Semantic Preserving Bijective Mappings for Representations of Special Functions between Computer Algebra Systems and Word Processors

André Greiner-Petter

Information Science Group, University of Konstanz, Germany
andre.greiner-petter@t-online.de

Abstract

This Master's thesis presents the development of a translation tool between representations of semantically enriched mathematical formulae in a Word Processor (WP) and its corresponding representations in a Computer Algebra System (CAS). The representatives for the CAS are Maple and Mathematica, while the representative WP is LaTeX. Semantic information of a mathematical formula in LaTeX is rather given in the context than in the formula itself. Extracting the information is complicated and sometimes even impossible. However, a translation to a CAS is only achievable, if the semantic of the formula is unique. The National Institute of Standards and Technology (NIST) in the U.S. has developed a set of semantic LaTeX macros for all Orthogonal Polynomials and Special Functions (OPSF) defined in the Digital Library of Mathematical Functions (DLMF). Using these macros can disambiguate formulae and provide access to the semantic information. Even if the semantics of a representation of one formula is unique, it does not need to match the semantics in another representation of the formula. For example, there are may be differences in the domains or a function in a CAS is normalized for a better performance. An important concept for complex and multivalued functions are *branch cuts*. However, the positions of these cuts are not consistently defined and can vary from system to system. Therefore, a CAS user needs to fully understand the properties and definitions of the used functions in the CAS. Hence, a manual translation from one system to another is not only laborious, but also prone to errors. This thesis explains the realization of the translator, and discusses and suggests the approaches to the problems mentioned above.

1 Introduction

A typical workflow of a scientist who writes a scientific publication is to use a Word Processor (WP) to write the publication and one or more Computer Algebra System (CAS) for Verification, Analysis and Visualization, among other tasks. Especially in Science, Technology, Engineering and Mathematics (STEM), LaTeX[1] has become the de facto standard for writing scientific publications over the last 30 years [1, 10, p. 559, 11]. LaTeX enables printing of mathematical formulae in a structure similar to handwritten style. For example, consider a Jacobi polynomial [7, (18.3), table 1]

$$P_n^{(\alpha,\beta)}(\cos(a\Theta)). \tag{1}$$

This formula is written in LaTeX as

```
P_n^{(\alpha,\beta)}(\cos(a\Theta)).        (2)
```

While LaTeX focuses on displaying mathematics, CAS focus on computations and user friendly syntax. Therefore, each system (such as LaTeX, Maple or Mathematica) uses its own representation and syntax. Hence, a writer needs to constantly translate mathematical expressions from one representation to another and back again. Table 1 shows four different representations for the same formula (1).

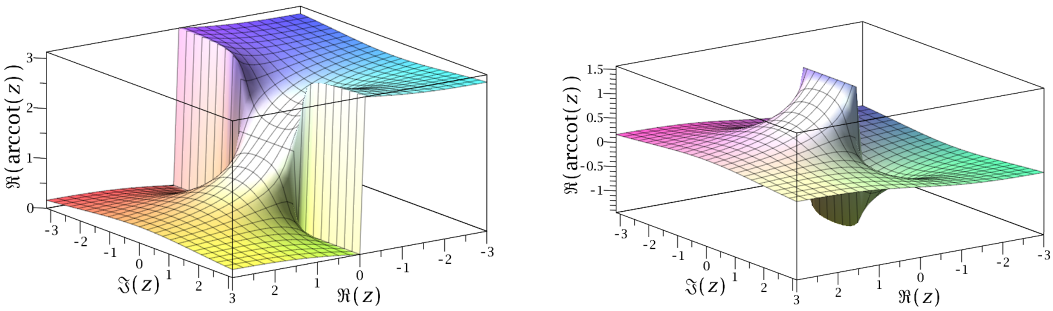
Systems	Representations
Generic LaTeX	`P_n^{(\alpha,\beta)}(\cos(a\Theta))`
Semantic LaTeX	`\JacobiP{\alpha}{\beta}{n}@{\cos@{a\Theta}}`
Maple	`JacobiP(n,alpha,beta,cos(a*Theta))`
Mathematica	`JacobiP[n,\[Alpha],\[Beta],Cos[a \[CapitalTheta]]]`

Table 1: Different representations for the same mathematical formula (1).

