

December 2017

Volume 8 Issue 3

JOCSE

Journal Of Computational Science Education

Promoting the Use of
Computational Science
Through Education

ISSN 2153-4136 (online)

JOCSE

Journal Of Computational Science Education

Editor: Steven Gordon
Associate Editors: Thomas Hacker, Holly Hirst, David Joiner,
Ashok Krishnamurthy, Robert Panoff,
Helen Piontkivska, Susan Ragan, Shawn Sendlinger,
D.E. Stevenson, Mayya Tokman, Theresa Windus

CSERD Project Manager: Jennifer Houchins **Managing Editor:** Jennifer Houchins. **Web Development:** Jennifer Houchins, Aaron Weeden, Joel Col-dren. **Graphics:** Stephen Behun, Heather Marvin.

The Journal Of Computational Science Education (JOCSE), ISSN 2153-4136, is published quarterly in online form, with more frequent releases if submission quantity warrants, and is a supported publication of the Shodor Education Foundation Incorporated. Materials accepted by JOCSE will be hosted on the JOCSE website, and will be catalogued by the Computational Science Education Reference Desk (CSERD) for inclusion in the National Science Digital Library (NSDL).

Subscription: JOCSE is a freely available online peer-reviewed publication which can be accessed at <http://jocse.org>.

Copyright ©JOCSE 2017 by the Journal Of Computational Science Education, a supported publication of the Shodor Education Foundation Incorporated.

Contents

Introduction to Volume 8 Issue 3 <i>Steven I. Gordon, Editor</i>	1
Using Inexpensive Microclusters and Accessible Materials for Cost-Effective Parallel and Distributed Computing Education <i>Joel Adams, Suzanne J. Matthews, Elizabeth Shoop, David Toth, and James Wolfer</i>	2
Innovative Model, Tools, and Learning Environments to Promote Active Learning for Undergraduates in Computational Science and Engineering <i>Hong Liu, Andrei Ludu, Jerry Klein, Michael Spector, and Matthew Ikle</i>	11
Computational Approaches to Scattering by Microspheres <i>Reed Hodges, Kelvin Rosado-Ayala, and Maxim Durach</i>	19
Toward simulating Black Widow binaries with CASTRO <i>Platon I. Karpov, Maria Barrios Sazo, Michael Zingale, Weiqun Zhang, and Alan C. Calder</i>	25
Using Blue Waters To Assess Non-Tornadic Outbreak Forecast Capability by Lead Time <i>Taylor Prislowsky and Andrew Mercer</i>	30
Blue Waters Supercomputing Applications in Climate Modeling with the WRF Model <i>Morgan Smith and Andrew Mercer</i>	36

Introduction to Volume 8 Issue 3

Steven I. Gordon
Editor
Ohio Supercomputer Center
Columbus, OH
sgordon@osc.edu

Forward

This issue begins with an article by Adams et. al. which describes a number of different microclusters that are being used by educators to introduce their students to parallel computing. They provide a review of various projects that have used different hardware in classroom settings, providing an assessment of the pros and cons of each. They then go on to discuss several strategies for using microclusters as well as options for inserting a course into the undergraduate curriculum.

Liu et. al. provides a summary of a coalition of universities used to introduce computational science courses and skills to their students. Using a combination of online instruction and local lab experiments, three campuses shared their resources and instruction to teach three courses: two in mathematical modeling and one in data mining. The project provided a way for smaller, teaching oriented institutions to provide computational science skills for their students.

This issue also has four articles by students describing their internship experiences. Hodges, Rosado-Ayala, and Durach compared models of electromagnetic fields produced using both Mathematica and Fortran95. Karpov et. al. simulated black widow pulsar systems using the adaptive-mesh astrophysical simulation code Castro. Prislowsky and Mercer used the WRF code to simulate five derecho events in an effort to explore the forecast quality and lead time for such events. Finally, Smith and Mercer used the WRF model to explore climate-scale interannual variability patterns that impact climate forecasting efforts.

Using Inexpensive Microclusters and Accessible Materials for Cost-Effective Parallel and Distributed Computing Education

Joel C. Adams
Department of Computer
Science
Calvin College
adams@calvin.edu

Suzanne J. Matthews
Department of EE & CS
United States Military
Academy
suzanne.matthews@usma.edu

Elizabeth Shoop
Department of Mathematics,
Statistics, & Computer
Science
Macalester College
shoop@macalester.edu

David Toth
Department of Computer
Science
Centre College
david.toth@centre.edu

James Wolfer
Department of Computer &
Information Sciences
Indiana University, South Bend
wolfer@jwolfer.net

ABSTRACT

With parallel and distributed computing (PDC) now in the core CS curriculum, CS educators are building new pedagogical tools to teach their students about this cutting-edge area of computing. In this paper, we present an innovative approach we call *microclusters* – personal, portable Beowulf clusters – that provide students with hands-on PDC learning experiences. We present several different microclusters, each built using a different combination of single board computers (SBCs) as its compute nodes, including various ODROID models, Nvidia’s Jetson TK1, Adapteva’s Parallella, and the Raspberry Pi. We explore different ways that CS educators are using these systems in their teaching, and describe specific courses in which CS educators have used microclusters. Finally, we present an overview of sources of free PDC pedagogical materials that can be used with microclusters.

Keywords

Beowulf clusters, microcluster, Computer Science, Distributed, Education, Parallel, Teaching

1. INTRODUCTION

Prior to 2005, parallel and distributed computing (PDC) were elective topics in the computer science (CS) curriculum, optional, and rarely covered at the undergraduate level. While parallel programming libraries such as OpenMP [21] and MPI [24, 23] existed, the expense of parallel hardware and the difficulty in accessing high performance resources made these concepts difficult to teach. Over the last decade,

this has changed largely as a result of two watershed events: the birth of multicore architecture and cloud computing. Both of these innovations have greatly decreased the cost and increased the accessibility of programming parallel architectures.

Intel and AMD released the first of many commercial multicore CPUs in 2006. This changed the foundation of commodity hardware, because each core in a multicore CPU could run a program simultaneously (i.e., in parallel). Dual-core CPUs were followed by quad-core, then hexa-core, then octa-core, and so on. Today Intel offers 22-core Xeon and 72-core Xeon Phi CPUs [26]. Traditional sequential programs use only one of a CPU’s cores. Software must be intentionally designed and written as parallel software, if it is to fully leverage multicore CPUs. In addition, Nvidia released the CUDA [35] library in 2007, spurring the birth of GPGPU and manycore computing.

Amazon introduced its *Elastic Compute Cloud* (EC2) in 2006, allowing users of its service to run their applications on rented virtual machines (VMs) on Amazon’s infrastructure. Users can vary the number of VMs they rent (the “elastic” aspect of Amazon’s service), allowing them to write distributed applications that run at varying scales. This ability to access scalable computing facilities without having to maintain the hardware infrastructure has proven highly attractive to the business community. This popularity has necessitated a shift in program design: Since a traditional sequential program runs on just one machine, software must be designed and written as parallel and distributed software to run across multiple VMs, if it is to take advantage of the scalability of a cloud service like EC2.

Recognizing the implications of these events – that all CS students now need to learn about PDC – both the *IEEE TCP/ Curriculum Recommendations* [40] and the *ACM/IEEE CS 2013 Curriculum Recommendations* [44] moved PDC topics into the core CS curriculum.

These changes raise many questions, including:

What hardware, software, and teaching resources should we use to teach students about PDC?

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, 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.

Given the ubiquity of the multicore CPU, just about any computer has hardware on which students might learn about parallel computing, but not distributed computing. Distributed computing by definition involves a computation being distributed across multiple machines. Hardware platforms that might be used for teaching distributed computing include a network of workstations (NoW), a Beowulf cluster [42], or a cloud system such as Amazon's EC2. A computer lab can be configured as a NoW, and so can be used to teach PDC. However, when students launch distributed computations in a heavily-used lab, each student's computational performance will suffer as the computations compete for the lab's limited CPU resources. A cloud service such as Amazon's EC2 can be used to teach PDC, but the processing all takes place at a distance from the student, resulting in a loss of immediacy in the student's learning experience. High performance Beowulf clusters can be used, but they are relatively expensive to build and maintain.

The widespread availability of inexpensive single-board computers (SBCs) provides a different option for teaching PDC. Just as inexpensive integrated circuits made the *microcomputer* possible, inexpensive SBCs make it possible to build a **microcluster** – a personal, portable, Beowulf cluster – on which students learn about PDC.

In this paper, we present several different microclusters that CS educators have built for teaching their students about PDC, and explore the different ways these educators are using them in their classes. In the next section, we present some background information on microclusters. We examine the above-mentioned microclusters in detail in Section 3. In Section 4, we describe our experiences using these microclusters for teaching, research, and outreach. In Section 5, we discuss some of the software and teaching resources that can be used on these systems to help students learn about PDC. We conclude with some observations in Section 6.

2. BACKGROUND

Wiglaf, the first Beowulf cluster, was built by Donald Becker and Thomas Sterling at NASA in 1994 [42]. *Wiglaf* was a small cabinet containing sixteen motherboards with 80486 CPUs, which communicated through 10Mbps Ethernet. Unlike most of its successors, *Wiglaf* might be considered a microcluster by today's standards, since the entire cluster fit into one cabinet with its monitor and keyboard on top. In contrast, its immediate successor *Hrothgar* consisted of three shelves containing sixteen Pentium PCs connected with 100Mbps Ethernet, initiating the "lots of boxes on shelves" model used by many subsequent Beowulf clusters.

Almost immediately after *Wiglaf*'s creation, people began building microclusters. In the rest of this section, we describe a few of these early examples of microclusters, as the context for the systems described in Section 3.

2.1 TTL_Papers

The term "microcluster" was coined by Hank Dietz and his students at Purdue University. They debuted their *TTL_Papers* microcluster at the 1994 Supercomputing conference [22]. It consisted of four nodes, each with a 25-MHz 80486 processor, that communicated through their parallel ports via a custom-built interconnect unit. The entire cluster weighed 30 pounds and fit within a 1 foot (30.48 cm) cube.

2.2 SETI Stacks

In 1999, the SETI@Home project [29] released a distributed computing client that, running on a personal computer, would: (1) download a batch of signals received by a radio telescope; (2) analyze those signals, looking amid the noise for regular patterns that might be evidence of intelligent communication; and then (3) report its findings back to the project.

In the hope of being the first to find evidence of extraterrestrial intelligence, many enthusiasts built clusters dedicated to running the SETI@Home client. At least 26 of these were microclusters that were dubbed "SETI Stacks", consisting of stacked motherboards with Pentium-era CPUs, communicating via original Ethernet. Many of these systems had colorful names like "*Crunchenstein Stack*", "*Stomp Monster*", "*SetiCruncher*", and so on [33].

2.3 Ultimate Linux Lunchbox

In the early 2000s, Ron Minnich and Mitch Williams built a series of microclusters at Sandia and Los Alamos National Labs. These may have been the first clusters built using SBCs, and their efforts culminated in 2005 with the *Ultimate Linux Lunchbox*, a lunchbox-sized microcluster consisting of sixteen Technologic Systems TS-7200 SBC nodes, connected using 100Mbps Ethernet [34]. These SBCs had StrongARM CPUs, a precursor to the ARM processors used by the microclusters we present in Section 3.

2.4 LittleFe

In 2005, Paul Grey, Tom Murphy, and Charlie Peck built *LittleFe*, a six-node cluster-in-a-suitcase small enough to take on the road to conferences, workshops, and high school outreach events [14]. The name is a pun, as "LittleFe" is the opposite of "Big Iron".

The initial *LittleFe* was a wooden frame housing six microATX motherboards with 1-GHz x86 CPUs, 100Mbps Ethernet, a shared hard drive, and a customized Linux distribution. Subsequent versions replaced the wooden frame with a machined metal frame, upgraded the network to Gigabit Ethernet, and upgraded the CPUs to multicore CPUs with integrated GPUs. Each version cost less than \$2800, which includes a Pelican case rugged enough to let its *LittleFe* survive an airline's baggage-handling system.

In 2010, Charlie Peck obtained a grant from Intel to fund the first *LittleFe Buildout*, a conference event at which attendees (generally faculty) were given the raw materials for a *LittleFe* and spent a day assembling them into a working cluster, aided by Charlie and his students from Earlham College. At the end of the day, participants took the system they had built back to their home institutions, at no cost to them or their institutions. The beauty of this approach is that in building their own cluster, the participants acquired much of the knowledge needed to maintain it. Subsequent support from Intel, the National Science Foundation, and XSEDE has funded a total of 7 *LittleFe Buildouts*, resulting in 106 *LittleFe* clusters being built and used to teach parallel computing in colleges and universities throughout the U.S. Most of the systems in this paper were inspired by *LittleFe*.

2.5 Microwulf

After seeing the first version of *LittleFe* at a 2006 conference, Joel Adams and his student Tim Brom built a "personal, portable, Beowulf cluster" at Calvin College. Their goal was to generate as much computational performance as



Figure 1: Parallella in case (left). 4-node Parallella Beowulf Cluster (right).

possible within a \$2500 budget. To accomplish this, Adams designed this cluster in keeping with “Amdahl’s Other Law” [13]; balancing its CPU resources, main memory, and network bandwidth, to keep its computations from being CPU-, memory-, or network-bound.

The result was *Microwulf* [9], a microcluster of four nodes, each with a dual-core AMD Athlon64 CPU, 2GB of RAM, and two Gigabit Ethernet adaptors. For less than \$2500 in early 2007, *Microwulf* achieved 26.25 Gflops on the HP-Linpack supercomputing benchmark [38]. This gave *Microwulf* a price/performance ratio of \$94/Gflop, making it the first Beowulf cluster to break the \$100/Gflop barrier. By August 2007, the price of its components had dropped to \$1256, improving its price/performance ratio to \$48/Gflop.

Adams used *Microwulf* in his Fall 2007 *High Performance Computing* (CS 374) course at Calvin College, to provide students with hands-on experience developing and running MPI, OpenMP, and MPI+OpenMP programs.

3. MICROCLUSTERS FOR TEACHING PDC

In 2012, the Raspberry Pi Foundation released the Raspberry Pi, a \$35 credit-card sized SBC. Its low cost catalyzed many CS educators and hobbyists to build Raspberry Pi microclusters. Notable early efforts with the Raspberry Pi include *IridisPi* [20], *PiCloud* [47], and efforts at Boise State [28] and FUB [8]. Despite its low cost, the Raspberry Pi’s relatively weak CPU (700 Mhz ARM 11) and small on-board memory (512 MB) encouraged educators to turn to other SBCs for creating microclusters to teach their students about PDC¹. In this section, we briefly give an overview of four recent parallel efforts at the authors’ institutions.

3.1 StudentParallella

West Point was perhaps the first to incorporate the Parallella in the classroom [31]. Introduced by Adapteva via Kickstarter in 2013, the Parallella is a credit-card sized SBC with a dual-core Zynq ARM A9 CPU, 1 GB of RAM, gigabit ethernet, and a 16-core Epiphany co-processor. The board is extremely power efficient, requiring only a 5V/2A power supply, similar to a Raspberry Pi. The desktop edition of the board retails for \$145.00, while the microserver edition retails for \$99.00.

At West Point, Suzanne Matthews introduced undergraduates to the Parallella in a parallel computing elective course

¹The release of the Raspberry Pi 2 in 2015 updated the board to a 900 MHz quad-core ARM Cortex A7 CPU, making it possible to use the board as a standalone unit for teaching PDC concepts

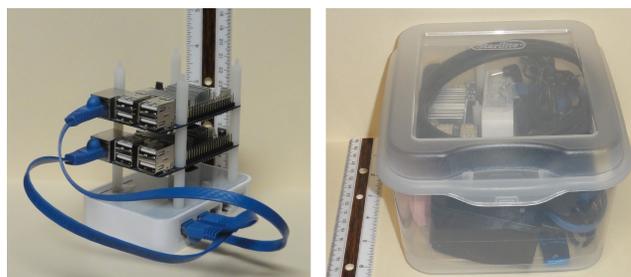


Figure 2: The 2-node HSC 6 Cluster (left). “Half-Shoobox” case (right).

in the Spring 2015 semester. Each student had their own desktop edition of Parallella board. Epiphany programming was one of five programming modules covered in the course, along with C, Pthreads, OpenMP, and MPI. Matthews open-sourced [31, 30] her Parallella teaching materials, disk images, and setup guides in 2015, facilitating others to use the Parallella in the classroom. A custom case for the Parallella that enables it to be used on its own or configured into a *StudentParallella* microcluster was designed and open-sourced by Matthews and Blackmon [32]. In Figure 1, we show the Parallella in case assigned to each student, as well as a 4-node microcluster configuration.

Pros and Cons of Parallella-based Clusters.

