Teaching, Learning and Collaborating through Cloud Computing Online Classes

Judy Qiu*, Supun Kamburugamuve*, Hyungro Lee*, Jerome Mitchell*,
Rebecca Caldwell†, Gina Bullock‡, Linda Hayden§
*School of Informatics, Computing, and Engineering, Indiana University Bloomignton
{xqiu, skamburu, lee212, jeromitc}@indiana.edu
†Winston-Salem State University
caldwellr@wssu.edu
‡North Carolina Agricultural and Technical State University
glbulloc@ncat.edu
§Elizabeth City State University
haydenl@mindspring.com

***Abstract*—A knowledge of parallel and distributed computing is important for students needing to address big data problems in later jobs in industry or academia. However, many campuses do not offer courses in these important areas due to curriculum limitations, a lack of faculty expertise, and limited instructional computing resources. MOOCs and Clouds provide an opportunity to scale learning environments and to help institutions needing an advanced curriculum. In this paper, we discuss a course offered at Indiana University and use it as a model for improving curriculum at institutions who cannot easily provide the needed courses themselves.**

Keywords—Online Education, Cloud Computing, Parallel and Distributed Computing.

I. Introduction

Parallel and distributed computing is becoming ever more important with the exponential growth of data production in areas such as the web and Internet of Things. Further modern computers are equipped with multiple processors that need to be utilized efficiently. On the other hand, clouds are becoming the standard computing platform for running both applications as well as data analytics. With these trends it becomes increasingly important for the next generation of software engineers and researchers to be familiar with distributed and cloud computing paradigms and how they can be applied in practice and often in parallel fashion. Unlike academia where one focuses on the fundamental computer science problems, cloud computing involves many technologies and software tools that are widely used by industry and academia for real-world applications that are now part of everyday life for billions of people. These include Internet-scale web search, e-mail, online commerce, social networks, geo-location and map services, photo sharing, automated natural language translation, document preparation and collaboration, media distribution, teleconferencing and online gaming. However the underlying fundamentals of these techniques are coming from different computer science disciplines including distributed and parallel computing, databases and computer systems architecture. A well-rounded course of cloud computing should cover each of these areas and explain them in the context of cloud computing. To gain practical experience on cloud computing, a student has to master many different technologies that are based on these principles.

In order to facilitate such a learning environment, Indiana University developed the Cloud Computing online course [1]. This course has been taught by faculty for several years both in-class for residential students and for online students. The course is offered as part of the curriculum for Computer Science graduate program at Indiana University and students from the Data Science graduate program. Intelligent System Engineering and Library Science are also given the opportunity to take the course. The population of online students is geographically located world-wide, from London, France, Germany, India to Indianapolis. Most of the students are professionals who take online classes to update their knowledge and skills or earn a degree.

A primary goal of the course is to maintain the same standard as the residential course for the online course. Since this is a programming intensive systems course, it is especially challenging due to limitations on the face to face interactions with the online students, diverse background of students and the deep technical knowledge required by the course. The students are expected to have general programming experience with Linux and proficiency in the Java (2-3 years) programming language and scripting. A background in parallel and cluster computing is considered a plus but not required. The statistics present in this paper are related to the latest version of the online course which saw the largest attendance so far with about 160 students, where 100 were residential students and the rest were online students. The popularity of the cloud computing topic follows from major corporations including Microsoft, Amazon, Google, IBM, Facebook and Twitter, which provide infrastructure, tools or applications in Clouds. Business, government, academia and individuals use public or private cloud-based solutions for storage and applications.

The course has been taken as a model by other institutions to introduce cloud computing to their own students. This is facilitated by the availability of online course materials. This

[1] http://cloudmooc.appspot.com/preview

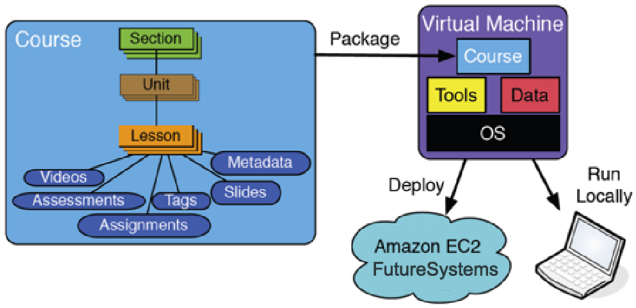

Fig. 1. Model for the MOOC Course Content and Delivery using Cloud

provides a unique opportunity for collaboration between Elizabeth City State University (ECSU) and Indiana University (IU) in remote sensing of the environment using Cloud Computing technology and involve faculty and students from Minority Serving Institutions (MSI) by exploiting enhancements using Cloud Computing technology. Computational Science and Data Science are important areas that have the capability to host both parallel computation (using MPI and Hadoop) and learning resources (online MOOC), making it an attractive focus for universities without a major research history looking to participate on an equal footing with research intense universities.

