Program Assessment and Program Improvement: Closing the Loop

Neelam Soundarajan

Computer & Information Science Department,

Ohio State University

2015 Neil Avenue #395, Columbus, OH 43210, USA

e-mail: neelam@cis.ohio-state.edu

Program Assessment and Program Improvement:
Closing the Loop

Abstract

Engineering Criteria 2000 (EC 2000), the recently revised set of accreditation criteria for engineering programs in the US, places considerable stress on outcomes assessments. EC 2000 requires the assessment results to be used to identify program improvements, and for such usage and the resulting improvements to be documented.

While numerous assessment instruments have been developed and discussed in the literature, less attention has been paid to the question of how to use these to improve the programs or to document this usage. In this paper, we present an approach that serves both to identify possible improvements based on the results of assessments, as well as to provide high-quality documentation. As an added bonus, it also helps incoming students and new faculty to get a good understanding of the structure and evolution of the program.

Keywords: EC 2000, Program assessments, Program improvements,

Closing the loop, Documentation of results.

1 Introduction

Engineering Criteria 2000 (EC 2000) [4] is the recently revised set of accreditation criteria of the Engineering Accreditation Commission (EAC) for engineering programs in the US. The key difference with EAC's earlier set of criteria is the importance that EC 2000 attaches to programs having well defined objectives and outcomes. EC 2000 further requires programs to perform regular assessments to see how well the outcomes are being achieved. But of course assessments are not an end in themselves. Another key requirement is '*to close the loop*'; i.e., programs are required to use the results of the assessments to identify and implement improvements. Perhaps even more challenging, EC 2000 requires programs to *document* such usage as well as the resulting improvements. Specifically, Criterion 3 reads, in part, "Each program must have an assessment process with documented results. Evidence must be given that the results are applied to the further development and improvement of the program." While numerous assessment instruments, from learning portfolios to surveys of various constituent groups, have been developed and discussed in detail in the literature, see for example [6, 8, 3, 12, 14], less attention seems to have been paid to the question of how exactly to use these results to improve the engineering programs or how to document this usage. This paper presents an approach adopted by the Computer Science and Engineering program at the author's university that has helped address both of these questions; as we will see, the approach offers a number of other advantages as well.

Pomerantz [11], in discussing the importance of 'closing the loop', suggests, "we may have been focusing on the accountability dimension of assessment to the neglect of its other important dimension: improvement." Angelo [2] presents a compelling argument that the primary purpose of assessment ought to be program improvement; he is concerned that "some view assessment as a mechanistic, technical process - a collection of monitoring devices . . . Others as a necessary, periodic bother . . . In either case, assessment is seen as something that might result in small changes, usually adding data-collection and reporting processes here and

2

there." Similar concerns have been expressed by others; for example, Peterson and Einarson [10] wonder to what degree assessment efforts have influenced decision making or produced discernible impacts on various programs. Further, perhaps somewhat indirect, evidence that assessment results are not being used as effectively as they should be to identify and effect program improvements comes from the web page of the American Society for Engineering Education (ASEE) at http://www.asee.org/colleges/assessment.cfm which provides links to the assessment-related activities of numerous programs. While most of the programs listed on that page provide detailed information about the assessment tools they use as well as summary data obtained from the assessments carried out using these tools, there is little mention of program improvements that could be attributed to these assessment activities.

Why is this? Why are assessment results not being used to effect program improvements? There seem to be two key reasons. First, improvements in courses or groups of courses are often the result of many different factors, such as changes in the field, informal feedback from students who have taken the course, formal or informal data from multiple assessments over a period of time, interactions with on-campus recruiters, etc. Second, and essentially as a result of the first point, *documenting* such improvements and tracing them or attributing them to specific data obtained during specific assessment activities is quite difficult. The *course group report* mechanism described in this paper is a powerful tool for addressing these considerations. As we will see later in the paper, this mechanism provides a number of advantages for the program, the students, and the faculty:

1. It enables, indeed requires, the group of faculty most directly connected with each group of related courses to carefully consider and produce a report that addresses various important issues, including information from recent assessment activities, actual performance of students in the individual courses, informal feedback that may have been received from various constituents, evaluate possible changes in the courses and their potential impact both on other courses that depend directly upon the courses in

this group, as well as on the program as a whole; and to do this on a regular basis.

2. It generates clear and reliable documentation of the various improvements resulting from the assessment activities. This documentation will be of great value during accreditation evaluations.