Students were initially very excited by the Parallella. The small form factor and powerful 16-core co-processor were extremely motivating to students early in the course. However, their enthusiasm waned as the course went on, due to difficulties with the Epiphany Software Development Kit (eSDK) [31]. Students were largely dependent on Matthews’ guide to the eSDK [30] to complete their co-processor assignment due to their lack of experience reading user manuals. The dual-core ARM CPU also limited students’ speedup analyses. It is worth noting the Zynq SoC also has an FPGA which can be used in a parallel computing course, though Matthews did not do so.

Matthews concluded that while using a credit-card sized computer was very motivating for her students, it was unclear if the Parallella was the best option. Programming the co-processor is not significantly easier than programming a GPU, and some students at the end of the course expressed a desire to have learned CUDA instead [31]. While the Parallella system shows much promise, existing software packages and APIs could use more maturity before the Parallella is really ready for integration into an academic course.

3.2 Half Shoobox Clusters

David Toth of Centre College built the first *Half Shoobox Cluster* (HSC) in 2014 [45]. The project was motivated from a desire to build a low-cost, portable cluster for students to learn about PThreads, OpenMP, and MPI, necessitating a dual-core SBC.

The first HSC consisted of two dual-core Cubieboard2 [4] SBCs. After discovering the ODROID [3] SBC, six subsequent HSCs were built with various ODROID SBCs as they were released, beginning with the ODROID U3, and continuing with the C1, XU3-Lite, C1+, XU4, and most recently, the C2. The U3, C1, C1+, and C2 all have quad

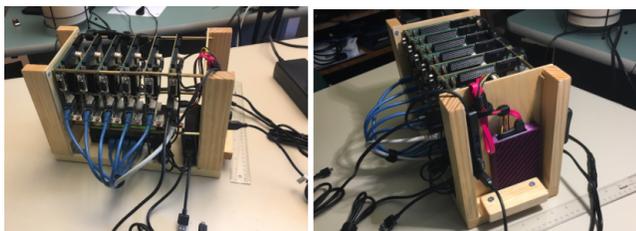


Figure 3: Rosie, a 6-node Nvidia Jetson TK1 cluster

core CPUs, while the XU3-Lite and XU4 have 8-core ARM CPUs. The HSCs with the XU3-Lite and XU4 nodes also support OpenCL programming. The least expensive HSC costs about \$150. Figure 2 shows the most recent HSC.

Toth maintains a web site [46] with disk images for the HSCs and a parts list so people can build their own HSCs and start using them immediately. HSC 6 is shown in Figure 2. The cluster is 2.75" wide by 4.25" long by 4.25" high. The box is 6.75" wide x 7.5" deep x 4.5" high. All other HSCs fit in the same box, but some HSC boards are slightly bigger than HSC 6's boards.

Pros and Cons of Half Shoebox Clusters.

Students loved having their own clusters. Students not enrolled in the course were very curious about the clusters and in some cases, envious of the students in the course. The ODROID nodes also have GPIO pins like a Raspberry Pi, allowing the nodes to be used in an Internet of Things course. The forum on the web site of the manufacturer of the ODROID nodes have a vast assortment of questions and contributions by users. In addition, questions posted to the forums get answered quickly, and there is a free ODROID magazine with lots of projects and ideas. In short, much like there is a large Raspberry Pi presence and community on the Internet, such a presence and community exist for the ODROID systems, too.

Some disadvantages of the HSCs are that only some of them (XU3-Lite and XU4 nodes) support GPU programming with OpenCL, and none support CUDA.

3.3 Rosie

In April 2014, Nvidia released a development board called the Jetson TK1, featuring their Tegra TK1 processor. The board consists of a quad-core ARM Cortex-A15 processor and an integrated Kepler GPU with 192 cores. A single Jetson board costs \$192.00, or one dollar per core. This low cost, together with its capability of easily being flashed with an Ubuntu Linux distribution from Nvidia, makes these Jetson boards good candidates for building an inexpensive cluster with a great deal of computing power; this six node cluster provides $(6 \times 4) = 24$ CPU cores plus $(6 \times 192) = 1152$ CUDA-capable GPU cores.

Libby Shoop and a Macalester College undergraduate connected six of these boards via a Gigabit Ethernet switch to create a six-node microcluster named *Rosie* (see Figure 3). Rosie features an NFS-mounted disk that all nodes share, and each node is CUDA-capable, thanks to its Kepler GPU and Nvidia Linux distribution. Because each board has multicore and GPU capability, this cluster has been used to teach heterogeneous computing techniques with MPI and either OpenMP or CUDA.

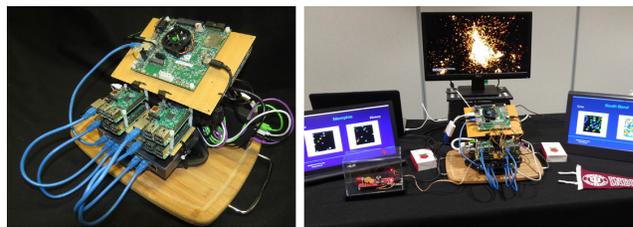


Figure 4: Cu-T-Pi

Pros and Cons of Jetson-based Clusters.

Rosie is placed on a cart and rolled into class to demonstrate how this small cluster models larger supercomputers. Students gather around the hardware and are excited to see its pieces. Instructors can explain sources of overhead by pointing out the distance over the network that data must travel when using MPI for distributed programs. This visceral interaction with the hardware is valuable.

This value has a price: at \$192, the Jetson is the most expensive of the SBCs. Setting up such a cluster also requires some effort. NVidia provides software for flashing the operating system onto these boards (including CUDA and MPI), but you need another Ubuntu Linux machine to do so. Afterward, the cluster network, network file system (NFS), passwordless ssh, and accounts must be set up for student users of the system to use MPI. For six nodes, this process took an undergraduate student under Shoop's supervision a few days. To make this easier, complete instructions, including a parts list, are available on [6].

3.4 Cu-T-Pi

Cu-T-Pi, shown in Figure 4, is a microcluster named for some of the technologies it aggregates (CUDA, Nvidia Tk-1, and Raspberry Pi). Created by James Wolfer [48] at Indiana University, South Bend (IUSB), this cluster is designed to be highly visible, portable, and have hardware and software architecture consistent with contemporary heterogeneous systems. The cluster consists of four Model B+ Raspberry Pi worker nodes and one Nvidia Jetson Tk-1 head node, connected through a gigabit Ethernet switch, all mounted in a terraced arrangement for instructional visibility. The resulting system provides eight ARM cores and 192 CUDA-capable GPU cores, supporting demonstrations and development using the OpenMP, MPI, and CUDA platforms.

In addition to its use for introducing parallel concepts, Wolfer has used CU-T-Pi to demonstrate benchmarking concepts in his *Parallel Computing* course. By using the HPL benchmark adapted for the Raspberry Pi [38, 15], we can observe and quantify the impact of asymmetric communication speeds. Details can be found in [48].

Pros and Cons of CU-T-Pi.

Positive aspects of this heterogeneous system include modeling current HPC architecture, supporting heterogeneous software development. Asymmetric speeds between component nodes offer unique opportunities for MPI benchmarking. Each node includes GPIO capabilities allowing interfacing with custom hardware.

Limitations include the relatively slow speed of the Raspberry Pi, and incompatible GPIO voltage/current requirements between the Raspberry Pi and TK-1 computers.

4. TEACHING, RESEARCH, & OUTREACH EXPERIENCES

Microclusters are useful for introducing PDC concepts inside and outside of the classroom. In this section, we describe the various strategies we have used to introduce students to parallel computing using microclusters. We describe best practice strategies when assigning each student their own cluster compared to having a single cluster for the entire classroom. We also discuss our experiences in parallel computing electives and core computer science courses. Additionally, we discuss our efforts using microclusters to engage undergraduates in research and outreach events.

4.1 Strategies for Introducing Microclusters

The primary way in which the authors have used microclusters is to introduce PDC concepts to undergraduates. CS Education research advocates the use of “hands-on experiential learning” [19] for teaching PDC concepts, while recommending a high degree of interactivity in the classroom [19, 17]. In all cases, microclusters had a definite “cool” factor that encouraged student engagement and enthusiasm for PDC concepts. There were two primary ways students engaged with microclusters in the classroom. In the first, students shared access to a common microcluster. In the second, each student had their own microcluster. Each strategy has benefits and draw-backs.

4.1.1 One Cluster per Class

The “One Cluster per Class” can take a variety of forms ranging from units designed for student access to classroom demonstration machines. Rosie was designed for both roles, being used at Macalester College in an upper-level *Parallel Computing* course first as a demonstration machine, and then for individual student access for in-class activities, homework problems, and course projects. Students are given accounts on the head node, which is connected to Macalester’s network and given an internal IP address. They can then log in remotely for class activities and work on graded assignments. Cu-T-Pi was designed as a classroom demonstration machine; to provide a portable, pre-configured, “ready-to-rock” device that can move from class to class and provide an introduction to parallel computing.

Both Cu-Ti-Pi and Rosie enable instructors to expose their students to OpenMP, MPI, and CUDA programming. CUDA concepts can be taught via demonstration code provided by Nvidia CUDA SDK, or through open-source teaching materials such as those available through CSinParallel or CDER (see Section 5). For example, Macalester uses the modules available through CSinParallel to teach students CUDA, OpenMP, and MPI programming.

Pros and Cons of “One Cluster Per Class”.

Students have a very positive reaction to seeing the cluster hardware, and its physical nature generates student curiosity and motivates discussion of system components. For example, students can observe communication “overhead” by monitoring the Ethernet lights on both the SBCs and the network switch. Rosie’s NFS-shared hard disk allows for discussion of Network File Systems, and lets students see the storage device. By contrast, traditional HPC or cloud systems hide such details from students, which can hinder their ability to fully understand such mechanisms.

The portability of the “One Cluster Per Class” units encourages their use in classes across the spectrum, from first-year to graduate levels, inviting comments like “...brought in a mini super-computer to demonstrate the power of parallel programming” in a freshman class.

The gains in portability and instructional flexibility are balanced by its limitations. Specifically, except when connected online, these systems are not directly available to students, limiting first-hand experience—only the students involved in building Rosie, for example, gained system-level experience. When used outside class, demand is high at certain times and students compete for cluster cycles, perhaps limiting this approach to smaller institutions.

4.1.2 One Cluster per Student

West Point and Centre College both adopted the “one cluster per student” approach. The decreasing cost of SBC hardware enables each student to purchase SBCs of their own, or sign a cluster out as individual lab equipment. There are some immediate benefits to this approach, including eliminating resource contention between students and maximizing the individual learning experience. A microSD card pre-loaded with necessary course materials can be shared with students at the beginning of the course. Updates may be downloaded by students from a class website. Students may connect to their clusters using SSH or a more familiar desktop environment.

Both Centre College and West Point use this approach to teach students about the MPI, OpenMP, and Pthreads libraries. West Point students also learned about accelerator programming using the Parallella’s Epiphany co-processor. Centre College maximizes the one cluster per student approach by assigning each student their own two-node cluster. In contrast, West Point assigns each student a single Parallella; as an in-class lab, students network their boards together to form a cluster [31].

Pros and Cons of “One Cluster Per Student”.

Students at both institutions were very excited to have their own clusters. At West Point, students opted to primarily connect to their clusters via SSH, owing to the time and number of peripherals (e.g. monitor, keyboard, mouse) required for access. At Centre College, students typically brought their clusters to the dedicated computer science lab and hooked them up to large monitors, keyboards, and mice there. At both places, students were unable to connect their clusters to the Internet, due to institutional IT policies. This is consistent with experience of others experimenting with SBCs at their own institutions [43]. Instead, students transferred files between their laptops and their clusters via SCP/SFTP or USB flash drives.

A significant benefit of the one cluster per student model is that it removes the system administration work from the faculty member - a node crashing only affects a student (who can reboot it themselves). Additionally, when disk images are provided, a faculty member new to teaching PDC can be up and running quickly, without having to learn about configuring the hardware. The one cluster per student model also enables students to operate in isolation from each other, thus allowing them to measure the performance of their sequential and parallel versions of their programs without other students’ tests influencing their results.

4.2 Courses Using Microclusters

A natural place to use microclusters in the classroom is in a *Parallel Computing* elective course. Such courses introduce students to PDC concepts in a self-contained package. The drawback to electives is that they cannot guarantee that every student gets exposure to PDC concepts. The *ACM CS 2013* [44] report introduced the PDC knowledge area and recommended that 15 core tier-one and tier-two hours be included in the core undergraduate computer science curriculum. Likewise, the *NSF/IEEE TCPP Curriculum Initiative on Parallel and Distributed Computing* (NSF/IEEE TCPP) [40] recommends that PDC concepts be introduced in required computer science “systems” courses. The authors have used their microclusters to introduce students to PDC in required courses such as *Computer Organization* and *Computer Architecture*, and to delve more deeply in *Parallel Computing* electives.

4.2.1 Uses in Parallel Computing Electives

At West Point, the students used the Parallella to complete homework assignments. Epiphany co-processor programming was covered along with OpenMP, MPI, and Pthreads. For each course assignment, students wrote a C program and a parallel equivalent in one or two other libraries and performed timing studies to quantify their performance gains. Students were also required to complete a final project and summarize their results in a research paper. In addition, they were assigned 10 topics papers in which the summarized and reflected upon CACM articles covering PDC topics.

At Centre College, students used their HSC systems to learn about Pthreads, OpenMP, and MPI. Students chose a problem and over the course of the semester, they created sequential, OpenMP, MPI, and MPI+OpenMP programs to solve their problem. The programs were then tested and their performance was compared. The students then created posters about their projects, wrote a paper summarizing the project and the results, and gave presentations about their programs and what they had learned.

At Macalester College, students in the elective course used Rosie for an MPI homework assignment and for heterogeneous computing activities with MPI+CUDA and MPI+OpenMP. A Monte Carlo solution for modeling a pandemic was used as a class activity. As the final project for the course, students chose a project for a particular architecture; those who chose a distributed computing solution used Rosie as their platform to explore and test for scalability.

Cu-Ti-Pi has also supported classroom demonstrations in the senior/graduate level *Parallel Computing* course at IUSB.

4.2.2 Uses in “Systems” Courses

At West Point, the Parallella architecture is briefly discussed as part of a larger lesson on hardware acceleration in the *Computer Organization* course. Due to the importance of the x86 architecture in West Point’s CS curriculum, the custom Parallella architecture was eschewed in favor of traditional x86 multicore servers for the hardware base of the course.

At IUSB, Cu-Ti-Pi has supported the *Operating Systems* class by providing remote access to an interfaced Geiger counter, as the basis for a random number serving file system project [27]. It has also been used for classroom demonstrations in the *Computer Organization* course.

Rosie has served as a demonstration platform in the *Computer Systems* course at Macalester College, where it is wheeled into class and used to demonstrate examples that illustrate scalability and speedup in a distributed system. The students work with Pthreads and OpenMP on shared-memory multicore machines first, so Rosie is shown as a contrasting architecture for distributed parallel computing. Plans in progress at Macalester are to include hands-on activities with new Raspberry Pi-based clusters by shortening some units earlier in the course to free up time for distributed programming at the end of the course.

4.3 Research & Outreach Experiences

Microclusters have been used at each institution to inspire students and faculty about parallel computing. For example, Cu-Ti-Pi has been used for demonstrations in a general education computer literacy course.

At West Point, a student project used a cluster of Raspberry Pi B+s to simulate a remotely operated “smart” mortar system [41] in 2014. The same cluster was used by another West Point student in 2015 to study password cracking for their final project in their parallel computing course. In 2017, students ascertained a Raspberry Pi 2 cluster’s ability to detect power grid anomalies [18].

At Macalester College, the building of Rosie itself was a very beneficial summer research experience for undergraduate students. The experience of students has been passed on so that current students have not only rebuilt Rosie but also built new clusters based on newer Jetson boards and on the relatively new Raspberry Pi 3 boards.

The microclusters mentioned in this paper have also been demonstrated at the SIGCSE Technical Symposium [11, 12].

5. TEACHING MATERIALS

As described previously, traditional parallel libraries such as Pthreads, OpenMP, and MPI can all be taught using microclusters. For example, the Pacheco parallel programming textbook [37] and his earlier book for MPI programming [36] have been used at Calvin, Macalester, and West Point as a main text for their parallel computing electives.

Given the advances in computer architecture in the last ten years, a chief complaint is the scarcity of textbooks for teaching modern parallel computing concepts to undergraduates. In this section, we describe two NSF funded initiatives, CSinParallel and CDER, that aim to alleviate this shortfall. CSinParallel offers a series of parallel “modules” that can be used in the context of a course in place of a textbook. CDER has an initiative to build a modern textbook. We also discuss other sources of PDC educational materials. All the authors have used various combinations of these materials to supplement their coverage of PDC concepts in their courses.