The rest of paper is organized with section II curriculum development and course organization, followed in section III course scaling and techniques, in section IV evaluations of the course outcome and knowledge growth for students, and section V ADMI Cloud for scaling the model to MSI institutions. Finally, in section VI we summarize the challenges, impact and future work in modernizing curriculum and workforce development.

II. Curriculum Development and Course Organization

The course is aimed at teaching the basic principles of parallel and distributed computing and explore the application of these in practice in cloud environments. This is a graduate level course with large emphasis on programming and expects prior knowledge of programming in order to be successful. The course follows the cloud computing text book [1]. By the end of this course, students are expected to learn key concepts in cloud computing and have enough hands-on programming to be able to solve data analysis problems on their own. The organization of the course is shown in Fig. 1.

A. Course Content

The course uses the Google Course builder as the content hosting platform. Google Course builder provides an easy way to host course content. Its source code is distributed under the Apache License version 2 and is free to modify and redistribute. An individual instructor can develop a course quickly with the features provided by this out of the box software. Since Course builder is open source, an instructor can modify the source code to create a more personalized version of the course. The final completed course should be deployed in Google infrastructure using the Google App Engine.

The course content is mainly lecture videos hosted in YouTube. Text version of the content is also possible. The course has been structured as a set of units. Each unit contains a set of lessons. Each lesson is a video plus some text description. Each lesson can be followed by a simple activity. The instructor creates an activity as a JavaScript file. The activity contains simple multiple-choice questions or text based answer questions with specific answers. Between units there can be course assessments. These assessments can be quizzes, midterm exam and final exam. They also have the same format as activities followed by lessons and features multiple-choice questions and simple text based answer questions. The activities and assessments can be graded and the scores are displayed in the student profile.

The course consists of six units starting with cloud computing fundamentals and then move on to infrastructure as a service (IaaS), Platform as a service (PaaS) and cloud data storage and Internet of Things applications. Each unit consists of multiple lectures with videos.

The videos were recorded by the instructor with the help of a professional staff for video recording and editing. It took a lot of effort and time to get the videos properly recorded in the first time of offering the course. After the initial videos were finalized it was relatively easy to add more content or update the videos for later offerings of the course.

B. Projects

The course was offered with a comprehensive set of cloud application projects that are interlinked together. The overall goal is to build a web search engine from scratch. Students can use various tools and build the system one component at a time using cloud based data analytic platforms. The projects use Hadoop [2], HDFS [3] and HBase [4] as main technologies. The data set used by the projects is ClueWeb09 [5] available for educational purposes. We only use a moderate size data set from the original because of the resource constrains.

The projects are packaged into a virtual machine and a student can download this to run the projects on their home machine or on a cloud provider if they choose to do so. The course expects students to run the projects on their own local machines at the start and then move to production distributed environments. Each project is accompanied by a video that explains the project in detail and show some of the steps required to build and run the project.

The projects starts with a small activity that involves configuring Hadoop and running a simple Hadoop program. The first building block of the search engine expects students to write a pagerank [6] algorithm in Hadoop to assign an importance to web pages. Next the HBase distributed storage is introduced to the students and the course expects them to write a program to load the data into HBase as well as create an inverted index from word to page to facilitate the search. The next step is to combine the results from pagerank and use the inverted index to do actual searches. Apart from these projects the students are expected to implement a standard machine learning algorithm using the Harp [7] machine learning platform developed at Indiana University. These projects are aimed at teaching students about complex data analytics

