

# Linear and Dynamic Programming Applications in Stock Portfolio Optimization

Zhe Wang

August 2025

## 1 Introduction

Portfolio optimization seeks to allocate capital among assets to achieve objectives like maximizing expected return or minimizing risk. While the seminal Markowitz mean-variance framework employs quadratic programming [? ], many real-world risk measures and constraints yield linear or dynamic programming formulations. Linear programming enables efficient single-period optimization under piecewise-linear risk measures (e.g., CVaR) and linear constraints, whereas dynamic programming handles intertemporal decision-making under uncertainty by recursively solving Bellman equations [? ]. This report surveys key LP and DP approaches and provides comparative analysis.

## 2 Linear Programming Formulations

### 2.1 CVaR Minimization

Conditional Value-at-Risk (CVaR) quantifies the expected loss in the worst  $1 - \alpha$  tail of the return distribution. Rockafellar and Uryasev [? ] demonstrated that CVaR minimization admits a linear programming formulation by introducing an auxiliary variable  $\xi$  and scenario slacks  $\zeta_s$  representing tail losses. The LP:

$$\begin{aligned} \min_{w, \xi, \zeta} & \xi + \frac{1}{(1 - \alpha)S} \sum_{s=1}^S \zeta_s, \\ \text{s.t.;} & -w^T r_s - \xi \leq \zeta_s, \quad \zeta_s \geq 0, \quad s = 1, \dots, S, \\ & \sum_i w_i = 1, \quad w_i \geq 0. \end{aligned}$$

Here,  $r_s$  are scenario returns and  $w$  the weight vector. This formulation is piecewise-linear and solved in polynomial time using interior-point methods [? ]. Complexity grows with the number of scenarios  $S$  and assets  $n$ , typically  $O((n + S)^3)$  in worst-case solver time [? ]. CVaR LP is inherently single-period since it optimizes static weights over a fixed return distribution.

## 2.2 Sector and Transaction-Cost Constraints

Real-world portfolio rules—such as sector exposure limits or proportional transaction costs—can be encoded as linear constraints. For maximum sector weight  $L_k$  in sector  $k$ :

$$\sum_{i \in \mathcal{S}_k} w_i \leq L_k,$$

and transaction costs  $c_i$  per dollar traded introduce terms  $c_i|\Delta w_i|$ , linearized by splitting  $\Delta w_i$  into positive and negative parts [? ]. These constraints preserve convexity and allow LP solvers like CPLEX to compute solutions efficiently [? ]. The tractability depends on the total number of constraints, but remains polynomial-time, with complexity  $O(m^{2.5}n)$  where  $m$  is constraint count and  $n$  variable count [? ].

## 3 Dynamic Programming Approaches

### 3.1 Multi-Period Rebalancing

Dynamic programming formulates portfolio rebalancing over multiple periods via Bellman recursion [? ]. Let  $V_t(x)$  denote the maximal expected utility from period  $t$  onward given wealth  $x$ . Under log-utility and discrete returns  $R_{t+1}$ :

$$V_t(x) = \max_{w_t} \mathbb{E}[\ln(w_t^T R_{t+1}x) + V_{t+1}(w_t^T R_{t+1}x)].$$

Solving this requires discretizing wealth and weight spaces, leading to  $O(T \times |X| \times |W|)$  complexity, where  $T$  is number of periods,  $|X|$  grid size, and  $|W|$  action count. The curse of dimensionality arises as  $|X|$  and  $|W|$  grow exponentially with problem granularity [? ]. Approximate dynamic programming or value-function approximation [? ] mitigates this by projecting  $V_t$  onto a lower-dimensional basis.

