Has the Moment Passed for Classical Solutions? Definitely Yes and Definitely No

Marvin E. Criswell
Department of Civil Engineering, Colorado State University,
Fort Collins, CO 80512-1372

Abstract

Today's computing tools facilitate rapid numerical solutions carried out to a precision unimaginable only a few decades ago. An initially logical conclusion flowing from this capability may be that the "classical" solution methods once used extensively in past professional practice for many engineering problems can now be declared obsolete and then discarded. But, many of these methods have purposes in the classroom and in engineering practice other than just producing numerical answers. They can be very useful in teaching behavior, carrying out preliminary design, and checking software output for reasonableness. In what can be considered a paradigm shift, today's young engineers are themselves no longer the primary producers of numerical solutions. They are increasingly users and managers of powerful software which carry out nearly all computationally-intensive tasks. We have also largely passed through a second stage computational maturity when engineers more often needed to prepare computer programs to carryout analysis and sometimes design. In the present third state, engineering educators need to prepare the graduates to be knowledgeable, efficient and safe designers in a professional environment where commercial software is pervasive. Classical methods, including moment distribution, and many approximate techniques have a reduced role in giving the graduate the skill to produce "exact" numerical solutions, but can and should have an even increased role, when appropriately used in synergy with computer resources, in equipping the graduate with the understanding and wisdom needed to be a successful practitioners in the present and future professional environment.

Introduction

The engineering student and young engineer today almost takes for granted today's computing environment which facilitates almost immediate numerical solutions to complex problems and with a computational precision unimaginable only a few decades ago. The necessity of using the "classical" (and often approximate) hand methods to get primary analysis results and design answers is in the past. It might be considered natural and logical that the old classical methods (both numerical and graphical) be completely replaced with "modern" methods. This conclusion uses the philosophy of why continue to teach methods that have become unnecessary and are thus obsolete. But let us not be so hasty. What are all the applications of these methods in the classroom? Can they be used to help the students to understand how structures and other engineered artifacts and systems behave, and can they help designers create and complete effective designs? Structural analysis using moment distribution is used as a primary example in this largely philosophical paper, with some other "classical" methods also noted.

Many classical methods, such as moment distribution, can be judged as no longer needed and no longer the best way to obtain design information in this age of powerful finite element and other design/analysis software. However, their simplicity can be a very significant advantage in their use to help teach basic behaviors. They also can be very valuable "back-of-the-envelope" sources of preliminary design information and checks of reasonableness of "computer" solutions.

Increasingly, a primary task of our graduates in their role of young engineers is to be intelligent users and managers of design and analysis software. If the pre-computer stage of 35 to 75 years ago is considered the "first stage" in modern civil engineering analysis, we have already largely passed the "second stage" when we expected our graduates to routinely work directly in setting up and solving matrix-based structural problems using computer program languages such as Fortran or Pascal or C. We are now well into a third stage where designers in practice have access to powerful software prepared by others, software representing an expenditure of programming effort impractical for the individual engineer not in software production to try to duplicate. Experienced engineers are extremely concerned with the increasing problem of new engineers misusing software and not understanding basic behaviors well enough to produce practical and realistic preliminary designs. Engineers at all experience levels find that the use of quick approximate solutions to check the reasonableness of computer results can be critical.

Although today's software tools frees the capable civil engineer from the tedium of detailed analysis computations and allows more complex designs to be considered, design also includes preliminary design and assessment/evaluation of designs. The moment for classical solutions has not passed, but their proper role has certainly changed. Rather than discarding these classical methods, we need to explore what their new role should be, now that they are free from the former emphasis being their sometimes tedious use to produce numerical solutions.

Background

Engineering design includes several steps – ranging from the creative conceptual and preliminary design stages, detailed analysis and component/member design, evaluation (assessment and refinement), and then to construction and beyond. It is in the steps of detailed analysis and component/member design that today's "modern" computer-based methods, increasingly embedded in powerful commercial software, have made the largest impact. Many of the "classical methods" which were the only tools available, or at least practical, before computers (BC?) for obtaining detailed analysis and component/member design are no longer routinely used in practice for these purposes. However, before we declare these classical methods to be obsolete and relegate them the past, let us proceed to examine what all are their uses in both engineering education and the practice of engineering.

