# Goergen Teaching Award Essay

Michael E. Gage
Department of Mathematics
University of Rochester

September 1998

## 1 Introduction

What excites me about teaching is the organization and presentation of ideas. It always seems like I learned this stuff the hard way and I want to explain how easy (and fun) it really is. There is some self-delusion in this attitude – usually everyone has to learn the hard way – but a little self-delusion can be useful in teaching.

Let's divide the knowledge business into three parts: research, scholarship and teaching. Research consists of making new discoveries and proving new theorems. Scholarship consolidates these discoveries; creating unifying theories and isolating the new insights which made the discoveries and theorems possible. Teaching passes the knowledge, insights and some of the discoveries on to the next generation of knowledge workers.

I had mathematics and the sciences in mind when I made this division, but I think it works for the humanities as well. For example during the process of presenting orchestral music, the composer does research, the musical theorists and historians do scholarship while the conductor and musicians do the presenting (teaching). And these last two do scholarship as well, witness the conductor who researches and presents something besides the current classical top 40. All of these activities are frequently performed by one person and in all cases this compartmentalization is not very precise; there is a great deal of overlap.

Mathematical scholarship can be rather invisible. Research – brand new results, even relatively predictable results – lead to publication, and one knows how to count that (although most people, inside and outside mathematics, are hard pressed to define concretely in layman's terms what mathematical research is). Teaching in front of a classroom is frequently under rewarded, but it's clear what one is doing, or trying to do, or at least that one is doing something. Mathematical scholarship: reworking known mathematics, some of it decades old, reorganizing it, explaining it from a modern perspective, making connections across the subfields of mathematics and sometimes other fields, this happens most frequently in departmental coffee rooms, graduate seminars and courses and expository lectures and articles. It happens also in undergraduate classes, but not as frequently.

I've tried to apply mathematical scholarship seriously to teaching my undergraduate courses and I've found the greatest opportunities in those courses lying outside the core of the standard mathematics graduate school preparation curriculum: numerical analysis, discrete mathematics, differential equations, non-Euclidean and projective geometries. Many of these courses have been taught routinely for many years, the majority of the standard texts are routine, and perhaps because the audience is largely non-math majors, both audience and instructors have been content with the status quo.

For good or ill, I'm seldom content with the status quo.

What about calculus? This is the most frequently examined undergraduate math course, in fact with the possible exception of freshman English and "western civilization", it's the most frequently examined undergraduate course period. I see great consensus on how this subject is to be understood – all the commotion (read reform calculus) is over how it should be presented. (In contrast, a move to base undergraduate calculus courses on non-standard analysis – a genuinely different approach to calculus based on Abraham Robinson's results from the '60s – was seriously considered by some during the late '70s. That movement is now quiescent if not abandoned. ) I'm largely in sympathy with the goals of the reform calculus movement (not all of them – it is a rather diverse group) but I find it useful to recognize that they largely concern the communication or transmission of the subject rather than its conceptualization. The act of teaching the subject, getting the students to learn and understand the material, requires that a connection be made, and the emphasis is on both the students and the subject matter. Motivational skills, empathy, showmanship, interpersonal skills and wide range of other factors not connected to the subject matter come into play in this activity. Important as these are, its the reformulation of the subject itself that fascinates me most.

In using computer graphics programs for calculus and differential equation classes, and in helping to design and develop WeBWorK, I've made my own contribution to changing the presentation of calculus, and I'll say more about this below. Meanwhile, in terms of the conceptualization of calculus, I agree with the consensus about how calculus should be understood, even though my own conversion as an undergraduate from physicist to mathematician (logician even!) came from being introduced to non-standard analysis. Maybe some year, if there is enough slack, I'll get out my old notes and design and teach a course.... watch out!

With calculus out of the way what else is there to know about undergraduate mathematics? Well everyone should have some idea of what mathematics and mathematicians have done, what the mathematical community is attempting to do now, what it can hope to accomplish, and just as important what it is not attempting and cannot accomplish. Addressing this problem is enormously important for society and for the mathematics and scientific community, and the mathematics community, somewhat belatedly, is coming to realize this. Using the musical analogy this means giving concerts and teaching music appreciation courses in addition to giving piano lessons. The courses I discuss in detail below, however, are piano lessons, not music appreciation; the goal is for the student to be able to perform at some level even though the student does not plan a career as a concert pianist, but as a music teacher, composer, elementary school teacher or (best of all) as an amateur musician. One might argue (most mathematicians, including myself, have) that this is in fact the best kind of mathematical appreciation class, but it's not for everyone.

