The impact of MOOC methodology on the scalability, accessibility and development of HPC education and training

Julia Mullen
MIT Lincoln Laboratory
Lexington, MA
jsm@ll.mit.edu

Weronika Filinger
EPCC, The University of Edinburgh
Edinburgh, United Kingdom
w.filinger@epcc.ed.ac.uk

Lauren Milechin
Massachusetts Institute of Technology
Cambridge, MA
lauren.milechin@mit.edu

David Henty
EPCC, The University of Edinburgh
Edinburgh, United Kingdom
d.henty@epcc.ed.ac.uk

ABSTRACT

This work explores the applicability of Massively Open Online Courses (MOOCs) for scaling High Performance Computing (HPC) training and education. Most HPC centers recognize the need to provide their users with HPC training; however, the current educational structure and accessibility prevents many scientists and engineers who need HPC knowledge and skills from becoming HPC practitioners. To provide more accessible and scalable learning paths toward HPC expertise, the authors explore MOOCs and their related technologies and teaching approaches. In this paper the authors outline how MOOC courses differ from face-to-face training, video-capturing of live events, webinars, and other established teaching methods with respect to pedagogical design, development issues and deployment concerns. The work proceeds to explore two MOOC case studies, including the design decisions, pedagogy and delivery. The MOOC development methods discussed are universal and easily replicated by educators and trainers in any field; however, HPC has specific technical needs and concerns not encountered in other online courses. Strategies for addressing these HPC concerns are discussed throughout the work.

KEYWORDS

Supercomputing, HPC, MOOCs, HPC Education, HPC Training

1 INTRODUCTION

Traditionally, HPC concepts have been taught in formal academic settings as part of an undergraduate or graduate degree, or as highly condensed one-off on-site training courses lasting anywhere between a half-day to a few days. For students pursing research requiring significant computing resources, HPC education may be integrated into courses in a student's subject domain, e.g. computational engineering or physics; however, the number of institutions offering HPC coursework is low. The challenges associated with creating effective and personalized educational experiences for HPC fall into two primary areas: a limited pool of trainers leading to a limited number of workshop offerings over a calendar year and the diversity of the subdomains of interest within the HPC community.

Consider the diversity of subdomains. While there is some overlap of content and skills that HPC practitioners must learn, each subdomain within the HPC ecosystem has a different focus. Additionally, as research applications in medicine, social science and biology become more complex and require more extensive computing power, a new cohort of students with limited computer literacy are searching for a pathway to HPC expertise. This variety in focus leads to a vibrant community, but it requires that HPC educators create effective educational materials capable of speaking to a range of disciplines and computer literacy levels. Unlike past decades when HPC educators and trainers could assume that new graduates had basic computer literacy upon which to build HPC expertise, educators need to add basics to the training program, which in turn affects the number and type of workshops that can be held within a given year. Furthermore, for professionals in need of training, workshops appropriate to their learning needs may not be available in a time frame appropriate to their project and needs. To address these challenges and diversify, the community requires developing scalable, personalized and accessible training to provide multiple learning paths to build HPC expertise.

2 MOOC OVERVIEW

HPC has always been about scaling, developing new technologies, and pushing the boundaries of existing systems. One educational approach that combines these traits is Massively Open Online Courses, or MOOCs. At the end of 2017 there were 81 million MOOC learners spread across the major MOOC providers [14] and another 13 million across 1500 individual sites running their own Open edX platforms [13] in support of their own organizations [7]. Clearly MOOCs offer a means of reaching diverse communities beyond the traditional university cohort of the developed world. However, unlike standard university courses that are part of a program, MOOC courses are open to everyone, generally without pre-requisites, are asynchronous, to support "Just In Time" learning, and compete with work and life responsibilities for learner attention without the promise of a degree or credential.

These characteristics make MOOCs notably different from other more traditional teaching methods, and those differences require

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Copyright ©JOCSE, a supported publication of the Shodor Education Foundation Inc.

