

## Search Systems for Mathematical Equations

Abdou Youssef

Department of Computer Science  
The George Washington University  
Washington, DC 20052

To process and disseminate technical knowledge more effectively, efforts are underway worldwide to create and codify Web-accessible digital libraries of mathematical contents. Notable examples include the Digital Library of Mathematical Functions (DLMF) project at the National Institute of Standards and Technology (NIST), and the markup languages MathML and OpenMath. To benefit from such digital libraries, users should be able to search not only for text, but also for equations and other math constructs.

Searching can be divided into three broad classes of increasing complexity: (1) keyword based, (2) structural, and (3) semantic. Text search technology has matured at the keyword level, has made significant progress at the structural level (phrase and contiguity search, and XML-based search of structured documents), and is starting to push towards semantic search.

Applying text search technology to math search of equations and other math constructs faces serious problems, even at the keyword and structural levels. The first problem is that mathematical contents often involve symbols, as in  $P_n(x)$  and  $d^2y/dx^2 - x = 0$ , that are misinterpreted or unrecognized by text search systems. The second problem is that mathematical expressions have rich structures whose semantics are undetected by text search system. For example,  $\sin(x + \log x)$  is no different to a text search system than  $\sin x + \log x$ , and  $x(y + z)$  is misinterpreted as  $x y + z$ , if interpreted at all. The third problem is mathematical equivalence (“synonyms”). A sum or a product of several terms can be expressed in many equivalent ways; numbers can be represented in multiple forms (e.g.,  $1/2$  vs.  $0.5$  vs.  $2^{-1}$ ); polynomials can be expressed in many factored and unfactored forms; and so on. Standard thesaurus-based approaches in text search are not adequate for searching for mathematically equivalent forms. The fourth problem is the issue of levels of abstraction in mathematical contents and queries. Should not a document containing  $f(x)$  match a query “ $f(t)$ ”, or a document containing  $f^2 + g^2$  match a query “ $f(x)^2 + g(x)^2$ ”? The fifth problem is that of notational ambiguity in mathematics. Is  $dy$  the product  $d \times y$  or the differential  $dy$ ? Is  $x(t + s)$  the product  $x \times (t + s)$  or the function  $x$  applied at  $t + s$ ? Markup languages can eliminate ambiguity in the database contents, but not in users’ queries.

In this talk, we will discuss the efforts and approaches for addressing those problems and, generally, for developing math search techniques and systems. At the keyword and structural levels of math search, two methodological strategies will be covered. The first is the evolutionary strategy of building a math search engine on top of a text search engine, and translating math contents (including the symbolic and the structural) in the database and in queries into textual counterparts for text search. The second strategy is to develop new schemes to index equations and structures directly, using parse trees and possibly other models, to increase the precision of searching. Common to both strategies is the necessity of an effective and easy-to-use interface for entering and editing math queries. The math search system being developed at NIST for the DLMF project will serve as an illustrative focus.

The talk will also cover pertinent aspects of the activities, projects, and products that are related to, or have impact upon, math search. These include math markup languages (OpenMath, MathML, OMDOC), math ontologies and metadata, math languages/editors (EzMath, MathType, MathWriter, etc.), Math on the Web (MathWeb, MONET, the Esprit Project), and industry support.