Civil engineering practice is no more than two centuries removed from a period when the available analysis tools were very limited – moment distribution was not defined by Professor Hardy Cross[1] until in the early 1930's, soon after the completion of the Empire State Building in New York City, and many routine mechanics of materials equations and graphical analysis tools date back to no earlier than the mid-1800's. The creation of notable structures – such as gothic cathedrals, Eiffel Tower, and the Brooklyn Bridge – was not postponed until all these "classical" tools become available. Design was typically based on adapting successful design to new circumstances based on experience, experimentation, and extrapolation – with failures not unknown, and on subjective evaluation. Experience is valuable only if it leads to and is accompanied by insight, understanding, judgment and intuition – qualities that some individuals seem to inherently have more than do others. These attributes can be somewhat developed in the classroom, and they are certainly enhanced by on-the-job experiences. Our tasks in engineering education include helping our graduates to develop that elusive quality of experience in this broad sense of it including understanding, intuition and an ability to visualize and critique.

Improved analysis techniques, whether moment distribution in the early 1900's or computer-based matrix structural analysis techniques more recently, have primarily improved the evaluation process and moved it from being a qualitative assessment of the outcomes during and after construction to a largely quantitative assessment within the overall design process. Experimentation and the effects of extrapolation can increasingly be evaluated prior to construction, rather than by the lack or presence of physical distress and/or failure. Experience (including understanding, intuition, etc.) remains a very key component both in the creative portion of design (including in describing something which can then be analyzed) and in the overall evaluation and acceptance of designs. The roles of the classical methods in helping develop designer experience in the broad sense just noted cannot and should not be ignored, as they surely will be if these classical methods are quickly declared obsolete and discarded.

In an 1994 editorial page in the ASCE Civil Engineering magazine titled "Computer Analysis No Substitute for Experience", Anton Tedesko, the designer of many notable shell structures in the US and Europe, noted[2] that: "The use of computers has not diminished the value of back-of-an-envelope calculations. Intuition and experience guide a quick calculation, which may reveal the reasonableness or ridiculousness of a design before it gets too far". He further observed the dangers of elaborate analyses based on assumptions that were in error, neglected critical conditions, or used incorrect information; and were then accepted. Relative to overall design success and reliability, he notes "that God is almost always on the side of those with good judgment, talent and experience. Designers need the ability to understand basic behaviors and judge reasonableness".

What is a classical method or solution for the purposes of this paper? Primary emphasis will be given to approximate, often iterative, methods for modeling systems and producing answers by hand calculations or graphical constructions. Some, such as moment distribution applied to structural analysis, converge to the correct numerical solution with more iterations. Others entail assumptions known to be approximate, but useful to obtain an inexact but generally adequate answer – an example being the use of assumed moment contraflexure (M= 0) locations in the approximate analysis of laterally-loaded rigid frames. Classical methods can be considered to also include some computational "tricks" (now of limited use) and basic requirements of equilibrium and the mechanics of materials equations (of lasting importance). The basic principles of mechanics and basic mechanics of materials equations will not be considered among the classical solution methods/tools which are candidates to be discarded, although the software available today can facilitate design without their user having to directly utilize these basic tools.

One Classical Method - Moment Distribution

Moment distribution is one of several "classical" methods for conducting structural analyses within the computational limitations of the "pre-computer" design environment. This method is not as old as many might expect, as it is younger than many notable structures, including the skyscrapers of the early 1900's. It was defined by Professor Hardy Cross[1] in the early 1930's) as a way to get what could be considered "exact" analysis results using an iterative (relaxation) solution technique that converged to the correct solution within the common assumption of shear and axial deformations having negligible effects on bending moments and flexural shear. Other methods then existed (e.g., slope-deflection, direct integration) which for a "reasonable sized real structure" quickly produce a set of simultaneous equations that was impractical to nearly impossible to solve by hand methods. Practical analyses before moment distribution necessarily had to be of an approximate nature for most structures.

