

Spectral Optimization

Dr. Tamás Nagy

Dr. Tamás Nagy

tamas@thel latent.space

Skeleton

Abstract

Spectral Optimization — ProofEnv proofs (condition number, spectral step rate).

This paper presents 71 machine-verified theorems. All results are formally verified in the Platonic proof kernel (137 verification units, 72 proved statements) and exportable to Lean 4.

1. Introduction

2. Further Results

Theorem (condition_ge_one). *Condition Ge One*. [Platonic: condition_ge_one, domain: spectral_optimization]

Theorem (rate_in_unit). *Rate In Unit*. [Platonic: rate_in_unit, domain: spectral_optimization]

Theorem (condition_implies_eigen_order). *Condition Implies Eigen Order*. [Platonic: condition_implies_eigen_order, domain: spectral_optimization]

Theorem (better_condition_faster). *Better Condition Faster*. [Platonic: better_condition_faster, domain: spectral_optimization]

Theorem (portfolio_sgd_rate). *Portfolio Sgd Rate*. [Platonic: portfolio_sgd_rate, domain: spectral_optimization]

Theorem (kappa_mul_symm). *Kappa Mul Symm*. [Platonic: kappa_mul_symm, domain: spectral_optimization]

Theorem (w2_kappa_nonneg_of_ge_one). *W2 Kappa Nonneg Of Ge One*. [Platonic: w2_kappa_nonneg_of_ge_one, domain: spectral_optimization]

Theorem (w2_lmin_nonneg_of_pos). *W2 Lmin Nonneg Of Pos*. [Platonic: w2_lmin_nonneg_of_pos, domain: spectral_optimization]

Theorem (w2_gap_same). *W2 Gap Same*. [Platonic: w2_gap_same, domain: spectral_optimization]

Theorem (w2_add_monotone). *W2 Add Monotone*. [Platonic: w2_add_monotone, domain: spectral_optimization]

Theorem (w2_sub_flip_order). *W2 Sub Flip Order*. [Platonic: w2_sub_flip_order, domain: spectral_optimization]

Theorem (w2_cheb_num_nonneg). *W2 Cheb Num Nonneg*. [Platonic: w2_cheb_num_nonneg, domain: spectral_optimization]

Theorem (w2_cheb_denom_pos). *W2 Cheb Denom Pos*. [Platonic: w2_cheb_denom_pos, domain: spectral_optimization]

Theorem (w2_rate_nonneg_only). *W2 Rate Nonneg Only*. [Platonic: w2_rate_nonneg_only, domain: spectral_optimization]

Theorem (w2_rate_lt_one_only). *W2 Rate Lt One Only*. [Platonic: w2_rate_lt_one_only, domain: spectral_optimization]

Theorem (w2_kappa_plus_comm). *W2 Kappa Plus Comm*. [Platonic: w2_kappa_plus_comm, domain: spectral_optimization]

Theorem (w2_mul_sub_distrib). *W2 Mul Sub Distrib*. [Platonic: w2_mul_sub_distrib, domain: spectral_optimization]

Theorem (w2_sub_sub_assoc). *W2 Sub Sub Assoc*. [Platonic: w2_sub_sub_assoc, domain: spectral_optimization]

Theorem (w2_condition_product_nonneg). *W2 Condition Product Nonneg*. [Platonic: w2_condition_product_nonneg, domain: spectral_optimization]

Theorem (w2_eigen_max_ge_min_repeat). *W2 Eigen Max Ge Min Repeat*. [Platonic: w2_eigen_max_ge_min_repeat, domain: spectral_optimization]

Theorem (w2_neff_le_one). *W2 Neff Le One*. [Platonic: w2_neff_le_one, domain: spectral_optimization]

Theorem (w2_denom_order). *W2 Denom Order*. [Platonic: w2_denom_order, domain: spectral_optimization]

Theorem (w2_nlin_mono_product). *W2 Nlin Mono Product*. [Platonic: w2_nlin_mono_product, domain: spectral_optimization]

Theorem (w2_step_nonneg_scale). *W2 Step Nonneg Scale*. [Platonic: w2_step_nonneg_scale, domain: spectral_optimization]

