120th ASEE Annual Conference & Exposition
FRANKLY, WE DO GIVE A D*MN
June 23-26, 2013

Paper ID #5714

Interdisciplinary Problems and Numerical Analysis: 10 Things We Wish We Knew 20 Years Ago

Prof. James C. Squire, Virginia Military Institute

James Squire is a Professor of Electrical Engineering at the Virginia Military Institute. Dr. Squire received a B.S. from the United States Military Academy and his Ph.D. from the Massachusetts Institute of Technology. He was awarded a Bronze Star in the Army during Desert Storm and was selected as Virginia's Rising Star professor in 2004. He is a licensed Professional Engineer in Massachusetts and Virginia and maintains an active consulting practice.

Prof. Turk McCleskey, Virginia Military Institute

Turk McCleskey earned a Ph.D. in colonial American history from the College of William and Mary. His research focuses on frontier societies in North America. He is a tenured professor in the Department of History, Virginia Military Institute, where he teaches introductory courses in American history and historical methods, upper division elective courses in colonial American history, and senior capstone seminars on the American Revolution and on North American Indians.

Dr. Elizabeth White Baker, Wake Forest University

Elizabeth White Baker is a visiting assistant professor at Wake Forest University Schools of Business, teaching courses on Information Systems Management, Strategic Information Systems and Entrepreneurship. Her research interests include Information Systems Mergers and Acquisitions, Cross-Cultural Information Systems and Commercialization of Intellectual Property. Baker's research has been published in a variety of academic journals including the Information Systems Journal, Journal of Global Information Management, Journal of Information Privacy and Security, and Journal of Decision Systems.

Prof. Anthony Evan English, Western New England University

Anthony English received the Ph.D. degree in medical engineering and medical physics from the Harvard-MIT division of Health Sciences and Technology in 1996, where he received a Raytheon Fellowship. He is currently a professor at the Western New England University where he teaches biomedical engineering. His interests include the application of non-linear control theory to soft biomaterial phase transitions and cellular level phenomena. Prof. English is the recipient of a National Science Foundation CAREER Award.

Interdisciplinary Problems and Numerical Analysis: 10 Things We Wish We Knew 20 Years Ago

Abstract

Non-engineering faculty often find the engineering quantitative mindset and ability to conduct numeric analysis helpful in their research, yielding valuable results not otherwise discernible by either specialty alone. New engineering faculty can find such work helpful to launch their careers by exposing them to a wealth of productive research topics relatively untouched by single-discipline researchers, as well as providing opportunities to get to know many faculty and be exposed to a variety of research methods, writing styles, and grant sources. Despite the demonstrable benefits of such collaborations there also are pitfalls, especially for new engineering faculty who have little experience coordinating complex interdisciplinary projects. This paper describes observations on interdisciplinary collaborations based upon and referenced to several dozen interdisciplinary papers the authors have published with faculty from clinical medicine, bioengineering, finance, educational psychology, colonial history, business, sports medicine, and seismology. The paper includes five reasons to seek opportunities to apply numerical analysis to interdisciplinary problems, three common pitfalls of work in such interdisciplinary projects, and ten best practices for conducting numerical analysis of interdisciplinary problems.

I. Reasons to seek interdisciplinary numerical analysis opportunities

Interdisciplinary research often reveals low-hanging fruit

As a graduate student, one of the authors was the lone electrical engineer in a biomedical center that had a predominantly molecular chemistry emphasis. His specialty was analog hardware design, but naturally anything that had to do with signal analysis, data analysis, or more generally mathematics fell on his desk as well. As he worked with the various biology post-doctoral scientists he found that several relatively elementary engineering analysis techniques could be used to solve problems they encountered in novel ways, such as the use of finite-difference techniques to predict fluid flow[1], or strain fields in arterial tissue[2], or using relatively simple geometrical relationships to experimentally measure arterial surface strain[3]. The experimental papers that resulted were relatively fast to write since they were in largely-uninvestigated fields, in contrast to the depth required to publish in journals of comparable impact factor in his relatively well-researched area of sensor design[4]. This is a well-known relationship; the authors of popular Teaching Engineering text observe: “It is easiest to get results and write publications when you work on new ideas instead of following the well-beaten research track[5].” The ability to speed-publish such papers is especially important for new engineering professors given the reality that the tenure process places enormous pressure on new hires to publish quickly and in quantity[6,7,8,9].

