Diagnosing why Representation Remains Elusive at your University: Lessons Learned from the Center for Inclusive Computing's Site Visits

Carla E. Brodley
Khoury College of Computer Sciences
Northeastern University
Boston, MA, U.S.A.
c.brodley@northeastern.edu

Catherine Gill
Center for Inclusive Computing
Northeastern University
Boston, MA, U.S.A.
c.gill@northeastern.edu

Sally Wynn
Center for Inclusive Computing
Northeastern University
Boston, MA, U.S.A.
s.wynn@northeastern.edu

***Abstract*—While best practices to support women in undergraduate computing are known and are being adopted, departments can move faster and see better results by developing a deeper understanding of issues unique to their institution. Launched in 2019, the Center for Inclusive Computing (CIC) awards grants to colleges and universities to support the implementation of evidence-based approaches that significantly increase the representation of women in computing. As part of the grant application process, the CIC conducts a site visit to assess the probability of success and sustainability of proposed plans. After 22 site visits, we have developed a strategy to uncover institutional barriers and issues through targeted questioning of key stakeholders. The site visit reveals if the institution has accurately diagnosed the issues and if the proposed interventions are appropriately aligned with the identified problems. In this paper, we detail the structure and content of the site visits, thus enabling leaders in undergraduate computing programs to identify their unique challenges and position themselves to implement well-informed broadening participation strategies.**

***Index Terms*—Broadening participation in computing, undergraduate education, women**

I. Introduction

Universities are increasingly examining the demographic makeup of their computing programs[1] and are implementing strategies to improve the representation of groups historically underrepresented in tech.[2] While best practices for broadening participation in computing (BPC) have been developed and evaluated [1], specific best practices may not necessarily align with the set of challenges facing any singular institution. To understand which best practices to implement, universities must acknowledge and accurately diagnose the primary barriers to representation at their institution before investing resources into solutions that may not address their particular challenges.

In this paper, we present and discuss the data to collect and questions to ask to help a university self-diagnose why equitable representation remains elusive. The topics are informed by the Center for Inclusive Computing's (CIC) site visits, a series of interactive meetings at universities across the country, conducted during the application review phase of the CIC's grantmaking process. This paper provides context for the history and grantmaking of the CIC, details the strategic process of our site visits, offers categories of targeted questions to guide the senior leadership of undergraduate computing programs toward an accurate self-diagnosis, and discusses common institutional barriers.

II. The Center for Inclusive Computing

Housed at Northeastern University, the Center for Inclusive Computing (the CIC) was founded in 2019 to substantially increase the national representation of women majoring in computing in the United States. The CIC was established with the goal of transforming the national landscape of women in tech by providing grant funding that supports the implementation of best practices for broadening participation in computing at the undergraduate level. Grant recipients receive financial support and expert technical assistance from computing faculty across the nation who are established experts in BPC. Funded schools are required to collect in-depth data to assess whether the interventions are achieving the desired outcomes.

The CIC awards grants to colleges and universities committed to improving the representation of women graduating with computing degrees. *Best Practice Grants (BPG)* fund the implementation of evidence-based approaches that have been shown to substantially increase the representation of women in computing [2]. To qualify for a BPG, schools must graduate 200 or more computing majors annually. We prioritize applications that propose meaningful cultural, curricular and/or pedagogical improvements that lead to *systemic change*. Universities interested in applying for a grant should visit the CIC's webpage (https://cic.khoury.northeastern.edu/grants/best-practice-grants/), which provides application instructions and eligibility criteria.

[1] In this paper we use the term "computing" to refer primarily to computer science (CS) programs.

[2] Historically underrepresented groups in computing include women, Black/African American, Hispanic/LatinX, American Indian/Alaska Native, Native Hawaiian/other Pacific Islander, and people with disabilities.

BPGs offer schools the opportunity to implement change with substantial funding and technical assistance from expert computing faculty. As part of the application process, schools submit three years' worth of enrollment, graduation, and demographic data, as well as a Letter of Intent (LOI) detailing the proposed interventions that will be employed and the problems these strategies are designed to address. Based on the submitted data and the LOI, the CIC selects schools for a comprehensive site visit to assess the accuracy of the identified problems and the appropriateness of the proposed interventions.

After conducting 22 site visits at colleges and universities across the country between October 2019 and December 2020, the CIC has honed an approach to uncovering a school's institutional barriers to increasing representation and to pinpointing the specific places where the school is experiencing leaks in the pipeline to the computing degree. Through the site visit, the CIC works with the school to diagnose which interventions most align with the challenges facing that institution. In the next section we describe how we conduct site visits and the issues we seek to uncover through this process.