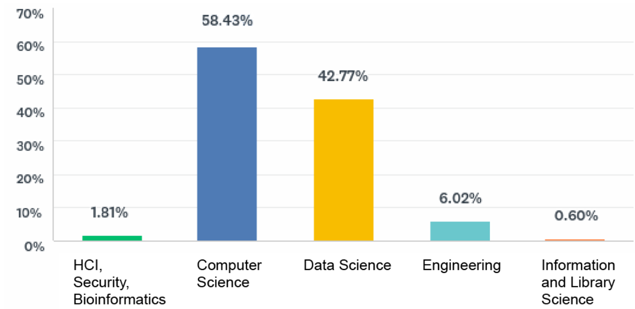

Fig. 2. Departments where the course is cross-listed among five different programs: Informatics, Computer Science, Data Science, Intelligent Systems Engineering, Information and Library Science

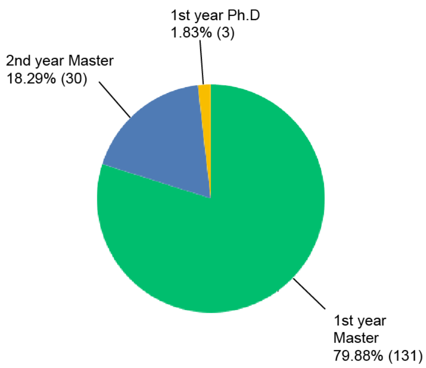

Fig. 3. Student's Level: 81% students in their first year, 19% students in their second year

and how to use parallel processing to speed up a sequential algorithm.

C. Assignments & Exams

Assignments were mainly focused on testing the basic knowledge about the subject matter. Most questions are selected from the text book. Assignments were given weekly or bi-weekly and had a turnaround time of week.

D. Student Evaluations

Students are evaluated based on their performance on eight programming projects, written assignments and two exams. The exams are focused on the core concepts of cloud computing and related underlying principles. For online students, the exams are conducted using the Canvas platform and the Adobe connect video conferencing.

III. Course Scaling and Techniques

A. Audience and Diverse Background

The course was targeted towards a wide audience coming from different backgrounds. As shown in Fig. 2, we found that the student distribution ranged from Informatics, Computer Science, Data Science, Engineering, Information and Library Science, to Industry with diverse knowledge and background about the subject matter and in general of the field.

Fig. 4. Students' Interests about the course in Cloud Word View

At the beginning of the class, we performed a survey about the course to understand students' background and expectations. The course is offered to the five different programs and therefore collecting survey data is necessary to estimate students' level and preparation for the class. Figures 2, 3, and 4 show that this course needs to explore several Hadoop-oriented technologies in dealing with Big Data on Cloud Computing. Although prior knowledge of the field is desirable, most students expressed their lack of experiences on these new technologies as they are in the first year of graduate study. We also observed that students eager to learn a wide range of knowledge and experiences about parallel computing with particular software such as Apache Hive, Spark, Pig and Lucene being of interest.

B. Forums

Since the course is offered for a large number of online and residential students from different time zones and different professions, providing interactive support of course materials, especially about hands-on projects with code implementations is one of the challenging tasks for instructors.

We experimented with several options for class forums which is a vital part of the course. Because the large class size, an instructor is not always possible to solve problems encountered by individual student in person. In previous years the course was run with Google Forums[2], Indiana University internal forums and Piazza[3] forums and we found Piazza to be the best option.

The web-based tool Piazza is mainly used for the communication between instructors and students and among students and our statistics indicate that 84% of questions received responses within 61 minutes in average. Fig. 5 shows an overall activities on Piazza in enabling online collaboration of the class.

C. Hands-on Labs

The course is organized with biweekly projects to encourage active developments in source code writing and connect between an literature in a textbook and the latest technologies. The fundamental pedagogy underlying these hands-on projects is to embrace new experiences in learning both theory and practice with minimal barriers, for example, learning a new programming language or preparing computing environments with recent software tools, which takes effort and time to

[2] https://groups.google.com
[3] https://piazza.com/

achieve. Fig. 8 gives an indication of students' programming ability associated with project developments of the class. Many students have at least 1 or 2 years language experiences among Java, C#, C++, C and Python which are abundant to start basic code developments in most assignments.

One of the challenging activities in teaching from previous classes is building a controlled experimental environment over different computing platforms. We built a virtual machine image to avoid such hassles and the choice of computing environments is given to students based on their confidence level. The VM image is able to run on the student's desktop via VirtualBox. The transition from using a desktop to run jobs on the cloud environment is a steep learning curve. The labs provide students with step-by-step instructions on how to install and configure the Hadoop cluster on OpenStack Kilo of FutureSystems cloud computing resources at Indiana University. Their applications can execute on production computer clusters for projects such as Hadoop PageRank, BLAST, WordCount, and Harp Mini Batch K-means.