In the moment distribution method, all joints of the structure are initially assumed to be fixed to prevent both rotation and translation. Next, the moments at the member fixed ends resulting from the applied loads acting on the beam are determined. In the usual case, the moments from all members entering a joint do not result in the joint being in equilibrium. The joint must physically rotate some from its fixed position to produce member curvatures and thus moments that provide equilibrium for the joint. An important component of the moment distribution method is the concept of the relative stiffness of the members entering the joint. If the members are prismatic and have their far end fixed, some constants drop out and the member flexural stiffness can be expressed as its EI/L (material modulus of elasticity times member bending moment of inertia divided by member length). The member relative stiffness is its EI/L divided by the sum of the EI/L of all members entering the joint under consideration.

The name "moment distribution" follows from any unbalanced joint moment being distributed among the members entering the joint in proportion to the individual member's relative stiffness. In the moment distribution method, one joint is released and equilibrium restored at that joint through this moment redistribution. The change in moment, and thus, change in rotation, causes an end moment at the far end of the member (for the usual prismatic member, the magnitude of this moment is one-half of that at the member end at the joint allowed to rotate). In the basic moment distribution method, joints are released and relocked one at a time, then each released and relocked one at a time in a second iteration cycle with generally smaller joint moment imbalances than in the previous cycle. This process continues until all joints are acceptably close to being in moment equilibrium. Variations arise to accommodate calculation efficiency and to permit the analysis of frames with sidesway. Moment distribution allows a more exact analysis of frames not having pinned connections than was previously computationally possible, but at the price of a possibly large number of fairly simple calculations. The moment distribution sheet or sheets for a middle-size multistory building could cover most or all of the drafting table and take days to complete.

This rather brief description of how the method works illustrates several important principles that are very obvious to the perceptive user of the method. These include, but are not limited to, (1) equilibrium must be satisfied for all members and at all joints, (2) the moments along a beam are basically those resulting from the loads on the member modified by the end moments, (3) moment (and forces) flow more to a member as that member's stiffness increases, (4) member stiffness depends upon material stiffness, member size, and member length in a simple relationship, and (5) a load in one location has an decreasing effect at members located further away. It can be argued that the physical demonstration of these principles is much more visible in the moment distribution method than in either the classical or the "matrix-based, computer-implemented modern" methods based on the creation, assembly, and solution of many simultaneously equations. The moment distribution method inherently teaches how structures behave, rather than emphasizing to its user primarily how to produce equations for solution.

The utility of the moment distribution method in today's classes on structural analysis includes that it remains a powerful tool for demonstrating and teaching structural behavior and it remains a practical method to quickly produce solutions for the quite simple structures comprised of only a small number of members, such of those that are typically used as example and homework problems in the first one or two analysis courses. Moment distribution is an efficient tool to address the effects of pattern and moving loads.

Moment distribution is also an analysis technique that remains supported by some of the design codes. The current ACI Code, ACI 318-2002, "Design Requirements for Structural Concrete"[3] continues long-standing Code content which describes assumptions and procedures compatible with implementation by moment distribution for continuous beams and for two-way slab systems. In both cases, when lateral bracing is provided by means other than rigid frame action (such as by shear walls), the beam or floor level under consideration may be modeled with the far ends of the columns extending above and below this level fixed and the rest of the structure ignored. For the equivalent frame method used for two-way slabs, the slab and column members are modeled as non-prismatic members, which moment distribution can address more simply than most methods once member stiffness characteristics are known. With the far ends of the columns assumed fixed, moment redistribution calculations need be done only at the joints along the beam or slab, and convergence is quick. Pattern load effects can be found quite quickly. The moment distribution method itself effectively demonstrates such key concepts as larger/stiffer columns result in the loadings on one beam span having less effects on adjacent spans, thus reducing the effects of pattern loads.