III. Site Visits: A Diagnostic Tool

Each site visit consists of a series of meetings with an applicant institution. Before and during the site visit we often ask for additional data; e.g., 1) the drop/fail/withdraw (DFW) rate for each class in the introductory course sequence broken down by demographics and 2) the demographics of students admitted from high school, transfering from community college, and transfering from another major (i.e., internal transfers).[3]

During the visit, we meet with all stakeholders including the project leads, women currently enrolled in the program, instructors who teach the introductory sequence, teaching assistants (TAs), academic advisors, and the Dean. In this section, we detail the purpose of each meeting with each stakeholder. The site visits are designed to assess the following questions:

1) Probability of success - How much change is possible over five years and how attainable are the stated goals?
2) Alignment - Are the problems identified accurately diagnosed and do they align with the proposed interventions?
3) Sustainability - If funded, how will the proposed interventions be sustained?

Project leads: This is our first meeting of the site visit. Our goals are to understand the different pathways that lead to a major in computing and to dig into the content and pedagogy underpinning the introductory sequence. In addition, we discuss the potential barriers that could impede the success of the proposed interventions. We meet with the project leads again at the end of the day to discuss any issues that arose in meetings with the other stakeholders that warrant more discussion, information or data.

[3] Because this data can be time consuming to collect we do not ask for it when schools submit their initial Letter of Intent.

Undergraduate students: In advance of the site visit, we ask that the project leads recruit a group of undergraduate women. Our goal is to gain insight into the culture of the department, as experienced by the students themselves. We seek to understand the reputation of computing at the university and the existing support structures for students. To this end, we ask the students how they arrived in the major (direct admission, internal/external transfer), whether they had prior experience in computing, and how they utilize help and advice from faculty, advisors, teaching assistants, tutoring resources, peer mentors and student groups. Note that for some universities not all of these resources are available.

Instructors on the introductory sequence: We meet with faculty who teach the introductory course sequence (typically the first three courses) based on the assumption that after a student successfully completes the first three courses, they rarely leave the major (e.g., as observed by [3]). Indeed, in 21 of the 22 site visits, the conversations with faculty supported our assumption. In our meeting with instructors, we seek to understand the culture among instructors on the introductory sequence. We discuss if the instructors collaborate to sync all sections of the same course, if they participate in anti-bias training, how they recruit their teaching assistants, and how they handle the bi-modal distribution of students with or without prior experience in the subject area that is often observed in these initial classes. We also ask questions to understand the DFW rates for the introductory courses and whether the DFW rate varies by race and gender. Throughout this diagnostic process, we assess the level of buy-in for the proposed interventions, because it is critically important that the instructors on the introductory course sequence believe in and care about making computing accessible to everyone.

Teaching assistants (TAs): Depending on the school this meeting can be undergraduate and/or graduate TAs. In the meeting, we seek to assess the quality of training and guidance received. Examples of questions we ask include: Do they receive anti-bias training? Are they trained to help students without doing the work for them? How are they selected to be TAs and how are they evaluated? What efforts are made to recruit undergraduate TAs from historically underrepresented groups in computing? In the majority of our site visits we have observed three challenges with regard to TAs: there is little to no training, they do not receive any formal (or even informal) evaluation, and the recruitment/selection of *undergraduate* TAs for each class is often left to the faculty member teaching that class (i.e., no centralized application process exists). Through our 22 site visits, we have observed that having faculty drive (or lead) the selection of undergraduate TAs, without explicit anti-bias training, is likely a contributor to the underrepresentation of certain demographic groups.

Academic advisors: In the meeting with academic advisors, we seek to understand the advisor to student ratio, the advising model (e.g., dedicated professional advising staff, faculty, or a combination of both), and the level of holistic support available to students. Speaking with advisors offers insight into student struggles, common academic and structural barriers, quality of support, and the degree to which students are taking advantage of resources.

Meeting with the dean: This is a one-on-one meeting between the executive director of the CIC and the dean (and in some cases with the Provost, depending on the project leads and the school's structure). Our goal is to assess the level of institutional support available to sustain the interventions after the grant period. We ask questions about the budget model to better understand any institutional constraints; e.g., for an RCM[4] school, it can be challenging to implement *CS+X*,[5] because it will have direct implications on the budget of other units. In addition, even in a non-RCM budget model, money follows enrollments (albeit more slowly), which might mean that other deans may resist the growth in computing enrollments. We assess whether the dean is in favor of enrollment caps, or whether they are willing to hire additional faculty to keep pace with enrollment. We also ask questions to understand whether the project leads have the necessary positional authority to make the proposed changes.