I've not taught a mathematical appreciation course, but I have made fledgling efforts to address this need through the organization of SUMS lectures and the math awareness week presentations. These co-curricular activity may really be the best place to address the mathematics appreciation problem within the university setting, since it does not require a long term commitment on the part of the audience. I don't know that anyone has yet a solid understanding of what it takes to make this approach effective and successful. There is a lot room for experimentation. At the very least it will take commitment from universities, colleges and mathematics community so that mathematicians and scientists will be willing to devote energy to this kind of communication and substantial advertising, perhaps even propoganda, will be required to convince the public that it is worth listening to the message.

I do worry that in the rush to make calculus accessible to everyone ( a worthy goal) the resources to present the wider view of mathematics are gobbled up. (Would you give piano lessons to everyone at the expense of concerts? )

For technical majors there is much more that they should know about mathematics than just calculus. (At the very least they need linear algebra and perhaps differential equations.) Particularly at a small research university such as Rochester the math department needs to fight to maintain the diversity in upper level mathematics offerings and make sure there is sufficient audience by insuring the recognition of the benefits that a broad diversity of courses offers for non-majors as well as majors. This diversity of upper level offerings is one of the important differences between a research school and an undergraduate institution. (Would you give only piano lessons to everyone at the expense of music lessons for amateurs on all the other instruments??)

Returning to my central theme, the aspect of teaching undergraduates which most interests me is the scholarship involved in designing new courses, or more frequently, redesigning old ones. In all fields of knowledge, and especially in the sciences, the frontiers keep pushing onward, and the background needed to make a contribution keeps increasing. This is a particular problem for the sciences and mathematics where knowledge is unusually hierarchical; you can not learn it in any order, and to learn advanced material usually requires mastery of a great many prerequisites. Fortunately, in the process of pushing the frontier onward, mathematical research also unearths new ways of conceptualizing existing results and sheds new light on traditional concepts. What was once difficult and exotic becomes obvious and fundamental. Not every new discovery follows this path; some move from the exotic to the esoteric to the irrelevant. The challenge is to recognize which new insights should be used to speed up and compress the mathematical education of new knowledge workers and to introduce these into the redesigned courses – the rest can be left for the experts. The challenge for mathematics is particularly daunting as the physical sciences continue to use more sophisticated mathematics and the social sciences, particularly economics, become more mathematically based. Both our calculus classes and our advanced classes have a greater diversity of students than ever before with widely varying backgrounds, needs and goals.

So where does technology come into this? I suppose I'm best known for my use of computers in teaching, and I've hardly mentioned them so far. The first thing to recognize about computers is that, in education at least, they are primarily communications devices, not number crunchers. I suppose that with the explosive growth of the world wide web this is now obvious to everyone, but even five years ago this was a novel idea to many people. I was fortunate to learn this viewpoint of computers in the early '70s, when I was graduate student at Stanford from contact with Douglas Engelbart. (Doug was one of the early implementers of hypertext, and someone whose novel ideas have still not been fully exploited – see http://www.bootstrap.org).

The new communication capabilities provided by computers, and particularly computer graphics, is one of my stimuli for rethinking both how to present material, where to present it (it's not just for classrooms any more) and more profoundly, which material it is now possible to present and in what order. Computer graphics can have a profound effect on the hierarchies of what must be learned first. Where one once needed to learn some technical skills before even being able to imagine the mathematical objects, such as the solution to a differential equation, now they can be presented, (although perhaps not really understood) in a pictorial way. This proves to be a powerful motivator, even if it doesn't replace the need to learn technical skills. At least now the audience knows in some intuitive way, what you are heading towards and perhaps is more interested and motivated as a result.

At this point I wish to discuss in some detail four projects that I have worked on in the last ten years:

- Using computer generated graphs in calculus classes

  This is a nice addition to the classroom presentation, but as I point out below, it doesn't really change the way calculus has been thought of for the last hundred years. It may make the presentation more effective.

- Differential equations

  Computer graphics is used here also, and it has enabled a fresh approach to conceptualizing the subject, one which brings the researchers point of view to even beginning students of the field.

- Numerical analysis

  Here I did some mathematical scholarship in unearthing underlying unifying principles for predicting the maximum possible error in a large class of numerical methods. I didn't do any original research, all the facts were known to the experts, but they were not widely known and were seldom presented to undergraduates, despite the fact that they were not complicated. As result most mathematicians, engineers and scientists don't know these principles either.

- WeBWorK

  Using the computer to deliver homework assignments and offer instant feedback, doesn't change the content of calculus courses, but it can make a difference in how easily and well the students learn.