© 2018 Journal of Computational Science Education
https://doi.org/10.22369/issn.2153-4136/x/x/x

specific approaches in the creation and delivery of teaching materials. For example, the open nature of the courses and lack of pre-requisites means that educators need to predict and design for the diverse background knowledge and potential knowledge gaps of their learners. The absence of credentialing equates to a range of learner motivations for joining and completing courses. Adult learners, in particular, bring a broad range of experience to their learning, need to be self-directed and are drawn to learning experiences that address a given problem or question that they are working to understand [9]. Some MOOC learners may be interested in only the initial overview material in order to get a glimpse of a domain; others may want to work with a few units in order to answer their questions or developed deeper understanding of a specific problem, while others may desire a whole course. In this learning environment, a learner that fully is engaged with only a fraction of course material may deem it more useful than a learner who completed the whole course. It is possible that traditional university students have a similar response to full courses, but until recently, when MOOCs began "unpacking" courses into modules that allow learners to determine when and how they they engage with educational material, this aspect of learner behavior has not been studied. As part of the effort to design engaging learning environments, this aspect of learner behavior is getting more attention.

With respect to educational design, MOOC courses build on pedagogical research demonstrating the value of segmenting content into concise and easily digestible chunks and providing frequent assessments to build mastery learning [5]. The fragmentation of content allows students to access the material in asynchronous, interruptive and often non-linear ways. Each content step should be self-contained, to the extent possible, fit in the overall narrative and bring something new, such as content or perspective, to the narrative. By design, learners can easily navigate between content units, revisiting material as necessary, eliminating the need for repeated content. While each step needs to be informative, it doesn't have to be comprehensive. Assessment questions are generally auto-graded and interspersed between units of video and text content. The interleaving and auto-grading provides students with immediate feedback to quickly resolve misconceptions and reaffirm learning. Finally, to support social learning, most MOOC platforms include a discussion forum. Many individual courses provide synchronous times for course interaction via video-conference tools and some even support smaller local gathering in a study group format, complete with questions to prompt discussion.

To better understand how MOOCs can be used to scale HPC education we consider two efforts. Section 3 describes a MOOC that has been offered three times over the past few years while Section 4 focuses on a Small Private Online Course (SPOC) that is being converted to a MOOC. While the two studies use different MOOC platforms, the designs rely on similar pedagogy, and the challenges listed in Section 1 and Section 2 are common.

3 SUPERCOMPUTING MOOC

3.1 Design

A course called "Supercomputing" [3], was developed on behalf of the Partnership for Advanced Computing in Europe (PRACE) by EPCC at The University of Edinburgh in collaboration with SURFsara from the Netherlands. Similar to many HPC centers, EPCC, as a part of the University of Edinburgh, the UK national HPC service provider and one of the PRACE Advanced Training Centers, offers a series of highly successful workshops and short courses to teach parallel and distributed computing concepts to students, faculty and researchers across the UK. However, despite years of increasing the number of training opportunities, meeting the training demands of the user community remains difficult. Considering these restrictions and the existing gaps in the available HPC training materials, a decision was made to create a course that would serve as an introduction to supercomputing, answering the what, why and how questions for a target audience of newcomers up through beginners in the field. The goal was to create an interesting course with an accessible introduction for newcomers while also building firmer foundations and fill-in the missing links for those with some degree of supercomputing knowledge. Concurrently, PRACE was exploring the MOOC approach to increase training accessibility and funded this project as one of two pilot MOOCs ("Supercomputing" and "Managing Big Data with R and Hadoop" [1]). PRACE selected the FutureLearn [8] platform as the host for the two courses. This discussion focuses on the former course, "Supercomputing".

The design of the FutureLearn platform is based around a pedagogical concept of social learning and follows three basic principles:

- telling stories,
- provoking conversation and
- celebrating progress. [2]

The cohesive 'story-like' narration is maintained by segmenting the course content into individually themed weeks. The FutureLearn platform follows the pedagogy described in Section 2 so that the content for a week is further segmented into a relatively large number of bite-size steps and delivered using a variety of content delivery modes, e.g. a mixture of videos, articles, discussions, exercises, quizzes and tests. Furthermore, the learning material for a week supports the learning outcomes through a set of activities with clearly defined learning goals associated with each unit or step. To help learners track their progress, work toward their goals and stay motivated, the FutureLearn platform presents clear progress updates in the student dashboard.