D. Online Meetings

To facilitate the questions from students regarding both course content as well as projects, online meet ups were conducted every week. These were one hour sessions mainly steered by associate instructors but participated in by the instructor. In early course offerings, with only a small number of students the Google Hangouts platform was the choice for online meetings. But with larger classes we switched to Adobe Connect platform, provided by Indiana University for online courses. Every such online meeting is recorded and available through YouTube for later viewing by the students. We find that such videos are helpful in the subsequent runs of the course as well.

Video conferencing tool Adobe Connect (now replaced by Zoom) is also provided for weekly class lab session and office hours to instruct how to complete course assignments with step-by-step tutorials and provide individual feedback. Our experience indicates that these tools ensure effective learning of students and productive course management for instructors. Fig. 6 is a sample screenshot that we captured during the normal video sessions. The chat window at the bottom allows us to have a public and one-to-one conversation among participants and the main window flips to either a presentation mode or screen sharing for lectures and tutorials. Also recordings for these sessions have been made for self-study in case that students need to re-visit materials covered in those sessions.

E. Content Repository

One innovation of this project is to build on our extensive experience with online education and its technologies to use MOOCs technologies and build an open source community X-MOOC repository to explore a modular and customizable process for storing, managing, and sharing course content and learning materials. The developer of the course found the need to share content among different courses run by different instructors. In order to do so, a MOOC platform should be able to share course content among different courses. As part of this course, we have developed technology on top of edX

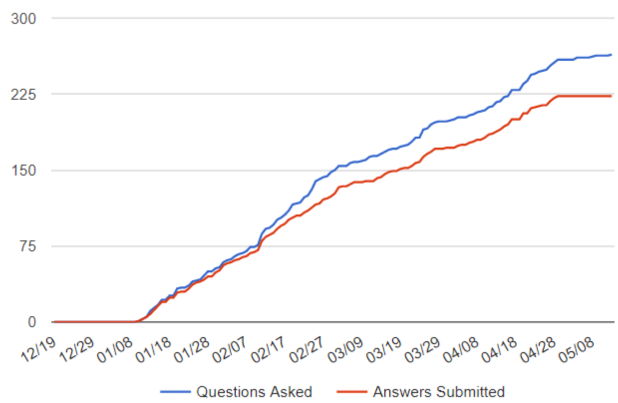

Fig. 5. Questions and responses received on Piazza over the semester

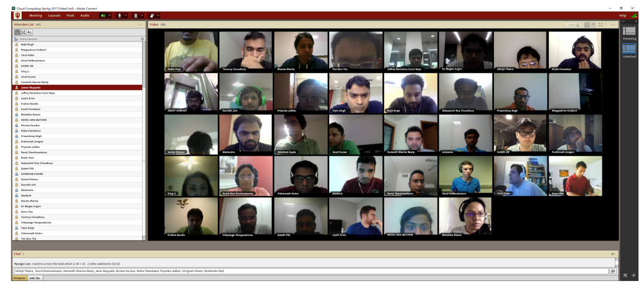

Fig. 6. Supporting remote students through video conferencing

MOOC platform to share content among different courses and we have moved the content of this course which is currently in Google Course Builder to the edX platform. This development will allow instructors from different universities to easily share course content and quickly create new courses that modify the old one.

Link function adds a link to related course videos in assessments. This function can navigate students through specific course content for review. It is particularly helpful when students make mistakes in a quiz or an exam but are unclear about the missing knowledge. The instructors can provide a list of keywords and their associated video tags for each question. If a student chooses an incorrect answer, these video links will automatically appear as hints (buttons) under the question.

We have created and implemented a *playlist* function on both edX and Google Course Builder. It allows customizable selection and arrangement of lessons according to preference. The UI provides a drag and drop function with easy interactions. Instructors can assemble a new course from modules in a shared repository on the fly. Students can use this function to navigate through the most important lessons in the course modules more efficiently.

Interoperability could include an automated process in sharing courses between edX and other course hosting platforms including Hubzero. This will allow the authors to freely move from one platform to another. The edX courses are already exported as XML content, so we need to find mechanisms to convert them to an accepted format for other online sites. By tagging those course modules with metadata, we can classify and organize shared course materials and make them easily searchable for others. We can further set up learning objectives and review, rate, and provide feedback on them to ensure a high quality online learning experience.