5.1 CSinParallel

The NSF-funded *CSinParallel* project [16] maintains a site, CSinParallel.org, which contains a comprehensive set of course modules for PDC education. Each module is designed to be used over a short period of time in any of several courses, depending on the curriculum and course structure at an instructor’s institution. Modules exist for various levels of experience, ranging from novices in introductory courses to experienced seniors in advanced electives. Modules also exist for various types of hardware and software. As

an example, for MPI, there are modules for distributed computing fundamentals, heterogeneous computing, a pandemic modeling exemplar using a Monte Carlo approach, and others. Modules can be used to briefly introduce students to parallel computing concepts, or provide them with in-depth exposure to programming examples and practices. CSinParallel modules have been used successfully at Macalester, St. Olaf, West Point, and other places, with students rating the material highly.

The patternlets module at CSinParallel.org is especially useful for teaching with microclusters. Patternlets are small code examples that demonstrate particular PDC topics, using well-known, tried-and-true parallel design patterns [10]. There are currently 21 MPI patternlets available for teaching distributed-memory parallel topics and 16 OpenMP patternlets for teaching shared-memory parallel topics, as well as patternlets for POSIX pthread multithreading and heterogeneous (MPI+OpenMP) computing.

5.2 CDER

The NSF-funded *Center for Parallel and Distributed Computing Curriculum Development and Educational Resources* (CDER) was created with mission of developing core PDC curricula that can be adopted at a wide array of institutions across industry and academia. CDER maintains a collection of instructor-produced PDC materials [25], and has synthesized several introductory topics into a book [39]. A second volume of the book will target students in upper-level computer science courses.

In addition to collecting PDC educational materials, CDER has also sponsored an Early Adopter Program to provide seed funding for faculty to integrate PDC concepts into their courses, and the EduPar workshop series to provide a venue for dissemination of these Early Adopters' results. The Early Adopter Program has funded over 100 institutions to date. CDER also provides access to parallel hardware platforms for teaching PDC topics.

5.3 Other Sources of Materials

There are many high quality PDC teaching materials available from other sources including HPCUniversity.org [2], the Computational Science Education Reference Desk [1], and Lawrence Livermore National Laboratory (LLNL) [7].

HPCUniversity.org has tutorials on programming languages such as C, C++, and Python, as well as UNIX and shell programming to help students acquire necessary background skills. There are also tutorials there on using OpenMP, MPI, MPI+OpenMP, the Intel Xeon Phi, GPGPU, and CUDA.

CSERD has a number of modules that show how to implement solutions to problems with different parallel computing libraries. These include classic examples like Conway's Game of Life and calculating the area under a curve to interdisciplinary applications of parallel computing such as biofilms and solving the party problem in mathematics.

LLNL is another source of useful tutorials, focused on Pthreads, MPI, and OpenMP.

6. CONCLUSIONS

We have described how microclusters can be used to introduce undergraduate students to PDC. The microclusters at Calvin College, Centre College, IUSB, Macalester College, and West Point have excited students about parallel computing concepts and applications. Microclusters embody the

"hands-on" learning of PDC clusters advocated by CS educators, are a fun way to introduce students of all levels to parallel and distributed computing, and provide small-scale models of larger HPC systems.

We also described different strategies for engaging students with microclusters, including different courses in which students can use microclusters, and microcluster-based research and outreach experiences. We noted free teaching materials that can be used in conjunction with microclusters. Some of our SBCs cost less than many course textbooks; with the availability of free, high-quality teaching materials, SBCs and SBC clusters can be individually purchased by students in lieu of a textbook.

While we highlighted the one cluster per student model and the one cluster per class model, Macalester College has recently begun moving towards an intermediate "several clusters per class" model. In this approach, students work in groups, with each group using their own microcluster to explore hands-on CSinParallel.org activities. These clusters, built in summer 2017 using Raspberry Pi 3 boards, will be used in the 2017-2018 academic year in the sophomore-level systems course. This model should provide better scalability than the "one cluster per class" approach.

The price/performance ratios of SBC architectures continue to improve, as new processors and boards are released. For example, Nvidia introduced the Jetson TX1 in 2015, based on the Tegra TX1 processor, containing an updated ARM CPU and 256 GPU cores, and now offers an updated Jetson TX2 developer kit board for an educational discount price of \$299.00. Macalester has recently built a new cluster using 4 TX2 boards. More affordable multicore SBCs, such as the 4-core Raspberry Pi 3 board (\$35.00 without a power adapter) are also now excellent candidates for affordable small clusters. The future of Adaptea and its Parallella board is in some doubt [5]; however, it is the only SBC that supports teaching about FPGAs and co-processors.

For each SBC, Table 1 summarizes the software libraries that can be covered, the hardware features, and the cost per node. Additional items that may be needed include network router and cables, power supply, monitor, and keyboard. For demonstrations and student research projects, more expensive and powerful clusters based on NVidia Jetson hardware may be preferable. The low-cost multicore ODROID and Raspberry Pi 3 hardware are excellent ways to get hardware in students' hands and avoid competition for a shared cluster. For these SBCs, downloadable images allow students to get started with these systems quickly.

The choice of hardware and teaching model depends on the course learning outcomes at a given institution. To illustrate, all of the microclusters presented in this paper support the coverage of learning outcomes related to OpenMP and MPI; to cover CUDA outcomes, a Jetson SBC is needed; to cover outcomes related to co-processors or FPGAs, a Parallella is needed; and so on. Likewise, the "one cluster per student" model scales well and is applicable at institutions of all sizes; the "one cluster per course" model may be limited to small institutions and/or demonstration machines.

Our collective experiences strongly suggest that microclusters are an inexpensive, accessible, cost-effective, and motivating way to introduce parallel computing concepts to undergraduates, in keeping with current CS curriculum guidelines. We hope that our positive experiences will inspire others to use microclusters to teach PDC concepts.

Features	Parallella	OROID XU4	NVidia Jetson TK1	Raspberry Pi 3
OpenMP	Y	Y	Y	Y
MPI	Y	Y	Y	Y
GPGPU	N	OpenCL	CUDA	N
Co-Processor	Y	N	N	N
FPGA	Y	N	N	N
Cores: CPU + GPU/Co-Processor	2 + 16	8 + 6	4 + 192	4 + 0
Cost (per node)	\$99.00	\$59.00	\$192.00	\$35.00

Table 1: Overview of node architectures and features.

7. ACKNOWLEDGEMENTS

The opinions in this work are solely of the authors and do not necessarily reflect those of the U.S. Military Academy, the U.S. Army, or the Department of Defense.

The authors also wish to thank the anonymous reviewers, whose comments were very helpful in finalizing this paper.

8. REFERENCES

- [1] CSERD: Home. Internet Website, last accessed 05-31-16, 1994. <http://www.shodor.org/cserd/>.
- [2] HPC university: Home. Internet Website, last accessed 05-31-16, 1994. <http://hpcuniversity.org/>.
- [3] OROID | hardkernel. Internet Website, last accessed 06-01-16, 2013. <http://www.hardkernel.com/main/main.php>.
- [4] Cubietech - open source hardware. Internet Website, last accessed 05-31-16, February 2014. <http://www.cubietech.com/>.
- [5] Adapteva status update. Internet Website, last accessed, August 2017. <http://www.adapteva.com/andreas-blog/adapteva-status/>.
- [6] How to Build a Jetson TK1 Cluster. Internet Website, last accessed, August 2017. <http://selkie.macalester.edu/csinparallel/modules/RosieCluster/build/html/>.
- [7] Lawrence livermore national labs high performance computing training. Internet Website, last accessed, August 2017. <https://hpc.llnl.gov/training/tutorials>.
- [8] P. Abrahamsson, S. Helmer, N. Phaphoom, L. Nicolodi, N. Preda, L. Miori, M. Angriman, J. Rikkila, X. Wang, K. Hamily, et al. Affordable and energy-efficient cloud computing clusters: The bolzano raspberry pi cloud cluster experiment. In *Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on*, volume 2, pages 170–175. IEEE, 2013.
- [9] J. Adams and T. Brom. Microwulf: A beowulf cluster for every desk. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, pages 121–125, 2008.
- [10] J. C. Adams. Patternlets: A teaching tool for introducing students to parallel design patterns. *Journal of Parallel and Distributed Computing*, 105:31–41, 2017.
- [11] J. C. Adams, J. Caswell, S. J. Matthews, C. Peck, E. Shoop, and D. Toth. Budget beowulfs: A showcase of inexpensive clusters for teaching PDC. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 344–345. ACM, 2015.
- [12] J. C. Adams, J. Caswell, S. J. Matthews, C. Peck, E. Shoop, D. Toth, and J. Wolfer. The micro-cluster showcase: 7 inexpensive beowulf clusters for teaching PDC. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pages 82–83. ACM, 2016.
- [13] G. Amdahl. Storage and I/O parameters and systems potential. In *Proceedings of the IEEE International Computer Group Conference (Memories, Terminals, and Peripherals)*, pages 371–372, 1970.
- [14] I. Babic, A. Weeden, M. Ludin, S. Thompson, C. Peck, K. Muterspaw, A. F. Gibbon, J. Houchins, and T. Murphy. LittleFe and BCCD as a successful on-ramp to hpc. In *Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment (XSEDE'14)*, 2014.
- [15] Baldzys. Bramble HPL (high-performance linpack benchmark) results. Internet Website, last accessed 2016-06-03, 2014. <https://www.raspberrypi.org/forums/viewtopic.php?t=33186&p=612733>.
- [16] R. Brown and E. Shoop. CSinParallel and synergy for rapid incremental addition of PDC into CS curricula. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*, pages 1329–1334. IEEE, 2012.
- [17] R. Brown, E. Shoop, J. Adams, C. Clifton, M. Gardner, M. Haupt, and P. Hinsbeck. Strategies for preparing computer science students for the multicore world. In *Proceedings of the 2010 ITiCSE Working Group Reports*, pages 97–115. ACM, 2010.
- [18] K. Candelario, C. Booth, A. St. Leger, and S. J. Matthews. Investigating a raspberry pi cluster for detecting anomalies in the smart grid. In *Proceedings of the 2017 IEEE Undergraduate Research and Technology Conference*, 2017.
- [19] R. A. Chesebrough and I. Turner. Parallel computing: at the interface of high school and industry. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education (SIGCSE 2010)*, pages 280–284. ACM, 2010.
- [20] S. J. Cox, J. T. Cox, R. P. Boardman, S. J. Johnston, M. Scott, and N. S. O'Zbrien. Iridis-pi: a low-cost, compact demonstration cluster. *Cluster Computing*, 17(2):349–358, 2014.
- [21] L. Dagum and R. Enon. OpenMP: an industry standard API for shared-memory programming. *Computational Science & Engineering, IEEE*, 5(1):46–55, 1998.

- [22] H. Dietz. TTL papers microcluster. Internet Website, last accessed 2016-05-25, 1995. <http://aggregate.org/EXHIBITS/sc95.html>.
- [23] E. Gabriel, G. E. Fagg, G. Bosilca, T. Angskun, J. J. Dongarra, J. M. Squyres, V. Sahay, P. Kambadur, B. Barrett, A. Lumsdaine, et al. Open MPI: Goals, concept, and design of a next generation MPI implementation. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, pages 97–104. Springer, 2004.
- [24] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing*, 22(6):789–828, 1996.
- [25] N.-T. C. Initiative. Courseware management. Internet Website, last accessed 2016-06-08, 2014. http://grid.cs.gsu.edu/tcpp/curriculum/?q=courseware_management.
- [26] Intel. Server processors. Internet Website, last accessed 2016-05-25, 2016. <http://ark.intel.com>.
- [27] W. J. Keeler and J. Wolfer. A raspberry pi cluster and geiger counter supporting random number acquisition in the cs operating systems class. In *Proceedings of the International Conference on Remote Engineering and Virtual Instrumentation (REV)*, pages 344–345, February 2016.
- [28] J. Kiepert. RPiCLUSTER: Creating a raspberry pi-based beowulf cluster. Technical report, Boise State University, 2013.
- [29] E. Korpela and D. Wertheimer. SETI@Home. Internet Website, last accessed 2016-05-25, 2016. <http://setiathome.ssl.berkeley.edu>.
- [30] S. J. Matthews. Technical musings. Internet Website, last accessed 06-01-16, May 2015. <http://suzannejmatthews.github.io>.
- [31] S. J. Matthews. Teaching with parallella: a first look in an undergraduate parallel computing course. *Journal of Computing Sciences in Colleges*, 31(3):18–27, 2016.
- [32] S. J. Matthews and W. Blackmon. Parallella case and cluster files. Internet Website, last accessed 05-31-16, June 2015. <http://www.thingiverse.com/thing:892684>.
- [33] L. Mester. SETI stack and farm systems. Internet Website, last accessed 2016-05-25, 1999. <https://web.archive.org/web/20130918044235/http://bhs.brook.k12.wv.us/homepage/staff/seti/farms.htm>.
- [34] R. Minnich. The ultimate linux lunchbox. *Linux Journal*, (11):3–7, 139. November, 2005.
- [35] NVIDIA. *CUDA Compute Unified Device Architecture : programming guide*. NVIDIA Corporation, 1st edition, 6 2007.
- [36] P. Pacheco. *Parallel Programming with MPI*. Morgan Kaufmann, San Francisco, Calif, 1st edition, Oct. 1996.
- [37] P. Pacheco. *An Introduction to Parallel Programming*. Morgan Kaufmann, Amsterdam : Boston, 1st edition, Jan. 2011.
- [38] A. Petitel, R. Whaley, J. Dongarra, and A. Cleary. Hpl - a portable implementation of the high-performance linpack benchmark for distributed-memory computers. Internet Website, last accessed 2016-05-25, 2008. <http://www.netlib.org/benchmark/hpl/>.
- [39] S. K. Prasad, A. Gupta, A. Rosenberg, A. Sussman, and C. Weems. *Topics in Parallel and Distributed Computing: Introducing Concurrency in Undergraduate Courses*. Morgan Kaufmann, 1st edition, Aug 2015.
- [40] S. K. Prasad, K. Kant, Y. Robert, A. Chtchelkanova, S. Das, A. La Salle, R. Le Blanc, A. Rosenberg, F. Dehne, M. Lumsdaine, et al. NSF/IEEE-TCCP curriculum initiative on parallel and distributed computing-core topics for undergraduates. In *42nd ACM Technical Symposium on Computer Science Education (SIGCSE 2011)*, 2011.
- [41] Z. J. Ramirez, R. W. Blaine, and S. J. Matthews. Augmenting the remotely operated automated mortar system with message passing. *CrossTalk, The Journal of Defense Software Engineering*, 28(6), November 2015.
- [42] T. Sterling, D. J. Becker, D. Savarese, J. E. Dorband, U. A. Ranawake, and C. V. Packer. Beowulf: A parallel workstation for scientific computation. In *Proceedings of the 24th International Conference on Parallel Processing*, pages 11–14. CRC Press, 1995.
- [43] D. Tarnoff. Integrating the arm-based raspberry pi into an architecture course. *Journal of Computing Sciences in Colleges*, 30(5):67–73, 2015.
- [44] The ACM/IEEE Joint Task Force on Computing Curricula. Computer science curricula 2013: Curriculum guidelines for undergraduate degree programs in computer science. December 2013.
- [45] D. Toth. A portable cluster for each student. In *Proceedings of the Fourth NSF/TCCP Workshop on Parallel and Distributed Computing Education (EduPar-14)*, May 2014.
- [46] D. Toth. Dave toth’s portable compute cluster page. Internet Website, last accessed 06-01-16, February 2016. <http://web.centre.edu/david.toth/portablecluster/index.html>.
- [47] F. P. Tso, D. R. White, S. Jouet, J. Singer, and D. P. Pezaros. The glasgow raspberry pi cloud: a scale model for cloud computing infrastructures. In *Distributed Computing Systems Workshops (ICDCSW), 2013 IEEE 33rd International Conference on*, pages 108–112. IEEE, 2013.
- [48] J. Wolfer. A heterogeneous supercomputer model for high-performance parallel computing pedagogy. In *Proceedings of the IEEE Global Engineering Education Conference (EDUCON/ITEP)*, pages 785–791, March 2015.