One of the key differentiators of the FutureLearn Platform is the emphasis on learning through social interactions and discussion. To encourage discussion, each step (video, text, exercise) includes space for learners to post comments and most steps include explicit calls to action to encourage conversations among learners and with educators. These discussion components allow learners to exchange opinions and verify their understanding of the covered material by providing learners the opportunity to reflect on their learning and share their insights with others. The importance and value of learners' contributions cannot be over-emphasized as they provide perspectives and motivation for other learners, prompting other students to join the discussion and share their own opinions on each topic. Equally, if not more important, are contributions from the educators. Once a course has begun, the instructor roles include replying to comments and questions and prompting discussions that share experiences, perspective and knowledge among the entire

cohort. The impact of this design element in the "Supercomputing" MOOC is presented in Section 3.3.

3.2 Conversion Challenges

Conversion from traditional course designs to a MOOC model generally requires a major effort to segment the material into bite-size steps that when combined provide a clear narrative. Similarly, the structure of the material typically covered during a two-day course was not easily aligned with any MOOC model. One key takeaway from this re-design effort was the amount of content from previous courses and workshops that could be trimmed because it didn't add new insight or deeper exploration into the concepts being presented within a section. Significant attention is paid to the idea that variety in delivery modes makes the content more accessible and easier for learners to absorb, but another takeaway is recognizing that content shouldn't be made to fit a specific delivery type but rather be presented in the simplest way that convenes the greatest meaning. In reaching a balance, experience suggests variety is good but simplicity is even better.

3.3 Results and Lessons Learned

One of the benefits of the segmented model used by MOOC platforms is that small content chunks are easy to modify and new content can be inserted into the narrative seamlessly. This course used this feature, and after each run of the course a number of small changes were made. Most were motivated by either the need to clarify or explain specific topics, or in response to suggestions made by learners or fellow educators. For example, we were surprised by the number of learners who confused supercomputing with Artificial Intelligence, AI. This lead to a new step titled "What supercomputing is not" introduced early in the course to clarify this misconception. Similarly, between runs of the course additional formative exercises, auxiliary content and "useful links", a collection of all the links referenced in discussions, were added to provide deeper experiences for the learners.

The discussions in which educators participate tend to be more animated than others and give learners certainty of being on-track with their understanding of the subject. The instructor team was pleasantly surprised by how well the discussions played out during the various course runs. The diverse background and different levels of familiarity with the topic led to many thought-provoking conversations. One compelling observation was that steps not containing any explicit questions or discussion topics had a much smaller number of comments. Figure 1 shows a number of discussion comments from the first run of the course for each step within week 3. The steps 3.1, 3.10 and 3.23, shown along the x axis in Figure 1 did not include any calls to action, which directly affected the number of learners who engaged in discussions. Including additional discussion prompts was one of the first modifications made between the first and second run of the course.

Table 1 presents the enrollment and engagement statistics collected by FutureLearn for the three runs of the Supercomputing MOOC. The term joiners is used to describe anyone who signed up for the course, learners are those people who actually accessed the content, active learners are those who track their progress through the content (i.e. mark steps are completed) and finally, social learners are those who are active in comment sections. Although, the numbers from the first two runs were relatively low by the FutureLearn and MOOC standards, they were not a source of concern for the educator team because the classes were significantly larger than a similar workshop. Additionally, lower numbers facilitated direct engagement between educators and students, increasing the opportunity for social learning. The third run showed that reaching out and attracting the right audience was one of the biggest challenges of the course. The enrollment numbers a week prior to the start of the fourth run, were even smaller. It seems that the active FutureLearn participants interested in the course have already taken it and advertising the conceptual no-programming introductory course within the HPC community has not brought the desired effects.