Notable is the inclusion of "moment distribution" as a topic within the "Basic Educational Curriculum" recently formulated and published by the National Council of Structural Engineering Associations[4]. The last five topics under the general heading "Analysis 2" (which includes basic mechanics and the beginning structural analysis topics) are slope-deflection method, moment distribution for beams and frames, virtual work, approximate methods, and influence lines. Matrix methods follow as another group of topics. Several of the last five topics of "Analysis 2" just noted above are too often omitted from the undergraduate structural analysis course as either being no longer needed or as being a more practical topic that should not displace topics which are more basic mechanics; influence lines and pattern loads often fall victim to this last assessment.

Examples of Other Useful "Obsolete" Classical Methods

The list of other "obsolete" but arguably still useful classical solution methods extends beyond the moment distribution method for structural analysis. Professor Cross developed the same sort of iterative analysis approach to determine flows in pipe networks. Parameters describing the relative ease of flow in each of the pipes entering a connection and the requirement of the total flow of the liquid into and out of the connection needing to balance replace the relative member stiffness and the joint moment equilibrium in this application. Similar observations about the characteristics of pipe network performance to those of structural behavior can be drawn, especially when this pipe network analysis is cast as a demonstration or virtual experiment.

Graphical solution methods have had a large role in structural analysis, with pinned trusses very commonly analyzed using graphical techniques building upon the basic requirement that the forces acting at a joint must satisfy equilibrium. This requires that the axial force vectors, when placed head-to-tail, of all members entering a joint must form a closed polygon. As each truss member enters two joints, force polygons for many joints can be superimposed, as can diagrams showing member elongations and joint deflections. These methods provide the "pictures" often worth many words or numbers in the understanding they convey. Instead of straight edges, scales, and protractors, computer software can now do the drawing of such graphical solutions, as well as to-scale and exaggerated deformed shapes.

Two approximate methods for lateral load resisting frames were in use before the moment distribution method was defined. Partly because of the practical problems in using moment distribution for multistory laterally unbraced frames, they remained in use as the primary analysis method for many multistory buildings well into the middle to second half of the twentieth century. In these methods, enough assumptions are made that the highly indeterminate frame is converted to a determinate frame easy to analyze. The portal method utilizes the observation that the columns and beams in the usual tier-type rigid frame subjected to only lateral loads display a doubly-curved deformed shape, with an inflection point (where the direction of member curvature changes) at near the middle of the member length. As the bending moment must be zero at these points of inflection, placing a physical hinge at their location does not change the overall pattern of moments within the structure. Placing enough hinges within the frame makes it statically determinate and easy to analyze.

The slightly different cantilever method entails similar types of assumptions and works better for tall, slender rigid frames under lateral loads. With some guidelines on where to place hinges in members near large changes in member end moments (including the first and top floors), an experienced analyst can produce approximate moments that are within a few percent of those found from more "exact" moment distribution and "modern" techniques. Today, structures more often have a complex geometry. While the "classical" portal and cantilever methods are more limited for these cases, they remain a way to provide reasonably close solutions to a broad class of practical structures.

The Mix of Classical and Computer Methods in the Classroom

The characteristics of the classical methods noted above lead to a conclusion that these classical methods remain useful and powerful tools to demonstrate many important aspects of structural behavior. They thus are important tools for helping students to develop a general understand of how structures behave. Even though these classical methods are no longer the "workhorses" of the detailed analyses in professional design practice, their use is still practical for the simple problems of the classroom, especially when carefully paired with computer-aided techniques.

Computer-implemented software can be used to introduce the more complicated problems and to do parameter studies, although their utilization for simple problems is often the first step. A common problem with the early introduction of "modern methods" is how much time is needed for the students to learn to use the software (which can results in significantly less time for basic class topics). A necessary tradeoff must be made between how extensive the direct use of the software by the student can be versus the use of the software in more a demonstration mode with the software more under the control of the instructor. The process of preparing input for the software usually provides little information to the user on "how structures behave"; the instructional value is largely limited to what can be obtained from a careful examination and display of the results, including results from parameter studies.