Rapid socialization

Numerous studies have shown that the fastest-starting new hires tend to be the ones that socialize the most quickly[10,11], yet the academic research environment tends to isolate new faculty[12]. Lending one's numerical and programming skills, and their associated viewpoints, to help solve what would otherwise be intractable problems is an excellent way to meet other junior and senior faculty. As an example of the former, one of the authors worked with a newly-hired mechanical engineer to build a device that uses seismic waves to communicate through the earth that resulted in several academic publications[13,14] and international media presentations[15,16] which were sufficiently fast-developing to be useful in his promotion and tenure.

New grant and publication resources

Different disciplines tend to have varied grant sources and publication sources, but it is clear that there are increasing expectations for new engineering faculty to bring in money[17]. Furthermore, the resources available to fund, publish, and train new professors are complex and often not well understood[18].

For example, basic science and medical research are often funded at the federal level[2,4,19] applied engineering by industry[13,14], and work in the liberal arts and social sciences we have found is often initially supplied by universities themselves[20,21] or by non-profit organizations [22]. These sources can provide a rich source of future funded projects, since many require or respond well to interdisciplinary components and publish. Regardless of the details, experienced faculty in varied departments will each be able to introduce funding and publication resources they have productively used in the past.

Exposure to varied analysis tools

Much as many disciplines have their own grant and publication sources, many also have different preferred methods of analysis. Early career exposure to an array of varied analysis tools can have deeper value than simply analysis power – they can shape the way we approach research in a manner like the positive spin on the old wag about when one only has a hammer, everything starts looking like a nail[23]. As one example, one of the authors learned about the utility of analysis of variance from work done with social scientists when investigating the effects of computer network latency on learning efficiency[24], and that method impacted the way I approached unrelated work in determining a generalized theory of limitations of stock market portfolio analysis based on historical technical versus fundamental data[20]. Being facile with a variety of analysis tools used in multidisciplinary projects not only helps research, but also teaching. It has been reported[25] that students respond far better to courses that use different methods and involve different disciplines than to "highly atomized" course content.

Specialized skills of the collaborator

If one benefit of interdisciplinary work involves internalizing other research methods described above, it is also profitable to externalize work through delegating to collaborators with skills indirectly linked to their subject areas. For instance this paper involved a colonial historian[26], whose editing skills far exceed the first author's, and it might seem trite but the authors have never had such perfectly-made posters for a conference as the ones fabricated by a surgeon[27]. A related issue involves the synergy of collaboration. Much has been written about the ability of two or more people to synthesize ideas that are more than the sum of each individual contributor, and although the process is not fully understood the effects are clear: in general it is more effective to work together than alone[28]. Our limited experience suggests that if combining different experiences and ideas is beneficial, the greater the divergence in backgrounds, the greater the benefit. For example, the authors worked on an insulin refrigerator that could contact geographically-distant family members if not accessed three times in a day. Adding a computer engineer to the team provided the insight that we could substitute wireless email notification capability for less money than paging capability, but adding a physician to the team provided the insight that the method could be generalized to sense any sort of activity by motion detection, and thus create an automatic version of the manually-activated Life Alert® system[29].

Shared responsibility

Finally, writing with someone else, whether as a lead or secondary author, adds a sense of responsibility. It is far easier when writing alone to say "I was going to submit this year to the ASEE conference, but my schedule is squashed…I'll save it for next year." If honest, the authors could reference most of our publications, and all of the ones unpublished, on this theme. This is of particular importance when conducting interdisciplinary research since it can be difficult to find an appropriate discipline-specific journal for publication. When this happens, having others involved has often made the difference on whether we find a solution or abandon the project entirely. For instance, when attempting to publish a paper describing a novel method to communicate to trapped underground miners using seismic waves we had a paper rejected three times because reviewers were from either signal processing or physical geology backgrounds and became confused about topics outside their area. The solution was to postpone academic publication and instead patent and license the invention[30]. Once licensed it was simple to publish in the leading technical trade journals[13,14].

