G Math – A New Paradigm for Mathematics

Mahesh Chugani and John Hanks

Introduction

LabVIEW from its inception has been developed primarily for the purpose of improving measurement tasks. Until recently, it was not clear that LabVIEW or the graphical programming technology could deal with extremely complex mathematical routines. There have been some examples created using the G programming language paradigm to implement mathematical algorithms, but most of these past examples were written only for demonstration purposes. However, some of these examples can now be considered the first examples of a new paradigm for mathematics – graphical programming, or G Math.

Until recently, most mathematical literature about numeric algorithms had been totally dominated by the classical procedural programming languages such as FORTRAN, C, Pascal, and so on. The new approach is to use LabVIEW with its graphical programming language, G. There are numerous advantages to using LabVIEW, some of which are:

- Visual programming language
- Data-driven system design
- Virtual instruments concept
- Platform independence

The question arises whether G has the flexibility and speed to build up a rich collection of important mathematical functions and procedures. The answer is relatively simple if G is compared to the classical manner of code implementation in languages such as C, FORTRAN, Pascal, and so on. Because LabVIEW is a full programming language, it is able to perform the same functionality as traditional text-based code. However, we believe that G, a well defined, data-driven, graphically oriented programming language, represents the most suitable description of a given numerical algorithm.

LabVIEW and the G Math Toolkit deliver a new paradigm for solving advanced mathematics and analysis problems. With the G Math Toolkit and the LabVIEW G programming language you can graphically assemble your mathematics program as a block diagram. In addition, the ability to integrate interactive front panels introduces a new level of interactivity to math and analysis problem solving. Overall, the compiled speed, connectivity, open architecture, increased productivity, and flexibility of LabVIEW and the G Math Toolkit delivers to you tremendous power for your applications.

The graphical user interface combined with the data driven paradigm make LabVIEW the perfect tool for complex measurement and analysis tasks. Mathematical operations and algorithms and measurement tasks have a lot in common. Therefore, LabVIEW and the G Math Toolkit can be used as a programming system for almost all kinds of numeric and symbolic mathematics.
Overview of the G Math Toolkit

The G Math Toolkit consists of a collection of interwoven libraries (Figure 1). The core of the G Math Toolkit are the VIs in the Parser Library, which make manipulating formulas on LabVIEW front panels possible. In addition, the G Math Toolkit VI libraries are interconnected, and leverage off each other. For example, the Ordinary Differential Equation (ODE) VI library is based on the Function and the Parser VI libraries. In addition, the G Math library relies on the Advanced Analysis Library, which is included in the LabVIEW Full Development System. The area of concentration of each of the G Math libraries is as follows:

PARSER – formula interpreter, used in all parts of the package. These VIs act as an interface between the end user and the programming system. They parse the user-written formula and convert it to a form that can be used for evaluating the results.

ZERO – root finding of 1D and nD real functions.

OPTI – optimization of 1D and nD real functions, complemented by linear programming and fitting algorithms.

1DEXPLO – investigation of 1D real functions, function graphs, extrema, zeroes, curve lengths and more.

2DEXPLO – investigation of 2D real functions, plots, extrema, zeroes, partial derivatives and more.

TRANS – collection of important signal transforms not implemented in the LabVIEW Advanced Analysis Library.

VISUALIZ – tools to depict function graphs, both in 2D and 3D.

FUNCTION – collection of commonly used function definitions not available in the LabVIEW Advanced Analysis Library.

EXAMPLE – real-world applications of the G Math Toolkit based on well developed user interfaces.
The interconnections between all the G Math libraries are numerous, and in some sense, this is the true strength of the G Math Toolkit. As an example, the 1DEXPLO library needs the VIs from the ZERO, OPTI, FUNCTION, VISUALIZ and PARSER libraries. The user has the freedom to build up his or her own solutions of any desired level of complexity. The entire package consists of more than 100 core VIs, supplemented by dozens of subVIs. All these VIs can be used directly by the user. Thus, the strength of the complete G Math Toolkit is not only the power of its individual VIs in performing mathematical calculations, but also the capability of combining the VIs to solve extremely complex math problems. In addition, you can interface real-world measurements to the mathematical algorithms in order to obtain practical solutions. This flexibility makes the toolkit an extremely powerful analysis tool.