Computer-implemented software is increasingly becoming more able to generate useful visualizations of results. It is almost always easier to interpret a well constructed picture (graph, contour plot, etc.) than a flock of numbers. Today's students are increasingly becoming more receptive to visual information, as they are having more experiences with visualizations through a variety of sources – including computer games, graphing calculators, etc. Visualizations of results are similarly important for those in practice.

It is practical for students to implement some of the "classical" methods within the computer environment. The basic repetitive steps involved in moment distribution are very compatible with the abilities of equation-solving software and even spreadsheets. The author recently had a M.S. student implement the buckling analysis of non-prismatic columns using the Newmark method and the spreadsheet program Excel – the resulting work replaced a many "page" Fortran computer program and is very user friendly. An application the author has not personally explored follows. Many of the classical graphical methods can be implemented within the calculation and display capabilities of spreadsheets and/or equation solving software such as MatLab. The key observation relevant here is that the "classical" methods often can be implemented in a computerized environment that retains their tutorial features. This often can be done more easily than can be the more "modern", powerful and general matrix-based frame and finite element theories utilized by most commercial general analysis software.

In the classroom, it is instructive to compare the results of the old "classical" approximate methods to those from the general analysis software. This comparison is useful to demonstrate that nothing magical is happening with the "modern" software analyses – the answers from the two sources will be at least close when the conditions and assumptions for the classical methods are met.

Preparation of the Graduate to be a Responsible User of Software

One of biggest concerns of experienced designers working with recent graduates is the young engineer who proves to be better at making software work than in making sure the results from the software are reasonable. Workshops with practitioners carried out as a part of a NSF Department Reform Planning Grant directed by N. Grigg and including the author[5] repeatedly included the observation that it is generally much easier to teach new engineers to use specialized software for specific applications than it is to teach them either (1) to appreciate that the numbers produced relate to real things and consequences, or (2) to be able to critique the results for general accuracy and appropriateness.

Another way of stating this concern is to note that increasingly, a primary task of our graduates in their role of young engineers is to be intelligent users and managers of design and analysis software. This change in the role of the individual designer task from being the direct producer of numerical solutions to the manager changed with utilizing software written by others to produce a correct numerical solution is a major shift, perhaps qualifying as a paradigm shift. How should we best prepare our graduates for this role, one so different than what at least us older faculty experienced as a student? First, we need to cover enough of the basic concepts that they understand what the commercial software is doing, thus turning the possible "black box" into a "gray box" for the vast majority of graduates who are not going to be preparing software or studying it in enough detail that the software can become a "clear box". Second, students and graduates need to know and appreciate the general limitations of software and be endowed with an appropriate combination of constructive professional skepticism and confidence that they can meaningfully critique the results of software.

At some point, the designer must build upon what others provide. It is increasingly not practical to expect the structural designer to fully understand all the details of how the software was written and how it works, just as the software writer does not need a complete understanding of how the programming language and computer logic implements the software instructions. This balance between using available tools and doing computational steps "from scratch" has been recently addressed at the level of "to use or not to use" built-in features of scientific/graphing calculators and equation-solving software (e.g. MatLab, Mathematica) by M. Rossow[6]. Many of his observations on error sources (including errors in transfer of information, conceptual errors, keystroke/input errors) and in error detection apply also for software use.

Students and young engineers need to avoid what I will refer to as GIFGO, an enhancement of GIGO meaning "garbage-in, fabulous garbage out" --- the sense that something magical must take place in the computer, namely that numbers so precise that they are given to ten or more significant digits must somehow be accurate, meaningful, and useful. This phenomenon has also been described as "Computer Rapture"[7], a "condition where otherwise rational people show a complete and unquestioning belief in anything that emanates from a silicone brain – more than just GIGO, it is a complete unwillingness (or inability) to do a sanity check on computer output".