II. Ten best practices for conducting numerical analysis of interdisciplinary problems

Specifying the analysis

Unlike many single-discipline projects, it is difficult to specify all of the numeric analyses required at the start of the project because intermediate results will determine how the paper dynamically develops. Instead, seek to initially fully specify just the first one or two experiments, meaning conceptual input/output (I/O) and detailed data format I/O, and then agree with your collaborators to a range of number of software programs you will create for the paper. I have found five to ten works well. Too few constrains the paper development excessively and creates the unrealistic expectation that there will be no dead-ends to proposed questions; too many encourages blind hunting of trends and mismatches the enormous amount of work required to fully verify analysis programs with the work of the collaborators. As the fields involved in the research become more disparate, expect the amount of time required to fully specify each analysis program to increase. Researchers in related technical fields of robotics, for instance will intuitively know how to specify simulation parameters to analyze optimal floor-covering mappings in the presence of noise for a robot that removes ticks from commercial properties[31,32], whereas it took many weeks with a historian to completely enumerate the data forms and manipulation required to investigate whether one's wealth impacted the likelihood of being called to jury service in colonial Virginia[22].

Decide who runs the analysis

There is considerable benefit to be gained by allowing our non-numerically trained colleagues, to whom I will refer to as users, run the programs independently. Doing so gives them the opportunity to experiment with different data sets and experimental protocols, explore relationships only apparent to one skilled in those arts, and feel less guilty about asking for "one more analysis[33,34]." Yet there is also a case for separating the roles of the person providing the data and the person performing the analysis; it makes our colleagues justify their questions and gives us the opportunity to find mathematical explanations governing unexpected behavior[35]. In general, I find that straightforward but data-intensive analyses are better handled directly by the user[20], and that separation of the roles is helpful in the more controversial analyses to promote problem-solving best done together by people on both sides of the numerical fence [22].

Code for user flexibility

Often numerical analysis is used to answer a series of related questions. For example "did colonial American service as a magistrate make one more likely to be chosen as a petit juror?" is similar to "did service as a grand juror make one more likely to be chosen as a petit juror?"[22] While it may be obvious to code these in such a way that we as programmers can benefit from code re-use, it may be less obvious that by standardizing on a particular data format that accommodates variable-length fields that the user may use the exact same program to do both analyses.

Use the right language for the job

Selecting a programming language involves a tradeoff between ease of use for the programmer and ease of use for the user. High-level languages such as Matlab, Mathcad, and Mathematica have been termed rapid application development (RAD) languages and provide built-in commands for reading formatted text files, performing matrix algebra, and plotting. In Matlab, for example, a program that queries the user for a text data file using a standard dialog box, convolves the data with an impulse response, and plots the result takes three lines of code.

Unfortunately all these RAD languages require the interpreter/compiler software to be installed and authorized on the user's machine which can be difficult or expensive to do. Depending on the language, this approach also may require the user to run the interpreter / compiler, and then load and run the program which adds a layer of complexity for a non-technical user.

We have found the speed of application development in a RAD language almost always justifies the amount of time it takes to install the compiler and train the user to use it. An exception to this rule occurs when the collaborator wishes to share the analysis ability with many other colleages[20] that requires an executable. In that case, it is better to code to target a Windows or

Apple platform based on popularity, and compile to an executable file using an object-oriented language such as C++, C#, or Java, and write one unique class per data file structure for reading and writing into expandable arrays. If the processing technique is unique to the data file, encapsulate the processing in an associated class; otherwise abstract it into its own class. If future programs need to use the same data file structure, extend its functionality using inheritance and polymorphism. Title the application with the version number to reduce re-versioning confusion. Anticipate email server security rules complicating distribution by email, by sharing either by embedding in a personally-owned website or by using a file-sharing location such as Dropbox®. We have found simpler measures, such as compressing or changing the file extension, only work through a subset of university email servers.

The remaining tips and best practices are less abstract than the above and more technical in nature. The canonical framework for developing numerical analysis programs is shown in Figure 1.