Such translations can become complex, if there are non-obvious differences between the used systems. For example, the CAS could use a different order of the arguments, define a normalization of the function for better performances or even use different domains for the variables. Multivalued functions are particularly difficult [6]. A CAS usually defines so called *branch cuts* to compute principal values of multivalued functions [8]. Thereby, implementations of multivalued functions are discontinuous. In general, positioning branch cuts follow some conventions. They are mostly defined conveniently, such as a straight line. However, since a branch cut is a curve, it also can be defined as a spiral or it can be defined for a different position in the complex plane. The position of branch cuts can therefore varies from CAS to CAS [5]. Figure 1 illustrates two examples of different branch cut positioning for the inverse trigonometric arccotangent function. While Maple defines the branch cut at $[-\infty i, -i]$, $[i, \infty i]$ (figure 1a), Mathematica defines the branch cut at $[-i, i]$ (figure 1b).

Hence, a CAS user needs to fully understand the properties and special definitions (such as the position of branch cuts) in the CAS to avoid mistakes during a translation [8]. In consequence, a manual translation process is not only laborious, but also prone to errors. Note that this general problem has been named to automatic Presentation-To-Computation (P2C) conversion [20].

[1]Note that technically LaTeX is not a WP (`https://www.latex-project.org/about/`, seen 07/2017) like *Microsoft Word*. However, since LaTeX (an extension of TeX) is the standard to write in STEM and we only focus on mathematical writings in this thesis, we categorize LaTeX as a WP as well.



(a) The real part of arccotangent with a branch cut at $[-\infty i, -i]$, $[i, \infty i]$.

(b) The real part of arccotangent with a branch cut at $[-i, i]$.

Figure 1: Two plots for the real part of the arccotangent function with a branch cut at $[-\infty i, -i]$, $[i, \infty i]$ in figure (a) and at $[-i, i]$ in figure (b), respectively. (Plotted with Maple 2016)

An automatic translation process is desirable. Providing translations for mathematical LaTeX expressions is difficult, because the semantic information is absent. Semantics is the meaning of an expression. Consider the example of the Jacobi polynomial in (1). A scientific reader might be able to conclude information like P indicating the Jacobi polynomial and n is being a non-negative integer. However, for a computer it is difficult to understand the expression (2). Since LaTeX is the de facto standard for mathematical expressions, most CAS provide import and export functions for LaTeX expressions [13, 17, 14, 19]. However, because of the explained difficulties, those functions are very limited. In fact, they only work for expressions with unique semantic information, but fail for more complex expressions, such as our Jacobi polynomial example.

The National Institute of Standards and Technology (NIST) has therefore developed a set of semantic LaTeX macros [15]. Each of these macros is tied with a unique well-defined mathematical object and linked to the corresponding definition in the Digital Library of Mathematical Functions (DLMF) [7, 12, 16]. The semantic macros are used to semantically enhance the DLMF. An outgrowth of the DLMF project is the Digital Repository of Mathematical Formulae (DRMF) [2, 3], which defines more semantic macros, especially for Orthogonal Polynomials and Special Functions (OPSF). In table 1 we have already seen an example of a semantic macro. We call LaTeX expressions with semantic macros *semantic LaTeX* and without *generic LaTeX*, respectively. With semantic LaTeX, it is possible to provide translations to CAS.

Note that partial results of this Master's thesis have already been published at the 10th Conference on Intelligent Computer Mathematics 2017 [4]. Techniques, approaches and results have been roughly introduced in the article and are discussed in more detail in this thesis. Pictures and ideas that have already been presented there are not explicitly referenced in the following.

Furthermore, note that the translator we will present in this thesis is based on not published code from the Mathematical Language Parser (MLP) project and the Part-of-Math (POM)-tagger. We will also use the mentioned set of DLMF/DRMF LaTeX macros, which are not published yet as well. Therefore, the current version of the translator is not online available yet. However, we are planning to publish the translator soon on the DRMF website[2].

Moreover, I will use 'we' rather than 'I' in the subsequent chapters of this thesis, since I published and discussed my ideas with others including my advisors Dr. Moritz Schubotz and Professor Abdou Youssef, the hosts during research visits Howard S. Cohl and Jürgen Gerhard, and the students Joon Bang and Kevin Chen.[3]

2 Goals of the project

In this thesis, we will present an automatic tool for translations between semantic LaTeX and the CAS Maple and Mathematica. The translation tool should be lite, in a sense that a user does not need a whole software package to use the program. It is also a part of the DRMF project and should provide interactive translations in the DRMF. Furthermore, it should provide additional information about the translation process and present solutions if an appropriate translation is not possible.

Another goal is the translator's extensibility in order to provide the basis for future work.

3 Structure

In the following chapters, we will discuss our translator and verification techniques. Prior to explaining implementation details, we need to introduce some basic background information in chapter **??**. We start this chapter with an overview of related work (section **??**). Multivalued functions and branch cuts, which were already mentioned, will be introduced in section **??**. Besides the mathematical background, we will discuss semantics in LaTeX in section **??**. A brief introduction to grammar in languages and how we will use it to understand mathematical expressions will be given in section **??**. The chapter is closed by some definitions for the following chapters.