The necessary role of the analyst is to be smarter, although more computationally limited, than the software he or she is using. They cannot allow themselves to be mesmerized by the computational elegance and colorful graphics of the tool. The user must be able to determine if the results are reasonable for the problem at hand, even when exact correctness cannot be validated by hand methods.

This returns us to the observation that the classical methods can be very useful to give the students at least two important abilities; (1) the general understanding of structures so that they can sense quickly whether a result is even close to reasonable and (2) the ability to formulate meaningful, albeit approximate, answers that should not differ greatly from the software results. Classical and computer solutions should be comparable, they are not solutions for two different, unrelated worlds. The software user also needs to be proficient at other basic checks such as overall and member equilibrium, and compatibility of shear and moment diagrams with the loading conditions. The simple fact that the member moment diagram shape and overall magnitude relate to the basic member length and loading seems to be too often overlooked by the neophyte software user.

It is important to differentiate between some common computer checks for solution convergence and solution accuracy in a mathematical sense and a second more general, higher-level assessment of the applicability of the overall results for the problem at hand, although these two checks can be easily confused. The first more addresses whether the numerical methods are accurate, assuming the input information and assumptions correctly represent the problem being analyzed. The second includes a check of whether the right problem is being addressed and if this problem has been correctly described to software properly selected to have the ability to model the behavior in an appropriate way.

The choice of the most appropriate analysis method for the problem at hand is another ability needed by graduates and young engineers. Classical methods and basic mechanics equations have a definite role in this respect. A computer program may not be the best solution for a simple problem. Some students tend to loose the ability to do even simple problems without a computer program. An extreme is using a finite element program for a beam flexural problem when the basic bending equation, flexural stress = My/I, is sufficient. This characteristic is indicative of the student who has difficulty critiquing software results for reasonableness, as the software tool has become the only tool in his or her analysis toolkit.

Other Components in Preparing Intelligent Designers

The computer-implemented analysis and design software now available to the design community, when supplemented by user-friendly spreadsheet and similar general purpose software, frees the designer from almost all tedious calculations. Not many decades ago, this task required a great proportion of the design engineer's time and it was properly given a great deal of attention in the analysis-oriented classes. The designer is increasingly being a manager and user of software for these computationally-intensive tasks. This should provide the designer with more available time for the higher-level creative tasks of conceptual and preliminary design and with large-scale error detection, critiquing and validation of software results being vital steps which must be carried out before the final design is accepted. The content of engineering educational programs is slowing changing to reflect this changing role of the typical design engineer regarding how and by what/whom the computational task is conducted.

While most of this paper notes the applicability of classical methods, along with basic equations and principles, as tools to give students more understanding of structural behavior and to be more effective future users of design software, there are other tools helpful in developing a more mature understanding of design and overall structural performance.

Additional tools of note are case studies, examinations of both exemplary structures and failures, and descriptive texts on how structures work. These three topics are often interrelated, and all can help the student to build on the successes and lessons learned by others in a way not fully provided by theory, problem sets, and even computer simulations. Books I have found to be excellent in providing students with a "big, overall picture" of structural design criteria, design principles, possible failure modes, and performance expectations are:

"Why Buildings Fall Down" by Matthys Levy and Mario Salvadori[8].
"Why Buildings Stand Up: The Strength of Architecture" by Mario Salvadori[9].
"Design Paradigms – Case Histories of Error & Judgment in Engineering" by Henry Petroski[10].
"The Tower and the Bridge" by David P. Billington[11].
"Karl Terzaghi – The Engineer as Artist" by Richard Goodman[12].

The possible instructional components just noted address at least four needed topics appreciated by successful designer that go beyond the narrow question of "what analysis tools are now appropriate for production analysis in practice". First, they emphasize that having a well developed sense of structural behavior is important in formulating a solution – in the conceptual and preliminary design stage. The modern analysis tools do not carry out the creative design tasks of the designer, and some structure must be described before the analysis has something to analyze. Second, they demonstrate that there are many design criteria, among them function, economy, aesthetics, and maintenance, that must be considered in addition to structural adequacy. Third, they show that past failures can be organized by pattern and used to help in error detection and failure avoidance. Fourth, they demonstrate that a primary role of the structural engineer is to identify and understand all the possible ways that an individual structure might fail, then to take the necessary steps to prevent all of these failures.