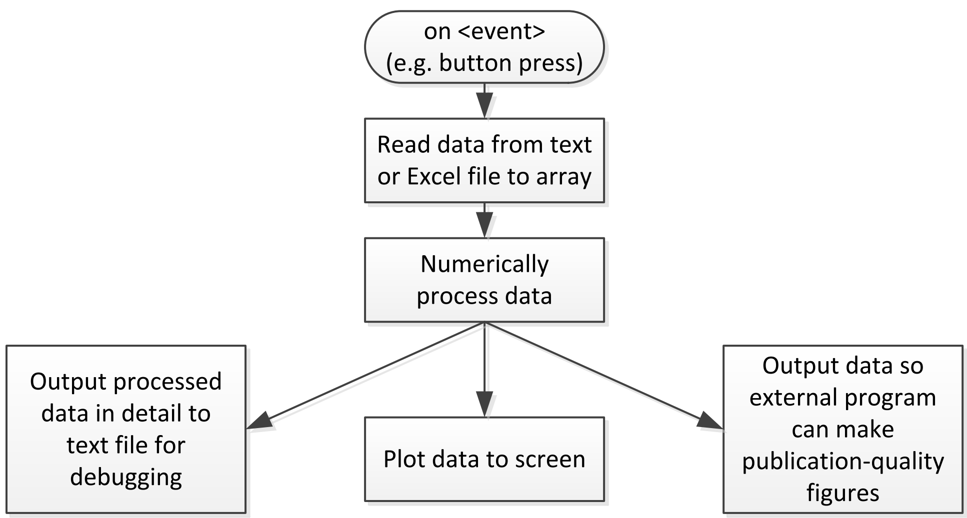

Figure 1: Canonical framework for numerical analysis routines.

The benefit of using a RAD language that provides built-in support for each of these blocks becomes apparent. The disadvantage of using traditional languages such as Java or C++ can be partially mitigated by developing or purchasing code library extensions. These libraries, however, still do not provide the development speed of a language with native support for these constructs.

Choose the best input data format for the job

The first step in Figure 1 involves reading the data. Numeric data processing often involves importing data files from other users whose native format may be open-source binary (e.g. some GIS data or PNG images), proprietary binary (e.g. native spreadsheets), ASCII, or ASCII's successor, Unicode UTF-8, or UTF-16. While it is possible to code readers that can take proprietary binary formats, it is almost always easier to request the user export the data in tab or comma-delimited ASCII. While this seems obvious in retrospect, I have lost many hours providing native-format import support for programs that offer export capability[22,30].

Verify data as it is read, with failures throwing an exception that includes the line number and instance of the offending data. This is clearly essential with hand-entered data, but also with computer-generated files that may include malformed tokens as a result of undocumented codes, incomplete data, or occasional Unicode characters when ASCII is expected.

If data involve strings, be aware when choosing export formats of the range of string content. Use of commas, for instance, will cause problems if exporting as a comma-delimited file. In a similar vein, be aware of the different standards for end of line characters among PCs, Macs, and Unix machines.

If the data are large (one author used data files well over 5GB in size[20]) consider reading the file in small chunks into a buffer, preprocessing them to eliminate unnecessary data, and then either store the result into an array in memory or into a new smaller file on disk. If a new data file is generated, consider either deleting it when the processing is finished or carefully naming it by date to prevent versioning problems when new data files are published.

Choose the right input database structure

Closely related to the previous tip, rather than spend programming time manipulating an awkward input data structure, have the user provide the data in a format that is easy to use. For example, instead of reading the data from an .xls file and operating on the first and third columns of elements if the second column begins with an "A" character and then sorting the result, have the user in Excel first sort on the second column and then save just the first and third columns of the "A" entries as comma delimited file. While it may seem obvious in retrospect, the time required to manipulating the data directly within the host database application is orders of magnitude smaller than programming and debugging customized code to do the same.

Debugging and testing

These best-practices follow from the excellent guide on writing code by Maguire[36]. Test analysis program accuracy by creating very small data sets so that results may be hand-verified, since errors may exist in either the code or in the input data. Data integrity should be checked in both the data-reading block and after the processing blocks of Figure 1.

Logic should be added in the data reading block to catch data format errors and to terminate the program after displaying the line number in the data file where error occurs. Common data errors include

- Bad characters. This is especially a likely culprit if the data is being entered on a different operating system than the processing program. For example, Windows, Apple, and Linux operating systems can create cross-compatibility issues.
- Missing data fields.
- Data fields are of wrong size or type, such as letters in a number field.
- Undocumented or unexpected codes to indicate special conditions. This is especially likely when working with large data sources, such as daily stock transaction histories[20] with

millions of records. The likelihood of incomplete data on any one record is low, but the likelihood of a data error somewhere is high.