# 2 Graphing calculators, calculus and calculus reform

Although computers and calculators don't seem to have affected what should be taught in calculus courses, they have influenced how it is taught due chiefly to the computer's ability to quickly graph functions and thus provide a concrete visual presentation of the mathematical object. The ability to visualize the graphs of functions separates those students who can effectively use calculus from those who can't. Graphing calculators and computers serve as an effective aid for those who initially find this visualization difficult. Currently the range of options for graphing functions ranges from graphing calculators, to small graphing programs (such as Xfunctions, mathPad, NuCalc or Graphing Calculator) to full fledged numerical and symbolic manipulation packages (Mathematica, Maple, MathCad, MatLab). I've chosen to use the middle option, since I find graphing calculators limited in their input and output capabilities (but they are portable and cheap), while the symbolic manipulation packages have such steep learning curves that they distract the student from the immediate problem at hand: calculus. (I do use symbolic manipulation packages in more advanced courses.) While computer graphics has provided new opportunities for instructing calculus, the conceptualization hasn't changed much. I find the following summary of the situation enlightening and humbling:

> Of the various branches of collegiate mathematics doubtless none have received so much attention on the pedagogical side during recent years as the Calculus. ... The present situation may perhaps be described by saying that those pupils, relatively few in number, who really have the power to think may indeed understand the calculus now if they choose, but that the pupil who has only a fair measure of natural talent still finds it essentially difficult ... This may necessarily be so on account of the subtleties of the subject, but I am inclined to the belief that without seriously impairing the dignity or the value of the calculus, it may and should be brought still closer to the comprehension of the average pupil. In considering how this may be done, I am led to the idea which in substance is that the ordinary first course in calculus should be more largely graphical... The question which the beginner desires answered is, what is the calculus *about* — what does it *do*? The attitude is the same for any branch of scientific study when first approached and it is only after such questions have been answered that there is a natural inquiry into or need for the really exact. ... we should give a larger recognition to the merits of the so-called graphical calculus. We should also recognize more fully that it is the facts of the calculus and not the logical coherence of it that appeal primarily to the beginner, and this in turn necessitates a fuller recognition on the pedagogical side of the principle of successive approximation.

I quite agree with this formulation of reform calculus – I first read this passage reprinted in the Comments section of the February (Vol 100 #2) 1993 American Math Monthly. The words were written by W. B. Ford of the University of Michigan in 1910. (Monthly, 17(1910), 77–81) . Given a description of this unsolved problem and a potential strategy for solving it which are still accurate 90 years later, it seems to me that the calculus pedagogical problem deserves our healthy respect. The pedagogy of calculus has a long history, and while I always believe incremental improvements are possible, I doubt the prospects of finding a magical final solution any time soon.

## 3 Differential equations

The situation with differential equations is a bit different. Here the advent of computers, combined with the influence of the dynamical systems crowd, has allowed a significantly new way to conceptualize differential equations. Well, it's not new to researchers in the field, but it is relatively new viewpoint for undergraduate courses in the field.

Traditionally a first course in differential equations emphasizes the tricks required to analytically solve a handful of useful first order equations and second order linear equations. The properties of the solution are then deduced from this analytic representation. A second course in differential equations gives proofs of the existence and uniqueness of solutions and shows how to determine the qualitative behavior of solutions even for those cases where no analytic solution exists.

Leaving the proof of existence and uniqueness until the second course still seems like a good idea, but studying the qualitative behavior of differential equations should be emphasized from the beginning, for it turns out that even when you can find an analytic representation of the solution, it is easier to understand the solutions properties from the differential equation itself.

Two examples:

(1) The family of functions which solves the differential equation $y' = y$ is the family of exponential functions $y(t) = Ae^t$, and these functions are increasing when they are positive. But this fact is obvious from the differential equation, since the solution, what ever it is, has a positive derivative (slope) whenever it is positive, and hence must be increasing.

(2) The second example is not as obvious: Any family of functions satisfying the differential equation $y'' = -y$ must satisfy certain identities, such as $(y')^2 + y^2 = \mathit{constant}$. The solutions are combinations of sine and cosine and $\cos^2(t) + \sin^2(t) = \mathit{constant}$, but the point is that this fact can be determined **before** the solution is known explicitly. You can calculate the formula for the sin or cosine of the sum of two angles directly from the differential equation as well.

When you think about it, it is not clear how you calculate exponential, sine and cosine (sure they are buttons on your calculator, but how does the calculator calculate them?), and their properties are really determined by the fact that they satisfy certain differential equations, not the other way around.

The core of this qualitative approach is to take the graphical interpretation of the solution of a differential equation seriously. This means using direction fields and phase plane diagrams from the beginning of the course. The use of computers as communications facilitators i.e. their graphing capability, makes this possible at an early stage, without an initial emphasis on analytic computation. The graphical techniques work equally well for linear and nonlinear equations, even when there is no analytic method for obtaining explicit representations of the solutions of the latter. The growing use of nonlinear equations in physics, chemistry, biology and economics and the existence of computer programs such as Mathematica which can easily find exact analytic solutions to linear equations, increase the importance of understanding the underlying principles.