3. Perhaps even more important, over time, the set of reports provides current and incoming students as well as new faculty and other interested constituents, an easily accessible and succinct record of how the program has evolved and the rationale behind that evolution.

The rest of the paper is organized as follows: Section 2 summarizes the assessment instruments used in the author's program and relates them to the two-loop model [1]. Section 3 describes the Course Group Report (CGR) mechanism and how it helps "close the loop". Section 4 evaluates the approach by comparing it with some other approaches that have been presented in the literature. The final section summarizes the work and conclude with a program-level question and a possible way to address it.

2 Assessment Instruments

Most engineering programs (the sites linked to by the ASEE web page mentioned earlier provide numerous examples) use a variety of assessment mechanisms ranging from standard examinations in individual courses to surveys of alumni and employers. The following assessment instruments are used in the author's program:

1. Assignments and examinations: Assignments, including programming projects, and examinations are standard in most courses.

2. Student Evaluations of Teaching (SETs): In each course, students have the opportunity to provide feedback on the effectiveness of the the course via anonymous SETs. The
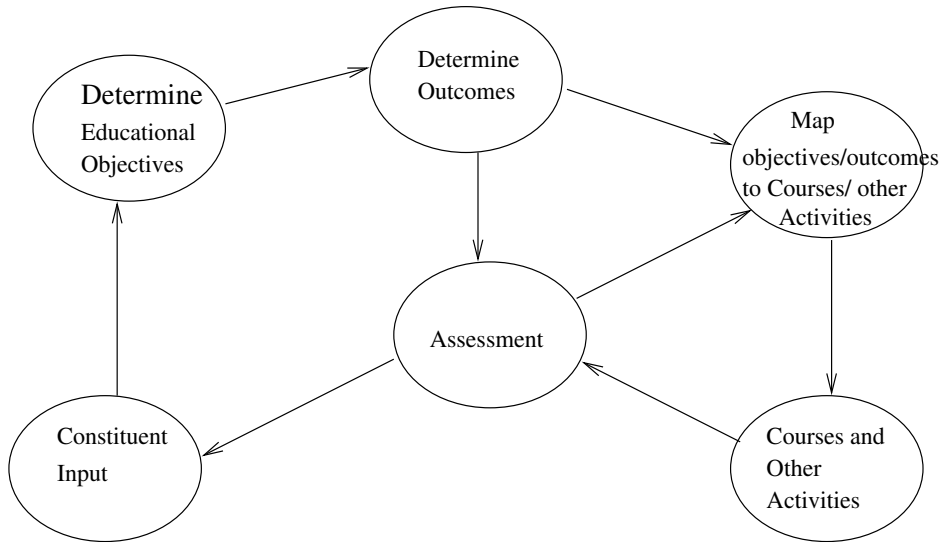
Figure 1: Two-loop Model of Assessment

SETs ask the respondent to evaluate both the quality of instruction as well as the quality of the notes and other instructional material, the assignments, the projects, etc.

3. Exit surveys: All students who are near graduation are required to complete an *exit survey.* The survey allows the student to rank, for each of the objectives and outcomes of the program, how well that objective or outcome was met, as well as how important the objective or outcome is.

4. Alumni surveys: These are similar to the *exit surveys* and allow recent alumni to rank the importance of each objectives and outcomes in their current professional position, and rank how well they felt they were prepared by the program with respect to that objective or outcome.

5. Supervisor/Manager surveys: This again is similar; the respondents –alumni who graduated several years ago and can now be expected to be in supervisory positions– are asked to rank the importance of each objective and outcome and how well the program prepared the respondent's recent supervisees with respect to each.

These assessment instruments can be related to the *two-loop* model [1] shown in Fig. 1. Different programs have used slight variations of the model. The key point is that there are two loops in the system, one that is primarily concerned with determining program objectives and outcomes, and the other that is concerned with the actual courses and other components of the program having to do with the activities that students engage in. Data obtained from the assessment activities is central to both loops, and is what drives the program toward improvement. In terms of this model, the exit, alumni, and supervisor surveys, items (3), (4), and (5), in the foregoing list of assessment instruments, are mainly part of the loop on the left. Items (1) and (2), respectively assignments/examinations in individual courses and student evaluation of teaching, are mainly part of the loop on the right.