After the processing section, logic should be added to catch and identify data errors or likely data errors. For instance, data could be sorted and checked for duplicate entries, or to ensure numeric results are within known possible bounds.

Include a tabular data output option

There are a variety of mathematics packages such as Excel, Matlab, and Mathematica that produce high quality graphics so there is no need to duplicate that ability to have publication-ready artwork by the data processing program if it dumps the raw numerical data in ASCII or comma-delimited form. It is useful to provide an option to make the tabular data output more verbose than required, both to aid during testing/debugging, and to enable the user some degree of autonomy in data mining the processed results in ways not anticipated when designing the program. For example, a historian may seek to understand if the selection of petit jury members in a colonial American court was influenced by previous service of the jury member as a magistrate. By providing a verbose tabular data export option, the historian can then examine what characteristics magistrates turned petit-jurors have in common.

Include a graphical plot data output

In apparent contrast to the previous tip, consider adding automatic data plotting of the processed output. It does not have to be of publication-quality since the user has the option to export data to a graphics plotting package, but it should provide an intuitive overview of the data. This helps identify programming logic errors, assists the user interactively check his or her ideas, and also find hidden relationships among the data especially if you provide multiple ways to process or sort the data. For example, one may explore similarities among the pricing stocks with various characteristics[20]. Allowing the user to sort among different industries, volumes, and trade dates, and provide instant graphical feedback of similarities among pricing trends lets one uncover relationships orders of magnitude faster than if the data had to be exported into an external plotting package at each step.

Avoid saving derived data

Often programs are best written in a serial chain format, where the output of one stage is used as the input of another. The intermediate data is called “derived data”, and this approach is common when the derived data is important in its own right, when there are several possible different analyses that must be applied to the derived data, or when it is useful to access it for debugging and verification purposes. In these circumstances, it is tempting to save the derived data for later processing by other programs. This is dangerous since changes to the original data file will not be reflected in the saved derived data, complicating re-versioning of both the original data and the program used to create the derived data. Instead, it is preferable to call the first processing program from the later processing program, to ensure the latest version of the data and software are always being used.

III. Pitfalls

As exciting and fruitful as this style of interdisciplinary research is, its collaborative interdisciplinary style is also known to be associated with several potential problems, especially for relatively new faculty[37,38].

When things don't work out

Often research does not develop as originally envisioned. While this can be a good thing when discoveries are made, research sometimes just simply does not work out. A hidden relationship thought to exist may not, and may yield no insights as the researchers to continue to throw time and energy into a rabbit hole leading nowhere. At these times new faculty, especially if working in a highly interdisciplinary field with senior colleagues, may feel uncomfortable saying "we need to pull back and reevaluate our approach." This is a tough problem to handle alone, let alone with senior colleagues, and although much has been written about it[39] there are no universal answers when working with people or with research.

Connotations and denotations

Words may be loaded with field-specific connotations and field-specific denotations. Projecting data, for instance, means something completely different when speaking to a historian than to a mathematician. This can cause misunderstanding between the programmer and client as the project is being developed, as well as between the author and journal reviewer. Similarly, some terms of art are so commonplace in one field that they may not seem to require definition, yet are quite foreign to another. As others have noted[40], it is important to seek in-house reviewers for a paper from as varied an audience as the journal's audience before submitting an article for review. One technique the authors have found that breaches cross-disciplinary boundaries and helps reduce confusion between the programmer and the client is to sketch the expected form of figures desired in the resulting publication.

Where to publish

Highly interdisciplinary research can be fruitful from a research perspective but paradoxically can sometimes be harder to publish if it falls between the cracks of established journals. This is especially true if the intended journal audience is not comfortable with numeric analysis tools. The authors have found it simpler to find a venue eager to publish an econometrics paper employing digital filter optimization techniques[20], for example, than a paper discussing colonial American litigation practices that borrowed techniques from control system identification[22], despite the latter generating a large amount of interest at conferences. Even in the cases where it is difficult to locate a suitable journal, universities still overwhelmingly consider multidisciplinary work to be "hard" and not "soft" research, as reported by Boyer[41], and thus valuable for new educators in the tenure-seeking process.

