OMSCS is a new type of program and has been very successful in many areas. The trick with OMSCS is, because of the scale involved, *everything* associated with the program has to be offered in a way that takes advantage of its size, through a system that becomes more effective the larger it grows.

There are a couple of common forms this has taken so far. One is **a shift from centralized operations to peer support structures**. For example, the program now provides many of its own TAs in classes like Knowledge-Based Artificial Intelligence (KBAI), Computational Complexity and Algorithms (CCA), and even Educational Technology (EdTech) itself – this is possible because a class of two to four hundred students is virtually guaranteed to have at least 10-20 very good ones who are very interested in the subject and can be brought on to TA the next semester. The other is **a shift from following traditions to making decisions based on data gathered from students**. For example, Dr. David Joyner's controlled experiments on grader efficacy (performed in conjunction with the summer offering of KBAI) attempt to isolate factors governing the quality and speed of feedback, so as to develop grading "best practices" which could be applied more generally in the program.

That said, there are some remaining challenges that have not yet been adequately addressed in the new format. Academic advising is one of these - in fact, as early as Spring 2014 Dr. Charles Isbell flagged it as one of the leading problems the program must seek to address[1]. The program has two advisors, responsible for approximately 1000-2000 students each; this is not a favorable ratio and in practice does not allow for individualized professional advising as one might enjoy in a smaller program.

Advising in OMSCS appears to have begun the transition towards peer-reliant, high-data solutions. Many OMSCS students currently read the unofficial course review spreadsheet and ask their peers or the Google+ group for advice on which courses they ought to take next. To a casual observer these solutions might seem to leverage the program's size fairly well, and indeed they are significantly more feasible than a traditional advising structure in the context of OMSCS. However, because a student is still limited to manually sifting through posts or reviews, the process becomes less efficient, not more, as the program scales up; this is costly to both advisors and advisees. Furthermore, the quality of advice a student receives varies quite a bit depending on when that student asks, who happens to read and respond, which

[1] E.g., slide 23 of http://www.ecedha.org/docs/default-source/2014-annual-conference/charles-lee-isbell-jr-.pdf?sfvrsn=0

details of academic background are included in the inquiry, and so forth.

For this reason, I argue that we still have further to go. I propose to build a web application that will *systematically* elicit the most relevant advising inputs from students, perform the (objectively verifiable) task of suggesting a sequence of courses that will satisfy a student's desired specialization, and provide a framework for identifying (more subjective) potential factors governing student success and adjusting course recommendations accordingly. This project, which I've dubbed omscs-advisor, will have four major components[2]:

- User profile & history: allows the student to input (at a minimum) their previously-completed courses, desired specialization, timetable for graduation, and study hours per week. Eventually, may serve as the basis for social features (sharing one's class schedule with others to plan study groups).
- Planner: considers the user's profile and suggests appropriate courses and enrollment dates; allows the user to save their plan and tweak it later, or start over from scratch if they want to make a major change (e.g., taking a term off or switching specializations).
- Course browser: allows the user to access additional course information that may help in evaluating how appropriate the planner's suggestions are for their unique interests and circumstances.
- Review system: allows the user to give feedback on courses they've completed, to benefit both the system (which will use some of this data, e.g. effort estimates, in recommendations) as well as other students (who will look at their predecessors' opinions when deciding whether to take a course).

The first three functions map to functions that are also performed by a traditional advisor – while not every advisor has a formal checklist for this process, all rely on this cycle of gathering student data, acquiring and providing additional information, and offering recommendations based on both. Many advisors also rely on student word-of-mouth to evaluate the difficulty or value of courses, but this process is usually very informal; the review system will draw its conceptual inspiration from the existing course review spreadsheet.[3]

[2] This breakdown is essentially paraphrased from an earlier work at https://github.com/Arkaaito/omscs-advisor.

[3] This one: https://docs.google.com/spreadsheets/d/1zJp3Bu2Rb4DRpHLYL2r4nZ210Y2u8eFY7xgQxO4mq7o/edit?usp=sharing.