IV. Course Outcomes and Evaluations

In addition to standard institutional class evaluations, we conducted a post course survey to gather feedback on the details of course content and measure the growth of students in obtaining the knowledge and skills. For the cloud computing course we wanted to know the preference of students in using a VM instead of a distributed environment for the projects. As Fig 7 shown, the majority preferred the VM. Such a preference can be stemming from the fact that VM is easier to setup and run programs rather than using a distributed cluster.

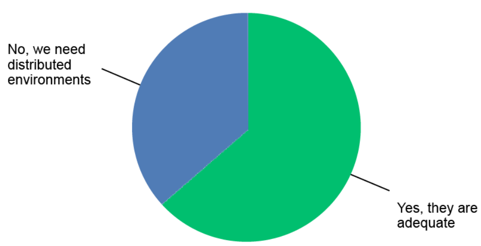

Fig. 7. Survey on single VM environment and preference

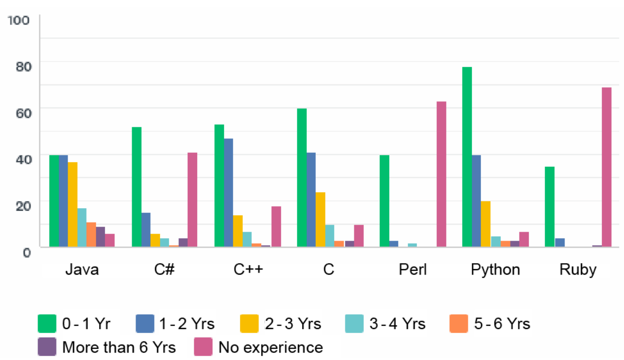

Fig. 8. Student Programming Language Experience. The y axis shows the number of student who are proficient in a given language and number of years.

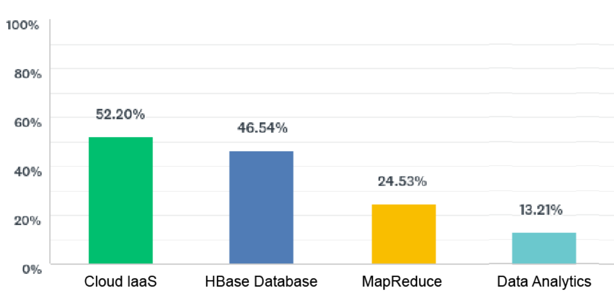

Fig. 9. What Do Students Already know? - Prior Knowledge at the First Day of Class; IaaS (i.e. AWS, Azure, GCE, OpenStack) is one of the existing skills whereas MapReduce, Iterative MR Model (i.e. Twister, Harp), and Data Analytics are desired knowledge to learn during the course.

The survey asked the students to rate their preparation and growth in different cloud technologies related to the projects. It is observed in Fig. 8 that many students have 0, 1-2 years or no programming experience, although Java is a familiar language to students. Half of the class are able to use VMs and Cloud but lack of experience in using tools in distributed and parallel computing environments. Data analytics skills are desirable in particular as shown in Fig. 9. Fig. 10 illustrates the knowledge growth of students as seen by themselves in different areas in a scale of 1 to 5. Fig. 11 shows the average score of projects related to these areas. It is hard to master these cloud technologies in a semester as they require vast knowledge and experience. To bridge the gap, having an average knowledge after taking the course seems to be a reasonable expectation.

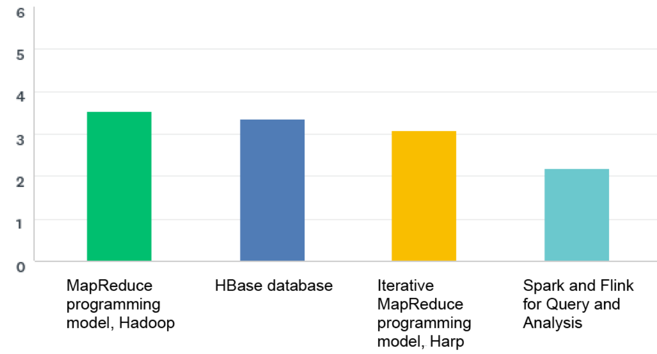

Fig. 10. Student knowledge growth in different cloud technologies on a scale of 1 to 5 from low to high. The averages are shown.