A second concept I've tried to introduce is to explicitly discuss the actions of scaling on solutions to a differential equation (or in fancier, more general terms, the actions of Lie groups on the solutions of a particular differential equation). This is a method of creating new solutions to an equation from old solutions and is often unconsciously practiced by those applying differential equations. It is the motivation for much of Lie's work at the turn of this century. Examining this idea explicitly leads to the discovery of the sum formulas for sines and cosines mentioned above (and the laws of exponents as well).

I've played the role of a conductor, not a scholar, in this endeavor. Scholars such as Prof. Beverly West, Prof. Hubbard and Prof Devaney had already begun to point out how the graphical view point emphasized in the research into dynamical systems could be used to revolutionize the conceptualization and presentation of this traditional undergraduate subject. But the role of the conductor is an important one – without me, or someone like me, the insights and scholarship of Hubbard, West and Devaney would have remained available only to students at Boston University and Cornell and those few undergraduates who stumbled across their texts on their own.

Nothing in these "innovations" would surprise an expert in the field. It is choosing which new insights to bring into the undergraduate curriculum which requires artistry. And the taste of the conductor is important too, not every new thing is worth importing, I hope that I have chosen well. A university or college needs good conductors among its faculty – no institution can hope to invent everything.

I used Differential Equations by Prof. Hubbard and West, one of the earliest and most radical books of this movement, in teaching the honors calculus course in 1994. Since then many books and software tools have emerged, including a differential equations book by Prof. Devaney which finds middle ground between the new and old approaches to a first course in differential equations.

To see more of what can be done in a first semester differential equation course look at Prof. Devaney's review of "Interactive Differential Equations" (IDE), a CD disk by West, Strogatz, McDill and Cantwell which Prof. Adrian Nachman of our department has been using in his courses. (This review appears in the American Math Monthly, Vol 105 #7 p687-689). Here's an excerpt

> ...Better yet, open the Parameter Plane Animation tool and watch how the phase plane evolves dynamically as the parameters move around the trace-determinant plane. This is a topic that I never dared to introduce into my ODE course until IDE became available. Now, however, it is not only a staple of the course, but one of the real highlights for the students as well. They really "get it."

I hope I've interested you.

# 4 Numerical Analysis

A common view of numerical analysis, held by both engineers and mathematicians, is that numerical analysis consists of a collection of routine procedures. Titles such as "Numerical Recipes" certainly illustrate this attitude and the wide spread usefulness of numerical analysis stems from the accuracy of this viewpoint; you can make good use of the numerical methods without understanding them, just as you can cook well, up to a point, by closely following a cookbook. Understanding why these methods work is another matter.

I came to the subject fresh – I had never taken a course in numerical analysis nor used it in my research. My research into motion by mean curvature lent itself to computer graphics movies illustrating the motion of curves in the plane whose velocity is proportional to their curvature. Others had made such movies (it requires a certain amount of numerical analysis expertise) and I wanted to understand the principles behind how it was done. I decided to teach the numerical analysis class to learn about the subject.

Understanding what the numerical methods do is not hard, and finding a proof that they will work is only slightly more difficult, but neither is very enlightening. The explanations I found in the popular elementary textbooks were ad hoc – each method was justified in different, but similar ways. I felt that one should be able to describe the subject in terms of three or four basic principles and the rest could be viewed as variations on these basic themes. After teaching this course several times I feel that I have made significant progress towards this goal, not from any original discoveries, but from unearthing and presenting unifying principles, many discovered over 80 years ago, which are simple and enlightening, and known to experts in the subject, but which have failed, until recently to become part of the undergraduate curriculum.

## 4.1 The technical part

How do you integrate a function numerically? Integration, "finding the area under the curve" starts with a function (represented by a graph) and ends with a number (the area). It is found numerically and theoretically by approximating the area by many rectangles, or trapezoids or by other shapes. While there are many methods for finding this area, almost all of them share a common set of properties called "linearity". The Peano theorem uses only these linear properties, so in one sense it is possible to understand the theorem without understanding any details about the theory of integration. This illustrates decisively the power of abstraction in problem solving. Since only these "linearity" properties are involved, Peano's theorem can be applied to any operation which is linear (and there are many of these). The demonstration below also illustrates what an abstract understanding alone does not accomplish – understanding Peano's theorem at the "abstract" or logical level, which can be done with out any knowledge of calculus, will not be very productive unless one also has some knowledge of the methods to which this principle can be applied and why the resulting information is useful. As is the case for the duller books on differential equations, it's been the failure to take both abstraction and practice seriously which has caused the collection of bad books in numerical analysis.