### 3.2 Stochastic Control and HJB Equations

In continuous time, dynamic programming yields Hamilton–Jacobi–Bellman (HJB) partial differential equations for the value function  $V(t, x)$  [? ? ]:

$$0 = V_t + \sup_{\pi} \left\{ \mu \pi x V_x + \frac{1}{2} \sigma^2 \pi^2 x^2 V_{xx} \right\},$$

where  $\pi$  is the fraction invested, and  $\mu, \sigma$  the asset drift and volatility. Solving HJB PDEs via finite-difference or finite-element methods has complexity  $O(n_x^3)$  per time step in  $n_x$  state dimensions [? ]. Analytical solutions exist under CRRA utility [? ], but more realistic models require high-dimensional numerical PDE solvers, limiting practical dimensionality to three or four state variables.

## 4 Comparative Analysis

Table contrasts key LP and DP characteristics: Linear programs like CVaR minimization optimize a static weight vector under fixed return scenarios, enabling polynomial-time solutions via

Method	Scope	Complexity
CVaR LP	Single-period risk control	$O((n + S)^3)$ [?] ]
Sector LP	Constraint handling	$O(m^{2.5}n)$ [?] ]
Multi-period DP	Intertemporal optimization	Exponential in $ X ,  W $ [?] ]
Stochastic Control	Continuous-time models	$O(n_x^3)$ per step [?] ]

Table 1: Comparison of LP and DP Methods

interior-point algorithms [? ]. Sector constraint LPs scale similarly, with complexity determined by constraint count [? ]. Dynamic programming’s multi-stage optimization requires discretizing state-action spaces, leading to an exponential growth in computation known as the curse of dimensionality [? ? ]. Continuous-time stochastic control methods solve high-dimensional HJB PDEs, with computation cost scaling cubically in state dimensions and grid resolution [? ]. These complexities guide method selection based on horizon length, risk preferences, and computational resources.

## 5 Implementation Considerations

Practitioners should ensure high-quality scenario generation for LP-based CVaR optimization and calibrate transaction-cost parameters. For DP, approximate methods such as basis function approximation, policy rollout, and actor-critic algorithms can reduce dimensionality [? ]. Hybrid frameworks using LP at each DP stage or combining DP-derived insights with LP solvers show promise [? ].

## 6 Conclusion

Linear programming provides tractable single-period optimization with robust constraint handling, making it well-suited for static portfolio problems. Dynamic programming extends optimization across time horizons but incurs significant computational burdens due to state-action discretization and PDE solving. Selecting between LP and DP approaches depends on investment horizon, model complexity, and available computational power. Hybrid strategies that leverage the strengths of both paradigms can deliver flexible, high-performance portfolio solutions.

## References

- [1] H. Markowitz, *Portfolio Selection*, The Journal of Finance, 1952.
- [2] R. Bellman, *Dynamic Programming*, Princeton University Press, 1957.
- [3] R. T. Rockafellar and S. Uryasev, *Optimization of Conditional Value-at-Risk*, Journal of Risk, 2000.
- [4] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*, Wiley, 1988.
- [5] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, 1997.

- [6] F. J. Fabozzi, P. N. Kolm, D. Pachamanova, and S. Focardi, *Robust Portfolio Optimization and Management*, Wiley, 2007.
- [7] IBM, *IBM ILOG CPLEX Optimizer*, [<https://www.ibm.com/products/ilog-cplex-optimization-studio>]
- [8] V. Chvatal, *Linear Programming*, Freeman, 1983.
- [9] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Wiley, 2011.
- [10] R. C. Merton, *Optimum Consumption and Portfolio Rules in a Continuous-Time Model*, Journal of Economic Theory, 1971.
- [11] W. H. Fleming and H. M. Soner, *Controlled Markov Processes and Viscosity Solutions*, Springer, 2006.
- [12] P. A. Forsyth and K. R. Vetzal, *Quadratic Convergence for Valuing American Options using a Penalty Method*, SIAM Journal on Scientific Computing, 1999.
- [13] J. Hu and W. B. Powell, *Hybrid Hidden Markov Model Approach for Portfolio Optimization*, INFORMS Journal on Computing, 2012.