But, as noted in Section 1, EC 2000 does not just require that the data from the assessments be used to improve the program, but also that the program provide documented evidence that this is being done. It is here that there is an important problem. To quote Lohmann [7], "Rarely are curriculum changes and other major academic policy shifts the result of a single survey, one meeting of an advisory board, etc. They are typically the result of input from many sources, often over a period of time, and frequently involve many faculty meetings and committees. As such, ... document[ing] the connection from a given result to a specific change was very difficult." Indeed, many of the improvements in individual courses tend to be triggered by spontaneous discussions between various faculty and between faculty and students in hallways and classrooms, rather than as a result of inputs gathered via formal instruments such as surveys. How then is the requirement of creating, on a continuing basis, documented evidence of the improvements in the program resulting from the assessment activities, to be met?

3 Course Group Reports

In order to deal with the dual challenges of using the data obtained from the assessments to improve the program and of documenting these improvements, we adopted the following approach in our program. The entire *Computer Science and Engineering* curriculum is organized into seven or eight groups of related courses. One group, for example, is the Software Engineering course group; another is the Networking course group; etc. For each course in a given group, the faculty coordinator for that course along with the other faculty members most directly involved with that course have the responsibility of identifying the specific objectives of that course, the corresponding learning outcomes, as well as the relation of these objectives and outcomes to the objectives and outcomes of the program as a whole. In addition, the faculty are also required to map the objectives and outcomes of the course to the EC 2000 Criterion 3 outcomes. This mapping is specified in terms of a precise scale to characterize this relationship. But since the focus in this paper is on how assessment results are used to identify possible improvements to the program and how to document these, the details of this scale are omitted.

At least once every two years, the faculty coordinators of the courses in each course group are responsible for producing a report, the *Course Group Report (CGR)* for that particular group of courses. The CGR is expected to address such questions as whether prerequisites for courses in the group are appropriate, and whether the courses in the group are helping meet the overall program outcomes and objectives as intended (and as specified in the mapping described above). More specifically, the CGR addresses the following items with respect to each course in the corresponding group:

- Are the course objectives appropriate and up-to-date? With the current pace of change in many disciplines, it is important to update course objectives in a timely manner.

- Are the current prerequisites appropriate? Are the prerequisite courses not covering

any material that they ought to cover? Or covering material that should be left to this course?

- Is the current textbook suitable? Has anything better become available?

- Student reactions: Does the average student in the course seem prepared for it? Do most students understand the main ideas and acquire the skills that the course is meant to equip them with? Do they seem to enjoy the course? Anything unusual about how student reactions were assessed, i.e., were anything beyond exams, assignments, and SETs used for assessments?

- Relation to rest of the program: Is the course helping meet the overall program objectives in the way it was meant to? Does it fit well with the other courses in the program?

In answering these questions, the group is expected to draw upon (and document in the CGR) the inputs obtained from the various assessment mechanisms listed in Section 2, as well as inputs that may have been obtained by other possibly ad-hoc means. For example, the CGR on the Programming Languages group noted the feedback from students, and alumni, as well as from on-campus recruiters, concerning the importance of providing opportunities to students to learn recent languages used widely in industry, such as *Perl* and *Python*. In response to this, the group proposed the creation of a new 1-credit course on *Perl* and this course is now offered regularly.

One point to note is that although each of these questions applies to each course in the group, the close relations among these courses –indeed that is why they were all grouped together in the first place– means that the answers for the different courses are often strongly tied to each other. For example, a change in one course could well trigger the need for changes in one or more of the other courses in the group. Thus the introduction of the course on *Perl* led to some changes in the *Concepts of Programming Languages* course since one of the main

8

topics that this latter course deals with has to do with ideas underlying implementations of languages, and the approach used in the implementation of *Perl* is rather different from that of most other languages. This change was not reported in the last Programming Languages CGR since it happened after that report was prepared. It will be reported in the next CGR for the group; and the report will document not just the change, but relate it to the previous change, in the form of the introduction of the new *Perl* course and the reason for that change.

Once the faculty members involved with the courses in the given group have prepared a CGR, they present it to the departmental Curriculum Committee. This presentation serves two important and related purposes. First, since the members of the Curriculum Committee hear the presentation of each CGR, they are able to form a 'global' picture of the program and see how the program as a whole is evolving. Second, this global perspective enables these members to suggest further ideas for changes and improvements in this group of courses based on how the rest of the program is evolving. Thus while the faculty most directly involved with the given group of courses are the ones who are primarily responsible for changes in these courses, the involvement of the other faculty, in the form of the members of the Curriculum Committee, with knowledge of and expertise in the rest of the program, helps ensure that these courses benefit from changes in the courses in the other groups, and in turn that courses in the other groups benefit from changes in this group.

