Int. J. Engng Ed. Vol. 16, No. 4, pp. 292–300, 2000
Printed in Great Britain.

0949-149X/91 $3.00+0.00

A New Metric for Assessing Quality in Advanced Graduate Courses in Computer Science and Engineering*

SUMIT GHOSH
Department of Computer Science & Engineering, Arizona State University, Tempe, Arizona 85287, USA.
E-mail: sumit.ghosh@asu.edu

This paper presents a philosophy that underlies the design of a few advanced graduate courses at ASU in the sub-disciplines of hardware description languages, communications networks, computer-aided design of digital systems, distributed systems, distributed algorithms, and modeling and simulation. From the philosophy, a new metric emerges: the extent and significance of the knowledge 'discovered' by the students, towards evaluating the quality of such courses. Discovery refers to the knowledge that is brought out into the open, through logical reasoning from the first principles, by the student for himselfl herself. It is significant in that it becomes an integral part of the individual who not only gains invaluable insight and confidence in the subject matter, but can improvise, reason, and apply it to other areas in creative ways. The choice of the metric is influenced by the author's experience as a doctoral student at Stanford and as a faculty, first at Brown and currently at ASU, as well as the candid comments and feedback from full-time graduate students and graduate students coming from industry. The paper illustrates the application of the metric through a number of actual cases encountered during teaching at ASU. It also presents a list of the desirable attributes of the underlying educational environment to ensure success in the design and delivery of such courses.

INTRODUCTION

IN GRADUATE SCHOOLS across the USA, the teaching of graduate courses continues to constitute a significant component of graduate education. In general, graduate courses are organized into introductory and advanced levels. The introductory courses are designed primarily to bring the knowledge level of the incoming graduate students from different institutions and disciplines such as mathematics and electrical engineering, to a common standard set by the individual institution concerned. Such a course is similar to a traditional undergraduate course in that it prescribes a textbook, follows a well defined syllabus, utilizes conventional homeworks and examinations, and resorts to the usual grading mechanisms. In contrast, the primary goal of the advanced graduate courses is to prepare full-time graduate students to undertake highest-quality Ph.D. dissertation research and to impart advanced knowledge to the students coming from industry to enable them to engage in frontier research projects at work. In essence, the objective is to create first-rate, original thinkers of the future. In many universities in the US, including ASU, the quality of the graduate courses designed and taught by a faculty member, plays a vital role in defining the quality of the graduate program and in promotion and tenure considerations. Since research, by definition, implies unknown knowledge, a priori, and as knowledge in the CSE discipline is continuing to experience rapid advancement, the design of advanced graduate courses and the metrics for assessing them must clearly be different.

There is a general belief among computer science and engineering (CS&E) educators that the courses at the undergraduate level must focus on imparting the fundamentals, core knowledge, and information to the students to prepare the majority of them for industry. In contrast, the goal of the graduate courses must be to foster critical analysis, reasoning, interpreting data and results, and thinking. A few educators believe, the author included, that these goals must be extended even to undergraduate courses. Graduate students are assumed to be more responsible and are, therefore, expected to learn material on their own. Lee and Messerschmitt [7] observe the rapidly advancing nature of the electrical and computer engineering (ECE) field and note that the breadth of the required fundamentals has expanded significantly. In proposing a new curriculum, they suggest that the MS program be designed to instill deeper technical knowledge in the students and prepare them for design careers while the Ph.D. program design ought to focus on creating teachers, researchers, and technical experts' who would work in the frontiers and extremely high-performance technologies in the industry.

* Accepted 9 September 1999.

A key issue with the design of courses in the academic setting, especially graduate courses, the subject of this paper, is their assessment or evaluation. The principal reasons underlying the evaluation are three-fold. The first and most important reason is to measure the amount of knowledge and extent of learning that is imparted to the students. This will, understandably, translate into the students' successful careers. This paper will focus primarily on this aspect. Second, the results of the evaluation may provide feedback to the administration and help initiate the necessary curriculum modifications and improvements. Third, in the current academic environment, teaching evaluations serve as important criteria for the administration to guide its decisions on tenure, promotion, and post-tenure review, relative to the instructor.

The traditional teaching assessment techniques consist of:

1. Student evaluations, where the students fill out a questionnaire at the end of the semester, and from which a score is computed.
2. Peer evaluations, where a peer of the instructor attends a prescribed number of classes, analyzes the course material, and determines a score.