From the very beginning the number of students participating in the Supercomputing course was low by MOOC standards. However, it never was the primary indicator of course success. It has been shown that on average only about 5 percent of joiners actually complete massive open online courses [10]. Most of the time this is not a reflection of the course quality or learners' satisfaction with it. Due to the nature of adult online learning, taking place after hours and competing with life obligations, and the wide spectrum of learner motivations, it is hard to evaluate the success of an online course offering. Just because a learner did not go through the entire course does not mean they did not meet their learning objective, find the information they needed or fully engage with the content they completed. Also, not every learner likes or feels the need to engage socially so the number of comments is not a clear indicator of success. Although MOOC platforms collect data on student performance, engagement and demographics, this is only part of the impact story. Being able to reach learners from over 140 countries is a fantastic achievement in itself and being able to hear or meet someone who was inspired by it is even better. One success story was a Nepalese woman whose participation in the Supercomputing course motivated her to attend ISC'18.

4 UNDERSTANDING HPC WORKFLOWS AND HOW TO EXPLOIT THEM

4.1 Design

The Lincoln Laboratory Supercomputing Center (LLSC) Team pioneered an on-boarding process for professional engineers and scientists that flattens the HPC learning curve. As part of this on-boarding process, a Computational Science and Engineering (CSE) Team member meets with each new user to provide a targeted introduction to HPC and discussion of strategies for scaling the user's workflow. While this form of consulting is highly effective, it is not scalable, especially as support needs expand to include members of the MIT campus community. To scale the on-boarding process, the team leveraged the Open edX platform for the delivery of a MOOC course. The Open edX platform was selected because as an open source product it affords

- the ability to modify and extend the platform to provide tools that support the our teaching
- access to all of the student data which is used for
  - continuous course improvements

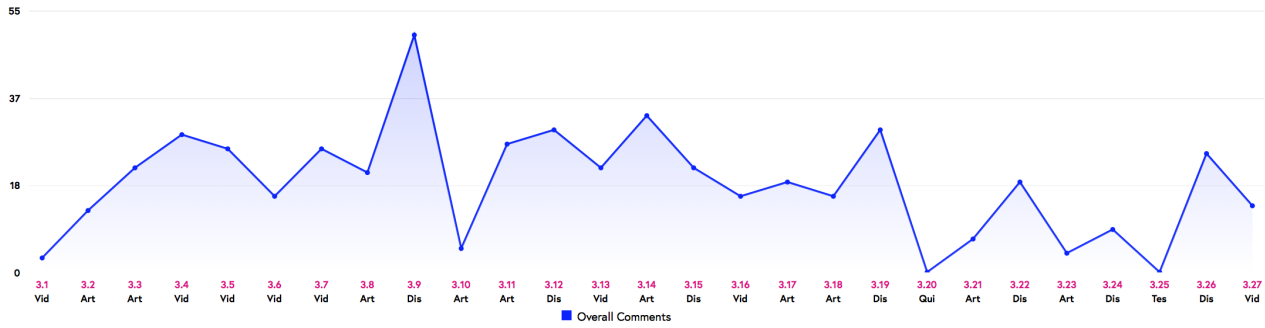

Figure 1: Comments for steps within Week 3 of the first run of Supercomputing.

Table 1: Enrollment and engagement statistics for the first three instances of the Supercomputing MOOC. The percentage of leaners is calculated using joiners as a basis, active learners using learners as a basis, and social learners and step completion using active learners.

Run	Joiners	Learners	Active Learners	Social Learners	Learners completing ≥ 50%	Learners completing ≥ 90%
1 (6 Mar 2017)	3,263	1,722 (52.8 %)	1,138 (66.1 %)	379 (22 %)	337 (19.6 %)	211 (12.3 %)
2 (28 Aug 2017)	3,100	1,910 (61.6 %)	1,285 (67.3 %)	403 (21.1%)	362 (19 %)	186 (9.7 %)
3 (15 Jan 2018)	1,825	1,218 (66.7 %)	761 (62.5 %)	219 (18 %)	207 (17 %)	115 (9.4 %)

– learning analytics

The initial pilot included in the on-boarding process was intended to teach users how to map their serial or small scale application onto a set of common HPC workflows, e.g. high throughput, MapReduce, Leader-Worker and full MPI based parallel applications. Once students understand where their applications fit in a set of canonical workflows, they can focus on techniques to effectively refactor or extend their application to make it suitable for an HPC system [11, 12].