Here is how Peano's theorem works: Let's assume that the integration operation $I$ finds the area under a curve $f$ on the interval from 0 to 1. The curve $f$ might be defined by a formula such as $f(x) = x^2$ or $f(x) = x^3 - 3x + 1$.

The integral has these properties with respect to its input:

1. $I(0) = 0$ (the graph of the constant function 0 has area 0.)

2. $I(f + g) = I(f) + I(g)$ (if we make a new function by adding the two functions $f$ and $g$, the area under this new function will be the sum of the areas associated with the original functions.)

3. $I(mf) = mI(f)$ (If we make a new function by multiplying the function $f$ by a constant $m$, then the area of the new function will be $m$ times the area of the old function.)

These three properties make the integral a "linear" function. The name comes because it is shared by the function $h(x) = mx$ which has the graph of a straight line which goes through the origin. It is a little misleading however, since the technical meaning of this term does not apply to all straight lines, only those which go through the origin.

More interesting examples come from matrix multiplication, (when x is a vector and m is a matrix) and from applications such as integration or differentiation where the input object is not a number but a function.

Estimating the integral is also a linear operation. For example to estimate the integral by a single rectangle, we might choose a box whose height is the height of the graph at the midpoint. Our estimate becomes $L(f) = f(0.5)$. (The height times the width which is 1 in this case.) Or we could estimate it from the area of a trapezoid in which case the formula is $L(f) = (f(0) + f(1))/2$. These are the midpoint and trapezoid rules used for numerical methods. Many more examples are easily thought of and most of them are given by formulas of this type.

$$L(f) = a_0 f(x_0) + a_1 f(x_1) + \ldots + a_n f(x_n) + b_0 f'(x_0) \ldots b_k f'(x_k)$$

although one could also include higher derivatives of $f$ as well.

You can check that the function $L$ is also a linear function – it satisfies the same three rules that integration does. Further the difference between these two functions $E(f) = I(f) - L(f)$ is also a linear function since it satisfies the three rules.

Next comes the Taylor polynomial. The philosophy is that many useful functions can be represented pretty well by a long polynomial. This result is taught in the second semester of calculus, but not used extensively except in exercises. Peano's theorem illustrates the problem solving power that follows from the polynomial approximation of functions.

More precisely stated Taylor's theorem guarantees that many functions can be written in this form

$$f(x) = A + Bx + Cx^2 + Dx^3 + R(x)$$

where $A, B \ldots$ can be calculated from the derivatives of the function (e.g. $A = f(0)$, $B = f'(0)$, $C = f''(0)/2!$ and $D = f'''(0)/3!$.

The remainder $R$, the difference between the function $f$ and its polynomial approximation, can be written as an integral:

$$R(x) = \int_0^x f''''(t)(x-t)^3/3! dt = \int_0^1 f^{(4)}(t) p_3(x,t) dt$$

where the function $p_3(x,t)$ is 0 when $x - t$ is negative and equal to $(x-t)^3/3!$ when this quantity is greater than 0. The functions $p_3(x,t)$ are called truncated polynomials and play an important role in estimating the error of numerical methods. They are also related to the fundamental B-splines which form the basis of understanding splines. The above formula means that the remainder can be written as a superposition or weighted sum of these truncated polynomials. The weights are given by the fourth derivative of $f$.

Now comes the payoff. Suppose that we have constructed a method of integration which does a perfect job (no errors) of integrating polynomials of order 3 and lower (Simpson's rule is such a method). We want to calculate the potential error that such an integration method will have when applied to a larger class of functions. We want to calculate $E(f(x))$.

$$\begin{aligned} E(f(x)) &= E(A + Bx + Cx^2 + Dx^3 + R(f)) \\ &= E(A + Bx + Cx^2 + Dx^3) + E(R(f)(x)) \\ &= E(R(f)(x)) \end{aligned}$$

The second line follows from the first because the estimating error function is linear, i.e. $E(g+h) = E(g) + E(h)$. In this case $g$ is a polynomial of degree three, and the estimating error is zero for polynomials of degree three or less.

Next we need to know that

$$\begin{aligned}
E(R(f)) &= E\left(\int_0^1 f^{(4)}(t)(x-t)_+^3/3!dt\right) \\
&= \int_0^1 E(f^{(4)}(t)(x-t)_+^3/3!)dt \\
&= \int_0^1 f^{(4)}(t)E((x-t)_+^3/3!))dt \\
&= \int_0^1 f^{(4)}(t)K(t)))dt
\end{aligned}$$

where the symbol $K(t) = E((x-t)_+^3/3!)$. This calculation again uses the linearity properties of the estimating error function $E$.