Innovative Model, Tools, and Learning Environments to Promote Active Learning for Undergraduates in Computational Science & Engineering

Hong Liu, Andrei Ludu,
Jerry Klein
Embry-Riddle Aeronautical University
Daytona Beach, FL 32114 US
+1 386 226 7741
{liuho, ludua, kleinj7}@erau.edu

Michael Spector
North Texas University,
Denton, TX 76201
United States
+1 940 369 5070
Mike.spector@unt.edu

Matthew Ikle
Adams State University
Alamosa CO 81102
United States
+1 719 588 4487
moikle@adams.edu

ABSTRACT

This paper presents an innovative hybrid learning model as well as the tools, resources, and learning environment to promote active learning for both face-to-face students and online students. Most small universities in the United States lack adequate resources and cost justifiable enrollments to offer Computational Science and Engineering (CSE) courses. The goal of the project was to find an effective and affordable model for small universities to prepare underserved students with marketable analytical skills in CSE. As the primary outcome, the project created a cluster of collaborating institutions that combined students into common classes and used cyberlearning learning tools to deliver and manage instruction. The instrumental tools for educational technologies included Smart Podium, digital projector, teleconference systems such as AdobeConnect, auto tracking camera and high quality audio in both local and remote classrooms. As an innovative active learning environment, an R&D process was used to provide a coherent framework for designing instruction and assessing learning. Course design centered on model-based learning which proposes that students learn complex content by elaborating on their mental model, developing a conceptual model, refining a mathematical model, and conducting experiments to validate and revise their conceptual and mathematical models. A wave lab and underwater robotics lab were used to facilitate the experimental components of hands-on research projects. Course delivery included interactive live online help sessions, immediate feedback to students, peer support, and teamwork which were crucial for student success. Another key feature of instruction of the project was using emerging technologies such as HIMATT (Highly integrated model assessment technology and tools) [11] to evaluate how students think through and model complex, ill-defined and ill-structured realistic problems.

Categories and Subject Descriptors

K.3.1 [Computer Uses in Education]: *Collaborative learning, Distance learning*. K.3.2 [Computer and Information Science Education]: *Computer science education, Curriculum*.

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.

DOI: <https://doi.org/10.22369/issn.2153-4136/8/3/2>

General Terms

Experimentation, Verification.

Keywords

Hybrid Learning, Cyberlearning, Educational Technology, MOOC, Model-based Learning, and Computational Science & Engineering Education.

1. INTRODUCTION

The digital technology in the 21st century is characterized by omnipresent smart devices and ubiquitous computing that enables computing to occur anytime and anywhere. This contributes to big data challenges, increasing complexity and rapid changes in technologies. Consequently, marketable skills for technical careers emerge and rapidly change. To harness complex technologies and make sense of the big data, undergraduate students majoring in STEM (Science, Technology, Engineering, and Mathematics) need to learn how to model the associated problems in mathematical formalisms and leverage the computing resources to simulate the problems. In [18], Levy summarized one of three major educational initiatives of a SIAM (Society for Industrial and Applied Mathematics) as follows:

“The Modeling Across the Curriculum, was built around the idea that modeling can build many job skills students need and can be an important educational tool at not only the secondary and undergraduate levels, but throughout the educational experience”.

The SIAM-NSF-ASA Workshop on Modeling Cross the Curriculum II [13, 14, and 20] made the following two recommendations to math teachers and STEM education policy makers: Building a pipeline for K16 education in mathematical modeling and connecting math to reality. Computational Science and Engineering (CSE) is an emerging multidisciplinary field of study that incorporates both endeavors recommended above. CSE focuses on the integration of knowledge and methodologies from computer science, applied mathematics, engineering, and science to solve real-world problems. CSE provides the critical mathematical modeling skills and data analytical skills that apply to all STEM fields. Therefore, it is critical that CSE courses and curricula are a viable option for every undergraduate STEM major. [19]. Based on the literature above, the authors of this project identified the following two courses as the corner stones of a CSE curriculum: Mathematical Modeling and Simulation (MMS) and Data Mining and Visualization (DMV).

Many non-research focused higher-education institutions, including most minority-serving and small institutions that the authors of this paper are affiliated, lack the necessary resources and minimal enrollments to justify offering such CSE courses. To help provide CSE educational opportunities for students at small institutions, a proof-of-concept project was funded by National Science Foundation (NSF). The project created a cluster of collaborating institutions that combined students into common classes and used cyberlearning technologies [1] to deliver and manage instruction. The primary goal of the project was to offer three cyberlearning courses in CSE through a coalition of three small universities by sharing resources and pooling students. Another goal was to use short-term summer projects to reinforce student learning and assess student ability to solve real-world problems. The following three key activities were implemented: (a) establish a multi-institution coalition that collaboratively provides CSE courses and projects by combining students from different colleges into a single class and leveraging each college's capabilities, (b) provide general CSE courses organized by computational methods (e.g. mathematical modeling and simulation) rather than domain specific courses such as computational biology; and (c) use cyberlearning technologies to deliver courses. This project provided a feasible model and innovative configuration of educational technology to significantly increase student participation in CSE and a cost effective method to renew courses and curricula.

2. STRATEGY AND IMPLEMENTATION

This section presents the implementation details of the hybrid learning model, innovative educational toolset, new MOOCs (Massive Open Online Course) in CSE, and the active learning environment for both face-to-face and online students.

2.1 Infrastructure of Coalition of Universities and Hybrid Cyberlearning Model

The infrastructure of the coalition was built on the contract that all participating institutions share the workload, resources, and benefits equally through faculty partnership over a period of two years. The Coalition of Universities includes two campuses of Embry-Riddle Aeronautical University (ERAU) - Daytona Beach (DB) Campus in Florida and Prescott Campus in Arizona, and Adams State University (ASU) in Colorado. The authors of this paper who are associated with the coalition of universities took turns developing, reviewing and teaching the three courses in Mathematical Modeling (I and II) and Data Mining. In particular, Liu at ERAU developed the first draft of MMS course, Ikle at ASU reviewed, revised and adapted Liu's course for ASU one year later. Meanwhile, Ikle developed the first draft of DMV, then, Liu reviewed, revised and adapted the course for ERAU. While Liu offered the first course in MMS in spring 2014, Ikle sat with his ASU students and monitored the course delivery online. Reciprocally, when Ikle at ASU offered the DMV courses in fall 2014, Liu sat with his student at DB Campus of ERAU online. Moreover, Liu and Ikle exchanged courses to teach in the next round (e.g. Ikle taught MMS in ASU in spring 2015 and Liu taught DMV in fall 2015). Therefore, Liu and Ikle both served as the co-developers and co-teachers of each other's course. Consequently, both universities gained a new course in their curricula that were originally developed by a peer university in the coalition. This model not only provided students access to new CSE courses in each year but also provide a cost-effective method for faculty development and curriculum enrichment.

2.2 Cyberlearning Technology to Facilitate Tri-located Course Deliverance

Each of the three courses was taught by a professor in the classroom at one location with physically present students while a small number of students in the other two universities attended the same class in remote classrooms using live two-way communications. The approach of this project is actually a hybrid learning environment that combines traditional classroom and cyberlearning with live interactions between the classmates as well as students and instructors.

The courses at ERAU were delivered at a state-of-art Teleconference Classroom. The major hardware devices are: (1) HD voice audio system with one microphone sitting in front of the podium and two mics hanging on the ceiling of the classroom and multiple well positioned speakers, (2) an auto video tracking camera that was installed on the back of the classroom to video the instructor, (3) a SmartPodium, which integrated the whiteboard with the computer screen by SmartTechnology (home.smarttech.com) on the front podium, and a digital projector installed in the ceiling of the classroom. Besides standard software settings, the computer in this classroom installed the AdobeConnect teleconferencing system to support two-way live communications to the students in the remote classroom. Among all the devices mentioned above, the most important one is the reliable teleconference system that makes the two-way communication possible. The second most critical instrumental factor is the quality of the audio system. Poor quality mics and speakers can be very annoying to the students in remote campuses. Before a new semester starts, a new instructor for such a synchronous online learning course should practice setting up and testing all relevant systems under the supervision of an experienced member of the IT staff.

In our tri-located class, we have a local class and two remote classes. Two modes of the teleconference features most frequently used were the video mode that uses the full screen to show the videos of local and all remote classes and the presentation mode that use the whole screen to show the lecture notes or other software documents of the presenter. The instructor has the ability to promote a participant in a remote campus as the presenter. In presentation mode, about three quarters of the screen of the remote classes shares the screen of the presenter and the right panel of the other quarter screen shows the small size videos of the instructors and other classes. The CSE courses require the instructors to use the presentation mode most of the time. In this case, students in remote campuses are invisible to the instructor unless they ask questions. Because the online students do not have the physical presence in the classroom, students are reluctant to ask questions and consequently can be easily neglected by the inexperienced online teacher. Therefore, instructors of such a hybrid synchronous learning course need to learn how to periodically query the online students in remote campuses. An effective technique we learned was to set a laptop near our SmartPodium which in effect turns the local class into an additional virtual class. The laptop is used to connect the second video camera shooting towards to physical present students so that it enables students in the remote classes to see the students in the local classroom. While the main computer connected with Smart Podium is set in presentation mode, we can set the laptop in video mode to monitor the facial expressions of the remote students. In this mode, the screen of our virtual classroom is equally divided into four sections, one for the instructor shot by the auto tracking

camera, the second one is the students in local classroom shot by the second camera, and the other two display the two classes of remote students.

2.3 Innovative Learning Environment

An R&D process was used to provide a coherent framework for designing instruction and assessing learning in which the instructional and assessment methods were aligned with a common idea: Model-based learning and reasoning. Besides traditional grading, the project used emerging technologies such as HIMATT to evaluate how students think through and model complex, ill-defined and ill-structured realistic problems [16].

All courses employed model-based learning and reasoning from first principles and were aligned with SIAM's CSE educational goals [19]:

1. Students can construct qualitative models using first principles of the domain.
2. Students can translate their qualitative models to mathematical models which require them to understand the underlying equations governing first principles.
3. Students can use mathematical modeling and computational software tools to create their mathematical models, design data collection systems, and interpret experimental data.

The three cyberlearning courses were: (1) Mathematical Modeling and Simulation-1 (MMS-I). This course was often titled Introduction to Computational Science and is included in most CSE curricula. It was first taught at ERAU-Daytona Beach. (2) Mathematical Modeling and Simulation-2 (MMS-II). This course was a revised version of the computational physics course and was first taught at ERAU-Prescott. This course provided a test case of modifying a domain-specific CSE course and making it non-domain specific: structuring the course by computational method methods rather than by types of physics problems. (3) Data Mining and Visualization (DMV). This course reflected a change in CSE degree programs [2]. Data explosion is a major trend and the resulting challenges have created tremendous employment opportunities in data-driven business, scientific research, and counterterrorism. All courses consisted of modules organized by computational method rather than organized by applications. When the organizational unit was based on the method of computation rather than the domain of application, it made the development of course modules for distributed teaching much easier. Also, modularization provided the flexibility needed to adapt courses to the particular interests and needs of students. Each module included a recorded demonstration and lecture and interactive instructional activities.

The project has been committed to course-based research experiences (CURE) which connect math with reality [14]. The primary instructional strategy of the project was for students to work in study teams solving problems using online resources created or selected for the particular topic. Since students represented multiple academic disciplines, each module included problem sets and supporting materials for the different domains. Each module consists of instruction on (1) a prototype problem, (2) tools for creating a conceptual model, (3) mathematical modeling tools, and (4) resources for each of the various academic disciplines. Each module was designed by clarifying and defining objectives and selecting or developing problems and examples. After completing the team course projects the students in remote campuses had the opportunities to participate in a two-week long summer research workshop. The purposes of the workshop were to reinforce learning and assess the students' problem-solving

ability. The workshops were based on the ACE program: Analysis, Computation, and Experiment [5].

We will use a couple of examples from the MMS to illustrate how the course objectives were implemented and accomplished. MMS (<http://modelsim.wordpress.com>) was designed for college sophomore and junior students who have taken multivariate Calculus and are familiar with at least one programming language. The goal of the course was to learn how to use the advanced mathematics language such as matrix algebra and calculus as well as software tools to solve real-world problems. The topics of the course covered broad interdisciplinary problems whose solutions heavily depended on mathematical modeling and simulation. More specifically, objectives were to

1. Introduce the major categories of mathematical models as well as their modeling languages and tools
2. Expose students to a broad variety of real-world applications of computational mathematics
3. Train students how to follow mathematical modeling process to conceptualize problems, validate their models and verify their solutions
4. Improve students' capability to make judicious tradeoffs in their modeling assumptions to abstract complex problems
5. Provide students with hands-on experience in the use of computational software tools such as MATLAB, NetLogo, STELLA, etc. to model and simulate mathematical problems, and present visual representations of the problem space as well as alternative solutions.
6. Provide students with teamwork experience to solve problems beyond the scope of textbook exercises and typically beyond the scope of effort for one person.

The course includes five modules: 1: Model Classification and Modeling Methodology, 2, Matrix Algebra, 3: Data and Error Models, 4: Agent-based Modeling, and 5: Modeling System Dynamics. The system engineering modeling methodology in module 1 ([7]) emphasizes (1) building of conceptual models before the mathematical models to increase traceability of model assumptions and first principles, (2) separating concerns and refining models iteratively to divide and conquer complexity, and (3) verification and validation (V&V) based on empirical data and mathematical analysis. The other four modules were divided into three units and each unit starts with an interesting application and develops a core concept to model. For example, to illustrate how we designed the course to develop deep learning, but starting at low thresholds, the concept of Eigenvectors was revisited four times with increasing depths at each time. In the first unit, we adapted a module about the Leslie transition matrix and its applications to Biology from a paper by A. Shiflet and G. Shiflet [12]. The Eigenvector was informally and intuitively introduced as the stable population distributions of the Leslie population model. In the second unit, we inspected the direction changes of the vectors under geometric transformations such as mirror reflection and projection. The students found that most transformations have some invariant directions except rotational matrices. Therefore, the concept of Eigenvectors was identified informally again. We provided its formal definition in the third unit and adapted an example of its application to the Google page ranking problem from a popular paper [3] by Bryan and Leise. In the fifth module, the Eigenvectors and their geometric interpretations were identified again from the solutions of the initial value problem of a linear ODE system.

The third module demonstrates the uncertainty and inevitable errors for modeling real world applications. Instead of seeking

exact solutions, the students learned that when dealing with real world applications it is more practical to search for optimal solutions that minimize the estimated errors based on observable data. Random variables, Monte Carlo methods, Markov chain were introduced in the first unit. Students learned how to use the stochastic transition matrices to model and simulate the uncertainty of outcomes for real-world financial and business problems. Students also discovered that great differences of outcomes can be significantly mitigated after the corresponding model was simulated thousands of times. The second unit focused on multivariate data fitting techniques and applications of data-driven prediction models. The third unit presented the concepts of error models, linear and nonlinear Kalman filters, and their applications to GPS. MATLAB was used to simulate how the Kalman filters could help to pinpoint our positions in feet range error by using the signals from 4-6 semi-geosynchronous GPS satellites that were 25,000 km above the earth. Liu learned about this application at a conference in 2005 which was the major inspiration to initiate the MMS course [10]. It takes 27 class hours to cover all lessons of the first three modules. The course included a mid-term test and an individual conceptual modeling project was due two weeks later. In the last month of the course, the focus was shifted to team projects and the instructor met each team separately at least once a week to check their progress and answer questions. The instructor only gave lectures once a week to give an overview to both agent-based modeling and system dynamical modeling. The students were encouraged, but not mandated to learn the last two modules online based on the needs of their team projects.

2.4 Learning Assessment

Our summative learning assessments included: (1) student feedbacks and survey data, (2) peer reviews of the online published course materials and student paper work samples by external experts such as the external evaluator Dr. Michael Spector and instructional designer Dr. Jerry Klein, and (3) student research outcomes from coauthored publications and presentations. We used traditional assessment instruments to measure student learning including tests that required students to explain and predict events as well as rubrics to grade student conceptual and mathematical models. However, knowledge-based tests are not sufficient. These instruments do not measure how an individual thinks or might approach other projects and problems [4]. We created assessment instruments using the HIMATT learning assessment methods and tools [11] and [16]. HIMATT is a validated technology that essentially captures the student's conceptualization or model of a complex situation and compares it with a reference model that could be an expert's model, to assess progress towards expertise, or models of that student at an earlier time to assess progress from previous states of complex thinking. In order to implement this technology, the project faculty created problem scenarios that were open-ended and not fully specified so as to require students to think about the nature of the problem and possible alternative solution approaches but not so detailed that students could actually provide concrete solutions. Four questions were then asked of students and experts: (1) what are the key factors influencing this problem situation, (2) describe the nature of each factor, (3) how are the factors related, (4) describe the nature of each relationship among the factors. HIMATT uses analyzing pairs of resulting conceptualizations with regard to surface, structural and semantic similarity. In general terms, experts tend to see more relationships among factors and tend to identify key nodes or concepts that influence the situation in

comparison with fewer experienced persons, and that pattern held up in this study.