Theorem (w2_ratio_num_le). *W2 Ratio Num Le*. [Platonic: w2_ratio_num_le, domain: spectral_optimization]

Theorem (w2_kappa_order_trans). *W2 Kappa Order Trans*. [Platonic: w2_kappa_order_trans, domain: spectral_optimization]

Theorem (w2_one_minus_rate_pos). *W2 One Minus Rate Pos*. [Platonic: w2_one_minus_rate_pos, domain: spectral_optimization]

Theorem (w2_rate_def_sym). *W2 Rate Def Sym*. [Platonic: w2_rate_def_sym, domain: spectral_optimization]

Theorem (w2_lambda_order_nonneg_diff). *W2 Lambda Order Nonneg Diff*. [Platonic: w2_lambda_order_nonneg_diff, domain: spectral_optimization]

Theorem (w2_optimization_refl). *W2 Optimization Refl*. [Platonic: w2_optimization_refl, domain: spectral_optimization]

Theorem (sp_w4_cheb_num_times_lmin_nonneg). *Sp W4 Cheb Num Times Lmin Nonneg.* [Platonic: sp_w4_cheb_num_times_lmin_nonneg, domain: spectral_optimization]

Theorem (sp_w4_rate_times_denom_eq_num). *Sp W4 Rate Times Denom Eq Num.* [Platonic: sp_w4_rate_times_denom_eq_num, domain: spectral_optimization]

Theorem (sp_w4_kappa_plus_one_pos). *Sp W4 Kappa Plus One Pos.* [Platonic: sp_w4_kappa_plus_one_pos, domain: spectral_optimization]

Theorem (sp_w4_gap_nonneg_squared). *Sp W4 Gap Nonneg Squared.* [Platonic: sp_w4_gap_nonneg_squared, domain: spectral_optimization]

Theorem (w10_detection_obs_iff). *W10 Detection Obs Iff.* [Platonic: w10_detection_obs_iff, domain: spectral_optimization]

Theorem (w10_optimal_rate_simplification). *W10 Optimal Rate Simplification.* [Platonic: w10_optimal_rate_simplification, domain: spectral_optimization]

Theorem (w10_diagonal_condition_ge_one). *W10 Diagonal Condition Ge One.* [Platonic: w10_diagonal_condition_ge_one, domain: spectral_optimization]

Theorem (w10_decorrelation_speedup). *W10 Decorrelation Speedup.* [Platonic: w10_decorrelation_speedup, domain: spectral_optimization]

Theorem (w10_preprocessing_improves_rate). *W10 Preprocessing Improves Rate.* [Platonic: w10_preprocessing_improves_rate, domain: spectral_optimization]

Theorem (w10_rate_depends_on_kappa). *W10 Rate Depends On Kappa.* [Platonic: w10_rate_depends_on_kappa, domain: spectral_optimization]

Theorem (w10_worse_kappa_slower). *W10 Worse Kappa Slower.* [Platonic: w10_worse_kappa_slower, domain: spectral_optimization]

Theorem (w10_adding_mode_helps_scalar). *W10 Adding Mode Helps Scalar.* [Platonic: w10_adding_mode_helps_scalar, domain: spectral_optimization]

Theorem (w10_equicorr_min_pos). *W10 Equicorr Min Pos.* [Platonic: w10_equicorr_min_pos, domain: spectral_optimization]

Theorem (preconditioner_reduces_kappa). *Preconditioner Reduces Kappa.* [Platonic: preconditioner_reduces_kappa, domain: spectral_optimization]

Theorem (perfect_preconditioner). *Perfect Preconditioner.* [Platonic: perfect_preconditioner, domain: spectral_optimization]

Theorem (preconditioner_rate_improvement). *Preconditioner Rate Improvement.* [Platonic: preconditioner_rate_improvement, domain: spectral_optimization]

Theorem (perturbation_preserves_positivity). *Perturbation Preserves Positivity.* [Platonic: perturbation_preserves_positivity, domain: spectral_optimization]

Theorem (perturbation_widens_gap). *Perturbation Widens Gap.* [Platonic: perturbation_widens_gap, domain: spectral_optimization]