In general, the former technique is used more frequently than the latter. A third assessment technique, currently utilized within the computer science and engineering department at ASU, consists of the administration evaluating the course portfolio which includes the syllabus, copies of the homeworks and examinations, optional lecture material, prescribed textbook, and samples of student work. The issue of student evaluations and their use by administration to assess faculty teaching performance is a very sensitive one and, to date, lacks universal consensus. In addition, while student evaluations have been in use in undergraduate courses at many academic institutions for some time, their use in graduate courses demands more careful analysis. The greater care is warranted, given the difference in the nature and purpose of undergraduate and graduate courses, especially in light of the rapidly evolving nature of the technical knowledge.

The literature on assessing graduate courses, especially advanced graduate courses, is sparse. Based on an opinion poll of engineering employers, Henderson [1] argues for the inclusion of engineering knowledge, mathematics, and communication skills in graduate education. Denning [2] notes that rapid advances in networking and information technology will soon render the traditional university obsolete and calls for new thinking in graduate program design. Koehno [3] observes that, of the 11 ABET criteria, graduate engineering students and practitioners considered only three to be important:

- an ability to apply knowledge of mathematics, science, and engineering;
- an ability to design and conduct experiments, as well as to analyze and interpret data;
- an ability to identify, formulate, and solve engineering problems.

Doderer and Ciolma [4] note that research emphasis at leading engineering institutions is causing a paradigm shift from design and construction to discovery and understanding. Mackle [5] argues that quality in education should be defined from the perspective of the end-user and that a key component should be the ability of self-learning. Bendell [6] argues that quality in higher education must refer to clarity of thought, presentation, and purpose.

This paper:

- aims to address the issue of quality in graduate courses and to design metrics for evaluation;
- presents a philosophy underlying the advanced graduate courses that have been developed and utilized by the author;
- presents details on the metrics for evaluation of quality in the advanced graduate courses, accompanied by a number of illustrative examples;
- outlines the necessary attributes relative to the underlying educational environment.

PHILOSOPHY UNDERLYING THE NATURE OF ADVANCED GRADUATE COURSES IN CS&E

The philosophy underlying the design and delivery of the advanced graduate courses may be expressed through its four principal characteristics. The first and most fundamental characteristic is based on the belief that every aspect of knowledge is inherent in every individual and that the aim of advanced graduate education is to foster an environment where the knowledge is discovered, i.e. brought out into the open through logical reasoning from first principles by the student. Thus, the advanced graduate courses are the means through which every individual may discover knowledge for oneself. The significance of learning through discovery is that the knowledge becomes an integral part of the individual who not only gains invaluable insight and confidence in the subject matter but can improvise, reason, and apply it to other areas in creative ways. The plan for an advanced graduate course begins with a basic, common starting point and then develops logically and systematically, building on successive topics, and utilizing sound reasoning. The final result is a coherent body of knowledge wherein a continuous thread of reasoning spans from the simplest starting point to the most complex concept.

The second characteristic stems from the well-known fact that knowledge is continuously evolving and that our current knowledge is merely an approximation of the complete truth.

According to the late Prof. Syed M. Fasih of the University of Karachi in Pakistan [18], a book, in general, encapsulates the past knowledge. It should be used only as a guide to learning new things, as nature unfolds the creation, and one should never mistake it to represent the truth, forever. This is especially true in CS&E today where technical knowledge in many sub-disciplines including distributed algorithms, concurrency, hardware description languages, and parallel processing, is witnessing rapid evolution and the books are rapidly becoming outdated. Therefore, in the design of the advanced graduate courses, every concept—basic and especially advanced concepts, must be questioned and subject to rigorous analysis. Often, new and more precise understandings result from critical analysis of an apparently well established concept. Evidently, strong will-power and self-discipline are necessary to overcome inertia and fear of the unknown. In contrast, where concepts are accepted without examining them in the light of scientific experimentation, there arises the possibility that the educational system may degenerate into a 'self-propagating system' [8].

As an illustration, consider the field of fault simulation of digital systems, in which the author had introduced the concept of behavior-level fault models [9] for enhanced performance and to enable high-level comprehension of the fault simulation results, relative to the use of the traditional gate level stuck-at fault model. Through experiments, the behavior-level fault models were observed to correlate strongly with the gate-level models for a number of representative digital modules. Thus, it was reasoned that the behavior-level fault models should enjoy the same level of confidence as the gate-level fault models. Initial resistance included the view that, unlike the well-established gate-level stuck-at models, behavior-level fault models lack a sound formal basis. The criticism was quickly revealed as unfounded since even gate-level stuck-at models were shown as purely empirical in origin and that they totally lacked any formal basis.