The next chapter **??** is the main chapter of the thesis. We will explain implementations and discuss problems and solutions of and for the translator. It starts with an outline of the goals and our approaches in section **??**. While we will present implementation details for the forward translations in section **??**, we will present the developed backward translation in the following section **??**.

In chapter **??** we will discuss, how we can verify or validate a translated expression. The chapter presents some approaches, such as round trip tests (section **??**) and numerical tests (section **??**). Extracted formulae from the DLMF and DRMF created a comprehensive test suite for these approaches. Section **??** gives a summary of the results.

---

[2] `http://drmf.wmflabs.org`

[3] Special thanks to M. Schubotz to give me the permission for using this phrase from his doctoral thesis [18].

The results and the current status of the translation tool will be discussed in chapter **??**. In conclusion, chapter **??** explains some approaches to further improve the explained techniques in future work.

References

[1] Gaudeul Alex. "Do Open Source Developers Respond to Competition? The (La)TeX Case Study". In: *Review of Network Economics* 6.2 (June 2007), pp. 1–25. ISSN: 1446-9022. DOI: `10.2202/1446-9022.1119`. URL: `https://ideas.repec.org/a/bpj/rneart/v6y2007i2n9.html`.

[2] Howard S. Cohl, Marjorie A. McClain, Bonita V. Saunders, Moritz Schubotz, and Janelle C. Williams. "Digital Repository of Mathematical Formulae". In: *Intelligent Computer Mathematics - International Conference, CICM 2014, Coimbra, Portugal, July 7-11, 2014. Proceedings*. Ed. by Stephen M. Watt, James H. Davenport, Alan P. Sexton, Petr Sojka, and Josef Urban. Vol. 8543. Lecture Notes in Computer Science. Springer, 2014, pp. 419–422. ISBN: 978-3-319-08433-6. DOI: `10.1007/978-3-319-08434-3_30`. URL: `https://doi.org/10.1007/978-3-319-08434-3_30`.

[3] Howard S. Cohl, Moritz Schubotz, Marjorie A. McClain, Bonita V. Saunders, Cherry Y. Zou, Azeem S. Mohammed, and Alex A. Danoff. "Growing the Digital Repository of Mathematical Formulae with Generic LaTeX Sources". In: *Intelligent Computer Mathematics - International Conference, CICM 2015, Washington, DC, USA, July 13-17, 2015, Proceedings*. Ed. by Manfred Kerber, Jacques Carette, Cezary Kaliszyk, Florian Rabe, and Volker Sorge. Vol. 9150. Lecture Notes in Computer Science. Springer, 2015, pp. 280–287. ISBN: 978-3-319-20614-1. DOI: `10.1007/978-3-319-20615-8_18`. URL: `https://doi.org/10.1007/978-3-319-20615-8_18`.

[4] Howard S. Cohl, Moritz Schubotz, Abdou Youssef, André Greiner-Petter, Jürgen Gerhard, Bonita V. Saunders, Marjorie A. McClain, Joon Bang, and Kevin Chen. "Semantic Preserving Bijective Mappings of Mathematical Formulae Between Document Preparation Systems and Computer Algebra Systems". In: *Intelligent Computer Mathematics - 10th International Conference, CICM 2017, Edinburgh, UK, July 17-21, 2017, Proceedings*. Ed. by Herman Geuvers, Matthew England, Osman Hasan, Florian Rabe, and Olaf Teschke. Vol. 10383. Lecture Notes in Computer Science. Springer, 2017, pp. 115–131. ISBN: 978-3-319-62074-9. DOI: `10.1007/978-3-319-62075-6_9`. URL: `https://doi.org/10.1007/978-3-319-62075-6_9`.

[5] Robert M. Corless, David J. Jeffrey, Stephen M. Watt, and James H. Davenport. ""According to Abramowitz and Stegun" or arccoth needn't be uncouth". In: *ACM SIGSAM Bulletin* 34.2 (2000), pp. 58–65. DOI: `10.1145/362001.362023`. URL: `http://doi.acm.org/10.1145/362001.362023`.