Within the user population the domains of interest ranged from those traditionally associated with computing and HPC, e.g. physics, engineering, math and computer science, to more recent adopters in the biological and social sciences. One consequence of this domain range is a significant range of computational experience, from researchers with minimal experience developing computational workflows to researchers running commercial parallel codes to those comfortable developing new parallel solutions. Reviewing the on-boarding requirements for these professionals and campus researchers, the instructor team recognized the common concerns of both populations, namely:

- deep domain knowledge but lack of HPC specific training
- computational experience ranging from novice to advanced
- need for strategies rather than HPC tools

The first design challenge was to create a course that addressed the three concerns listed above by providing enough theory and practice to enable users to accomplish their HPC goals: faster time to solution and more research turns in a day. Considering the range of computational experience, the expectation was that advanced users needed a simple refresher on how to use the Laboratory HPC system combined with an understanding of updated best practices and techniques. Novice users needed a bootcamp whose learning outcomes included understanding their application type in the context of HPC and how to effectively use the HPC system for their application. Intermediate users fell between these two, and their needs, while more fluid, were covered by the material created for the other two user cohorts. The design challenge centered on creating multiple learning paths through the course, including sufficient content and guidance so that each user could "build their own adventure".

The second design challenge was mapping a decade of experience providing individual consultation to users, tutorials on pMatlab at numerous conferences, and presentations about interactive supercomputing, to the Open edX platform. The pedagogy of the Open edX platform centers on mastery learning through the joining of theory and practice. The theory is provided in small content chunks followed by practice in the form of quizzes, problems, essays or discussions. Aligning the existing material to address both design challenges resulted in a complete redesign of existing tutorials and presentations. The refactoring was accomplished through the use of the Cmap [?] concept mapping tool, to create a map of the concepts necessary to understanding and executing successful HPC strategies for a given application type. The concept map for the full course is shown in Figure 2. Concept maps, also known as knowledge graphs, are driven by a major question, and the concepts required to answer the question form a hierarchy of material that a student needs to understand in order to answer the major question. The major question for the pilot Introduction to HPC course was

"How do I convert my application to an HPC workflow and exploit concurrency?" The map in Figure 2 was designed to include all of the concepts that a novice user would need to understand in order to answer the question. The map is developed by considering questions that follow from the major question, e.g. "What is Scientific Computing?", "What is HPC?", "How do I get started using the supercomputing system?" and "How do I parallelize my application code?". Each of these questions corresponds to a topic area in a more traditional syllabus and yields additional concepts as illustrated by the hierarchy of concepts associated with "What is High Performance Scientific Computing?" The authors note that each of the other major topics has a similar set of hierarchical concepts that have been minimized in Figure 2 to provide clarity.

The creation of concept maps to visualize related topics offers advantages not seen in traditional linear syllabi

- links between concepts highlight interconnections that novices often miss but are usually important to the discipline
- concepts that do not link to other concepts highlight material that is not necessary for understanding
- the knowledge graph provides a starting point for crafting personalized learning paths for students
- the creation of the concept map automatically segments the material into bite-sized chunks

For the instructor team, the creation of the concept map for the HPC course addressed both the second design challenge, segmenting existing material in a coherent manner and the first design challenge, building a course appropriate to a range of experience levels and goals. As an example, consider the educational needs for a user with HPC experience who has not used the supercomputing system in years and brings an application that is new to him or her. This student needs to learn the strategies associated with parallelizing and running the new application on the HPC System. Reviewing the full concept map, it is clear that this student only needs two topic areas as illustrated in Figure 3.