The third characteristic aims to address a fundamental dilemma. On one hand, it seems logical to focus a student's training in a specific sub-discipline so that the student may be immediately effective in an appropriate industry. Although extensive knowledge may constitute a key asset in the pursuit of innovative solutions to problems, it is fundamentally unknown how to induce innovation in an individual, relative to a specific sub-discipline. On the other hand, the single-minded training in the sub-discipline may constitute a disservice to the student, especially if the importance of the subdiscipline diminishes in the near future—a likely possibility in today's fast changing world. Also, the interaction among many sub-disciplines in undeniable. Furthermore, a question is raised whether the concentrated training may cause irreversible harm to the intrinsic creative potential of the student. Barlow [17] documents an episode involving a child prodigy who could intuitively factorize very large numbers, correctly, but lost the ability when subject to training in the conventional mathematical methods.

Therefore, this paper reasons that the advanced graduate courses must be designed to provide the necessary background, fundamental principles, and specific examples, and the students must be motivated through thought-provoking questions and problems. While students may be exposed to the advanced courses in different sub-disciplines, ultimately, each student must be permitted to evolve individually, utilizing one's innate abilities, talents, perspectives, and inherent interests, into an original thinker in the discipline of one's choice. The evaluation process, in turn, will be based on whether the student has developed, relative to the mission of the student's choice, unique and innovative solutions, raised good and thought-provoking questions, and accomplished the mission goals. This characteristic is inspired by the belief that knowledge does not come compartmentalized but, in essence, constitutes one continuous thread that spans seemingly different sub-disciplines including networking, algorithm design, computer architecture, and hardware design. At the advanced graduate level, the conventional boundaries of the different sub-disciplines must be transcended and innovative ideas must be sought at the boundaries.

A related example consists of the leadership course at the United States Military Academy [21]. The fundamental premise of the course, initiated in the 1980s, was that leadership cannot be taught, i.e. one could not be taught how to become a leader. The course material and instruction were focused on providing case studies of successful leaders and a few key principles. These included the key characteristic of a good leader that one must be true to oneself and not pretend to be someone else. In the end, the students realized that they alone must evaluate themselves, their strengths and weaknesses, and develop into original leaders with unique metrics for assessing their individual leadership skills.

The fourth characteristic views as the most important outcome of the advanced graduate courses, new and original ideas, and innovative ways of viewing traditional concepts. It is based on the belief that creative ideas hold the key to the future and that there is no limit to the depth of knowledge.

METRICS FOR ASSESSING QUALITY OF ADVANCED GRADUATE COURSES IN CS&E

Undergraduate-level courses in basic areas such as physics, chemistry, engineering mathematics, logic design, and signal processing reached maturity many years ago and have essentially remained unchanged, thereby enabling the development and use of well-defined, quantitative metrics for assessment. Introductory graduate courses are similar to undergraduate courses in that they are also relatively stable and that they are designed to primarily impart knowledge and information and secondarily focus on equalizing the level of the entering graduate students from other universities with those of the native graduate students. In contrast, advanced graduate courses in the sub-disciplines of CS&E are witnessing rapid evolution and, thus, the development of well-defined quantitative metrics, is difficult. At their onset, fundamental advances in knowledge have often been observed to exhibit only subtle differences with the established principles. Since they assume the presence of a common and objective standard that applies uniformly to all cases, quantitative metrics may not constitute the ideal means to capture the subtle aspects of advanced knowledge as well as the elements of creativity and innovation. Thus, it may not be meaningful to define the metrics rigorously. Instead, they should essentially remain somewhat subjective. Therefore, to realize a broad assessment of the advanced graduate courses, while nurturing its growth at the same time, this paper proposes to focus on their quality aspect. In the field of education, quality refers to a unique and rare attribute that is difficult to define, resists quantitative measurement, and has to be ultimately understood intuitively. The manifestations of quality, however, may be more easily observed which, in turn, provide the motivation for the pursuit of quality in learning. Examples of high-quality knowledge include Copernicus' discovery that the earth rotates around the sun, Newton's discovery of gravitation, Einstein's discovery of the special theory of relativity, etc. For a better appreciation of the subtlety of fundamental knowledge, consider the argument that, for all practical purposes, there is no need to supersede the classical laws of motion with the special theory of relativity. Though approximate, the classical laws of motion are 'good enough'. Clearly, the errors resulting from the use of Newton's laws, for the ordinary velocities that one encounters, as opposed to the special theory of relativity, are negligible. However, fundamentally and philosophically, the approximate, classical laws of motion are incorrect.