One issue that often receives limited attention is the question of possible changes and improvements in courses taught in other departments that might have a direct impact on the effectiveness of key courses in the program. A case in point for the author's program is the discrete mathematics course (taught by the Mathematics department) which is an essential prerequisite for a number of computing courses, such as the required course on databases. The faculty involved with the database course had noticed for some time that students were not as well prepared as they ought to be with respect to some important concepts from mathematical logic that serve as a key underpinning for database systems, and that

should be presented in the discrete mathematics course. They had tried to bring this to the attention of the mathematics faculty involved with the discrete mathematics course but these attempts were mostly at an informal level and did not result in any improvements. The CGR provided the database faculty a formal and well-defined mechanism that allowed them to document the problem precisely and its impact on the database course. Following this, changes were made to the discrete mathematics course, and the students now seem better prepared for the database course.

Interestingly, one point that came up during the CGR presentation in the Curriculum Committee was that revising the discrete mathematics course to include a more thorough discussion of the relevant concepts from mathematical logic would mean *reducing* the focus on such topics as recurrence relations that are an essential prerequisite for the course on algorithm analysis, another required course! Fortunately, in the meantime, it had been independently decided that the program should require students to take *two* courses on discrete mathematics although at that point no decisions had been made about the specific topics that the second course would include. The presentation of the database CGR made it clear that a good candidate would be the topic of recurrence relations. It is true that the idea of moving this topic from the first discrete mathematics course to the newly-required second course is a natural one, and might well have occurred even in the absence of the database-CGR presentation at the Curriculum Committee and the ensuing discussion. But the fact remains that the CGR mechanism is well suited to identify such relations between different parts of the program and the resulting impacts that changes in one part have on another.

So far our discussion has focused on the changes and improvements in the program that are identified based on the assessment data. But occasionally we are faced with a different problem: one or more of our assessments suggest a particular change that the faculty, as educators and as computing professionals, feel would *not* be an improvement to

the program. The most common change of this kind has to do with suggestions to introduce courses focusing on a particular technology or set of tools that happens to be currently 'hot'. While it may be true that students trained in the particular technology would have an edge in the immediate job market, if introduction of such courses would take focus away from preparing them for long-term success in the profession, that would clearly reduce the value of the program. Often though, it is not a cut-and-dry question; if a particular 'hot' technology is used widely enough, it may indeed be appropriate to try to include some discussion of it in the program and to try to also include discussion of how it relates to the relevant portions of the rest of the curriculum. This is precisely what happened, as described earlier, with the popular programming language Perl; some coverage of this topic was introduced (in the form of a 1-credit course) in the curriculum, and it was related (in the Concepts of Programming Languages course) to the relevant part of the core program. Whichever of these actions is taken, the CGR provides faculty with an opportunity, and an obligation, to document the action and the rationale for it.

In summary, the Course Group Report mechanism has proved to be extremely valuable: First, it allows the most appropriate groups of faculty to summarize data collected from various assessment activities and present them in coherent contexts of groups of related courses, rather than as tables of raw numbers. Second, it ensures that faculty in related courses make it a point to interact with each other on a regular basis, when preparing the CGR for the particular group. Third, it enables faculty who are not directly involved with a given group of courses to still get a broad understanding of the courses in the given group; this is especially important in the case where courses in one group are prerequisites for, or otherwise strongly related to, courses in another group. Fourth, it allows students, as well as new faculty, to acquire a thorough understanding of the various courses in the curriculum, as well as how the curriculum evolved and the rationale behind that evolution, by perusing recent CGRs of relevant groups. And finally, the CGRs provide the evidence called for by EC 2000 to show how the assessment results are used to improve the effectiveness of the

program.

4 Evaluation

In this section, we compare the CGR-based approach with others that have been considered in the literature. Scoles *et al.* [14] describe the assessment instruments and the approach used in their program to arrive at improvements based on the results of the assessments. The assessment instruments are quite similar to the ones described in Section 2; one additional instrument they use is an annual survey that allows faculty to evaluate individual courses. The results of the assessments are reviewed by the department head and if the review leads to the conclusion that changes are needed in course contents, teaching method, or some other aspect of the program, then an action plan is drafted and put into place. The advantage of the course group-based approach in comparison with this approach is that the former allows the right group of faculty to analyze not only the numerical results from the assessment instruments but also account for any informal feedback that may be received and also to account for changes in the field; and the CGR allows not just changes to be documented but also the rationale for changes that one or more of the assessments may have suggested but that the group decided would not be appropriate. And this is all in the context of groups of related courses, rather than individual courses in isolation from each other.