Student feedback included one survey in the beginning and two evaluations in the middle and the end of the semester. The survey had 21 questions about the students' academic and demographic background as well as their beliefs about CSE courses, cyberlearning vs face-to-face learning, and team work. The two evaluations consisted of four items focusing on assessing the students learning outcomes and the need for improvement: (1) List the primary ways your learning has been enhanced in this course, (2) List the primary ways that your learning has been hindered in the course, (3) List the primary ways you could enhance your own learning in the course, and (4) Some recommendations. An independent evaluator conducted the evaluations, summarized the data, and then, reported to both the instructor and the external evaluator. The summary report included the samples and frequencies of positive and negative feedbacks as well as recommendations. It is difficult to quantify the success of this type of data and summarize the feedback of all 8 classes. It is obvious, however, that positive feedback comments were dominant, and constructive criticism was evident; in addition, there was a very low drop/fail rates (0 to 10%) of all classes comparing with other math courses at ERAU (10%-30%). Indeed, the evaluations and formative assessments were more helpful in making timely instructional adjustments than measuring success. For example, in the middle term evaluation of MMS I in the spring of 2014, the remote students complained that the poor audio quality hindered their learning. To respond to this issue, the instructor obtained advice from IT experts and purchased three sets of the high-end quality audio and mic systems called Konftel. In two weeks, the desktop computer based audio systems in all three classrooms were all replaced with the Konftel systems. As another example, we observed that students paid little attention to the lectures that seemed irrelevant to their own team projects, which were assigned in the last month of the course. Therefore, we changed the course delivery method for the last month accordingly in next term: Instead of lecturing to the whole class three hours per week, the class met once a week to address logistics and common issues and an hour mandatory meeting with the instructor was scheduled with each team to report on progress, obstacles, and work plans for the next week. Consequently, the students were more engaged and prepared for their meetings with the teacher.

Team projects assigned for the MMS and DMV courses are presented here to illustrate how course objectives were met. In the last month of a semester, the emphasis shifted to team projects. The intention of projects was to cultivate students' ability to solve real-world problems. Project problems and grading rubrics are similar to those of the Mathematical Contest in Modeling (MCM, <https://www.comap.com/>) sponsored by COMAP (Consortium for Mathematics and its Applications). After the 10th week of the semester, the instructor provided two or more open-ended problems for their team projects. One is a continuous modeling problem and the other is a discrete modeling problem. In particular, the rubrics of the project grading included: 20% paper presentation, 10% for oral presentation, 20% for conceptual and math modeling, 10% for simulation, 20% for mathematical analysis, and 20% for verification and validation of the model.

One MMS problem was to model and simulate a safe landing gear for aero robots. Students had two weeks to build conceptual models that captured the major factors and their qualitative relationships. The conceptual model required students to answer

the four questions mentioned listed above. A month before the end of the course, teams of 4 students were assigned to build the mathematical model and simulate for the problem using Stella. The won team of students had the opportunity to participate in a paid summer research workshop and conduct the experiments in ERAU's Wave Lab (see Figure 1, safe landing gear for weather balloon problem). An example of discrete modeling is students in MMS were asked to model and simulate the emergent evacuation of the vulnerable residents in a coastal city when a tsunami hits the city. The basic geographic information such as the elevation and the population density of each city district were given in a data sheet. The territories of the districts are approximated as grids of uniform sizes and the streets are horizontal and vertically lined up to separate districts. The students were given 40 school buses and asked to select only one exclusively used street from east to west to evacuate the vulnerable residents who could not drive or had no cars. Two teams chose this project and both used the agent-based modeling approach for the problem. One team wrote their own Java program, and the other team chose to use NetLogo to model and simulate their solutions. While EXCELL, MATLAB, Stella and NetLogo were frequently used in case studies and students learned how to read the programs and diagrams, the instructor did not teach students how to program in those languages and tools. Instead, free online tutorials were provided to students so they could learn as needed. Teams were assigned based on complementary knowledge and skills so that each team had at least one student proficient in programming. An observable outcome was that students often demonstrated their ability of learning-on-demand when inspired by interesting problems ([6]). For example, one student who used NetLogo to program their evacuation model learned how to use NetLogo from scratch and did an excellent job in less than a month. He is continuing his project and intends to submit a paper for publication.

The DMV (<http://datamininedvis.wordpress.com/>) instructional design and course delivery strategy are similar to that of the MMS. Since DMV is a course that is offered by most universities, the content selection was less challenging than the MMS course. Two aspects that made our DMV course differently were the team projects: (1) The teams sometimes had to collaborate with students in a remote campus and (2) some projects have an industrial co-mentor in addition to the instructor. We also allocated the last month of the course for the team projects. The instructor selected the won team project and encouraged the students to continue the project in order to earn internship or job opportunities. Of particular interests are two projects that were successfully completed in spring 2015 and spring 2016. The first project involved data mining for profiling incoming students and predicting student retention rate based on the authentic training dataset provided by the Office of Student Success and Retention at ERAU. A team of four students from ERAU Prescott selected the project and did an excellent job when they took the course offered at the Daytona Beach campus in spring term 2015. In middle May of 2015, the four students were invited to Daytona Beach in the summer research workshop and they were co-mentored by Liu and Mr. Steven Lehr who had over 15 years of industrial data analytics experiences. After the research outcomes were presented to administrators at Office of Student Success and Retention, they organized a committee to investigate the impact of gateway courses such as pre-calculus or calculus to the student retentions in 2016. The research results [14] was published in an IEEE conference proceeding in 2016. The second project was assigned to two students as a team project in spring 2016. The

project was to use the biomass data collected around the artificial reefs provided by the Beach Safety Office of Volusia County to predict the healthiness and effectiveness of the artificial reefs towards to marine ecosystem preservation. The team continued to improve the data mining results in the summer with support from the Prepare Industrial Career Math mini-grant. The project was co-mentored by Liu and a staff member at the Beach Safety Office. The Project was presented as a poster at the SIAM Annual Conference in 2016. Because of this project, one student found an internship opportunity and part time job in Volusia County government. In summary, the strategies we used to engage student learning were (1) teach mathematical concepts using relevant application contexts and (2) provide team research projects that will facilitate the development of marketable problem-solving skills.

3. OUTCOMES

As the project progressed, the following new educational resources were created: (1) Multimedia MOOC materials for three courses in Mathematical Modeling and Data Mining, (2) Documents for formative and summative learning assessments; and (3) Five peer-reviewed research papers coauthored by the undergraduate participants of three summer research workshops from 2013-2015.

Each college contributed one faculty member and one course resulting in each university in the coalition having three new courses. All courses were thoroughly reviewed by peer instructors and upgraded multiple times providing each university three high quality innovative cyberlearning courses. The students in each campus have more course choices while the campuses save costs by not having faculty members teaching extra courses: We simply shared resources and pooled students to make the class size reach the ideal number of students. The feedback collected formally and informally from students indicated that they benefited from the multimedia course website, which allowed them to learn the material at their own pace. They read textual materials for the theory and concepts and watched the videos for the examples. Three course websites were built using WordPress technology and contributed to the MOOC.

Three two weeks workshops were offered for a total of 18 students from three universities in summer 2013-2015. Five students in ASU and Prescott campus were funded by the NSF grant to participate in the summer workshops at Daytona Beach. All students started their mandatory team projects six weeks before they completed their CSE courses in spring semesters. The focus of course-based research experiences (CURE) is on the modeling, computation, simulation and analysis of open-ended application problems. All 18 students chose to continue their research projects in their CSE courses but shifted their focus towards model validation based on authentic data. The three teams of students used the ACE methodology [5] to model and analyze physics applications such as safe landing gears of weather balloon payload, buoy motions and underwater light scattering patterns in a Wave Tank. They obtained their data from hands-on experiments in the Wave Lab and the Leverage Robotics Lab. A team of five students from the DMV course used data from the ERAU admission and registration office to predict student retention and attrition. The course-based researcher and summer research workshops resulted in four student co-authored papers [6, 8, 9, and 17], three conference presentations, and two conference presentations. Figures 1-4 shows the five student coauthors of [9] from ASU designing and testing a safe landing system for weather balloon payload in summer 2015.

Because students of the cyberlearning courses represented more diverse STEM majors than the typical students at any single campus and course assessment approaches were quite different, there was not a proper control group to compare learning outcomes quantitatively. In addition, class sizes were between 12-20 students from three campuses and consequently sample sizes were too small to draw statistically significant conclusions. Suggested by our external evaluator Spector, we compared the learning outcomes of MMS with an Ordinary Differential Equation Course (ODE, MA345 in ERAU), which was one of most similar courses taught by the same instructor of the MMS course at the same time. The two courses were similar in these aspects, (1) same prerequisite, (2) similar contents, matrices algebra and system ODE, (3) similar small class sizes (17 vs 29 students). Besides the delivery method (MMS used cyberlearning & blended learning, ODE used traditional face-to-face lectures), the other major differences between MMS and ODE are: (1) ODE students were more homogeneously from physics and engineering majors and the students in MMS came from more diversified STEM majors including biology and meteorology majors, (2) MMS is a problem-based learning course with model-based instructional design and assessments, query guided learning, teamwork, etc., the ODE course was taught and tested in conventional methods, and (3) the focus of the MMS course was on the depth of their understanding of the key concepts and problem solving ability with the use of computational tools while students in ODE course covered a broad range of content and solved several types of ODEs by hand. Comparing the student evaluations of the two courses showed that the students in MMS were much more motivated. They gained more confidence from the MMS course to solve relevant problems to their careers by using system engineering methodology and computational tools. Grades of ODE in spring 2014, 7 A's, 7 B's, 10 C's, 2 F's, and 3 W's, which is a typical grade distribution at ERAU. The grades awarded for MMS in the same semester were 6 A's, 6 B's and 4 C's, and 1 W. Though the grades of the courses might not tell how much the students really learned, we believe that low attrition and failure rate of the classes is a good indicator of the course success. We offered DMV three times, the first time by IKle, the second time by Liu, and third time by Liu and Lehr. Not a single student did withdraw from the class. This is very rare for any other mathematics course: The typical withdrawal rate of similar level mathematics courses are 10-20% at ERAU. Our student feedback showed that most of them loved the courses because of the relevant and marketable skills they gained for industrial careers. The withdrawal (W) or failure (D or F) rates of the three MMS courses were very low: one to two students in each course.

Spector examined all the student evaluations reports presented in section 2.4 and also evaluated online course materials, sample student test papers, project reports, etc. for each course. He communicated with the instructors frequently so that the student concerns were addressed timely and the project efforts were aligned with the major goal of these courses: fostering student problem-solving ability and promoting deep learning. As a result, the students in MMS courses learned how to use the system engineering modeling methodology and procedures to translate real-world problems into mathematical models. Students also gained hands-on experience in using software tools such as MATLAB and STELLA to model and simulate real-world projects. A set of sample projects of the MMV-II indicates that students developed deep learning including: (1) A Model of the Rings of Saturn illustrating the appearance of the Cassini Division for certain parameters of the shepherd moon(s) and (2) Writing a

game of "Tanks" in MATLAB. The students especially enjoyed the team projects on the application of authentic weather data, genetic data, public health data, real-estate data, and student retention data.

Although the student evaluation metadata in section 2.4 had limited value in helping us to accurately measure our success, the numerous positive feedbacks including the constructive criticisms clearly indicated that all three courses were effectively delivered and improved in the second and third round of offerings. Many of these student comments were included in the final project evaluation reported by the external evaluator to NSF. In the final project evaluation to NSF, Spector presented the following evaluation summary:

"In terms of the three primary goals of this project (establish an initial cluster of three collaborating colleges; create and offer a computational science and engineering program at collaborating colleges; create the infrastructure and processes to extend the collaboration cluster), this program has been successful and all three goals have been achieved. Problems encountered have been addressed and refinements made. Students taking the collaboration courses at a distance are performing as well as students in comparable classes as indicated by the grades awarded. Understanding of complex computational engineering problems is occurring as shown by an analysis of problem conceptualizations and solutions previously reported. Interest is especially high as shown by voluntary participation in the two summer workshops, which should be better funded if such non-formal but highly productive efforts are to be continued."



Figure 1. Students are preparing the safe landing system

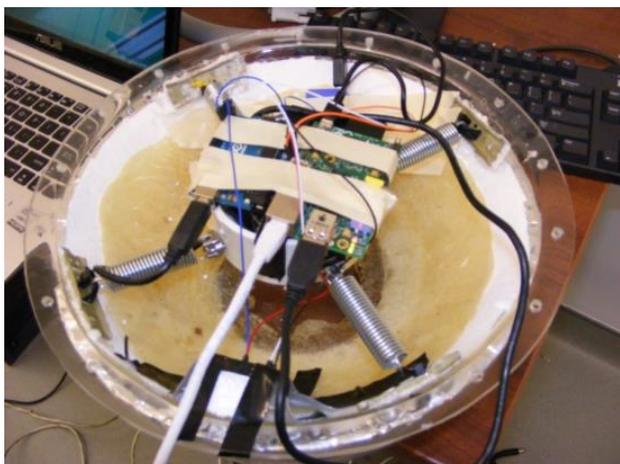


Figure 2. Sensor and Arduino board of the safe landing gear



Figure 3. Prepare test for landing on grasses



Figure 4. Captured the moment for landing on water

4. BROADER IMPACT AND FUTURE WORK

This project provided a feasible model to significantly increase student participation in CSE. Few small colleges have the resources to provide CSE courses and programs for undergraduate students. The project also demonstrated a viable method for scaling up: Adding more colleges to a cluster and then creating a network of clusters. Our strategies to advance CSE education can be straightforwardly extended to other disciplinary domains and other small universities.

In addition to the MOOC websites, this project resulted in the five publications [6, 8, 9, 15, and 17] (undergraduate coauthors are marked by asterisks). PIs conducted the following personal dissemination events: Liu and A. Shiflet co-chaired a miniSymposium on Educational Innovations in CSE Education in the SIAM Annual Conference, July 2014 in Chicago. Liu and Ludu Co-Chaired a miniSymposium in SIAM SEA Conference 2014 in Melbourne, FL, 2014. Spector and Liu Co-Chaired a miniSymposium on Cyberlearning Technology and Deep Learning in a CSE Conference in March 2015 in Salt Lake City, Utah. In addition, three students gave a presentation at the 2015 Kappa Mu Epsilon National Convention.

The social benefit to students included providing equal learning opportunities to under-represented students: Adams State University is a federally designated Hispanic Serving Institution (HSI). The summer research workshops brought minority students from Adams State and students from the Prescott campus in Arizona to meet the students from the Daytona Beach campus. These events not only facilitated student collaboration on research, but also helped them build friendships and develop the mutual understanding of other students from different cultural and socioeconomic backgrounds. In summary, the project is sustainable and scalable because it benefits small universities and provides a cost effective solution to advance CSE education.

As a continuation of this project, we have undertaken a new project that included more institutions, more STEM disciplines, and more students. In addition to ASU and ERAU, the newly funded IUSE project created a coalition that included two more institutional partners—Hampden-Sydney College (HSC) in Virginia and Bethune-Cookman University (BCU), a Historic Black University (HBCU) in Florida. The four small universities are collaborating to integrate CDSE (Computational Data-enabled Sciences and Engineering) coursework into the undergraduate curriculum and embed authentic research experiences based on a CURE pedagogical model. Four new courses were added: (a) Database Design, (b) Genomics and Bioinformatics, (c) Problems in Atmosphere and Hydrosphere, and (d) Advanced Computing Resources in Biology. Except for the Database course, the other three new courses were organized by applications instead of computational methods. We are also creating of a virtual educational observatory to ensure that the effort will be ongoing after project funding ends and consequently have an even broader impact, especially on small, regional, and minority-serving institutions. The continuous project will help us collect adequate survey data so that we can draw statistically significant conclusions using evidence based and data driven evaluations. For example, we have added more questions in the end of term evaluation so that we can compare changes in student beliefs about CSE and team work after they have completed the course. The end of term evaluation will also ask each student to rank his or her perceived importance and difficulty level of the course contents. In addition, the new project will adopt new technologies (e.g. educational data mining, learning analytics), use proven instructional approaches (e.g., experiential, problem-centered, collaborative learning), and integrate them into a flexible curriculum involving the co-creation of resources and activities by participating partners.

5. ACKNOWLEDGMENTS

The project for CSE education and summer REU workshops presented in this paper was sponsored by the National Science Foundation (NSF) of United States under the program of TUES Grant 1244967 from 2013-2016. The authors would like also give

thanks for the NSF of United States for the continuous support for the following-up project under the IUSE Grant 1626602 from 2017-2019.