(The fact that $E$ can be "moved inside the integral sign" is not too surprising. It would be true if the integral $I$ were defined by a finite sum as $L$ is. Since the integral is defined using limits as well as sums there is room for doubt since not everything that works well with sums (such as a linear function) will work in the same way with limits and sums or with infinite sums. Not withstanding this caution flag, a mathematician's or scientist's first guess, that what works in finite case will work in infinite cases as well, turns out to be completely correct in this case.)

What have we accomplished? By applying the integration method to the functions $(x-t)_+^3$ and calculating its error we obtain a new function $K(t)$. We can now calculate the error for EVERY function $f$ as the weighted sum of the errors arising in estimating each of the truncated polynomials.

If we can't perform this integration exactly, we can still get an estimate by using the largest value of the fourth derivative of $f$ and multiplying by $\int K(t)\,dt$ which can be calculated explicitly. Another possibility is to use the Schwartz inequality $((\int f^{(4)}K)^2 < \int |f^{(4)}|^2 \int |K|^2)$ to estimate the error in terms of the mean square norm of the fourth derivative of $f$. (This latter possibility is never mentioned in standard texts, but is an obvious consequence of this approach to error estimation.)

## 4.2 So what?

First – in page or so – I have presented a general method for calculating errors for nearly every method of calculating integrals. It applies also to methods for interpolating functions, and for numerically solving differential equations. (Runge-Kutta methods, which are not linear, are the only major exception to the applicability of Peano's theorem.) Many elementary texts present these error estimates individually, one for each integration method, each time using a different trick and each time requiring at least a page to present the proof. Here is a list of numerical methods, taken from a popular numerical analysis text, whose potential errors can all be estimated following the basic method outlined above:

1. Interpolating (approximating) functions by polynomials.

2. Hermite and cubic spline interpolations (and all other piecewise polynomial approximations).

3. Integration methods including trapezoid rule, midpoint rule, Simpson's rule, all other Newton-Coates integration and composite integration methods Romberg integration, Gaussian quadrature,

4. Differential equation solving methods including: Euler's method, higher order Taylor methods, multistep methods such as Adams-Bashforth and Adams-Moulton (but not Runge-Kutta which is nonlinear).

It's kind of amazing that one idea can apply in an essential way to nearly every method described in 200 pages (pg 78–281) of Burden and Faire's Numerical Analysis, 3rd edition and yet the idea doesn't appear at all in this and most other introductory numerical analysis books.

Peano's theorem and proof which I've provided is the key for understanding the common thread for all of these numerical methods. Those who have absorbed the argument will have little trouble deriving the error bounds for any of these methods. Since functions can also be represented by Fourier series (a series of trigonometric functions, rather than a series of polynomials) the abstract approach presented here automatically suggests that approximating integrals using trig functions as elementary functions rather than polynomials might be useful and also suggests how the errors in such an approach could be estimated.

### 4.3 Why hasn't Peano's theorem been used all along?

Who knows really? But I'll speculate anyway. I think that a large role has been played by the dichotimization of mathematical knowledge. These dichotomies are useful for understanding the field but they better represent the ends of a continuum rather than two disparate choices. Since numerical analysis is for applied sciences and mathematicians other mathematicians have not noticed how nicely it displays the usefulness of pure, theoretical results from linear algebra and functional analysis. Even if an instructor of, say a functional analysis course, is aware of the application there is never time to explain these applications in the pure mathematics courses (and in fact while the explanations can be understood in the abstract, their usefulness can't really be appreciated unless you also understand and work with some of the numerical methods. On the other hand those taking (and sometimes those teaching) the applied course just want to use these results, and resist spending the time to understand them better, even though that extra effort will lead to better more intelligent use of the numerical methods.

## 5 What is WeBWorK?

It's an internet based method for delivering homework problems to students. It provides them with immediate feedback about their answers.

Each WeBWorK problem set is individualized – each student has a different version of each problem. The student downloads a copy of the assignment, does as many problems as he or she wishes and then logs onto the internet and enters their answers into a web browser. The WeBWorK system responds telling the student whether an answer (or set of answers) is correct or incorrect. WeBWorK does not tell the student the correct answer, but he is allowed to try a problem as many times as he wishes until the due date.

You can try WeBWorK for yourself by using a world wide web browser such as Netscape or Internet Explorer and typing the address

http://www.math.rochester.edu/webwork

into the location/address field at the top of the browser window. From there you can try out one of the courses currently being offered as a guest or "practice" user. As a practice user your experience is the same as that of a student enrolled in the course, but your answers are not recorded.

Clicking on the "documentation" link, or typing the address

http://www.math.rochester.edu/webwork/docs

takes you to an extensive description of both the technical and educational aspects of WeBWorK, some of which I summarize in this essay.