The approach that Lyons and Bayoumi [8] describe is somewhat closer to the course group-based approach. In their approach, at the program level, the department head analyzes the results from surveys and forwards them to the appropriate departmental committees. The committees prepare written responses that are presented to the faculty as a whole. At the course level, *Program Area Teams*, the faculty being organized into these teams, are responsible for reviewing individual courses. The main instruments used by these teams are Course Portfolios and Course Surveys. Course Portfolios are maintained by the course

instructor(s) and include the course objectives, the strategies used to achieve the objectives, and a description of methods used for student assessment. The portfolio also includes representative samples of student work and results from Course Surveys of the course. The area team reviews the portfolio of each course once a year and recommends any needed changes.

The area team concept is similar to our course groups. One key difference is related to the fact that an area team is defined by the faculty members making up the team; hence if a given faculty member teaches two somewhat unrelated courses, both of these courses will become the responsibility of his or her team, although the other members of the team may have no interest in one of these courses. By contrast, each course in one of our course groups is closely related to each of the others in the group. This means that when reviewing the courses, an area team treats each course that is responsible for as a separate entity, rather than considering the different courses related to a given area as a whole. Considering these different courses together allows the faculty responsible for the given course group in our approach to analyze the effects of changes in one course on other courses in the group, to ensure that there is not unnecessary duplication of topics in different courses, etc. It also allows these faculty to take account of changes in the field as a whole, and explore possible ways that the course group can evolve to accomodate these changes. And the CGR provides a unified documentation of these considerations including, as noted earlier, the rationale for possible changes that the group decided *not* to implement.

Royer *et al.* [13] describe the approach used in their program centered on the concept of a *Course Assessment Plan* (CAP) for each course. A CAP consists of a Course Brief, a Course Log, and a Course Report. The Course Brief is presented by the faculty member responsible for the course, to the entire faculty. It details such items as the text book to be used in the course, major topics to be covered, and course objectives; the brief is required to address all concerns and recommendations from the previous offering of the course, as recorded in the Course Report from that offering. The Course Log is a record of the actual course delivery

containing information about the pace of the course, student and instructor feedback on assignments, information about student performance on exams, etc. The Course Report is completed after the end of the course and is a summary of how well the course objectives were met, a discussion summarizing the log entries, recommendations for the next offering, etc. This approach, especially the Course Report, has some clear similarities to the CGR, but there are also some important differences. First, the Course Report concerns an individual course whereas the CGR deals with a group of related courses. Second, the Course Report is based on a single offering of the course; and it does not integrate the feedback from other sources such as alumni or employers. Third, a Course Report is prepared by the instructor who taught that particular course on that particular occasion, wheras the CGR is prepared by all the faculty involved with the various courses in the group; indeed, the CGR is typically also revised to take account of comments from members of the department's Curriculum Committee when the CGR is presented to the committee. Further, the Course Report has to be prepared every time a course is taught whereas a CGR for a given group of courses is prepared once every two years or so, hence the former may require a greater commitment of resources; but at the same time, since the Course Report is prepared immediately after the completion of the course, it is likely to better reflect, for example, the actual performance of students in the classroom.

Finelli and Wicks [6] present an interesting approach to assessment that might be added to the CGR-based approach as a useful component. Their key observation is that two or more courses in the curriculum may be closely related to each other, indeed the first course might provide the essential foundation material for the second. In such cases, students' understanding of the foundational material often improves as they see it used in the second course. Thus it makes sense to assess the students' understanding of this material not just in the first course where it is the main topic, but also in the second course although it is not explicitly discussed in that course. In their experience, Finelli and Wicks did notice an increase in the students' understanding of the material covered in their first *Electrical*

Circuits course as the students progressed through the second *Circuits* course where the material was used. And in the cases where students seem to be having difficulty with a particular component of this material, either or both courses could be revised to provide better discussion of that component. Clearly a similar approach would be appropriate also for a *group* of courses because even if, unlike in Finelli and Wicks' case, the courses do not form a strict sequence, advanced courses in the group tend to build on material from earlier courses in the group. Thus, in the author's program, the course on *Concepts of Programming Languages* makes use of the material taught in the *Automata Theory* course although the former is not, in any sense, a continuation of the latter. And it has been observed that students' understanding of the automata theory material solidifies considerably as they see it applied in the Programming Languages course. Thus a test of the Automata Theory material in the Programming Languages course will help confirm this, and can help identify problems in how specific topics in this material are taught, and possible improvements to address these problems.