IV. Conclusion

There exist many reasons why new engineering faculty should consider teaming with members of interdisciplinary teams to apply their numerical analysis skills, including gathering quick-to-publish papers in areas not already extensively researched, rapid socialization with other junior and senior members of the university, availing oneself to new sources of grants and publication, and learning new research methods. By following the ten best practices and avoiding the pitfalls described in this paper it is possible for new faculty to parlay these advantages into part of a sustained and successful research career.

References

[1] Rogers C., Tseng D., Squire, J.C., and Edelman E.R. "Balloon-Artery Interactions During Stent Placement," *Circulation Research*, Vol. 84, pp. 378-83, 1999.

[2] Squire J.C., Rogers C., and Edelman E.R. "Measuring Arterial Strain Induced by Endovascular Stents," *Medical and Biological Engineering and Computing*, Vol. 11, pp. 87-93, 2000.

[3] Garassic J., Edelman E.R., Squire J., Seifert P., Williams M., and Rogers C. "Stent and Artery Geometry Determine Intimal Thickening Independent of Arterial Injury," *Circulation*, Vol. 101, pp. 812-818, 2000.

[4] English A.E., Squire J.C., Bodmer J.E., and Moy A.B. "Endothelial Cell Impedance Parameter Artifacts Produced by a Gold Electrode and Phase Sensitive Detection," *IEEE Transactions on Biomedical Engineering*, Vol. 54(5), p. 5, 2007.

[5] Wankat P.C. and Oreovicz F.S. *Teaching Engineering*, McGraw-Hill, NY, p. 23, 1993.

[6] Low H.M., Abdulla A., and Mohamed A.R. "Publish or Perish: Evaluations of Academicians' Journal Publications as a Way of Identifying Strategies for Improved Performance," *Clute International Academic Conference*, Las Vegas, Oct 15-17, 2012.

[7] Wankat P.C. and Oreovicz F.S. *Teaching Engineering*, McGraw-Hill, NY, p. 325, 1993.

[8] Sisson L.E. "Faculty evaluation and reward: Does effective teaching matter?" *Engineering Education*, p. 380, Feb. 1982.

[9] Addy A.L. and Dutton J.C. "Clearing the hurdles along the tenure track," *Proceedings ASEE Annual Conference*, ASEE, Washington, DC, p. 1266, 1987.

[10] Boice R. *Advice for New Faculty Members: Nihil Nimus*, Allyn and Bacon, Boston, p. 235, 2000.

[11] Gaff G. and Simpson R. "Faculty Development in the United States," *Innovative Higher Education*, Vol. 18 (3) pp. 167-176, 1994.

[12] Boice R. *Advice for New Faculty Members: Nihil Nimus*, Allyn and Bacon, Boston, p. 238, 2000.

[13] Squire J.C., Sullivan G.A., Baker E.W., and Flathers G.W. "Proof-of-concept testing of a deep seismic communication device," *Society for Mining, Metallurgy, and Exploration*, Vol. 326, pp. 97 – 100, 2009.

[14] Sullivan G.A., Squire J.C., Baker E.W., and Flathers, G.W. "To the Rescue," *World Coal*, Vol. 19, pp. 14 – 18, 2010.

[15] Squire J.C., Sullivan G.A. "Seismic Wave Communications," *Discovery Channel Canada*, "Planet Earth" feature segment with Dennis Porter, May 23, 2007.

[16] Squire J.C., Sullivan G.A. "Miracles of Nature, Episode 3, Season 1," *British Broadcast Company*, 2012.

[17] Boyer E.L. *Scholarship Reconsidered: Priorities of the Professoriate*, The Carnegie Foundation for the Advancement of Teaching, Princeton, p. 328, 1990.

[18] National Science Board. "The Science and Engineering Workforce: Realizing America's Potential, Report 03-69," *The National Science Foundation*, Arlington, p. 26, 2003.

[19] Squire J.C., Rogers C., Edelman E.R. "Stent Geometry Influences Later Restenosis," *Circulation*, Vol. 94, pp. I-259. (abstract) 1996.