4.2 Lessons Learned

Among the key benefits of MOOCs are

- the ability to track student activities
- insight into problem areas of the course
- ease of updating, modifying and extending content

These benefits allow instructor teams to better understand levels of student engagement with the material and to gain insight into potential content gaps or problems that need clarification. The combination of the platform design and the segmented content make it easy to close content gaps by adding and extending material. With an eye toward redesigning the pilot SPOC and creating a MOOC, there are three primary lessons that are being used to inform the new design; tracking student learning paths, modifying hands-on exercises to support open courses and building exercises for a student population that prefers web-based portals.

While designing a course where students can build their own path has great educational value, the basic premise renders it difficult to apply traditional academic metrics when attempting to determine the success of the course. By design each student should touch on the material that is appropriate to their situation. Even novice students aren't encouraged to consider all of the use cases, but rather to focus on the use case that best fits the application they are working with at the present time. The result is limited data on completion rates leaving the team to develop other approaches to evaluating success and recognizing gaps. Based on anecdotal data, the sections providing an overview of scientific computing, HPC and interactive computing are successful but student's understanding of the role and importance of the scheduler are less well understood. Students recognize the components of the supercomputing system but seem to have difficulty recognizing how different they are from the system on their desks, or why these differences are important. Finally, crafting personalized learning paths within a course offering is not fully supported within the standard Open edX platform. The pilot incorporated hands-on experience through programming assignments that could be developed and run on the LLSC and MIT supercomputing systems. While this is possible with a SPOC, providing access to supercomputing resources renders hands-on HPC experience difficult to include at the MOOC scale. Additionally, the hands-on components included instruction on how to run the applications on the local systems, using specific scheduler, file system and user account specifications. While videos demonstrating correct responses to job submissions and explanations of scheduler behavior are helpful for students, example application snippets that are too tightly coupled to a given system not only suffer from lack of portability but they quickly become outdated.

Finally, while the LLSC and Supercloud have pioneered interactive supercomputing and emphasize alternatives to batch processing, new users have a distinct preference for web-based portals to compute systems and application codes. The trend towards web-based access to systems is so ubiquitous that a Jupyter Notebook viewer and grader have been built for the Open edX platform. [4] As these students join the HPC environment, the LLSC and Supercloud CSE Teams have begun leveraging Jupyter Notebooks for both teaching and application development and recognize the need to integrate the notebooks into the next generation of the HPC MOOC.

4.3 New Design

Building on the lessons learned and the changes in student preparation described in Section 4.2, the MIT Supercloud HPC SPOC is being redesigned to separate the understanding of HPC from the job submission and monitoring concerns. Additionally, because the current state of the Open edX platform does not fully support personalized learning, the new design focuses on a set of "short courses". Short courses have the added benefit that they more closely align with the needs and time constraints of adult learners, described in Section 2, by providing the student with the material they need in a small bundle. Furthermore, the development of short courses follows a trend within edX, where there has been increased focus on micro-masters programs and an increase in the number of courses meant to run for three to four weeks rather than the ten to twelve weeks of earlier offerings [6]. The MIT Supercloud team split the course into two short courses "Understanding HPC Workflows and How to Exploit Them" and "Using the MIT Supercloud". The former covers the introduction to HPC concepts, and overview of the canonical workflows and short hands-on examples to explore

