Solving Hermite's Problem: Three Novel Approaches for Complete Characterization of Cubic Irrationals

Brandon Barclay
Email: barclaybrandon@hotmail.com

Abstract

Hermite's problem seeks an algorithm that characterizes cubic irrationals through periodicity, analogous to how continued fractions identify quadratic irrationals. We present a complete solution through three complementary approaches: (1) the Hermite Algorithm for Periodicity Detection (HAPD) operating in projective space, (2) a matrix-based characterization using companion matrices and trace sequence periodicity, and (3) a modified $\sin^2$-algorithm that handles complex conjugate roots via a phase-preserving floor function. Each method produces eventually periodic sequences precisely for cubic irrationals, including those with complex conjugate roots—previously an unsolved case. We rigorously prove the correctness of each approach, establish their mathematical equivalence, and provide comprehensive numerical validation. Our work creates a unified framework connecting periodicity to algebraic degree for cubic irrationals, resolving a long-standing problem in Diophantine approximation.

Keywords: Cubic irrationals, Hermite's problem, continued fractions, projective geometry, companion matrices, trace sequences, Diophantine approximation

The implementation code for all algorithms discussed in this paper is available at:
https://github.com/bbarclay/hermitesproblem
Interactive materials are available at: https://bbarclay.github.io/hermitesproblem/

1. Introduction

Hermite's problem, posed to Jacobi in 1848, sought a generalization of continued fractions that would characterize cubic irrationals through periodicity. Continued fractions produce eventually periodic sequences precisely for quadratic irrationals, but the cubic case with complex conjugate roots remained unsolved. We resolve Hermite's problem through three novel approaches: 1. HAPD algorithm in projective space, producing periodic sequences if and only if the input is cubic irrational 2. Matrix characterization using companion matrices and trace sequences with modular periodicity 3. Modified $\sin^2$-algorithm handling complex conjugate roots via phase-preserving floor functions

2. Equivalence of Algorithmic and Matrix Approaches

We establish formal equivalence between the HAPD algorithm and matrix-based characterizations of cubic irrationals. This equivalence proves our solution is robust and well-founded, with multiple complementary perspectives supporting the same conclusion.

2.1 Structural Equivalence

The analysis begins by proving that the structures underlying both approaches are fundamentally the same. **Theorem (Structural Equivalence):** The projective transformations in the HAPD algorithm correspond to matrix transformations in the companion matrix approach. Specifically, each iteration of the HAPD algorithm is equivalent to a matrix operation on the corresponding companion matrix. **Proof:** Consider a cubic irrational α with companion matrix C_α. The HAPD algorithm operates on triples (v■, v■, v■) in projective space P^2(■), where initially (v■, v■, v■) = (α, α^2, 1). For the companion matrix approach, trace sequences are computed as Tr(C_α^n). The initial triple (α, α^2, 1) corresponds to the powers α^1, α^2, α■. At each iteration, the HAPD algorithm computes integer parts (a■, a■) and remainders (r■, r■), then updates the triple. This operation corresponds to a specific transformation in the matrix approach, where the trace of C_α^n follows the recurrence relation derived from the minimal polynomial. The periodicity in the HAPD algorithm precisely corresponds to the periodicity in the trace sequence modulo certain integers, establishing the structural equivalence. This follows directly from the fact that both representations capture the full algebraic structure of ■(α).

3. Implementation Examples

All algorithms were implemented in Python with NumPy for numerical operations and SageMath for algebraic number field computations. def hapd_algorithm(alpha, max_iterations=100): v1, v2, v3 = alpha, alpha**2, 1 sequence = [] for i in range(max_iterations): a1 = math.floor(v1/v3) a2 = math.floor(v2/v3) sequence.append((a1, a2)) r1 = v1 - a1*v3 r2 = v2 - a2*v3 v3_new = v3 - a1*r1 - a2*r2 v1, v2, v3 = r1, r2, v3_new if v1 == 0 and v2 == 0 and v3 == 0: return "Periodic", sequence # Check for periodicity if detect_cycle(sequence): return "Periodic", get_period(sequence) return "Inconclusive", sequence Key implementation considerations: • High-precision arithmetic is essential for reliable periodicity detection • Normalization of triples improves numerical stability • Early termination conditions significantly reduce computation time The complete implementation with additional optimizations and test cases is available in our GitHub repository at https://github.com/bbarclay/hermitesproblem.

4. Conclusion

We have presented a comprehensive solution to Hermite's classical problem of characterizing cubic irrationals through periodicity. Our unified approach bridges algebraic number theory, projective geometry, and computational mathematics, resolving a question that has remained open since 1848. These interactive materials can be accessed at https://bbarclay.github.io/hermitesproblem/, while the complete source code, documentation, and additional examples are available in our GitHub repository at https://github.com/bbarclay/hermitesproblem. The repository follows best practices for scientific computing, including version control, continuous integration, and reproducible environments. We encourage interested readers to use these tools to develop intuition about the theoretical concepts, explore the algorithms' behavior with custom inputs, and build upon our work for further research and applications.