The general expectation is that, an individual when exposed to high quality education is likely to display uniqueness and originality in his/her work. Frequently, high quality education and research are associated with the discovery of innovative engineering principles, invention of unique products, and the creation of new disciplines. The issue of quality is frequently associated with excellence, depth of learning, and thinking. An unmistakable attribute of quality is that it is either present or not, which, in turn, may be viewed as being equivalent to the property that quality is not additive. Thus, while the combination of a number of efforts that lack quality may fail to solve a difficult problem, a single high-quality effort may lead to an innovative solution. The high-quality aspect of the advanced graduate courses is viewed as the key to creating first-rate, original thinkers. Such individuals are expected to either generate outstanding original Ph.D. dissertations or engage in innovative, frontier research in the industry.

Presently, while the body of advanced technical knowledge for many sub-disciplines within computer science and engineering is rapidly changing, books in the fields of distributed algorithms, networking, and architecture are becoming outdated, and the meaning and value of the traditional, standardized midterm and final examinations are increasingly being called into question. Also, the memorization of theory is viewed as being of little value and, in its place, understanding of concepts is being considered more important. Most problems of today are complex and do not lend themselves to a straightforward answer. Instead, they require complex design and trade-off analysis, and warrant the development of innovative perspectives. Therefore, as a metric to assess quality in advanced graduate courses, this paper proposes to utilize the extent and significance of discovery realized in the students. Discovery refers to new concepts, ideas, approaches, and knowledge that are acquired by a student as a result of critical analysis, imagination, and original thinking. Often, the discovered knowledge, though new to the student, has been previously published in the literature. However, in a few, rare cases, the discovered knowledge is indeed original. The discovery process may be triggered in a number of ways—through lectures, challenging examinations [10], one-on-one discussion between a teacher and a student, open-ended projects [11], and even thought-provoking Ph.D. qualifying examination questions. The late professor of physics at California Institute of Technology, Richard Feynman notes in [8] that he used to put a lot of thought into each lecture and writes [8] 'I start to figure out the motion of the rotating plate. I discover that when the angle is very slight, the medallion rotates twice as fast as the wobble rate—two to one. It came out of a complicated equation! Then I thought, is there some way I can see in a more fundamental way, by looking at the forces or the dynamics, why it's two to one?' The late Prof. R. L. Moore, a well-respected mathematician at the University of Texas at Austin, has been known to start his lectures by writing down a few concepts and then initiating a discussion [12]. As the discourse progressed, he would provide key details and guidance and, at the end, the class would converge, through logical reasoning, at the topic of the day. As a student, the author had encountered a number of similar courses, both at the Indian Institute of Technology at Kanpur and at Stanford University, that aimed at discovery through lectures and challenging examinations. In every one of these classes, the instructor would start with the most basic principles and, through logical reasoning, arrive at the highest-level, most complex concept. Clearly, the assessment of both the extent and the significance of the discovered knowledge, in any given advanced graduate course, is dependent on the current literature, and the quality of the instructor and the students and, is therefore, essentially subjective.

The data reported in the remainder of this section relate to the discovery of knowledge, triggered through lectures and challenging examinations, that are obtained from a total of five courses taught by the author at ASU, two of which relate to hardware description languages (HDL), a third to modeling and simulation of complex systems, and the remaining two relate to high-speed networks. The two hardware description language courses, offered in two consecutive years, had a total of 40 and 52 students, respectively. The strength of the modeling and simulation course was 12, while the two networking courses had strengths of 55 and 18, respectively. The teaching style in these courses were similar to the ones described earlier in this section. The instructor would state a topic of discussion for the day, start from the first principles, pause at critical points, initiate discussions by the students in the class, and gradually build up to the advanced topics through reasoning. The critical points would correspond to radical turns in the evolution of the discipline and the instructor would carefully encourage, but never insist, new ideas and directions, anticipating fundamentally new ways of thinking from the class. At times, one or more students would volunteer new ways of thinking on their own and take control of the discussion. Thus, the level of interaction would be significantly high and it would be common for the students in the class to ask as many as 20–30 questions during a 75-minute class, many of which would be highly thought- provoking. In his/her mind, the student would think that as a reasonably intelligent individual, he/she could follow from the first principles and think the material through, methodically and precisely. During a HDL class, near the beginning of the semester, the discussion focused on the evolution of HDLs and the instructor started by presenting material on ADLIB-SABLE, the first behavior-level HDI—. A question was then placed before the class for discussion: Given that ADLIB-SABLE lacks a language feature to provide the simulation time to the user, can one verify whether the setup and hold conditions are violated? This had been a key difficulty with ADLIB-SABLE for which a later, industrial version of ADLIB-SABLE, termed Helix, had to add a new facility to provide the current simulation time. Despite the seemingly impossible task, two students took the lead to come up with a simple, yet creative solution, that does not appear in the literature. In their approach, the value of the setup delay is examined and it is noted that it must constitute an integral multiple of the basic simulation time. The setup delay value is known to the designer a priori and let us assume that it is 4 ns. In the ADLIB comptype for the flip-flop, D and clock are input signals representing the D input and the clock. In addition, internal signals D1, D2, D3, and D4 are defined. These internal signals are derived from the D signal, subject to delays by 1 ns, 2 ns, 3 ns, and 4 ns, respectively. At the appropriate active clock transition, i.e. low to high say, the ADLIB code examines whether the value of the signals D1 through D4 are all identical, i.e. either all are low or all are high. If affirmative, the setup condition is satisfied. Otherwise, the setup requirement is violated. We need to ensure that the D input is stable, i.e. either at 0 or at 1, for at least 4 ns prior to the active clock edge. This is achieved by examining its value at the current simulation time and each of the points 1 ns, 2 ns, 3 ns, and 4 ns, prior to the current simulation time.

