf{z}) \\ + \|\mathbf{r}_d\|_\infty \|\mathbf{z} - \mathbf{z}^\star\|_1. \end{aligned} \tag{12}$$

The proof combines the first-order convexity inequality, feasibility of $\mathbf{z}^\star$, and Hölder's inequality. With an L1 diameter of the feasible set in place of the last norm, the right-hand side is computable at any iterate, so an early-stopped solve carries a rigorous bound.

Propositions 1 and 2, including the worst-realisation case, are machine-checked in Lean 4 (de Moura and Ullrich, 2021) with mathlib (The mathlib Community, 2020), relying only on the standard axioms. The correspondence between these statements and the numerical implementation is tested numerically and is not formally verified.

Empirical Results

All measurements use one thread pinned to one core of an Intel Core i5-7300U, with NumPy 2.5, SciPy 1.18, CVXPY 1.9, Clarabel 0.11, and Riskfolio-Lib 7.4 (Cajas, 2026). All data are synthetic. Times are medians of repeated runs, except single Riskfolio-Lib runs in Tables 2 and 3. Solutions were also compared with Riskfolio-Lib on each natively supported risk measure under minimum-risk, maximum-Sharpe, and utility objectives, and in each case the objective is within a relative 10^{-6} of Riskfolio-Lib's or better.

Factor-risk programs. Table 1 separates two effects at $n = 1{,}000$. Removing the expression graph and dense canonicalisation gives $13.3\times$, and replacing general conic factorisation with the Woodbury system gives a further $3.7\times$. Against the sparse model, the fairer comparison, the gain is about 3–4×. Weight differences near 5×10^{-4} mainly reflect the default quadratic soft-band penalty, with $\rho = 10^4$. Stopping early at six iterations and tolerance 10^{-3} takes 3.3 ms at $n = 300$ and 4.3 ms at $n = 1{,}000$, with objective gaps near 3×10^{-3} and weight differences up to 1.9×10^{-2}, which suits screening and warm starts only.

Table 1. Factor-risk programs at tolerance 10^{-8}. Each problem is rebuilt and solved end to end. The baseline is CVXPY with Clarabel (Diamond and Boyd, 2016; Goulart and Chen, 2024) or, for the 100-name problem, SLSQP, and accuracy is measured against it. "Sparse" is a hand-built sparse Clarabel model. Ladder accuracy is for the first rung. Dashes denote quantities not measured.

Workload	Solver path	Median (ms)	vs. baseline	vs. sparse	Objective gap	Max $\lvert\Delta x_i\rvert$
One-shot ($n = 300$, $r = 43$)	CVXPY + Clarabel	92.9	1.0×	—	0	0
	Sparse Clarabel	25.2	3.7×	1.0×	2.1e-9	1.2e-5
	This method	**8.8**	**10.6**×	**2.9**×	2.1e-6	4.4e-4
One-shot ($n = 1000$, $r = 43$)	CVXPY + Clarabel	1,165.3	1.0×	—	0	0
	Sparse Clarabel	87.9	13.3×	1.0×	2.1e-10	2.3e-7
	This method	**23.6**	**49.4**×	**3.7**×	5.3e-6	5.3e-4
Ladder ($n = 500$, 36 solves, per solve)	CVXPY rebuild	275.4	1.0×	—	0	0
	Sparse Clarabel	43.3	6.4×	1.0×	—	—
	This method, prepared	**12.1**	**22.8**×	**3.6**×	3.2e-6	5.5e-4
Market-neutral ($n = 200$)	CVXPY + Clarabel	52.2	1.0×	—	0	0
	This method	**8.7**	**6.0**×	—	5.3e-7	4.5e-4
Signed book ($n = 100$, dense risk)	SciPy SLSQP	133.9	1.0×	—	—	0
	This method	**35.0**	**3.8**×	—	—	7.4e-5

Table 2. Scenario models, $T = 10{,}000$, $n = 50$, long-only with no position cap, simulated from four common factors with Student-t shocks. Times in seconds. "Iter." counts interior-point iterations. All solves reach optimal status.

Measure	Objective	Ours	RL	Ratio	Iter.	Max $\lvert\Delta w\rvert$
CVaR	Min. risk	0.19	2.43	12.6	18	5.3e-8
	Sharpe	0.20	2.70	13.4	19	3.0e-7
WR	Min. risk	0.04	1.47	35.5	15	4.0e-8
	Sharpe	0.04	1.84	47.8	14	2.4e-7
MAD	Min. risk	0.22	2.29	10.6	15	1.6e-8
	Sharpe	0.23	2.74	11.8	16	1.7e-9
MSV	Min. risk	0.22	3.93	18.1	16	3.5e-8
	Sharpe	0.23	3.50	15.5	16	8.8e-8
MV	Min. risk	0.01	0.05	5.0	10	5.0e-6
	Sharpe	0.01	0.05	4.8	11	6.5e-7
CDaR	Min. risk	2.60	3.79	1.5	47	1.1e-9
	Sharpe	2.24	4.53	2.0	43	6.0e-8
MDD	Min. risk	0.85	3.11	3.7	28	7.1e-10
	Sharpe	0.72	3.31	4.6	24	1.5e-8
ADD	Min. risk	0.61	2.62	4.3	21	1.8e-9
	Sharpe	0.52	2.61	5.0	18	7.1e-8