6. REFERENCES

- [1] Borgman, C.L. (2008) *Fostering Learning in the Networked World: The Cyberlearning Opportunity and Challenge*. Report of the NSF Task Force on Cyberlearning.
- [2] Borne, K., Wallin, J. & Weigel, R. (2009), *The New Computational and Data Sciences Undergraduate Program at George Mason University ICCS 2009 Proceedings of the 9th International Conference on Computational Science*. 74-83.
- [3] Kurt Byran and Tanya Leise, (2006), *The \$25,000,000,000* Eigenvector the linear algebra behind Google*, www.rose-hulman.edu/~bryan/googleFinalVersionFixed.pdf.
- [4] Cognitive and Technology Group at Vanderbilt University, 1992. *The Jasper series as an example of anchored instruction: Theory, program description, and assessment data*. *Educational Psychologist* 27:291–315.
- [5] Liu, L. & Ludu, A. (2012) *ACE - A Model Centered REU Program Standing on the Three Legs of CSE: Analysis, Computation and Experiment*. *Proceeding of ICCS2012*, Omaha, NE.
- [6] Liu, H. Klein, J. (2013), *Using REU Project and Crowdsourcing to Facilitate Learning on Demand*, *Proceedings of IADIS International Conference on Cognition and Explorative Learning at Digit Ages*, Fort Worth, Texas, pp 251-258.
- [7] Hong Liu, Jayathi Raghavan, *A Mathematical Modeling Module with System Engineering Approach for Teaching Undergraduate Students to Conquer Complexity*, *The Proceedings of Conference ICCS 09 , Part II, LNCS5545*, pp 93-102, 2009.
- [8] Liu, H., Shi, X., Shao, J., Zhou, Q., Joseph-Ellison, S., Jaworski *, J., and Wen, C *, *The Mechatronic System of Eco-Dolphin - a Fleet of Autonomous Underwater Vehicles*, *Proceeding of International Conference of Advanced Mechatronics System 2015*, Beijing, August 24-27, 2015.
- [9] Lyons, T*, Ginther, M*. Mascarenas, P*. Rickard, E. Robinson*, J. Braeger*, J, Liu, H., and Ludu A. 2014, *Nonlinear Decelerator for Payloads in Aerial Delivery Systems. I: Design and Testing*, *Nonlinear Dynamics*, retrievable from <http://arxiv.org/abs/1408.4190>.
- [10] Pratap Misra, Per Enge, (2004), *Global Positioning System, Signals, Measurements, and Performance*, Ganga-Jamuna Press.
- [11] Pirnay-Dummer, P., Ifenthaler, D., & Spector, J. M. (2010), *Highly integrated model assessment technology and tools*, *Educational Technology Research & Development*, 58(1), 3-18.
- [12] Shiflet, Angela and George, (2013) Vol. 4, Issue 1, *Journal of Computational Science Education*, pp 11-15, http://www.shodor.org/media/content/jocse/volume4/issue1/hiflet_2013.
- [13] Society of Industrial and Applied Mathematics, *Modeling Across the Curriculum*, Report on a SIAM-NSF Workshop*, Arlington, Virginia, August 30–31, 2012 https://www.siam.org/reports/modeling_12.pdf.
- [14] Society for Industrial and Applied Mathematics. 2015. *Modeling across the Curriculum II: Report on the 2nd SIAM-NSF workshop*. Philadelphia, PA: Society for Industrial and Applied Mathematics.
- [15] Simpson, A. Ludu, H. J. Cho and H. Liu, 2014, *Experimental and theoretical studies on visible light attenuation in water, Atmospheric and Oceanic Physics*, retrievable from <http://arxiv.org/abs/1408.3883> .
- [16] Spector, M. (2010) *Assessing Progress of Learning in Complex Domains*. *The 11th International Conference on Education Research: New Educational Paradigm for Learning and Instruction*.
- [17] Steven Lehr, Hong Liu, Sean Klinglesmith*, Alex Konyha *, Natalia Robaszewska *, Jacob Medinilla *, *Use Educational Data Mining to Predict Undergraduate Retention*, Submitted to the 16th IEEE International Conference on Advanced Learning Technologies - ICALT2016, Austin, Texas.
- [18] *Transforming Post-Secondary Education in Mathematics (TPSE Math)*, Report of a Workshop Joint Mathematics, Meetings, San Antonio, Texas, January 2015,
- [19] Turner, P. & Petzold, L. (2011) *Undergraduate Computational Science and Engineering Education*, *SIAM Review*, 53 (3), 561–574.
- [20] Turner, P. Levy, R., and Fowler, K., *Collaboration in the Mathematical Sciences Community on Mathematical Modeling Across the Curriculum*, *Chance, Using Data to Advance Science, Education, and Society*, <http://chance.amstat.org/2015/11/math-modeling>.

Computational approaches to scattering by microspheres

Reed M. Hodges
Georgia Southern University
P.O. Box 8031
Statesboro, GA 30460
rh04272@georgiasouthern.edu

Kelvin Rosado-Ayala
Georgia Southern University
P.O. Box 8031
Statesboro, GA 30460
kr04537@georgiasouthern.edu

Maxim Durach^{*}
Georgia Southern University
P.O. Box 8031
Statesboro, GA 30460
mdurach@georgiasouthern.edu

ABSTRACT

Mie theory is used to model the scattering off of wavelength-sized microspheres. It has numerous applications for many different geometries of spheres. The calculations of the electromagnetic fields involve large sums over vector spherical harmonics. Thus, the simple task of calculating the fields, along with additional analytical tools such as cross sections and intensities, require large summations that are conducive to high performance computing. In this paper, we derive Mie theory from first principles, and detail the process and results of programming Mie theory physics in Fortran 95. We describe the theoretical background specific to the microspheres in our system and the procedure of translating functions to Fortran. We then outline the process of optimizing the code and parallelizing various functions, comparing efficiencies and runtimes. The shorter runtimes of the Fortran functions are then compared to their corresponding functions in Wolfram Mathematica. Fortran has shorter runtimes than Mathematica by between one and four orders of magnitude for our code. Parallelization further reduces the runtimes of the Fortran code for large jobs. Finally, various plots and data related to scattering by dielectric spheres are presented.

Keywords

Parallel computing, scattering, Mie theory, microspheres, Fortran, Mathematica

1. INTRODUCTION

Scattering by wavelength-sized microspheres involves a solution to Maxwell's equations derived by Gustav Mie and published in 1908. Several prominent books, such as those by Stratton [8] and Bohren & Huffman [1], provide a succinct derivation of Mie theory. But these derivations rely heavily on previous knowledge of the material and leave out

^{*}Corresponding author

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.

DOI: <https://doi.org/10.22369/issn.2153-4136/8/3/3>

many details. In this paper, we derive the theory from first principles and explicitly state most of the steps involved.

Microspheres exhibit many interesting photonic and plasmonic properties. When composed of pure dielectric and with radii on the order of the incident wavelength, they produce photonic nanojets in the shadow region [2]. These nanojets are regions of greatly increased intensity, and are reminiscent of a lensing effect in ray optics. Dielectric spheres can also be used in the design of efficient optical antennas [3], enhance two-photon fluorescence [6], and exhibit optical coupling and transport [2].

Wavelength-sized spheres have uses outside of the pure-dielectric regime as well. Metal microspheres can exhibit a plasmonic response at the interface [7]. Additionally, other effects can be produced as the geometries of the microspheres are altered. Chiral dielectric spheres can add angular momentum to the photonic nanojets [5]. Alternating layers of gold-dielectric concentric spheres can be designed to exhibit optical neutrality, or invisibility [4]. The many applications of the theory make modeling of electromagnetic scattering by microspheres with efficient code a beneficial endeavor.

Wolfram Mathematica and Fortran are two commonly used programming languages for computational physics. Mathematica has the advantage of being a simple-to-use symbolic language, with a small learning curve and a plethora of built-in functions. Fortran is more low-level, making it more difficult to learn and use but generally superior in speed. Here we outline the contents of the Fortran library developed specifically for Mie scattering by dielectric spheres, and compare the results with Mathematica code of the same purpose.

2. MIE THEORY AND CROSS SECTIONS

The solution begins by defining solutions to the vector wave equations, \mathbf{M} and \mathbf{N} [8].

$$\mathbf{M} = \nabla \times (\mathbf{r}\psi) \quad (1)$$

$$\mathbf{N} = \frac{1}{k} \nabla \times \mathbf{M} \quad (2)$$

The form of these spherical vector wave functions (SVWFs) is found with a scalar potential function ψ , which is a solution to the scalar wave equation. This equation assumes a time dependence of $\exp(-i\omega t)$.

$$\nabla^2 \psi + k_{ef}^2 \psi = 0 \quad (3)$$

It is solvable with separation of variables, yielding a solution

of the form:

$$\psi_{lm}^{(j)}(r, \theta, \phi) = (-1)^m z_l^{(j)}(kr) P_l^m(\cos \theta) e^{im\phi} \quad (4)$$

$$\psi_{elm}^{(j)}(r, \theta, \phi) = (-1)^m z_l^{(j)}(k_t r) P_l^m(\cos \theta) e^{im\phi} \quad (5)$$

Where $z_l^{(j)}(kr)$ are either spherical Bessel functions of the first kind ($j = 1$) or spherical Hankel functions of the first kind ($j = 3$), and $P_l^m(\cos \theta)$ are associated Legendre polynomials. The second form of this potential (Eqn. 5) is inside the sphere, with $k_t = k\sqrt{\epsilon_{sph}}$.

Performing the differentiation yields explicit forms for the SVWFs.

$$\mathbf{M}_{lm}^{(j)} = \frac{im}{\sin \theta} \psi_{lm}^{(j)} \hat{\mathbf{e}}_\theta - \frac{\partial \psi_{lm}^{(j)}}{\partial \theta} \hat{\mathbf{e}}_\phi \quad (6)$$

$$\mathbf{N}_{lm}^{(j)} = \frac{1}{k} \nabla \times \mathbf{M}_{lm}^{(j)} = \frac{l(l+1)}{kr} \psi_{lm}^{(j)} \hat{\mathbf{e}}_r + \frac{1}{kr} \frac{\partial^2}{\partial r \partial \theta} (r \psi_{lm}^{(j)}) \hat{\mathbf{e}}_\theta + \frac{1}{kr} \frac{im}{\sin \theta} \frac{\partial}{\partial r} (r \psi_{lm}^{(j)}) \hat{\mathbf{e}}_\phi \quad (7)$$

The electromagnetic field is then expressed as infinite sums of these functions, with the appropriate Mie coefficients. The field is broken down into three components: the incident field \mathbf{E}_{inc} , the scattered field \mathbf{E}_{sca} , and the field inside the dielectric sphere \mathbf{E}_{sph} .

$$\mathbf{E}_{inc} = \sum_{l=1}^{\infty} \sum_{m=-l}^l \left(p_{lm} \mathbf{M}_{lm}^{(1)} + q_{lm} \mathbf{N}_{lm}^{(1)} \right) \quad (8)$$

$$\mathbf{E}_{sca} = \sum_{l=1}^{\infty} \sum_{m=-l}^l \left(a_{lm} \mathbf{N}_{lm}^{(3)} + b_{lm} \mathbf{M}_{lm}^{(3)} \right) \quad (9)$$

$$\mathbf{E}_{sph} = \sum_{l=1}^{\infty} \sum_{m=-l}^l \left(c_{lm} \mathbf{N}_{elm}^{(1)} + d_{lm} \mathbf{M}_{olm}^{(1)} \right) \quad (10)$$

The magnetic fields are found by taking the curls of these expressions.

The Mie coefficients are determined with the boundary conditions that the tangential components of the electric fields be continuous at the interface, $r = a$, where a is the radius of the sphere. These boundary conditions arise from applying Stokes's theorem to irrotational fields. One such coefficient is:

$$a_{lm} = -p_{lm} \frac{j_l(k_0 r) \frac{1}{\epsilon_{sph}} \frac{\partial}{\partial r} [r j_l(k_t r)] - j_l(k_t r) \frac{\partial}{\partial r} [r j_l(k_0 r)]}{h_l^{(1)}(k_0 r) \frac{1}{\epsilon_{sph}} \frac{\partial}{\partial r} [r j_l(k_t r)] - j_l(k_t r) \frac{\partial}{\partial r} [r h_l^{(1)}(k_0 r)]} \quad (11)$$

The scattering and extinction cross sections (σ_{sca} and σ_{ext}) measure how much power is taken out of the incident wave. The scattering cross section only includes power loss due to scattering, while the extinction cross section also includes absorption.

$$\sigma_{sca} = \frac{2\pi}{k^2} Re \left[\sum_{l=1}^{\infty} (|a_l|^2 + |b_l|^2) \frac{2l(l+1)(l-1)!}{(2l+1)(l+1)!} \right] \quad (12)$$

$$\sigma_{ext} = \frac{\pi}{k^2} Re \left[\sum_{l=1}^{\infty} (p_l a_l^* + q_l b_l^* + b_l q_l^* + a_l p_l^*) \frac{2l(l+1)(l-1)!}{(2l+1)(l+1)!} \right] \quad (13)$$

A more explicit form of this derivation can be found in the Supplementary Materials.

3. FORTRAN MATHEMATICS LIBRARY

Once the analytical calculations for our project were completed, the challenge was to translate the various functions into code for numerical computation. The code to plot scattering by microspheres was written in Wolfram Mathematica, but Fortran was needed to improve speed and to be run on the Blue Waters supercomputer at the National Center for Supercomputing Applications. Fortran 95 was chosen as it is user-friendly while still being efficient.

The functions needed include the spherical Bessel functions, spherical Hankel functions, associated Legendre polynomials, and some miscellaneous functions. Few Fortran functions exist online that meet the needs: they either calculate regular Bessel functions but not spherical ones, calculate regular Legendre polynomials but not associated ones, can only be used up to $l = 10$, etc. Also, the desired precision was six-digit precision, to match the default precision of Wolfram Mathematica, and many of the functions found on the Internet lacked this. Recurrence relations or numerical solutions to differential equations can be used to approximate values for these functions, but these options involve recursion and thus are inefficient. Our library explicitly defines the spherical Bessel functions and associated Legendre polynomials up to the needed limits. Doing so comes at the cost of additional overhead due to the explicit writing of the table, but provides an efficient function as it only needs to look in the table for the appropriate formula and plug in the variables, as opposed to having to generate the formula dynamically. However, this also comes with the drawback of only being able to calculate a value up to a certain l , namely $l = 30$.

The library includes many other functions which simply called the Bessel and Legendre functions. Also, it has several miscellaneous functions and subroutines, such as a Kronecker delta function and a subroutine to convert between rectangular and spherical coordinates. The library is optimized to the extent of removing extraneous variables and combing loops to reduce the total number of iterations, and using Gfortran compilation options and constraints. These changes helped solve the initial segmentation faults and inaccuracies for low values of l or high values of kr . The library was tested for accuracy by comparing the results to the calculations performed by the Mathematica library. It is valid for values of l ranging from 1 to 30 and values of kr up to about 100. This is reasonable for the purposes of this project, since the values of the SVWFs are negligible for high l and the spheres considered are all wavelength-sized.

The completed library is used to calculate the electromagnetic fields and cross sections, and is available publicly on GitHub (<https://goo.gl/aRyScF>). See Algorithm 1 for the pseudocode used to calculate the fields from the functions in the library.

4. EFFICIENCY, PARALLELIZATION, AND RUNTIMES

Since several of the Fortran functions were written from scratch for the purposes of this project, it is useful to analyze their efficiencies via Big-O notation. The various unique functions and their efficiencies are summarized in Table 1. Four of the more rudimentary functions are either constant or linear in efficiency, which is advantageous because they are used in almost every other calculation. The rest of the

Algorithm 1: Calculation of electromagnetic fields

Input: Geometrical parameters and physical properties of the sphere, and physical properties of the incident light.

Output: A file containing position coordinates and intensities.

```

x = xmin;
z = zmin = xmin ;
for each x do
  for each z do
    initialize spherical coordinates ;
    initialize electromagnetic field vectors ;
    for each m value do
      for each l value do
        calculate all vector spherical wave
          functions ;
        add field term to net sum ;
      end
    end
    intensity(x,z) = sum of incident, scattered, and
      inside waves ;
    write in output file x, z, intensity(x,z)
  end
end
intensity(0,0) = 0 ;

```

Table 1: Fortran function complexities

FUNCTION	EFFICIENCY
δ_{lm}	$\mathcal{O}(1)$
$\tau_f(l, m), \pi_f(l, m)$	$\mathcal{O}(1)$
$n!$	$\mathcal{O}(n)$
q_{lm}, p_{lm}	$\mathcal{O}(m)$
$z_l^{(j)}(kr)$	$\mathcal{O}(l^2)$
$\partial_r z_l^{(j)}(k, r)$	$\mathcal{O}(l^2)$
$a_{lm}, b_{lm}, c_{lm}, d_{lm}$	$\mathcal{O}(l^2)$
$\psi_{lm}^{(j)}$	$\mathcal{O}(n^2)$
$\partial_\theta \psi_{lm}^{(j)}, \partial_r(r\psi_{lm}^{(j)}), \partial_{r\theta}^2(r\psi_{lm}^{(j)})$	$\mathcal{O}(n^2)$
$P_l^m(x)$	$\mathcal{O}(l+m) \approx \mathcal{O}(n^2)$

functions had quadratic efficiency.