[20] Hilliard J., Schwartz A., and Squire J.C. "A Test of Technical Analysis: Matching Time Displaced Generalized Patterns." Accepted at *Financial Management* for publication in Q1, 2013.

[21] Walsh V.K., Bush H.F., Squire J.C., Sullivan G.A. "A Multicenter Study of Students' Sensitivity To Screen-Update Delay," *Contemporary Issues in Education Research*, Vol. 4(9), June 2011.

[22] McCleskey, T. and Squire J.C. "Backcountry Virginia Connections to the Atlantic Economy, 1746-1755," *Virginia Forum Annual Meeting*, James Madison University, VA, March 2012.

[23] Wankat P.C. and Oreovicz F.S. *Teaching Engineering*, McGraw-Hill, NY, p.337, 1993.

[24] Bush H.F., Squire J.C., Sullivan G. Walsh V., English A. and Bolen R. "An Investigation of the Effect of Network Latency on Pedagogic Efficacy: A Comparison of Disciplines," *Contemporary Issues In Education Research*, Vol. 1, No. 4, pp. 11-25, 2008.

[25] National Academy of Engineering. "Educating the Educator of 2020: Adapting Engineering Education to the New Century," *The National Academies Press*, Washington, DC, p. 147, 2005.

[26] McCleskey, T. and Squire J.C. "Pennyslvainia Credit in the Virginia Backcountry, 1746-1755." Accepted at *Pennsylvania History* for publication in 2013.

[27] Tseng D.Y., Rogers C., Squire J.C., Edelman E.R. "Mechanisms of Endothelial Denudation During Endovascular Stent Deployment," *Proceeding of 23rd Annual Meeting of the Society for Biomaterials*, New Orleans, April 30-May 4, pp. 324-29, 1997.

[28] Boice R. *Advice for New Faculty Members: Nihil Nimus*, Allyn and Bacon, Boston, p. 223, 2000.

[29] Kuwik P.A., Largi T.A., York M.R., Crump D.J., Livingston D.L., and Squire J.C. "The Smart Medical Refrigerator," *IEEE Potentials*, pp. 42-45, Feb/Mar 2005.

[30] Squire J.C., Sullivan G.A, and Flathers G.W. United States Patent and Trademark Office Patent #7,843,768: *Method for Communication Using Seismic Waves*, 2010.

[31] Woulfe J.H., Crump D.J., Hammond G.B., Livingston D.L., and Squire J.C. "Autonomous Robot for the Collection and Extermination of Ticks," *NCUR 2005 Proceedings*, Lexington, VA, April 21, 2005.

[32] Squire J.C., Sullivan G.A., and Livingston D.W. United States Patent and Trademark Office. Patent #7,441,367: *Robotic Pest Removal System*, 2008.

[33] Squire J.C., Walsh V.K., Bush H.F., Sullivan G.A., and English A.E. "Pedagogical Effects of Network Latency on Learning," *Computers in Education Journal*, Vol. 18(1), p. 103, 2008.

[34] Sullivan J.C., Bush H.F., Squire J.C. and Walsh V.K. "The Results of a Longitudinal Study of the Effects of Network Delays On Learning," *Proceedings of the 2012 Las Vegas International Engineering Conference*, Las Vegas, October 2012.

[35] Hillard J.E., Schwartz A., and Squire J.C., "A Test of Technical Analysis: Matching Time-Displaced Generalized Patterns," *Eastern Finance Association 2001 Annual Meeting*, Savannah, April 13-16, 2011.

[36] Maguire, S. *Writing Solid Code*, Microsoft Programming Series, Washington, 1993.

[37] Hara N., Solomon P., Kim S.L., and Sonnenwald D. "An Emerging View of Scientific Collaboration: Scientists' Perspectives on Collaboration and Factors that Impact Collaboration," *Journal of the American Society for Information Science and Technology*, Vol. 54(10), pp. 952–965, 2003.

[38] Boice, R. *Advice for New Faculty Members: Nihil Nimus*, Allyn and Bacon, Boston, p. 184, 2000.

[39] *ibid*, p. 188-189.

[40] *ibid*, p. 183-186.

[41] Boyer E. *Scholarship Reconsidered: Priorities of the Professoriate*, The Carnegie Foundation for the Advancement of Teaching, Washington D.C., p. 20, 1997.