[6] James H. Davenport. "The Challenges of Multivalued "Functions"". In: *Intelligent Computer Mathematics, 10th International Conference, AISC 2010, 17th Symposium, Calculemus 2010, and 9th International Conference, MKM 2010, Paris, France, July 5-10, 2010. Proceedings*. Ed. by Serge Autexier, Jacques Calmet, David Delahaye, Patrick D. F. Ion,

Laurence Rideau, Renaud Rioboo, and Alan P. Sexton. Vol. 6167. Lecture Notes in Computer Science. Springer, 2010, pp. 1–12. ISBN: 978-3-642-14127-0. DOI: 10.1007/978-3-642-14128-7_1. URL: https://doi.org/10.1007/978-3-642-14128-7_1.

[7] F.W.J. Olver, A.B. Olde Daalhuis, D.W. Lozier, B.I. Schneider, R.F. Boisvert, C.W. Clark, B.R. Miller, and B.V. Saunders, eds. *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.0.14 of 2017-12-21. 2017.

[8] Matthew England, Edgardo S. Cheb-Terrab, Russell J. Bradford, James H. Davenport, and David J. Wilson. “Branch cuts in Maple 17”. In: *ACM Comm. Computer Algebra* 48.1/2 (2014), pp. 24–27. DOI: 10.1145/2644288.2644293. URL: http://doi.acm.org/10.1145/2644288.2644293.

[9] Herman Geuvers, Matthew England, Osman Hasan, Florian Rabe, and Olaf Teschke, eds. *Intelligent Computer Mathematics - 10th International Conference, CICM 2017, Edinburgh, UK, July 17-21, 2017, Proceedings*. Vol. 10383. Lecture Notes in Computer Science. Springer, 2017. ISBN: 978-3-319-62074-9. DOI: 10.1007/978-3-319-62075-6. URL: https://doi.org/10.1007/978-3-319-62075-6.

[10] Donald E. Knuth. *Digital Typography*. Reissue. Lecture Notes (Book 78). Center for the Study of Language and Information (CSLI), June 11, 1998. 685 pp. ISBN: 978-1575860107.

[11] Donald Ervin Knuth. *The Art of Computer Programming, Volume I: Fundamental Algorithms, 3rd Edition*. Addison-Wesley, 1997. ISBN: 0201896834. URL: http://www.worldcat.org/oclc/312910844.

[12] Daniel W. Lozier. “NIST Digital Library of Mathematical Functions”. In: *Ann. Math. Artif. Intell.* 38.1-3 (2003), pp. 105–119. DOI: 10.1023/A:1022915830921. URL: https://doi.org/10.1023/A:1022915830921.

[13] Maplesoft. *Produce output suitable for LaTeX 2e*. http://www.maplesoft.com/support/help/Maple/view.aspx?path=latex. Since version Maple 18, Seen 06/2017.

[14] MathWorks. *LaTeX form of symbolic expression in MATLAB*. https://www.mathworks.com/help/symbolic/latex.html. Seen 06/2017.

[15] Bruce R. Miller and Abdou Youssef. “Technical Aspects of the Digital Library of Mathematical Functions”. In: *Ann. Math. Artif. Intell.* 38.1-3 (2003), pp. 121–136. DOI: 10.1023/A:1022967814992. URL: https://doi.org/10.1023/A:1022967814992.

[16] Frank W. Olver, Daniel W. Lozier, Ronald F. Boisvert, and Charles W. Clark. *NIST Handbook of Mathematical Functions*. 1st. New York, NY, USA: Cambridge University Press, Apr. 30, 2010. 968 pp. ISBN: 9780521140638. URL: http://www.ebook.de/de/product/10464908/nist_handbook_of_mathematical_functions_paperback_and_cd_rom.html.

[17] Wolfram Research. *Generating and Importing TeX*. https://reference.wolfram.com/language/tutorial/GeneratingAndImportingTeX.html.

[18] Moritz Schubotz. “Augmenting Mathematical Formulae for More Effective Querying & Efficient Presentation”. PhD thesis. Technical University of Berlin, Germany, 2017. ISBN: 978-3-7450-6208-3. URL: http://d-nb.info/1135201722.

[19] William Stein. *Sage Math*. http://doc.sagemath.org/html/en/tutorial/latex.html. Seen 6/2017.

[20] Abdou Youssef. “Part-of-Math Tagging and Applications”. In: *Intelligent Computer Mathematics - 10th International Conference, CICM 2017, Edinburgh, UK, July 17-21, 2017, Proceedings*. Ed. by Herman Geuvers, Matthew England, Osman Hasan, Florian Rabe, and Olaf Teschke. Vol. 10383. Lecture Notes in Computer Science. Springer, 2017, pp. 356–374. ISBN: 978-3-319-62074-9. DOI: 10.1007/978-3-319-62075-6_25. URL: https://doi.org/10.1007/978-3-319-62075-6_25.