IV. Identifying Institutional Barriers

Several notable institutional barriers have emerged during the site visits. If an institution is unable to adequately answer or address the following, schools are highly encouraged to investigate how to remove these barriers and resubmit their proposal.

A. Addressing (Lack of) Prior Experience

Because computer science is not consistently offered, let alone required, across all school districts in the United States, students enter college with dramatically different levels of prior coding experience. Even when computer science is offered in a high school, it is most frequently offered as an elective. As a result, it has been observed that students from groups historically underrepresented in computing often enter college with little to no prior programming experience [5]–[7]. A bi-modal distribution of prior experience creates a challenge in the introductory course sequence in most universities. If ignored, it leads to a significantly higher DFW rate for students without prior experience and can create an inhospitable environment in the introductory sequence classes [8]. Best practices to deal with a bi-modal distribution of prior experience exist and should be implemented [9]. A second challenge for schools who have a large community college transfer population is whether the community college's computing courses have the same curriculum and preparation as courses offered at the university. Too often students from community colleges need to start almost at the beginning of the computing curriculum when they transfer to a 4-year institution because their community college may have used a different language on the computing introductory course sequence.

B. Covert Caps on Enrollment

We understand that capping enrollments may seem like a necessary policy as universities struggle with the booming enrollments in computing. However, it is well-established that explicit or implicit enrollment caps can hurt diversity [4]. In addition to explicit caps on enrollment, we have observed two types of *covert caps* that impede efforts to broadening participation.

Requiring a very high GPA in the introductory course sequence to get into the major. This policy favors students with prior computing experience and can create a toxic environment of competition. Most schools have some or all of the following ways in which students become computing majors: 1) direct admission to the major from high school; 2) direct admission from community college; 3) admission to a college (engineering, humanities, etc.) with the requirement that they must declare a major in their sophomore or junior year; or 4) internal transfers (i.e., when students want to transfer into computing from another major). An arbitrarily high GPA requirement for declaring a major or for internal transfers is a barrier to entry. Not only does it favor students with prior computing experience – thus skewing toward the same demographics observed in high school CS classes – but it can also lead to a level of competition among students that represents its own barrier to entry. Anecdotally, in our meetings with women students at universities with a GPA requirement for internal transfers, the women cited this as something that dissuaded their friends from even trying the first course in the introductory sequence.

Undeclared majors are unable to enroll in classes. At some universities, priority during class registration is given to students who were admitted to computing majors directly from high school or community college. Because students from groups historically underrepresented in computing often discover computing when they get to college, this places a barrier to internal transfers.

C. Where does the major sit?

The location of a computing major within a university varies; computing may be its own college or be a major within the college of engineering, natural sciences, or arts and sciences. Schools differ in whether all majors must reside in the academic unit that houses the computing faculty, or whether there are multiple pathways (e.g., many schools might have a B.S. in CS in engineering and a B.A. in CS in the humanities). When a major resides solely in engineering, we have identified two additional institutional barriers: 1) engineering programs currently graduate only 22% women

[4]Responsibility centered management (RCM) means that the unit's budget directly follows enrollment; i.e., in a given year, the school/college/department, receives all or a part of their budget directly tied to the revenues from enrollment for that year.

[5]*CS+X* refers to majors for which students take classes from CS and from major X, but are not required to double major. At Northeastern, these are called *Combined Majors*. Additionally, there is often one or more capstone courses that intergrate the two fields.

[10], which limits the number of women they can attract to computing, and 2) because of the additional math and science requirements in engineering (as part of ABET accreditation),[6] transferring to a computing major housed within the college of engineering may add significantly to the time to complete the degree, further reducing the attractiveness of the major.

D. Unsynced Introductory Courses

Schools vary in whether courses in the introductory sequence that have multiple sections are *synced*. At one end of the spectrum, all sections have identical assignments and exams. At the other end, instructors are given complete freedom with few guidelines other than outcomes such as understanding "while loops". For schools that are completely unsynced, we have observed that the DFW rates and performance upstream can depend more on who a student has as an instructor rather than the content of the coursework. Whether a school can sync depends on the appetite and willingness for instructors on the introductory course sequence to give up "academic freedom," as well as on structural issues in the university such as whether all students can take an exam at the exact same time.[7] These are issues where the positional authority of the project team and the buy-in from leadership become particularly important.