Examples

Here are some examples of how the G Math libraries can be used to solve mathematical problems of several levels of complexity.

1. Advanced Nonlinear Data Fitting

Often in practice, it is necessary to fit data with the help of given models. Sometimes these models are known in theory, and one can fit the data with well known models such as polynomials or trigonometric functions. In general, pairs of data \((x_i, y_i)\) are given, where \(y_i\) represent the measured data at time \(x_i\). The index \(i\) runs from 1 to the endpoint \(n\).

Consider the model equation \(y = f(a,x)\). The parameter \(a\) can be described by one or more components, i.e. it could be either a scalar or a vector. The goal is to find a value for \(a\), so that the sum is as small as possible; that is, one wants to minimize this sum.

\[
\sum_{i=1}^{n} |y_i - f(a,x_i)|^2
\]

One would usually solve this problem by using the Levenberg-Marquardt method (Reference 1), which is also useful in the case of nonlinear functions. The LabVIEW Advanced Analysis Library includes the Levenberg-Marquardt method. However, by using this, one has to fix the functional model \(f(a,x)\) before the runtime of the program. With the G Math Package one can enter the model equation in a formula directly on the front panel, thus providing the user a much greater degree of freedom. Differentiation between the model parameters and the independent variable is achieved by the components of a cluster on the front panel.

Figure 2 shows an example of a fitting problem. For clearer understanding, the data points were not obtained by a real measurement process. Instead, a special capability of the G Math Package was used, namely the generation of arbitrary function values based on a given formula. By using this procedure one can add stochastic terms to the definition, so that one can realize a real fitting problem. If the original function and the model equation have a similar structure, it is quite simple to compare the equality of the expected parameters with those of the given data.
Figure 2. With G Math, you can manipulate formulas during a fitting procedure. The model equation, the independent variable and the parameters can be entered in the cluster positioned at the left side. For clarity, the fitted data material is also produced by the help of a formula, in which an additional random term is used to simulate a real fitting problem.

2. Gibbs’ Phenomenon

The following example demonstrates the capability of the G Math Toolkit for educational purposes. Consider the Fourier series of periodic signals in the interval \([0, 2\pi]\). At the end of the last century, Michelson and Gibbs (Reference 2) realized that the Fourier series of any function forms an approximation of that function in a special sense. Viewing this convergence in greater detail, the approximation takes place in an integral sense but is not uniform for all points in the given interval. One can observe these effects by choosing simple discontinuous functions. An example is the step function, which can only have the values of 0 and 1.

Visualization of Gibbs’ Phenomenon can be easily shown with the G Math package (Figure 3). Not only does this package allow you to input a random function, but it also provides the ability to change the number of parameters and the order of the approximation (number of steps). One can notice the systematic behavior of the function in the region of the discontinuities. Increasing the order of the approximation does not help in reducing the magnitude of the deviation. Only the duration of this maximal magnitude becomes narrower.

The potential of using LabVIEW in the fields of education and simulation are tremendous. LabVIEW has many advantages over other math packages. With LabVIEW you can easily visualize data and connect the program to the real world with measured input data and data outputs. In addition, one has the G source code to customize algorithms or mathematical routines in the LabVIEW graphical programming language.

Another advantage is the simple and extremely easy program modification. This latter capability plays an important role in education and in simulation. Going back to the problem above, additional information about the absolute and square differences between a given function and its approximation can be depicted with minimal effort. In more complex problems, these experimental interests are even more strongly aided.
Gibbs' Phenomenon describes the systematic behavior of Fourier approximations of a function in the region of the discontinuities. Increasing the order of the approximation does not help in reducing the magnitude of the deviation.

Figure 3a.