Theorem (chebyshev_rate_better_than_gradient). *Chebyshev Rate Better Than Gradient.* [Platonic: chebyshev_rate_better_than_gradient, domain: spectral_optimization]

Theorem (acceleration_gap_nonneg). *Acceleration Gap Nonneg.* [Platonic: acceleration_gap_nonneg, domain: spectral_optimization]

Theorem (gd_iteration_complexity). *Gd Iteration Complexity.* [Platonic: gd_iteration_complexity, domain: spectral_optimization]

Theorem (agd_iteration_sqrt_kappa). *Agd Iteration Sqrt Kappa.* [Platonic: agd_iteration_sqrt_kappa, domain: spectral_optimization]

Theorem (condition_number_product). *Condition Number Product.* [Platonic: condition_number_product, domain: spectral_optimization]

Theorem (relative_gap_nonneg). *Relative Gap Nonneg.* [Platonic: relative_gap_nonneg, domain: spectral_optimization]

Theorem (eigenmode_descent_rate). *Eigenmode Descent Rate.* [Platonic: eigenmode_descent_rate, domain: spectral_optimization]

Theorem (eigenmode_contraction_strict). *Eigenmode Contraction Strict.* [Platonic: eigenmode_contraction_strict, domain: spectral_optimization]

Theorem (diversification_improves_conditioning). *Diversification Improves Conditioning.* [Platonic: diversification_improves_conditioning, domain: spectral_optimization]

3. Bounds and Estimates

Theorem (suboptimality_refl_bound). *Suboptimality Refl Bound.* [Platonic: suboptimality_refl_bound, domain: spectral_optimization]

Theorem (w2_half_life_bound). *W2 Half Life Bound.* [Platonic: w2_half_life_bound, domain: spectral_optimization]

Theorem (w10_equal_eigenvalues_minimize_bound). *W10 Equal Eigenvalues Minimize Bound.* [Platonic: w10_equal_eigenvalues_minimize_bound, domain: spectral_optimization]

Theorem (w10_doubling_T_halves_bound). *W10 Doubling T Halves Bound.* [Platonic: w10_doubling_T_halves_bound, domain: spectral_optimization]

Theorem (eigenvalue_perturbation_bound). *Eigenvalue Perturbation Bound.* [Platonic: eigenvalue_perturbation_bound, domain: spectral_optimization]

Theorem (step_size_bounded_by_inverse_lmax). *Step Size Bounded By Inverse Lmax.* [Platonic: step_size_bounded_by_inverse_lmax, domain: spectral_optimization]

Theorem (more_iterations_tighter_bound). *More Iterations Tighter Bound.* [Platonic: more_iterations_tighter_bound, domain: spectral_optimization]

Theorem (inverse_condition_bounded). *Inverse Condition Bounded.* [Platonic: inverse_condition_bounded, domain: spectral_optimization]

4. Spectral Theory

Theorem (spectral_gap_nonneg). *Spectral Gap Nonneg.* [Platonic: spectral_gap_nonneg, domain: spectral_optimization]

Theorem (w2_spectral_gap_sym). *W2 Spectral Gap Sym.* [Platonic: w2_spectral_gap_sym, domain: spectral_optimization]

Theorem (optimal_step_in_eigenvalue_range). *Optimal Step In Eigenvalue Range.* [Platonic: optimal_step_in_eigenvalue_range, domain: spectral_optimization]

Theorem (larger_eigenvalue_faster_descent). *Larger Eigenvalue Faster Descent.* [Platonic: larger_eigenvalue_faster_descent, domain: spectral_optimization]

Theorem (portfolio_condition_from_eigenvalues). *Portfolio Condition From Eigenvalues.* [Platonic: portfolio_condition_from_eigenvalues, domain: spectral_optimization]

5. Convergence Results

Theorem (cg_converges_in_kappa_steps). *Cg Converges In Kappa Steps.* [Platonic: cg_converges_in_kappa_steps, domain: spectral_optimization]

6. Proof Architecture

All proofs are implemented in the Platonic kernel (elysium/fields/spectral_optimization/).

File	Role
spectral_optimization_proof.py	

7. Discussion

References