During another class on HDLs, the instructor explained the three distinct functions of internal signals in VHDL. The first function is to interconnect two ports of two components. Second, a designer may also choose to use signals in a structure description to store data along with the timing information, unlike variables that do not carry any notion of timing. Third, signals ‘may be used to represent internal buses, to store a value internally at a specific time in the future, or to project an event in the future to serve as a reminder, in a behavior description. A student in the class made the astute observation that the use of a signal in an architecture description of type structure assumes the role of a global signal to all of the constituent components which, in turn, would cause difficulty in the simultaneous execution of the concurrent components.

During a class on modeling and distributed simulation, the discussion centered around the conditions under which an entity may be simulated, i.e. executed on the host computer. For accurate results, an entity must wait until the inputs are available and, only upon receipt of all of the necessary inputs, the entity is executed. Clearly, in a distributed environment, this may lead to deadlock [14] and starvation [13]. The instructor then paused and posed a question to the class: Is it absolutely essential to insist on receiving all the resources first and, if not, what are the implications? After some discussion, the class correctly observed that an entity may be executed for limited input stimulus but that when subsequent input stimuli are available, the correct output must overwrite the previously generated incorrect result. The class had not only discovered the time warp algorithm that had been proposed in the literature [15] but that they had also recognized the fundamental weakness of uncertainty inherent in this approach.

During another class on modeling and distributed simulation, the discussion focused on the intricacies of executing an asynchronous distributed algorithm on a loosely-coupled parallel processor system, the potential races that may result from timing errors, the problem of isolating races especially their propensity towards intermittentness, and the great difficulty associated with determining the true source of the errors. The instructor started by describing an actual problem that had occurred while executing behavior-level models [13] on the Bell Labs hypercube. Behavior models for the different components of a digital system were developed and each model was allocated to a unique processor of the 64-node hypercube loosely-coupled parallel processor system. The connectivity between the components of the digital system were represented through software connections between the corresponding processors. The overall execution was under the control of an asynchronous, distributed, discrete-event simulation algorithm. As a result, there was no notion of global simulation time and each model executed independently and concurrently. Upon receiving an external signal transition, a model, say M1, was initiated for execution, following which it would either generate or not generate an output response. When an output transition was generated, it was propagated to all other models that were connected to the output of M1. The communication primitives were non-blocking, implying that when a recipient model was busy but had adequate buffer space available, M1 would succeed in leaving the output transition in its buffer and then continue with its own execution. However, where the buffer of a recipient model, say M2, was full, for whatever reasons, M1 would fail in transferring the output signal transition to M2. Under these circumstances, M1 would continue on with its execution but first it would store the output transition and then re-attempt to propagate it to M2 at the next opportunity when presumably M2's buffer was no longer full. During execution, erroneous outputs were generated which were ultimately traced to a model, say M3. An initial examination of the behavior description of M3 yielded no errors. Next, a very limited number of print statements were included in the behavior description of M3 to provide a peek into its dynamic behavior. The error disappeared and correct results were obtained. However, as soon as the print statements were removed, the error reappeared. Thus, the error was clearly dependent on the timing and the addition of the print statements was causing the relative timing to be altered sufficiently to prevent the error from manifesting itself. The scenario fitted the classic Heisenberg uncertainty principle in that the use of a probe to observe the source of the error was affecting the very error itself. Every attempt to uncover the source of the error through the execution of the code and the observation of the execution results ended in failure. It had taken the instructor and another researcher at Bell Labs, a few years ago, a great deal of time and careful analysis to uncover the cause of the error. As the instructor was about to explain it to the class, one the students quickly reasoned from the fundamental principles of asynchronism and concurrency and made the following observation which, indeed, constitutes the correct explanation of the source of the error. Assume that the execution of M2 is slower than that of M1 and, as a result, one or more of Ml's messages are not successfully delivered to M2. That is, these messages are stored within M1 and, thereafter, M1 continues with its execution. During its subsequent execution, M1 generates another output transition which it immediately sends to M2, successfully. However, since the previously generated output transitions for M2 are still stored within M1, M2 ends up receiving transitions from M1 in incorrect order. To correct the error, therefore, the behavior of the model must be modified as follows. When a new transition is generated, it must first be stored in local storage in the correct order and then the model must attempt to propagate the entries to their correct destinations.