Figure 3b.

Figure 3. Gibbs' Phenomenon describes the systematic behavior of Fourier approximations of a function in the region of the discontinuities. Increasing the order of the approximation does not help in reducing the magnitude of the deviation.
3. Partial Differential Equations – the Wave Equation

It is possible to model physical quantities such as gas density, gas pressure, velocity fields or electrical and magnetic fields with the help of partial differential equations. One of the most important partial differential equations, in practice, is the wave equation, which describes the spreading of acoustic fields in homogeneous media.

In the simplest case of a one-dimensional homogeneous wave equation, one can find a comprehensive formula with a closed-form solution (Reference 3). One is then faced with the question of finding a suitable calculating tool for the study of these solutions. It is desirable to implement sources that are spread out. In the case of acoustic applications, these sources are placed at various positions. In general, the interpretation of the closed-form solution is not intuitive.

Figure 4 shows the solution of the wave equation based on G Math. The number of sources and their behavior in space and time can be controlled by the user. The exact time and space behavior of the resulting field can also be inspected. One of the possible applications is the development of an artificial source with the special property that it can be implemented as being spread out in space and consisting of the simple superposition of different sources.

Figure 4a.
Figure 4. More complicated simulations can be realized based on G Math. The examples in Figures 4a, 4b, and 4c demonstrate this for the calculation and representation of the so called wave equation.
4. Construction of Optimal FIR Filters

The design of optimal filters is an extremely important problem in the field of theoretical and applied signal theory. In finite impulse response (FIR) filter design, one tries to reconstruct a given frequency behavior with the relatively simple model

\[ G(\omega) = \sum_{n=0}^{M} a[n] \cos(n \omega) \]

in which the coefficients \( a[n] \) are calculated. The FIR filter works without feedback. In contrast, the complex infinite impulse response (IIR) theory uses previously calculated and filtered data.

The desired frequency behavior of a filter is usually given as a sequence of characteristic functions defined on frequency intervals. In other words, one can realize the filter as the superposition of bandpass filters.

The LabVIEW Full Development System provides the Parks-McClellan algorithm based on an exchange operation (Remez Exchange). However, from a purely design-oriented point of view, other optimization strategies are also well suited.

With the G Math Toolkit, which includes optimization tools, one can reformulate this FIR filter design problem as a well known linear programming problem. Here the desired frequency behavior is realized in the form of linear inequalities (Reference 4), wherein one tries to minimize the deviation.

Figure 5 demonstrates the solution of the FIR design problem.

![Figure 5a](image-url)
5. Differential Equations – a Practical Problem

Finally, a short description of the abilities of G Math in solving differential equations is demonstrated. G Math provides the Euler, Runge-Kutta and Cash-Karp methods, along with some numeric and symbolic routines for solving systems of linear differential equations and differential equations of the $n$th order, respectively.

This particular example describes the motion of a spinning top. The motion can be realized in 3-dimensional space with a fixed center of gravity (Reference 5). There are a lot of factors influencing the behavior of this toy. The parameters are arranged on the left part of Figure 6a, opposite the solution in 2-D phase space notation. Figure 6b presents the solution as a 3D curve.

The possibilities for inputting measurement data as parameters of differential equation models and returning the results via DA-transforms are additional important advantages of the G Math Toolkit used in connection with LabVIEW.
Figure 6. The solution of the spinning top problem. Figure 6a shows the input of the differential equations that are used, and the phase space representation of the solution. Figure 6b depicts the solution graphically in real 3D world coordinates.
Conclusion

The new areas of mathematics opened by the G Math Toolkit are a valuable addition to LabVIEW and its Advanced Analysis Library. In numerous important problems, the user can combine actual measurement processes with user-controlled data analysis. In the fields of mathematics, signal theory, and general technical education, G Math makes complex mathematics and simulation possible.

References

2 Alan V. Oppenheim, Alan S. Willsky; *Signals and Systems*, Prentice Hall, 1983, Chapter 4.3