Concluding Remarks

The uses of the "classical" solutions in the engineering classroom include at least these three – (1) as the tool useful to determine the "adequately correct" answers for design, (2) as a means to demonstrate and thus help student understand the physical phenomena and behaviors involved, and (3) as a practical way to reach "back of the envelope" answers to both assist in preliminary design and to help check more exact "computer" answers. Unlike in the "BC (before computers)" times, no longer are many classical methods needed by the practicing engineer for the expressed purpose of obtaining detailed numerical design solutions – in this sense, their time has passed. But, for the second two uses, their importance has not decreased. The importance of the third use definitely has increased. The approximate methods, including graphical techniques, can be very important in explaining how forces and moments flow through a structure (or how water flows through a pipe system), and thus, in assuring the designer selects a solution properly including these behaviors.

An important observation is that our educational approach should not be "classical methods or modern methods", as there is a role for both, along with basic theory, principles, and equations, and other educational resources such as case studies and descriptive texts on general design considerations, project performance, and failure analyses. The challenge is how to make optimum use of this increasing rich mixture of educational resources to best prepare our graduates to effectively function in the computer/software intensive design environment so that they will consistently produce efficient, creative, functional, and reliable engineered projects.

References

1. Cross, Hardy (1932). "Analysis of Continuous Frames by Distributing Fixed-End Moments, Transactions, ASCE, Vol 96, pp. 1 – 156.
2. Tedesko, Anton (1994). "Computer Analysis No Substitute for Experience", editorial in Civil Engineering, ASCE, February 1994, p. 6.
3. American Concrete Institute, Committee 318 (2002). Design Requirements for Structural Concrete (ACI 318-2002), American Concrete Institute, 350 pp.
4. Barnes, Craig E. (2004). "Education for the Structural Engineer – Basic Course Curriculum and Content, Structure, publication of NESEA, CASE and SEI, Vol. 11, No. 2, pp. 8 – 10. (See also www.structuremag.org)
5. Grigg, Neil S., Marvin E. Criswell, Darrell G. Fontane, and Thomas J. Siller (2004). "Information Technology in the Civil Engineering Curriculum", paper accepted for publication in ASCE Journal of Professional Issues in Engineering Education and Practice.
6. Rossow, Mark P. (2003). "Developing Students' Calculation Skills – A Defense of the Black-Box Approach", Computers in Education Journal, ASEE, Vol XIII, No. 4, Oct-Dec 2003, p. 68-77.
7. Scott, George (2004). "Eight Simple Rules for Mentoring My (Grand)Son", Test Engineering and Management, Febr/March 2004, p. 20.
8. Levy, Matthys and Mario Salvadori (1992, updated 2002). Why Buildings Fall Down, W. W. Norton & Company, NY, 330 pp.
9. Salvadori, Mario (1990). Why Buildings Stand Up: The Strength of Architecture, W. W. Norton & Company, 318 pp.
10. Petroski, Henry (1994). Design Paradigms – Case Histories of Error and Judgment in Engineering, University of Cambridge Press, Cambridge, UK, 201 pp.
11. Billington, David P. (1983). The Tower and the Bridge – The New Art of Structural Engineering, Princeton University Press, Princeton, NJ.
12. Goodman, Richard E. (1999). Karl Terzaghi – The Engineer as Artist, ASCE, 332 pp.

MARVIN E. CRISWELL, P.E. is a Professor of Civil Engineering and Associate Department Head in the Department of Civil Engineering, Colorado State University, Fort Collins, Colorado, where he teaches and conducts research primarily in structural engineering, materials, and engineering education. He has long been active in ASEE, especially in the Civil Engineering Division, with past service on the ASEE Board of Directors.