5   Conclusion

Our goal in this paper was to address the problem of 'closing the loop', i.e., of using the results obtained from various assessment activities to improve the program and to document such improvements. The Joint Task Force on Engineering Education Assessment, in its report [9], considers the problems related to achieving program improvements based on assessment results. The report cautions "Most, if not all, measures available are limited in their ability to assess program quality. ... Until the results of these studies [validating the relationships of measures to actual professional performance] are available, actions taken to improve programs on the basis of such measurements would be both premature and ill-advised." The key advantage of the Course Group Report mechanism is that it allows the integration of results from different assessment mechanisms and to compensate for any

inaccuracies that may be present in one or more of them; further, the report can include a narrative justification or explanation of the inaccuracy in question and perhaps suggest changes to the assessment instrument to eliminate or reduce the inaccuracy.

Olds and Miller suggest [3] that an important benefit of developing an assessment plan is that the process allows each faculty member to examine the entire curriculum and to see how his or her courses fit into the program's overall goals and objectives. On the other hand, just the development of the assessment plan and the associated process are not sufficient by themselves. As noted by the **ASME** report [5], "Several programs reported that compiling the information [obtained from the various assessments] did not help improve the programs." Rather, in the experience of the author's program, it is the active engagement of appropriate faculty in carefully prepared summary reports such as the Course Group Reports that helps improve the program and provide the documentation needed to show such improvement. Further, such a mechanism also allows the incorporation of feedback obtained from informal assessments such as personal meetings with employers; the importance of such assessments is also noted by the **ASME** report.

We conclude with a brief mention of a program-level issue. The *course groups* are carefully chosen so that only closely-related courses are grouped together in the same group. As a result, the CGRs tend to be quite cohesive. But this also means that these reports do not address program-level questions. It would seem it would be useful to have another 'higher-level' report that addresses these questions. But important questions such as who would be responsible for writing this report, precisely what the report would contain beyond summarizing the individual CGRs, remain to be addressed. One possible answer for the first question might be that the Curriculum Committee be charged with producing the report since the members of this committee have heard detailed presentations of the individual CGRs and hence have a thorough program-level understanding.

References

[1] M.D. Aldridge and L.D. Benefield. Assessing a specific program. In *How do you measure success*, pages 27–34. ASEE Professional Books, 1998.

[2] T. Angelo. Doing assessment as if learning matters most. *AAHE Bulletin*, 51(9):3–6, 1999.

[3] R. Miller B. Olds. An assessment matrix for evaluating engineering programs. *Journal of Engineering Education*, 87(2):172–179, 1998.

[4] Engineering Accreditation Commission. *Criteria for Accrediting Engineering Programs*. Accreditation Board for Engineering and Technology, 2002.

[5] R. Laurenson (ed.). Initial assessment of the impact of ABET/EC2000 implementation. Technical report, American Society of Mechanical Engineers, at: www.asme.org/education/enged/abet/reports/report1.pdf, 2001.

[6] C. Finelli and M. Wicks. An instrument for assessing the effectiveness of the circuits curriculum in an EE program. *IEEE Trans. on Education*, 43(2):137–142, 2000.

[7] J. Lohmann. EC2000: The Georgia Tech Experience. *Journal of Engineering Education*, pages 305–310, July 1999.

[8] J.S. Lyons and A.M. Bayoumi. Cqi processes, results, and program improvements for engineering design. *IEEE Trans. on Education*, 43(2):174–181, 2000.

[9] Joint Task Force on Eng. Education Assessment. A general assessment framework. In *How do you measure success*, pages 13–16. ASEE Professional Books, 1998.

[10] M. Peterson and M. Einarson. What are colleges doing about student assessment. *Journal of Higher Education*, 72(6):629–639, 2001.

[11] N. Pomerantz. Closing the loop: Program review in student affairs. Technical report, NASPA, at: http://www.naspa.org/netresults, 2003.

[12] R. Puerzer and D. Rooney. Alumni survey as an effective assessment tool for small engineering programs. *Journal of Engineering Education*, 91(1):109–115, 2002.

[13] E.G. Royer, C. Wright, and D. Peterson. Assessment for Electrical Engineering programs – processes implemented at the USAF. *IEEE Trans. on Education*, 43(2):132–136, 2000.

[14] K. Scoles, N. Bilgutay, and J. Good. A new course evaluation process. *IEEE Trans. on Education*, 43(2):125–131, 2000.