Scenario models. Table 2 compares the arrowhead kernel with Riskfolio-Lib (RL), which uses Clarabel through CVXPY. Objectives evaluated with common risk functions differ by at most 2.2×10^{-8} in relative terms, and maximum weight differences are near 5×10^{-6} at worst. Speedups depend on the data. Under a second data-generating process with weaker common factors and a 25% position cap, ratios range from 3.5 to 18.3 on CVaR, WR, MAD, and MSV, and from 1.2 to 5.2 on drawdown measures, with relative objective differences up to 7.9×10^{-7} and weight differences up to 1.1×10^{-4}. Drawdown measures are the slowest family because of their banded solves and higher iteration counts, which reach 47 here and 63 under the second process.

Table 3. Block-bootstrapped asset-allocation problem: 8 assets, 2,000 15-year monthly paths, 360,000 scenarios, 40% position cap. Times in seconds. "Rows" are scenarios retained by screening.

Model	Ours	RL	Ratio	Rows	Max $\lvert\Delta w\rvert$
CVaR min. risk	2.15	25.0	11.7	83,910	1.1e-8
CVaR Sharpe	1.11	19.2	17.4	47,466	5.7e-9
WR min. risk	0.035	7.13	203	128	2.8e-8
WR Sharpe	0.034	7.59	225	128	3.5e-10
MAD Sharpe	4.31	14.6	3.4	—	6.5e-6

Screening at scale. Table 3 reports a larger problem. Screening retains 13–23% of scenarios for CVaR within two rounds and 128 scenarios for worst realisation within one, with relative objective differences of at most 5×10^{-9}. Mean absolute deviation has no screening and gains far less.

Efficient frontier. We also trace a 20-point CVaR frontier on this problem, using evenly spaced risk caps between the minimum-risk and maximum-return portfolios. Of the 19 capped problems, 4 reach optimal status, 13 end optimal but inaccurate, and 2 stall with dual residuals of 2.6×10^{-7} and 4.3×10^{-7} and return no weights. Together they take 16.1 s. The same construction completes without a failed solve on the first 200,000 scenarios. Risk caps at large T are therefore the least reliable case for this method, although the failures are detected and reported.

Limitations

First, all benchmarks use synthetic data and one machine, so absolute times and ratios will differ elsewhere, and the two data-generating processes above show that ratios are sensitive to the data. Second, these results measure computational performance and say nothing about investment outcomes. Third, the factor-risk kernel assumes sign-fixed

weights, so free-sign support selection with minimum positions needs an outer ladder or a first-stage sign assignment, and its L1 turnover and soft-band treatments are approximations. Fourth, measures requiring exponential, power, or second-order cones, such as entropic value at risk and tracking-error constraints, are not supported, nor are integer constraints. Fifth, accelerator backends are not evaluated here.

Code Availability

The solver code is not publicly available.

References

Cajas, Dany, 2026, Riskfolio-Lib, version 7.4, Python software.

Charnes, Abraham, and William W. Cooper, 1962, Programming with linear fractional functionals, *Naval Research Logistics Quarterly* 9, 181–186.

Chekhlov, Alexei, Stanislav Uryasev, and Michael Zabarankin, 2005, Drawdown measure in portfolio optimization, *International Journal of Theoretical and Applied Finance* 8, 13–58.

Cuthill, Elizabeth, and James McKee, 1969, Reducing the bandwidth of sparse symmetric matrices, *Proceedings of the 24th National Conference of the ACM*, 157–172.

de Moura, Leonardo, and Sebastian Ullrich, 2021, The Lean 4 theorem prover and programming language, *Automated Deduction – CADE 28*, 625–635.

Diamond, Steven, and Stephen Boyd, 2016, CVXPY: A Python-embedded modeling language for convex optimization, *Journal of Machine Learning Research* 17, 1–5.

Gondzio, Jacek, 1996, Multiple centrality corrections in a primal-dual method for linear programming, *Computational Optimization and Applications* 6, 137–156.

Goulart, Paul J., and Yuwen Chen, 2024, Clarabel: An interior-point solver for conic programs with quadratic objectives, arXiv preprint arXiv:2405.12762.

Konno, Hiroshi, and Hiroaki Yamazaki, 1991, Mean-absolute deviation portfolio optimization model and its applications to Tokyo stock market, *Management Science* 37, 519–531.

Mehrotra, Sanjay, 1992, On the implementation of a primal-dual interior point method, *SIAM Journal on Optimization* 2, 575–601.

Rockafellar, R. Tyrrell, and Stanislav Uryasev, 2000, Optimization of conditional value-at-risk, *Journal of Risk* 2, 21–42.

The mathlib Community, 2020, The Lean mathematical library, *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*, 367–381.

Woodbury, Max A., 1950, Inverting modified matrices, Statistical Research Group Memorandum Report 42, Princeton University.

Young, Martin R., 1998, A minimax portfolio selection rule with linear programming solution, *Management Science* 44, 673–683.