Necessity of Using Abstract Numerical Values for Unicodes in Computer Science

Pu Justin Scarfy Yang

July 29, 2024

1 Introduction

The traditional Unicode standard assigns specific numerical values to characters, primarily focusing on linguistic and symbolic representation. However, as computer science increasingly intersects with advanced mathematics, there is a growing need to extend this concept to include abstract numerical values representing complex mathematical objects such as automorphic forms, L-functions, and motives. This paper discusses the necessity of using abstract numerical values for Unicodes in computer science.

2 Enhanced Mathematical Representation

2.1 Formalization of Mathematics

As computer science and mathematics become more intertwined, formalizing and digitizing complex mathematical objects is crucial. This extension can facilitate more precise and robust mathematical computations and proofs in areas such as automated theorem proving, symbolic computation, and computer algebra systems.

2.2 UniMath Integration

Using UniMath codes as abstract numerical values aligns with the ongoing effort to formalize all mathematics. It provides a standardized way to encode mathematical concepts digitally, enhancing their usability in computational environments.

3 Interoperability and Standardization

3.1 Standard Representation

A standardized representation of complex mathematical objects improves interoperability between different software systems and tools. This standardization

enables seamless data exchange and integration, essential for collaborative research and multi-disciplinary applications.

3.2 Cross-Disciplinary Applications

Fields such as cryptography, computational number theory, and scientific computing benefit from a unified representation of mathematical objects. This approach enhances cross-disciplinary research and applications, fostering innovation and discovery.

4 Advanced Computational Techniques

4.1 Efficient Encoding and Processing

Representing mathematical objects as abstract numerical values leads to more efficient encoding, storage, and processing techniques. These improvements are crucial for large-scale computations and simulations, making advanced computational techniques more accessible and effective.

4.2 Next-Generation Cryptography

Developing encryption schemes based on advanced mathematical frameworks, such as topos theory, may require a more abstract and flexible encoding system. This necessity drives the adoption of abstract numerical values for Unicodes, enabling next-generation cryptographic methods.

5 Educational and Research Tools

5.1 Improved Learning Tools

Standardized encoding and manipulation of mathematical objects enhance educational tools and platforms. This standardization aids in the teaching and learning of advanced mathematics, making complex concepts more accessible to students and educators.

5.2 Research Collaboration

Facilitating easier sharing and collaboration on complex mathematical research accelerates advancements and discoveries. A unified representation of mathematical objects supports this goal, promoting efficient and effective research collaboration.

6 Is UniMath the Best Choice?

6.1 Advantages of UniMath

- **Formalization**: UniMath is designed to formalize a large body of mathematics using the Coq proof assistant, ensuring rigor and correctness.
- **Community and Support**: UniMath has an active community of researchers and contributors, providing ongoing development and support.
- **Integration with Coq**: UniMath builds on the Coq proof assistant, which is widely used in formal verification and theorem proving.

6.2 Potential Alternatives

- **Lean**: The Lean theorem prover and its mathematical library, mathlib, are also robust choices. Lean is known for its user-friendly syntax and growing community support.
- **Isabelle/HOL**: Isabelle is another popular proof assistant with a rich set of libraries for formalized mathematics.
- **HOL Light**: This is a lightweight version of the HOL theorem prover, which is also used for formalizing mathematics and has a simpler, more minimalist approach.
- **Mizar**: Mizar focuses on a language close to mathematical vernacular and has a large library of formalized mathematics.
- **Metamath**: An extremely lightweight language for developing rigorously verified mathematics, with a small core and extensive database of proofs.

6.3 Conclusion on UniMath

While UniMath is a strong and well-supported choice, the "best" choice depends on the specific requirements and preferences of the user or organization. Each alternative has its own strengths, and the decision should consider factors such as ease of use, community support, existing libraries, and compatibility with other tools. For many advanced mathematical formalization tasks, UniMath is indeed a top choice due to its rigorous approach and strong community. However, exploring other options like Lean or Isabelle might be beneficial depending on the specific context and needs.

7 Conclusion

While it may not be immediately necessary for all areas of computer science, using abstract numerical values for Unicodes to represent complex mathematical objects is increasingly important. This approach addresses specific needs in

advanced computational mathematics, interoperability, and the formalization of mathematical knowledge, aligning with ongoing trends and future developments in the field.