E. Buy-in from Leadership and Faculty

Buy-in and commitment from institutional leadership and departmental/college faculty are critical to achieving systemic change. To even collect the demographic data on student retention needed to accurately diagnose where the primary barriers are for women typically requires influence from a chair or dean. If institutional leadership does not take ownership of their department's role in the pipeline to a computing major, meaningful improvements to representation will be challenging. Additionally, the instructors who teach the introductory sequence must believe in the importance of BPC and do the necessary work required to implement best practices.

V. Retention versus Attraction

A key goal of the site visit is to understand whether the university has a retention problem, an attraction problem, or a combination of the two. A retention problem exists when one or more demographic groups are observed to be leaving the major at higher rates than the majority demographic group. Retention issues can be found in the data that the school collects on who is leaving the major, the DFW rates of courses, and/or persistence of each demographic group in continuing to the next class of the introductory course sequence. An attraction problem exists when students from other majors don't even want to try a computing course. What we have observed is that schools that have a retention problem almost always have an attraction problem as well. However, schools that do not have a retention problem can have an attraction problem (often caused by a high required GPA to major, or inability to enroll in the introductory sequence classes as mentioned in the previous section).

[6] https://www.abet.org/accreditation/

[7] To prevent cheating, offering the same exam across multiple sections requires a single exam time.

VI. Conclusion

As we continue to learn more about the best ways to ensure that undergraduate computing is inclusive, it is essential that universities have the capacity and the commitment to understanding their data and diagnosing their specific challenges. Furthermore, universities must have the ability to welcome students regardless of their experience prior to entering college. Too often "the pipeline" is used as an excuse for inaction or a place to lay blame for the inadequate representation of women at the undergraduate level. Indeed, during site visits we have consistently observed various stakeholders blame the pipeline from high school rather than look critically at the institutional barriers or pipeline issues within their own program. By accurately diagnosing the problems and properly aligning interventions to the problems, universities can institute meaningful cultural and pedagogical changes to improve the experience of students traditionally excluded from computing.

Acknowledgment

The CIC is grateful for the contributions of Megan Giordano and Jessica Traynor, as well as the expertise from our Advisory Council and Technical Advisors. We want to thank Tracy Camp and Jodi Tims for their comments.

References

[1] Barker, L. and Cohoon, J. (2009) *Key Practices for Retaining Undergraduates in Computing* National Center for Women and Information Technology. https://www.ncwit.org/reainundergrads

[2] Wittemyer, R., Nowski, T., Ellingrud, K., Conway, M. and Jalbert, C. (2018) *Rebooting Representation: Closing the Tech Gender Gap through Philanthropy and Corporate Social Responsibility*, rebootrepresentation.org, 2018.

[3] Babe-Vroman, M., Juniewicz, I., Lucarelli, B., Fox, N., Nguyen, T., and Tjang, A. (2017) "Exploring gender diversity in CS at a large public R1 research university," in *SIGCSE '17: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. pp. 51–56 ACM. https://doi.org/10.7282/t3-q228-jn34

[4] National Academies of Sciences, Engineering, and Medicine. (2017) *Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments*. The National Academies Press, Washington, DC.

[5] DuBow, W., Weidler-Lewis J., and Kaminsky, A. (2016) "Multiple factors converge to influence women's persistence in computing: A qualitative analysis of persisters and nonpersisters," in *2016 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, Atlanta, GA, 2016, pp. 1–7.

[6] Margolis, J., and Fisher, A. (2002) *Unlocking the Clubhouse: Women in Computing.* Cambridge: MIT Press.

[7] Margolis, J., Estrella, R., Goode, J., Jellison-Holme, J., and Nao, K. (2017). *Stuck in the Shallow End: Education, Race, and Computing (2nd ed.).* MIT Press: Cambridge, MA.

[8] Decker, A., Egert, C.,and Cascioli, E. (2020) "Cohorting incoming students in a CS1 course: Experiences and reflections from the first year of implementation," *Journal of Computing Sciences in Colleges*, 35 (8), pp. 186–197.

[9] Cohoon, J., and Luther, T. (2011) "Analysis of a CS1 approach for attracting Diverse and inexperienced students to computing majors," in *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*, March 2011, pp. 165–170, Dallas, TX.

[10] American Society for Engineering Education. (2020). *Engineering and Engineering Technology by the Numbers 2019.* Washington, DC.