In the same class on modeling and distributed simulation, a discussion was centered around a student's project on modeling the foraging behavior of ants and understanding the emergent behavior of a colony whose constituents are driven by a simple set of rules. The discussion evolved and eventually led to a scientific, empirical understanding of creativity through modeling and simulation [20].

During a class on networking, the discussion focused on security issues in data and ATM networks. The instructor presented the traditional view of security which observes that the weakest link determines the overall system security and requires that all nodes and links of a network be equally secure. A question was placed before the class: Is a different paradigm for security feasible, today or in the future, and, if affirmative, what would it require of networks? One of the students in the class took the lead and came up with a radically new and unprecedented concept—security on demand [16]. The approach is realizable in ATM networks. In this approach, a channel with unique security attributes is created for each user, at the conceptual level, and the user's messages are transported through it.

Clearly, the extent and significance of the discovered knowledge in these examples is appreciably high. The students in these classes revealed both a deep confidence in themselves and a fundamental appreciation for the depth of knowledge. A sample of their candid comments include the following:

- The class was very thought provoking.
- I would have never believed that anything could ever interest me in hardware design languages.

- I have never had a course like this one before to compare.
- I took this course not because I had to but because I wanted to and actually looked forward to every class
- I truly learned the fundamentals of concurrency and timing.

One student comment is particularly elaborate and especially revealing: When I am encouraged to question the book, the literature, and the instructor, where I am granted a wide latitude relative to the questions, and when my questions are answered logically, from the fundamental principles, I gain a tremendous amount of confidence. While this self-confidence is very important to me as a graduate student, it may hold immense value to a beginning graduate student. The sooner one gets exposed to this style of learning, the quicker and deeper the student can engage in critical thinking and generate different approaches to problems.

ISSUES RELATED TO THE ENVIRONMENT UNDERLYING THE ADVANCED GRADUATE COURSES IN CS&E

The environment underlying the advanced graduate courses include the students, instructor, course material, and other factors that affect the design and delivery of the course. Given that (i) the nature of advanced knowledge is subtle, (ii) the metric—discovered knowledge—is subjective, i.e. its assessment is up to the instructor, and (iii) that the focus of the metric is on the quality of the courses, the success of such courses requires a critical balance of the key attributes of the environment. The key issues are six-fold:

1. The desirable qualities of the instructor may be described as follows. The instructor should conduct research in the field, possess practical experience and, where possible, industrial experience, expend considerable time thinking about the issues in fundamental ways, be dedicated, and maintain enthusiasm in learning new things in new ways. By way of suggestions to develop, within oneself, the art of thinking fundamentally, Prof. Feynman [8] relates two personal experiences—one where he states that he was 'afraid to read it [a paper] thinking it was too difficult,' and another where he states that, while engaged in explaining something in physics, his mind is so completely occupied by physics that he would feel immune even to nervousness. The eagerness to teach should be natural, not forced, should stem from within, and the instructor's love for sharing knowledge should be intrinsic. The teaching ought be for the sake of knowledge and must be motivated by the instructor's genuine obligation towards and concern for the long-term welfare of the students. The instructor ought to be constructive and patient with the students, extend wide latitude to encourage them to be innovative, never discouraging, and always respectful of their views. Even when a student's view is incorrect, the instructor ought to reveal the flaw through careful reasoning, while maintaining a good sense of humor. The students should be assured that in the course of learning to think, their grades will not suffer and that they would be evaluated based on their depth and quality of knowledge. This, in general, may require considerable effort.
2. For their part, the students ought to be sincere, hard working, willing to learn and grow, possess passion for comprehension, and extend respect, trust, and faith in the instructor's teaching. For such courses to be successful, the students should possess adequate maturity to recognize the value of such courses, distinguish between fundamental thinking and overtly vocational skills [7], recognize the long-term benefits of fundamentals as opposed to grades, and realize that knowledge is constantly evolving with the unfolding of nature. The students should neither be afraid to face challenges nor hesitate to challenge established knowledge and the instructor. They should be capable of verifying that the instructor is sincere about respecting and learning from the students.
3. Both the instructor and the students should approach such courses with the spirit of curiosity, adventure, pursuit of scholarship, and a deep appreciation for philosophy and beauty in nature. For true success in such courses, neither the instructor nor the students should view each other as an adversary but as a collaborator in a valuable experiment.
4. The role of the administration overseeing the educational environment is critical to the success of advanced graduate courses. The role should be one of encouragement, nurturing, understanding, patience, and a genuine commitment to help create first-rate, original thinkers. The administration's foresight will constitute an asset, not only to the university but, ultimately, the entire society.
5. Although grades are secondary to learning, the assignment of the grades at the end of the semester should correspond first to the quality and second to the quantity of the discoveries made by each student. The aim, however, should be to help develop the insights in every individual such that they all receive the highest grade. It is pointed out from experience that once a student has understood the subtleties of thinking, he/she gains an immediate understanding of its enormous value and the relative unimportance of grades.
6. While the number of students interested in such courses is likely to be modest at leading research universities, especially in urban settings

with a strong industrial base, that at smaller institutions may fall below the critical value. Nevertheless, the design and delivery of such courses ought to be encouraged at every institution, where appropriate.

Uses of the metric

Once a metric is generated for a given course, instructor, and student population, it should be analyzed and interpreted by enlightened academicians and administrators and even then with great caution. The metric is qualitative and subjective and its measure is susceptible to significant changes under different settings, instructors, and student populations. The metric may be used to determine whether the instructor has been successful, if the quality of the student population is high and they have demonstrated their capacity to think creatively, whether it has been worthwhile for the department to have offered such a course and should the course be offered again, and to characterize and document any new ideas generated from the course.

CONCLUSIONS

This paper has presented a philosophy underlying the design of a few advanced graduate courses at ASU in the sub-disciplines of hardware description languages, communications networks, computer-aided design of digital systems, distributed systems, distributed algorithms, and modeling and simulation. The philosophy has given rise to a new metric—the extent and significance of the knowledge 'discovered' by the students, towards evaluating the quality of such courses. Discovery refers to the knowledge that is brought out into the open by the student for himself/herself and it is significant in that it becomes an integral part of the individual who not only gains invaluable insight and confidence in the subject matter but can improvise, reason, and apply it to other areas in creative ways. The paper illustrates the application of the metric through a number of actual cases encountered during teaching at ASU. It also presents a list of the desirable attributes of the surrounding educational environment to ensure success in the design and realization of such courses. A comparative evaluation of the performance of the students exposed to discovery-based learning versus those exposed to the traditional style, is difficult because a student's subsequent environment and the opportunities therein are hard to control. Nevertheless, based on informal interviews with the students, the increase in the students' self-confidence and the enhanced ability towards critical analysis, are real and undeniable. One clear evidence of the superior learning lies in the discoveries themselves. One individual's self assessment reflects the general consensus, 'Previously, I looked at issues in general and my understanding was cluttered, neither solid nor comprehensive. Now, after having taken the course, I start with what I know and then build successively on top of that to gain a complete and thorough understanding of the issue, from the finest detail to the most condensed, highest-level concept. I can now see the practicality of the issues and their connection with the real world. This course has changed my attitude towards learning and research.'

Acknowledgments—The author gratefully acknowledges every student with whom he has had the privilege to learn together and specially thanks H. Jerry Schumacher and Qutaiba Razouqi of ASU, Dr. Seong-Soon Joo of ETRI, Korea, and Prof. Jeff Capone of ASU for discussions that have benefited the paper immensely.