## 5.1 Educationally what are the goals of using a web based homework delivery system?

WeBWorK is not an attempt at Computer Assisted Instruction (CAI) in the usual sense. The program does not teach; it merely gives immediate feedback as to whether or not the student has submitted the correct answer. If the student is unable to find and correct their mistake in a reasonable amount of time, the student is encouraged to seek help elsewhere, from fellow students, from TA's or from the instructor (either in person or via e-mail). Hints can be provided based on the type of wrong answer, but I prefer not to do this. It is not clear to me that such hints would help the learning process. It's preferable that when a student is having difficulty he or she should be encouraged to seek help from humans. ("When a student appears to need help, an instructor should appear to help him.")

In many ways WeBWorK has a conservative educational philosophy – roughly based on the premise that students who do their homework learn more. WeBWorK acts as a kind of surrogate tutor, it won't tell the answer, but it will tell whether the problem has been answered correctly, and that information is enough to motivate most students to keep trying until they get it right. Most students in our classes (more than 80%) get all of the answers correct before the due date. Without WeBWorK most students will make a calculation, get an answer, and forget about the problem until their homework is graded and returned a week (or more) later. In this sense WeBWorK promotes "mastery" of the material, not just familiarity. Students cannot mislead themselves into thinking they understand a problem. To misquote Tom Leherer – with WeBWorK "It is important to understand what you are doing AND to get the right answer."

It's also clear that WeBWorK is not appropriate and will not work for every type of homework assignment. For our calculus courses we assign both WeBWorK and traditional text problems as homework. WeBWorK does work very effectively for the many problem solving and knowledge demonstration questions which have a correct answer or a small number of correct answers.

## 5.2 What are the educational effects of WeBWorK?

There has been no comparative study of the use of WeBWorK, but there is no doubt, from anecdotal evidence alone, that using WeBWorK changes the social workings of the classroom. Students frequently work together to find answers to "tough" questions that no one seems to be able to get the computer to accept. Occasionally students show that they are right and the computer is wrong. (This was particularly common when the WeBWorK project was just beginning.) The lively e-mail correspondence between the student and instructor encouraged by WeBWorK testifies to a new dynamic in the learning process. A new communications and learning channel between students and their instructors has been created.

Because the problems are individualized (for example the numerical values in the problem will be different) students cannot merely copy answers from another student – they almost certainly have to learn something in order to get their answer right. Students can (and do) learn general methods for solving these problems from each other, but at this stage in their education that is exactly what we are trying to teach them! Coaching on how to invent original methods of solving problems is not something that WeBWorK can do, but with the exception of the mathematics honors and Quest courses and the Putnam exam practice sessions, this is not a major focus of our first year offerings.

Another anecdotal observation is that students are much more focused in the questions they ask TA's and instructors. Usually there is a specific part of a problem causing them difficulty and since they have already worked hard on the problem, they are more likely to really understand the instructor's explanation.

Finally, by giving "WeBWorK" exams where students receive partial credit for successfully answering via WeBWorK questions they missed on the regular in-class exam, it is possible to turn exams into a powerful learning experience while at the same time giving students a second chance to do better on exams. Here is how it works: the students take a regular in-class exam; questions involving essay answers or "show your work" types of questions are graded as usual with partial credit; questions with numerical or one word answers are graded without partial credit, but individualized versions of these questions appear on a WeBWorK assignment. Students who missed these questions on the in-class test obtain partial credit by correctly answering individualized version of the question on WeBWorK. This helps students to really think about their incorrect answers on the exam and to learn from their mistakes. They receive partial credit, not for exhibiting a hazy knowledge about the problem, but for getting the answer completely right, albeit a bit late.

## 5.3 A brief history of WeBWorK:

In 1995 the physics department, in particular Prof. Frank Wolfs, asked that the mathematics department offer a calculus course (MTH161) for incoming freshmen linked to the freshman mechanics course (PHY121). Students would register for both courses and the material would be loosely coordinated. In addition both courses would use a homework delivery program called CAPA, designed and programmed at Michigan State University, which allowed students to use computer terminals (but not web browsers) to answer homework questions and receive immediate feedback. Given my interests in using new technology in education, I was asked to teach the math section of the course in the fall of MTH161.

Educationally CAPA has all of the assets (and liabilities) I have described above for WeBWorK. However I found writing new mathematics problems in CAPA quite difficult and sometimes impossible; particularly when I tried to write problems that were somewhat different from the word problems originally envisioned by the CAPA designers. Since CAPA is a closed system, I could not change it myself, but had to wait for the MSU programmers to fix bugs and add features. After attempting to write "front end" programs which would make writing CAPA problems easier, and exchanging ideas with a mathematician at MSU who had also tried to write such programs, I decided instead to write my own version of CAPA. Another mathematics professor, Arnie Pizer, joined me the next semester and WeBWorK was born.