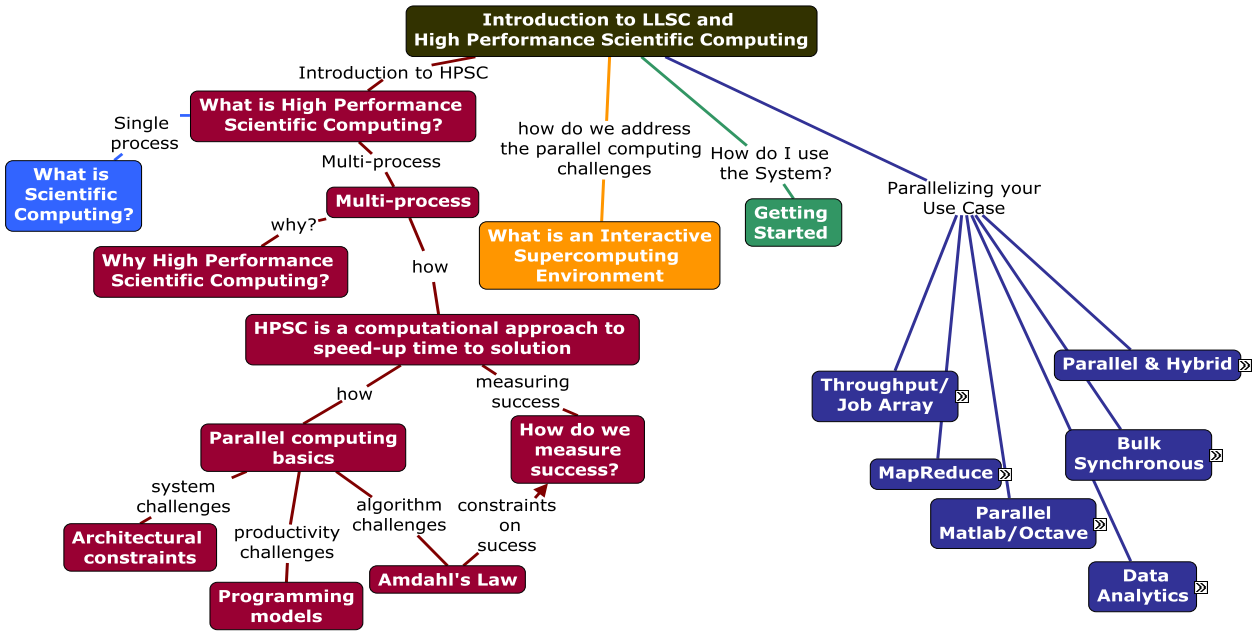

Figure 2: Concept Map for Introduction to HPC Pilot MOOC.

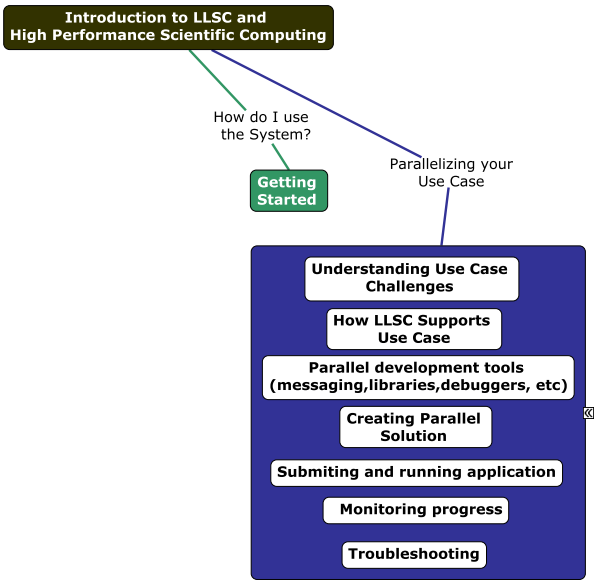

Figure 3: Concept Map for Expert Student in Introduction to HPC Pilot MOOC.

HPC workflows. The hands-ons examples include thought exercises and small programming exercises contained in Jupyter Notebooks that can be run on any system with multiple cores. The latter mini-course, "Using the MIT Supercloud" provides training on how to use the HPC system, including how to launch and monitor jobs, and Best Practices for using each system. The hands-on exercises using the MIT Supercloud provide students a chance to develop their competencies. This modular development maximizes re-use for both the LLSC team and the larger HPC education and training community.

5 CONCLUSIONS

It is clear that though MOOCs provide a means for scaling and expanding the accessibility of HPC education and training, there are significant challenges that need to be addressed. MOOCs cannot be viewed as an inexpensive alternative to face-to-face training, because of the time and effort required to develop and facilitate a course. There are best practices for segmenting content into small concept sized chunks which can reduce the effort and the long range benefit is a collection of small teaching units that can be recombined and reused for a range in a range of educational contexts..