REFERENCES

1. K. Henderson, Educating electrical and electronic engineers, *IEE Eng. Sc. Educ. J.*, **6,** 3 (June 1997), pp. 95–98.
2. P. Denning, New university, *Leading Edge*, **16,** 7 (July 1997), Society of Exploration Geophysicists, pp. 1013–1016.
3. E. Koehno, Engineering perceptions of ABET accreditation criteria, *ASCE J. Professional Issues in Eng. Educ. Practice*, **123,** 2 (April 1997), pp. 66–70.
4. E. S. Doderer and P. I. Giolma, If you want to teach engineering, *ASEE 1995 Annual Conf. Proc., June 25–28*, Anaheim, CA, Vol. 1 (1995) pp. 1351–1356.
5. D. Mackle, Quality—an end-user's perspective, *IEE Science, Education, and Technology Division Colloquium on Quality in Higher Education*, *April 26, 1995*, IEE, London, UK, pp. 5/13/1.
6. T. Bendell, Quality in higher education: a confusion of concepts, *IEE Science, Education, and Technology Division Colloquium on Quality in Higher Education, April 26, 1995*, IEE, London, UK, pp. 4/14/3.
7. E. A. Lee and D. G. Messerschmitt, Engineering, an education for the future, *IEEE Computer*, **31,** 1 (January 1998), pp. 77–85.
8. R. P. Feyriman, *Surely you are joking, Mr. Feyriman!* W.W. Norton and Company, 1985.
9. Sumit Ghosh and Tapan J. Chakraborty, On behavior-level fault modeling of digital designs, *Int. J. Electron. Testing, JETTA*, **2,** 2 (June 1991) Kluwer Publishers, Boston, pp. 135–151.
10. Sumit Ghosh, An exercise in inducing creativity in undergraduate engineering students through challenging examinations and open-ended design problems, *IEEE Trans. Educ.*, *Special Issue on Computation and Computers in Electrical Engineering Education*, **36,** 1 (February 1993) pp. 113–119.
11. Sumit Ghosh, Experiences with undergraduate honors theses in a computer engineering program, *Proc. 1998 ASEE/IEEE/IEEE-CS Frontiers in Education Conference*, Tempe, AZ, Nov 4–7, 1998.

12. Private Communications with Dr Paul Thurston, December 1997, formerly at Cornell University, Department of Mathematics.
13. S. Ghosh and M.-L. Yu, An asynchronous distributed approach for the simulation and verification of behavior-level models on parallel processors, *IEEE Trans. Parallel and Distributed Systems*, **6,** 6, (June 1995) pp. 639–652.
14. K. M. Chandy and J. Misra, Asynchronous distributed simulation via a sequence of parallel computations, *Communications of the ACM*, **24,** 4 (April 1981) pp. 198–206.
15. D. Jefferson, Virtual time, *ACM Transactions on Programming Languages*, **7,** 3 (July 1985) pp. 404–425.
16. H. Jerry Schumacher and Sumit Ghosh, An integrated approach to security on demand in ATM networks, *J. Inf. Sys. Secur.*, Auerbach Publications, Winter (January) 1998, pp. 10–21.
17. Fred Barlow, *Mental prodigies: an enquiry into the faculties of arithmetical, chess, and musical prodigies, famous memorizers, precocious children and the like*, Greenwood Press, New York, 1969.
18. Private communications with Dr Sohail Malik, Associate Research Fellow, Kimberly-Clark Corporation, Georgia, July 1998.
19. Sumit Ghosh, New metrics for assessing quality in graduate courses in computer science and engineering, *Proc. 1998 ASEE/IEEE/IEEE-CS Frontiers in Education Conference*, Tempe, AZ, Nov 4–7, 1998.
20. Peter S. Heck and Sumit Ghosh, A study of the meta-level, emergent behavior in a synthetic ant colony through modeling and simulation, Proceedings of *ANTS'98—From Ant Colonies to Artificial Ants: First International Workshop on Ant Colony Optimization*, Brussels, Belgium, October 15–16, 1998.
21. Private communications with Major Jerry Schumacher, ASU, July 1998.

Sumit Ghosh currently serves as an associate chair for research and graduate studies in the Computer Science and Engineering Department at Arizona State University. He had received his B.Tech. degree from IIT Kanpur, India, and his MS and Ph.D. degrees from Stanford University, California. Prior to his current assignment, Sumit had been on the faculty at Brown University, Rhode Island, and before that he had worked at Bell Labs Research in Holmdel, New Jersey. Dr Ghosh's research interests are in fundamental problems from the disciplines of asynchronous distributed algorithms, simulation, networking, computer-aided design of digital systems, and qualitative metrics for evaluating advanced graduate courses. Presently, he serves on the editorial board of the IEEE Press Book Series on Microelectronic Systems Principles and Practice. He has written three books—(1) Hardware Description Languages: Concepts and Principles (IEEE Press, 2000), (2) Intelligent Transportation Systems: New Principles and Architectures (CRC Press, 2000), and (3) Modeling and Asynchronous Distributed Simulation: Analyzing Complex Systems (IEEE Press, 2000).