We have been enormously aided in developing WeBWorK by our undergraduate interns Basem Moussa, Scott Douglass, Leeza Pachepsky and Jason Sundrum who have worked both during the summers on development and during the school year as TA's. We have received strong support, both money and in encouragement, from the College, especially from Dean Green; from the Mathematics Department and its chairs Joe Neisendorfer and Doug Ravenel; and from ResNet and the University Computing Center (UCC).

Since the chief difference between CAPA and WeBWorK is not in the educational philosophy but in the software design and implementation, let me say a few words about my ideals for software design, particularly educational software design, and how WeBWorK measures up to these ideals.

## 5.4 "Don't fence me in."

That's among my top demands of educational software. (Along with reliability, ease-of-use, etc... there's actually a long list.) I want open ended educational software, the kind that can be extended by the user, or combined with other software, and with non-electronic methods of communication. Otherwise the instructor is locked into the educational model imagined by the software's author and this does not lead to the best form of education. The constraints placed on a course by using a specific text are rigid enough, and I prefer that electronic aids to education be less restrictive than texts, not more.

Not surprisingly then, I designed WeBWorK from the beginning to be extensible, particularly in the way that the homework problems are designed and programmed. Much like TeX, the mathematics typesetting program widely used among mathematicians and physicists, WeBWorK provides an underlying structure which is very powerful, allows many choices and has a substantial learning curve. In WeBWorK's case the underlying structure is a substantial portion of Perl, a marvelous scripting language which is widely used to author dynamic web sites. On top of this basic programming structure is placed a layer of macros (similar to the Latex flavor of TeX) which reduces the number of choices, simplifies the creation of problems and reduces the initial learning curve considerably.

This layering of the capabilities of WeBWorK allows an instructor unfamiliar with WeBWorK to at first just choose problems which have already been created to use in his or her course. Soon he or she will begin to tweak these problems to suit his needs and will begin to gain an understanding of how they are written. Eventually (I hope) the instructor conceives of new problem types problem types which are more educationally effective than those imagined by the WeBWorK authors. The instructor can construct these questions himself using the underlying Perl language structure. It is seldom necessary to redesign the WeBWorK core or consult with the original programmers in order to add new capabilities. If the new problem types are widely useful, the instructor may author new macros that make it easier for the next instructor to write these new types of problems.

CAPA is a self-contained program and the advances in processing must come from improvements to CAPA itself. WeBWorK builds on the freely available technology of others, knitting together existing web programs, leveraging the work done by the programming community of the internet in order to accomplish the same goals as CAPA. One consequence of this is that any improvement of WeBWorK components (for example the switch from Perl4 to Perl5 scripting language, the introduction of the Microsoft web browser or the new revisions to Latex2HTML) has an immediate effect on WeBWorK as well. Usually, little or no change is needed within WeBWorK itself to take advantage of the improvements. Tying WeBWorK to the world wide web (which was fairly young when WeBWorK started in 1995 – now look at it!) allows any computer which can work with the internet to work with WeBWorK. It is because of this use of commonly available web programs that WeBWorK has been able to match and in places exceed CAPA's capabilities in a fairly short span of time. The pattern of community development of software, already well developed by the group surrounding Perl, is now being widely discussed on the internet. It's an important point to consider for educators planning to build educational software.(See Eric Raymond, the Cathedral and the Bazaar, http://sagan.earthspace.net/esr/writings/cathedral-bazaar .)

WeBWorK doesn't (yet) meet my own criteria for excellent software. In particular it is too difficult to set up initially on a new computer system and while there is a lot of documentation it is not complete and needs better organization. Right at the moment WeBWorK is quite easy for students to use, not too hard to use for instructors, especially if they use existing problems, and fairly difficult even for Unix gurus to set up. ( But they're used to it :-) ) WeBWorK is undergoing constant development by Prof. Pizer, our team of student interns and myself in order to remove these deficiencies and improve the effectiveness and ease of use of the program.

Instructors at the University of Rochester have a decided advantage since the program is already set up and running at our university and the WeBWorK authors are close by to provide aid and advice. Anyone interested in learning more about WeBWorK, particularly those who might like to use it in their own classes, should visit the web site at http://www.math.rochester.edu/webwork to try out the program directly and contact either Professor Pizer (apizer@math.rochester.edu) or myself (gage@math.rochester.edu) for more information.

This fall (1998) WeBWorK will be used in most of the first year calculus courses taught at the University of Rochester (over 700 students). Other institutions, notably Kansas State University and The Johns Hopkins University, are beginning evaluate WeBWorK as well.