In addition, HPC has unique requirements for access to parallel and distributed systems for courses where hands-on activities are to be included. The case studies presented here addressed this challenge by re-factoring the hands-on components to engage learners in thought experiments. In each case the expectation is that the student will be better prepared to utilize site and system specific training to develop and execute their applications. The benefit of creating a general course or series of mini-courses is that the material is accessible to a wider audience, e.g. a group of researchers who are beginner programmers, pre-university students, teachers and corporate and government decision makers. An additional benefit of segmenting the theory and thought exercises from the system details is that the resulting course components are general and can easily be reused by other centers, universities and laboratories. For example, new learners could take the Supercomputing MOOC described in Section 3 and follow that up with the MOOC described in Section 4 so that when they have access to an HPC system they are familiar with the basics of supercomputing, understand where their application fits in the larger HPC application landscape and know what questions to ask in order to get an efficient solution.

A REPRODUCIBILTY

This paper has examined methods of designing and developing MOOC courses for HPC education and training. All of these methods are reproducible by using standard instructional design methods for designing courseware and can easily be implemented on any online course platform. The key discussion here is re-use and each of the MOOC courses is modular such that it can be re-used either in full or part.

ACKNOWLEDGMENTS

The authors acknowledge funding from PRACE via the EU's Horizon 2020 Research and Innovation Programme (2014-2020) under grant agreements 653838 and 730913 for the Supercomputing MOOC. The authors would also like to thank the members of the MIT Lincoln Laboratory Supercomputing Center and MIT Supercloud Teams who support the deployment of a unified educational environment, including an Open edX instance and the supercomputing system used for HPC practice for the MIT MOOC.

REFERENCES

[1] [n. d.]. Managing Big Data with R and Hadoop. Retrieved August 2, 2018 from https://www.futurelearn.com/courses/big-data-r-hadoop
[2] [n. d.]. The Pedagogy of FutureLearn, How our learners learn. Retrieved September 22, 2018 from https://ugc-about.futurelearn.com/wp-content/uploads/FL-pedagogy-RGB.pdf
[3] [n. d.]. Supercomputing. Retrieved August 2, 2018 from https://www.futurelearn.com/courses/supercomputing
[4] Lorena Barba and Miguel Amigot. 2018. Jupyter-based courses in Open edX - Authoring and grading with notebooks. Video. Retrieved August 3, 2018 from https://www.youtube.com/watch?v=Qml1x_SbVzc
[5] R.C. Clark and R. Mayer. 2011. *E-Learning and the Science of Instruction, Third Edition.* Pfeiffer, A Wiley Imprint.
[6] edX [n. d.]. Retrieved September 27, 2018 from edx.org/course
[7] Open edX. 2018. Open edX Conference 2018 The State of Open edX. Video. Retrieved August 3, 2018 from https://www.youtube.com/watch?v=HyK2YoKKaGQ
[8] FutureLearn [n. d.]. Retrieved August 3, 2018 from https://www.futurelearn.com/
[9] M.S. Knowles. 1980. *The Modern Practice of Adult Education, From Pedagogy to Andragogy, 2nd Ed.* Cambridge Books.
[10] Daphne Koller, Andrew Ng, and Zhenghao Chen. 2013. Rentention and Intention in Massive Open Online Courses: In Depth. Retrieved September 27, 2018 from https://er.educause.edu/articles/2013/6/retention-and-intention-in-massive-open-online-courses-in-depth
[11] J. Mullen, Chansup Byun, Vijay Gadepally, Siddharth Samsi, Albert Reuther, and Jeremy Kepner. 2017. Learning by Doing, High Performance Computing education in the MOOC era. *J. Parallel and Distrib. Comput.* 105 (July 2017), 105–115. https://doi.org/10.1016/j.jpdc.2017.01.015
[12] J. Mullen and et. al. 2016. Designing a New High Performance Computing Education Strategy for Professional Scientists and Engineers. http://ieee-hpec.org/2016/techprog2016/index_htm_files/R-Mullen-HPEC-16.pdf
[13] Open edX [n. d.]. Retrieved August 2, 2018 from openedx.org
[14] Dhawal Shah. 2018. By The Numbers: MOOCs in 2017. Retrieved August 2, 2018 from https://www.class-central.com/report/mooc-stats-2017/