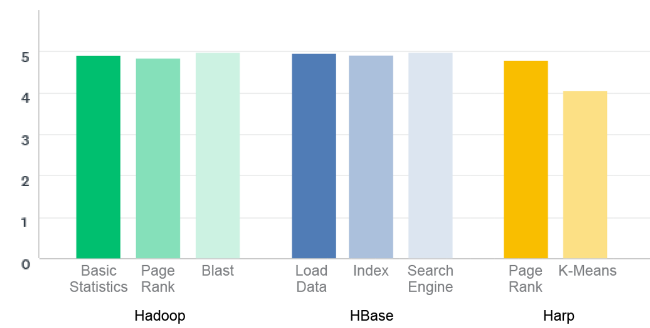

Fig. 11. Student grades in projects related to Hadoop, HBase and Harp. The marks are scaled 1 to 5 from low to high. Eight projects are listed in a row showing the marks for each project given on a specific topic.

We are evaluating the course each year and try to add new content as well as changing the projects and assignments to facilitate the integration of new knowledge and fast evolving technologies. The course also offers students with extra motivation to take on research projects with the instructor to further enhance their knowledge in the field.

V. ADMI Cloud

The Association of Computer and Information Science/Engineering Departments at Minority Institutions (ADMI) cloud attempts to develop curriculum and research for a remote sensing course with shared modules at a minority serving institution. Although the faculty is not well versed in parallel and distributed computing, training seminars were introduced, which allowed them to become familiar. Also, support throughout the year is granted to ensure successful course completion.

A pilot course is hosted by faculty at the Elizabeth City State University Department of Mathematics and Computer Science with the course focus on RS 506 The Principles of Microwave Remote Sensing. RS 506 introduces spaceborne remote sensing of the Earth's atmosphere, land, and oceans. The primary methods and applications of microwave remote sensing are considered, with both active (radar) and passive (radiometry) techniques discussed.

There are computer science and computational science (domain science) undergraduate research activities involving Clouds. The computer science focus includes a set of topics leveraging research from Indiana University programming models (Hadoop and MPI), storage, cloud environments, performance, and integration with sensor devices. The domain science approach utilizes polar science applications. Clouds provide an venue to store domain data and support multidisciplinary work. For example, the polar science community has built noninstrusive radars capable of surveying the polar ice sheets. As a result, they have collected terabytes of data from past surveys. They are increasing their repository every year as signal processing techniques improve and the cost of hard drives decreases, enabling new generation of high resolution ice thickness and accumulation maps. Manually extracting features from an enormous corpus of data is time consuming and requires sparse hand-selection, so developing image processing techniques to automatically aid in the discovery of knowledge of high importance.

In order to provide a scalable model in the targeted universities, it is essential to involve MSI faculty so they can teach classes and mentor/perform research, with the central responsibility being the modular Remote Sensing curriculum using the Cloud Computing electronic site. This technique is well-established and often termed teach the teachers or train the trainers and is studied in the context of professional development for teachers [8] [9] [10] [11] [12].

ADMI faculty participated in a summer 2016 training program, which they were trained to use MOOCs, as a first phase of the project. These training exercises were conducted during a three day session at Elizabeth City State University. During the session, participants were encouraged to develop initial content of classes taught at their home institutions. IU faculty and graduate students provided instructions using a mix of virtual and residential modes. The workshop activities include development and delivery using MOOCs for targeted ADMI computer science courses. Two follow up professional development activities were further provided at the ADMI meetings. One is an ADMI Curriculum Enhancement using Cloud Computing and MOOC Workshop, another is the ADMI conference in 2017. In the meetings, the concepts of cloud computing were presented in order to provide information for hosting and sharing new courses, and the ADMI faculty continued discussions from the summer to help exchange ideas about how to implement MOOCs for their classrooms.

The scaling courses are collaboration with faculty at North Carolina AT (NCAT) and Winston-Salem State University (WSSU), where Cloud Computing components can be added to their programs to enhance existing curricula for multiple classes. The skills and techniques of using Cloud will support economic development by preparing students for the many jobs becoming available in the Computer Science and Data Science areas.

VI. Discussion and Future Work

The Cloud Computing course has been offered for many years to the residential computer science graduate students at the Indiana University and has observed high enrollments each time. The course offers a mix of core concepts of distributed and parallel computing along with their practical applicability. This combination has been proven to be successful in teaching a diverse group of students who are primarily looking towards industry which increasingly demands engineers with experience in distributed and parallel computing domains. Faculty and IU support have helped develop a curriculum for remote sensing materials and this will allow other institutions within the AMDI community to reuse existing materials in order to foster a community of learning.