All of these functions were previously programmed in Wolfram Mathematica, a symbolic computation program. Fortran is orders of magnitude faster than Mathematica in computing the same data. Here we compare the mean runtimes of 30 trials between the two languages for the same calculations on the same computer. Table 2 shows the means, standard deviations, and percent differences between Fortran and Mathematica. Each of the more basic mathematical functions, $j_l(kr)$, $h_l(kr)$, and $P_l^m(\cos\theta)$, were around two orders of magnitude faster in Fortran than in Mathematica over calculations of 50,000 values of kr . The spherical Hankel function is comparatively slower in Fortran at only 218.4% faster, likely because it has to call the two types of spherical Bessel functions. The true speed of Fortran is revealed when the larger functions $\mathbf{M}_{lm}^{(1)}$ and $\mathbf{N}_{lm}^{(1)}$ are investigated. They are four orders of magnitude faster than Mathematica over calculations for 1200 values of k . Thus, it is clear that our Fortran library for Mie theory has runtimes less than our Mathematica library.

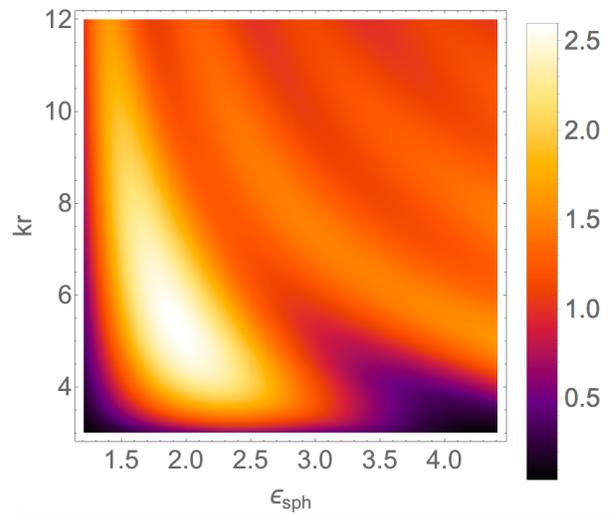


Figure 1: A plot of the scattering cross section as a function of kr and ϵ_{sph}

Parallelization further increases the benefits of coding Mie theory in Fortran for use on large computers like Blue Waters at the National Center for Supercomputing Applications or the Talon cluster at Georgia Southern University. The cross sections and intensities then can be calculated for many different permittivities, wavelengths, or sphere radii simultaneously by executing the do loops in parallel. We parallelized a program calculating the scattering cross section for many different values of ϵ_{sph} and kr , using the application programming interface OpenMP. The ϵ_{sph} do loop was divided to be worked on in 32 threads, and the runtime of the program was greatly reduced: the data contained in Fig. 1 can be generated in 15 seconds, while the same program requires 158 seconds in serial.

5. RESULTS

The developed Fortran library can be used to generate data related to scattering by dielectric microspheres. The Fortran outputs the data into comma separated variables files, which can then be quickly visualized in Mathematica. For example, Fig. 1 shows a density plot of the scattering cross section as a function of kr and ϵ_{sph} .

Graphs like in Fig. 1 reveal where there is heavy scattering by a sphere for a particular incident wavelength. Most of the forward scattering comes in the form of a photonic nanojet, a region of increased intensity on the shadow side of the sphere. Plots of the electric field intensity reveal this jet; Fig. 2 shows such plots for spheres of radius 250 nm and 400 nm, with incident wavelength 700 nm. The plots reveal that the photonic nanojet has a greater intensity and larger relative size when the sphere radius is closer to the order of the incident wavelength. This is corroborated by Fig. 1, which showed that the scattering cross section reaches a maximum near $kr \approx 6$.

6. DISCUSSION AND CONCLUSIONS

In this paper we outlined the fundamentals of Mie theory, the physical framework for electromagnetic scattering by wavelength-sized spheres that is derived from Maxwell's

Table 2: Comparison between Fortran and Mathematica

FUNCTION	FORT. MEAN (s)	FORT. ST. DEV. (s)	MATH. MEAN (s)	MATH. ST. DEV. (s)	% DIFF.
$j_l(kr)$	0.746267	0.0189863	5.66427	0.364201	759.9%
$h_l(kr)$	1.58893	0.13657	3.4708	0.158145	218.4%
$P_l^m(\cos\theta)$	2.07347	0.121895	14.5652	0.217958	702.5%
$M_{lm}^{(1)}$	0.310133	0.010061	45.9435	0.454602	14810%
$N_{lm}^{(1)}$	0.212133	0.00196053	68.748	1.03978	32410%

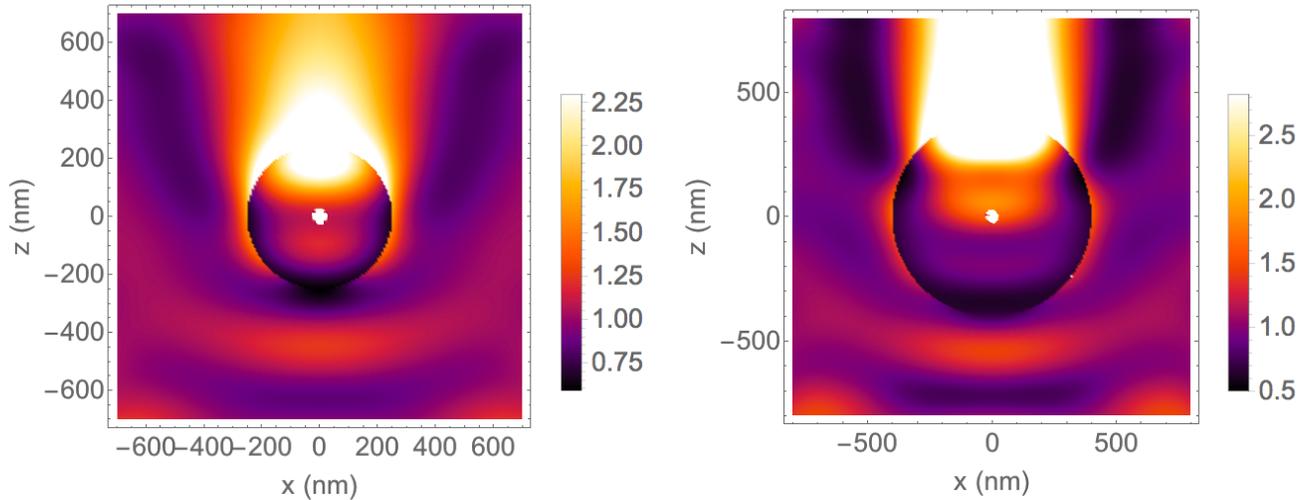


Figure 2: A plot of the electric field intensity, where $|E_0|^2 = 1$, for dielectric spheres of radii 250 nm (left) and 400 nm (right), displaying photonic nanojets. The incident wavelength is 700 nm and the effective permittivity is $\epsilon_{sph} = 1.33^2$.

equations. The solutions to the differential equations involve spherical Bessel functions of the first kind, spherical Hankel functions of the first kind, and associated Legendre polynomials. The electromagnetic fields themselves are large sums of these functions with the appropriate Mie coefficients. Also, the expressions for the scattering and extinction cross sections were derived, which also involve summations over l and provide a useful tool for analyzing the scattering.

We detailed our efforts to develop, code, and apply a Fortran mathematics library comprised of the necessary functions. Several problems related to the accuracy of the functions for certain extreme inputs were addressed. The final functions and subroutines had efficiencies no worse than $\mathcal{O}(n^2)$. The calculation of the electromagnetic field summations was achieved with nested for loops. These for loops were conducive to parallelization with OpenMP, which reduced the runtimes of certain jobs by about 10 times in our case. Furthermore, our serial Fortran code had runtimes orders of magnitude lower than the same tasks in our Mathematica code. Both of these facts reveal some of the benefits of using Fortran for Mie theory calculations rather than the more symbolic-based Mathematica.

Finally, we presented visualization of some of the data produced in Fortran. The plot of the scattering cross section as a function of the relative permittivity ϵ_{sph} and radial function argument kr informs where the maxima and minima of scattering occur. This helps optimize the characteristics of the photonic nanojets, plots of which were shown in Fig. 2.

They reveal that the relative intensity and size of the nanojet are optimized when the sphere diameter is on the order of the incident wavelength.

Although Fortran has proven to be much faster in performing these calculations compared to Mathematica, there are significant drawbacks which may limit its practicality. One drawback is that the upfront coding of the programs takes much longer in Fortran than in Mathematica. When coding the library in Fortran, we had to write functions to perform the vast majority of tasks from square one. In contrast, Mathematica has many of the needed functions predefined. The second drawback is in defining the functions. Writing Mathematica code is simpler than writing Fortran code, due to Mathematica's emphasis on symbolic computation. In Mathematica, one can just translate the mathematical formulae directly into code and be able to easily see what is written; in Fortran the code is less readable. The third drawback is that debugging in Fortran proved to be more difficult than in Mathematica. Unlike Mathematica, Fortran requires the coder to understand how the code works and manage the various underlying processes. One big example of this is in parallelization with OpenMP. Parallelization in Mathematica is as simple as adding the word "Parallel" in front of certain functions, whereas in Fortran one must include options in the compiler to enable OpenMP, call the appropriate additional functions in the code, and include the OpenMP directives. In addition, this does not protect against race conditions, so the programmer must manually adjust the code to ensure results obtained from

parallel execution are correct.

7. REFLECTIONS

R.H. completed this research project as part of the Blue Waters Student Internship Program with the Shodor Education Foundation, Inc. Below are some of his final reflections on the experience:

I began this internship opportunity with little prior experience with coding or computing. As such, it was challenging for me to learn how to program for scientific research, but the two-week workshop at the University of Illinois at Urbana-Champaign was undoubtedly helpful in teaching the skills and knowledge required. Not only did it teach me the fundamentals of high-performance computing, but it also taught me how to think about scientific problems from a computing perspective. This was invaluable to being successful in this project. For example, the errors encountered in coding the mathematics library were overcome by thinking of alternative approaches to calculating functions in Fortran.

As a whole, this project helped further my education in both computing and in physics. I learned how to use parallel computing to improve my coding through great practice. The background research into Mie theory also gave me foundational insights into electromagnetic theory and partial differential equations, which has helped me achieve a better understanding of topics covered in my undergraduate classes. Also, this project impacted my career outlook by further cementing my desire to research this subject in graduate school and beyond. I will use the skills gained through this experience for the remainder of my academic career.

APPENDIX

A. SUPPLEMENTAL MATERIALS

The general Mie theory solution begins with Maxwell's equations.

$$\nabla \cdot \mathbf{E} = 4\pi\rho \quad (14)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (15)$$

$$\nabla \times \mathbf{E} = -\frac{1}{c} \frac{\partial \mathbf{B}}{\partial t} \quad (16)$$

$$\nabla \times \mathbf{B} = \frac{4\pi}{c} \mathbf{J} + \frac{1}{c} \frac{\partial \mathbf{E}}{\partial t} \quad (17)$$

The vector wave equation is found by taking the curl of Eqn. 16.

$$\nabla \times (\nabla \times \mathbf{E}) = -\frac{1}{c} \frac{\partial}{\partial t} (\nabla \times \mathbf{B}) = -\frac{1}{c} \frac{\partial}{\partial t} \left(\frac{4\pi}{c} \mathbf{J} + \frac{1}{c} \frac{\partial \mathbf{E}}{\partial t} \right) \quad (18)$$

Plugging in Ohm's law, $\mathbf{J} = \sigma \mathbf{E}$, where σ is the conductivity of the medium, and the vector calculus identity $\nabla \times (\nabla \times \mathbf{E}) = -\nabla^2 \mathbf{E} + \nabla(\nabla \cdot \mathbf{E})$ we get:

$$-\nabla^2 \mathbf{E} + \nabla(\nabla \cdot \mathbf{E}) = -\frac{4\pi\sigma}{c^2} \frac{\partial \mathbf{E}}{\partial t} - \frac{1}{c^2} \frac{\partial^2 \mathbf{E}}{\partial t^2} \quad (19)$$

If we assume the volume charge density is homogeneous and the electric field has a time dependence of $e^{-i\omega t}$, we can write

$$\nabla^2 \mathbf{E} - \frac{4\pi\sigma}{c^2} (-i\omega \mathbf{E}) - \frac{1}{c^2} (-\omega^2 \mathbf{E}) = 0$$

$$\nabla^2 \mathbf{E} + \left(\frac{4\pi\sigma i\omega}{c^2} + \frac{\omega^2}{c^2} \right) \mathbf{E} = 0 \quad (20)$$

Defining the imaginary wave number as $k^2 = \frac{4\pi\sigma i\omega}{c^2} + \frac{\omega^2}{c^2}$, we have the vector wave equation.

$$\nabla^2 \mathbf{E} + k^2 \mathbf{E} = 0 \quad (21)$$

The same can be done for the magnetic field. These equations have three possible solutions [8]:

$$\mathbf{L} = \nabla \psi \quad (22)$$

$$\mathbf{M} = \nabla \times (\mathbf{r}\psi) \quad (23)$$

$$\mathbf{L} = \frac{1}{k} \nabla \times \mathbf{M} \quad (24)$$

Provided that the scalar function ψ satisfies the scalar wave equation.

$$\nabla^2 \psi + k^2 \psi = 0 \quad (25)$$

Solving for these functions yields the fields given in Eqns. 4-7. The incident plane wave coefficients are given as follows.

$$p_{lm} = -i^l \frac{2l+1}{l(l+1)} \frac{(l-m)!}{(l+m)!} [\tau_{lm}(\alpha) \sin \gamma + i\pi_{lm}(\alpha) \cos \gamma] \quad (26)$$

$$q_{lm} = i^l \frac{2l+1}{l(l+1)} \frac{(l-m)!}{(l+m)!} [\pi_{lm}(\alpha) \sin \gamma + i\tau_{lm}(\alpha) \cos \gamma] \quad (27)$$

Where

$$\tau_{lm}(\alpha) = -\frac{1}{\sin \alpha} P_l^m(\cos \alpha) \quad (28)$$

$$\pi_{lm}(\alpha) = -\frac{\partial}{\partial \alpha} P_l^m(\cos \alpha) \quad (29)$$

Next, the Mie coefficients are found using the boundary condition that the tangential component of the electric field be continuous at the surface of the sphere. This leads to four equations, one of which is derived below.

$$E_{inc,\theta,TE} + E_{sca,\theta,TE} = E_{sph,\theta,TE}$$

$$q_{lm} \mathbf{M}_{lm,\theta}^{(1)} + b_{lm} \mathbf{M}_{lm,\theta}^{(3)} = d_{lm} \mathbf{M}_{olm,\theta}^{(1)}$$

$$q_{lm} \left(\frac{im}{\sin \theta} \psi_{lm}^{(1)} \right) \Big|_{r=a} + b_{lm} \left(\frac{im}{\sin \theta} \psi_{lm}^{(3)} \right) \Big|_{r=a} = d_{lm} \left(\frac{im}{\sin \theta} \psi_{olm}^{(1)} \right) \Big|_{r=a}$$

$$q_{lm} j_l(k_0 a) + b_{lm} h_l^{(1)}(k_0 a) = d_{lm} j_l(k_t a) \quad (30)$$

Three other equations are found similarly, and they can be summarized with two matrix equations.

$$\begin{bmatrix} h_l^{(1)}(k_0 a) & -j_l(k_t a) \\ \partial_r(r h_l^{(1)}(k_0 r)) & -\epsilon_{sph}^{-1} \partial_r(r j_l(k_t r)) \end{bmatrix} \begin{bmatrix} a_{lm} \\ c_{lm} \end{bmatrix} = \begin{bmatrix} -p_{lm} j_l(k_0 a) \\ -p_{lm} \partial_r(r j_l(k_0 r)) \end{bmatrix} \quad (31)$$

$$\begin{bmatrix} h_l^{(1)}(k_0 a) & -j_l(k_t a) \\ \partial_r(r h_l^{(1)}(k_0 r)) & -\partial_r(r j_l(k_t r)) \end{bmatrix} \begin{bmatrix} b_{lm} \\ d_{lm} \end{bmatrix} = \begin{bmatrix} -q_{lm} j_l(k_0 a) \\ -q_{lm} \partial_r(r j_l(k_0 r)) \end{bmatrix} \quad (32)$$

The coefficients can then be solved for as a system of linear equations. The fields are then completely known and can be plotted.