Clouds and online MOOCs offer cutting-edge technologies to enhance traditional computational science curriculum and research with next-generation learning metaphors. There are many challenges in scaling the course and providing a robust learning environment. We have developed specific methods for effective teaching of large classes (hands-on labs), accommodation for individual student needs (forums, online meetings), and customization for interdisciplinary collaboration (content repository), as well as extensive engagement, outreach and training for a broader community.

This project builds off existing Indiana University activities, involving REUs for ADMI and other undergraduates, two cloud-related courses offered in Computer Science and Data Science programs. The project activities will include course development and delivery using MOOCs for a cloud-enhanced classes taught by ECSU and other institutions and IU faculty with a mix of virtual and residential modes. The course outcomes will be evaluated to understand the best practices of such shared curriculum across multiple disciplines and institutions. The prototyped cloud-based course modules are made available as examples of open source community X-MOOC repository [13]. For future work, we will continue modernizing curricula that is suitable for our next generation workforce development and connect to the community by systematically introducing multiple courses, teacher training, research support and electronic resources across the ADMI MSI and other teaching university networks.

Acknowledgment

The authors are grateful for the generous support from NSF EAGER Grants 1550784 and 1550720 on Remote Sensing Curriculum Enhancement using Cloud Computing. Google grant on Customizable MOOC for Cloud Computing supported the initial development and offering of the online course. The Harp open source software has been developed and used by students for their course projects, and we gratefully acknowledge generous support from the Intel Parallel Computing Center (IPCC) grant, NSF OCI-114932 (Career: Programming Environments and Runtime for Data Enabled Science), NSF DIBBS 143054: Middleware and High Performance Analytics

Libraries for Scalable Data Science. We would like to thank the students who participated in the surveys to provide feedback on the course. We would like to extend our gratitude to associate instructors who worked on this course over the years.

REFERENCES

[1] K. Hwang, J. Dongarra, and G. C. Fox, *Distributed and cloud computing: from parallel processing to the internet of things*. Morgan Kaufmann, 2013.

[2] T. White, *Hadoop: The Definitive Guide*, 1st ed. Sebastopol, CA, USA: O'Reilly Media, Inc., 2009.

[3] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, May 2010, pp. 1–10.

[4] L. George, *HBase: the definitive guide: random access to your planet-size data*. " O'Reilly Media, Inc.", 2011.

[5] J. Callan, M. Hoy, C. Yoo, and L. Zhao, "Clueweb09 data set," 2009.

[6] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[7] B. Zhang, Y. Ruan, and J. Qiu, "harp: Collective communication on hadoop," in *2015 IEEE International Conference on Cloud Engineering*.

[8] J. Van Orshoven, R. Wawer, and K. Duytschaever, "Effectiveness of a train-the-trainer initiative dealing with free and open source software for geomatics," in *Proceedings (J.-H. Haunert, B. Kieler and J. Milde, Eds.) of the 12th AGILE International Conference on Geographic information Science*, 2009.

[9] B. Fishman, S. Best, J. Foster, and R. Marx, "Fostering teacher learning in systemic reform: a design proposal for developing professional development." 2000.

[10] J. H. van Driel, D. Beijaard, and N. Verloop, "Professional development and reform in science education: The role of teachers' practical knowledge," *Journal of Research in Science Teaching*, vol. 38, no. 2, pp. 137–158, 2001. [Online]. Available: http://dx.doi.org/10.1002/1098-2736(200102)38:2¡137::AID-TEA1001¿3.0.CO;2-U

[11] D. Hestenes, "Toward a modeling theory of physics instruction," *American Journal of Physics*, vol. 55, no. 5, pp. 440–454, 1987. [Online]. Available: http://dx.doi.org/10.1119/1.15129

[12] H. Borko, "Professional development and teacher learning: Mapping the terrain," *Educational Researcher*, vol. 33, no. 8, pp. 3–15, 2004. [Online]. Available: http://dx.doi.org/10.3102/0013189X033008003

[13] X-mooc repository: Curriculum enhancements with cloud and mooc for online learning. [Online]. Available: http://cloudmooc2.soic.indiana.edu/