The scattering and extinction cross sections are found by integrating the radial component of the time-averaged Poynting vector over the surface of the sphere.

$$\sigma_{ext} - \sigma_{sca} = \frac{8\pi}{c} \int S_r r^2 d\Omega = \int Re[\mathbf{E} \times \mathbf{H}^*]_r r^2 d\Omega \quad (33)$$

The evaluation of this the cross sections requires approximations of the spherical Bessel and spherical Hankel functions for large r .

$$j_l(kr) \approx \frac{1}{kr} \cos\left(kr - \frac{l+1}{2}\pi\right) \quad (34)$$

$$h_l^{(1)}(kr) \approx \frac{1}{kr} (-i)^{l+1} e^{ikr} \quad (35)$$

Also, for normal incidence and linear polarization, we set $\alpha = \gamma = 0$, and only include the $m = -1$ mode. Plugging in the fields as in Eqns. 8-10 and simplifying using the identity

$$\int_0^\pi \left(\frac{dP_l^m}{d\theta} \frac{dP_n^m}{d\theta} + \frac{1}{\sin^2\theta} P_l^m P_n^m \right) \sin\theta d\theta = \frac{2l(l+1)}{2l+1} \frac{(l+m)!}{(l-m)!} \delta_{ln} \quad (36)$$

This leads to the expressions for the scattering and extinction cross sections.

$$\sigma_{sca} = \frac{2\pi}{k^2} Re \left[\sum_{l=1}^{\infty} (|a_l|^2 + |b_l|^2) \frac{2l(l+1)(l-1)!}{(2l+1)(l+1)!} \right] \quad (37)$$

$$\sigma_{ext} = \frac{\pi}{k^2} Re \left[\sum_{l=1}^{\infty} (p_l a_l^* + q_l b_l^* + b_l q_l^* + a_l p_l^*) \frac{2l(l+1)(l-1)!}{(2l+1)(l+1)!} \right] \quad (38)$$

B. ACKNOWLEDGEMENTS

R.H. and M.D. are grateful for the support of the Shodor Education Foundation and the Blue Waters Student Internship Program, including the two-week petascale computing workshop at the University of Illinois at Urbana-Champaign.

C. REFERENCES

- [1] C. F. Bohren and D. R. Huffman. *Absorption and scattering of light by small particles*. John Wiley & Sons, 2008.
- [2] Z. Chen, A. Taflove, and V. Backman. Highly efficient optical coupling and transport phenomena in chains of dielectric microspheres. *Optics Letters*, 31(3):389–391, 2006.
- [3] A. Devilez, N. Bonod, and B. Stout. Compact metallo-dielectric optical antenna for ultra directional and enhanced radiative emission. *arXiv preprint arXiv:1003.4880*, 2010.
- [4] R. Hodges, C. Dean, and M. Durach. Optical neutrality: invisibility without cloaking. *Optics Letters*, 42(4):691–694, 2017.
- [5] R. Hodges and M. Durach. Angular momentum in photonic nanojets due to chirality. In *COUR Symposium 2017*, Statesboro, GA, Apr. 13 2017.
- [6] S. Lecler, S. Haacke, N. Lecong, O. Crégut, J.-L. Rehspringer, and C. Hirlimann. Photonic jet driven non-linear optics: example of two-photon fluorescence enhancement by dielectric microspheres. *Optics Express*, 15(8):4935–4942, 2007.

- [7] S. A. Maier. *Plasmonics: fundamentals and applications*. Springer Science & Business Media, 2007.
- [8] J. A. Stratton. *Electromagnetic theory*. John Wiley & Sons, 2007.

Toward simulating Black Widow binaries with CASTRO

Platon I. Karpov^{* †}
 Department of Physics and
 Astronomy
 Stony Brook University
 Stony Brook, New York 11794
 plkarpov@ucsc.edu

Maria Barrios Sazo
 Department of Physics and
 Astronomy
 Stony Brook University
 Stony Brook, New York 11794
 maria.barriossazo
 @stonybrook.edu

Michael Zingale
 Department of Physics and
 Astronomy
 Stony Brook University
 Stony Brook, New York 11794
 michael.zingale
 @stonybrook.edu

Weiqun Zhang
 Center for Computational
 Sciences and Engineering
 Lawrence Berkeley National
 Laboratory
 Berkeley, CA 94720
 weiqunzhang@lbl.gov

Alan C. Calder
 Department of Physics and
 Astronomy
 Institute for Advanced
 Computational Sciences
 Stony Brook University
 Stony Brook, New York 11794
 alan.calder@stonybrook.edu

ABSTRACT

We present results and lessons learned from a 2015-2016 Blue Waters Student Internship. The project was to perform preliminary simulations of an astrophysics application, Black Widow binary systems, with the adaptive-mesh simulation code *Castro*. The process involved updating the code as needed to run on Blue Waters, constructing initial conditions, and performing scaling tests exploring *Castro*'s hybrid message passing/threaded architecture.

CCS Concepts

•Applied computing → Astronomy;

Keywords

pulsars:general – stars:evolution – stars:neutron – radiative transfer – methods:numerical

1. INTRODUCTION

Black Widow Pulsar Systems.

Black Widow Pulsar systems (BWPs), also known as Black Widow binaries, are binary star systems consisting of a millisecond pulsar (MSP) along with a substellar mass companion star that is being ablated by the pulsar's wind. Pulsars are highly-magnetic rotating neutron stars (NSs) that emit

^{*}2015-2016 Blue Waters Intern.

[†]Now at the Department of Astronomy & Astrophysics, UC Santa Cruz, Santa Cruz, CA 95064

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.

DOI: <https://doi.org/10.22369/issn.2153-4136/8/3/4>

beamed electromagnetic radiation, and MSPs are extremely rapidly rotating pulsars that have been spun up by accreting angular momentum from a close binary companion and are sometimes referred to as “recycled” pulsars. Observed BWPs mostly reside in globular clusters [?] and the systems have orbital periods of less than about 10 hours. An example of a system clearly demonstrating ablation is B1957+20 [?], and at least 27 systems are currently known with more being found as searches continue.

BWP systems may be the final stage in the evolution of the low mass X-ray binaries (see, e.g. [?]), which are NSs accreting from a low mass companion overflowing its Roche lobe. However, [?] argue that some ablating secondaries require a companion exchange in a dense environment. In any event, the low-mass companions are indeed ablating: about half the known BWPs exhibit a radio eclipse at low frequencies (e.g., [?]) due to a low density plasma cloud surrounding the secondary. The shape of the low frequency attenuation suggests a comet-like shape to the plasma cloud. In the case of PSR B1957+20, in addition to the radio eclipse, a plasma tail counter to the proper motion is observed in X-rays [?]. Eventually, the companion will be completely ablated away.

An interesting aspect of BWPs is that the pulsars seem to be relatively massive, possibly due to accretion of substantial amounts of the companions. Very few of these have been analyzed with corresponding optical observations that allow estimation of system masses, but those that have are intriguing. In at least five cases, NS masses have been estimated:

- B1957+20 [?] $M = 2.4 \pm 0.3 M_{\odot}$, the original BWP.
- PSR J1311-3430 [?] $M = 2.55 \pm 0.50 M_{\odot}$.
- PSR J1544+4937 [?] $M = 2.06 \pm 0.56 M_{\odot}$.
- PSR 2FGL J1653.6-0159 [?] $M > f(M_2)/\sin^3 i \gtrsim 1.96 M_{\odot}$. The largest measured companion mass function, $f(M_2)$, to date.
- PSR J1227-4859 [?] $M = 2.2 \pm 0.8 M_{\odot}$, a related system, a “redback” pulsar.

Uncertainties in the size and shape of the ablating companion, however, preclude an exact determination of the pulsar’s mass. If the companion is point-like, the inferred mass is small. If the companion is able to fill its Roche Lobe, allowing its mass to spill onto the pulsar, the inferred mass would be much larger. Improving estimates will require sophisticated radiation hydrodynamics simulations of the ablation of the companion to thereby inform interpretation of the observations. The work proposed for this Blue Waters Internship represents the first steps along this path.

Proposed BWP Work for the Blue Waters Internship:

The proposed work for the internship was for the intern to perform preliminary simulations of BWPs with the adaptive-mesh astrophysical simulation code *Castro* [?, ?, ?]¹. Initial simulations were to assume two-dimensional axisymmetry with a strong radiation field coming from the boundary to understand how the energy deposition disrupts the star, with full three-dimensional studies eventually following. Meaningful scaling tests required three-dimensional simulations, however, so full three-dimensional simulations were performed for the internship.

The radiation hydrodynamics module for *Castro* consists of radiation transport in the flux-limited diffusion (FLD) approximation, which is not sufficient for the BWP problem because the absence of the directional information in FLD makes it insufficient for modeling the ablation of the stellar companion. Preliminary simulations exploring the dynamics of BWP systems, however, are intended to yield important information about the relevant hydrodynamic and radiation hydrodynamic time scales and allow construction of initial models.

Further development of the radiation hydrodynamics module in *Castro* was expected to occur during the performance period of the internship by a then-unnamed graduate student in collaboration with *Castro* developers at the Lawrence Berkeley National Laboratory. The core solvers in *Castro* (an interface to the algebraic multigrid solvers in the *hyre* package [?]) support a more general system, and expected development for the project includes implementing an M_1 approximation, like that used in [?], to better describe the radiation. In contrast to FLD, the M_1 approximation includes an angular dependence enabling it to “cast a shadow”—it doesn’t artificially diffuse where it shouldn’t. This improvement will allow for accurate simulation of ablation, but the time required for this development was known to be far longer than the performance period of this internship.

The intern, Karpov, was expected to contribute to all aspects of the preliminary problem, including exercising *Castro* at scale on Blue Waters, constructing the model star and initial conditions, performing the preliminary simulations, and as possible contributing to the development of the M_1 transport module. During the internship, a graduate student (Barrios Sazo) became involved and began working on the project along with the intern.

2. BLACK WIDOW BINARIES

2.1 Setup

¹*Castro* is freely available from <https://github.com/BoxLib-Codes/Castro>.

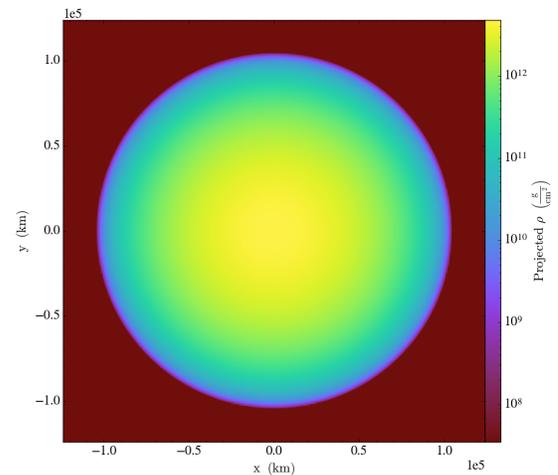


Figure 1: Rendering of the G type companion star to be ablated. The initial conditions included constructing this three-dimensional hydrostatic model.

Our model consists of a Solar-like companion star with radially varying density, being exposed to a “wind” of constant radiation coming from a single direction, as if from the accompanying neutron star (Fig. 1). As simulation time progresses, the ablation process would be visible as the wind interacts with the star.

2.2 Results

The XE node of Blue Waters contains 4 non-uniform memory access (NUMA) domains, hence we focused on this optimal configuration of runs performed with 4 MPI-ranks and 8 OpenMP threads per rank, to utilize all 32 cores on each node. When we tested this setup with the runs of 5120 cores and 256³ resolution for the BWP model, we found it to have the best performance out of any other MPI-OpenMP hybrid configuration, as shown in Fig. 2.

This model was expanded onto a wider range of cores (1280, 2560, 10240, 20480, 40960, and 81920). In Fig. 3 we can observe the effects of message passing (MPI) and threaded (OpenMP) parallelism within the hybrid architecture for our BWP simulations with *Castro*. The figure only compares the results of the MPI-only runs to MPI-OpenMP hybrid runs with optimal thread configuration, as stated above². In the case of 40960 and 81920 cores, the time required for I/O exceeded the maximum wall-clock time allowed in the message passing only case so we were only able to make the comparison out to 20480 cores. The addition of threads was observed to increase the calculation time at each step up to 20480 cores (Fig. 3(a)). The I/O time for each checkpoint, however, was significantly decreased with the addition of threads (Fig. 3(b)). We found an optimal combination of the two effects, providing the best total simulation time (Fig. 3(b)).

Even though this scenario might seem as the most optimized for our BWP simulation, further studies showed a deviation from this model. *Castro* utilizes the *BoxLib* adaptive mesh library, and the main metric is the number of boxes

²Other configurations had been tested (eg. 8, 16 MPI-ranks with 4, 2 OpenMP threads per rank respectively), and they showed the same trends, but worse performance overall.

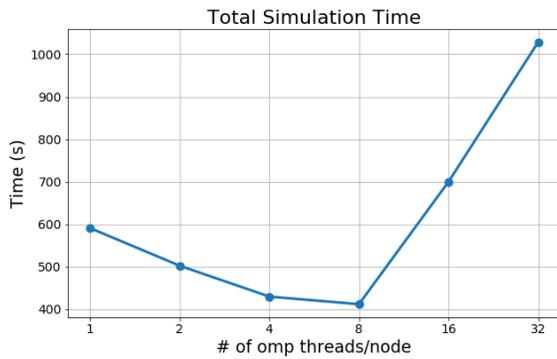


Figure 2: An example of MPI-OpenMP hybrid for a run on 5120 cores and 256^3 resolution.

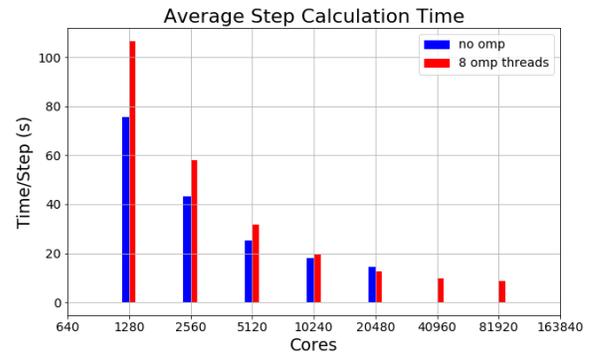
per MPI task. Too few, and a particular job is work-starved and does not perform well. Fig. 3(c) shows that running with 8 threads at higher core count slows down in its total simulation time. Further, Fig. 4 presents the result that *Castro* demands more threads as we increase the number of cores in order to keep up with the performance. That being said, the total simulation time appears to stagnate at the core counts above 10240, with 8 threads per rank setup.

Given the limited amount of computing time (20,000 node hours), the further study of ablation processes onto the companion were not possible. The MPI-OpenMP hybrid performance study left no computing time to evolve computationally expensive Black Widow binary system enough to see significant effects of ablation due to radiation.

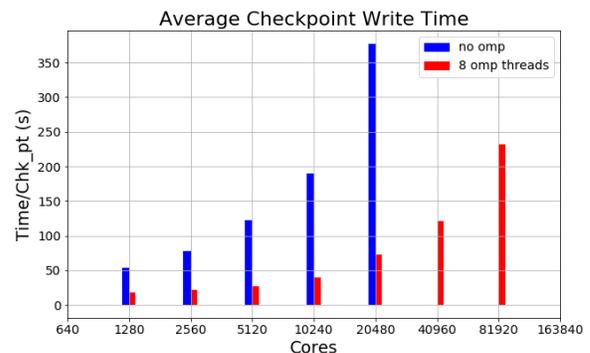
2.3 Lessons Learned

The study presented here was initiated as a project for an intern in the Blue Waters Student Internship Program for 2015-2016 (Karpov). The problem of BWPs is of interest to the nuclear astrophysics research group at Stony Brook, including the mentor (Calder) and the *Castro* expert (Zingale). And, as mentioned above, during the course of the internship, a graduate student (Barrios Sazo) began working on the problem as well. Also, one of the developers of the radiation hydrodynamics module (Zhang) kindly assisted as well. Thus the results presented in this paper show the combined effort of a team, and discussion of contributions of members and the education of each during the process is warranted.

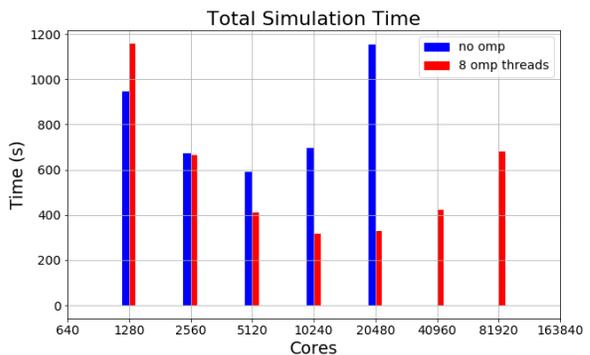
The intern came to this project with experience from a computational physics class, PHY 277 at Stony Brook University, that included experience with Linux workstations and programming, but little experience with supercomputing. Accordingly, the first lessons learned were associated with using a supercomputer such as Blue Waters: the difference between login and compute nodes, the difference between local scratch space, home directories, and mass storage, and the process of submitting jobs for execution through a queuing system. Next came lessons associated with large simulation codes: understanding the software architecture, constructing a new problem setup consisting of routines for constructing the initial conditions and setting parameters, identifying, installing as needed, and linking to requisite libraries, assessing parallel performance through scaling studies, and the necessity of platform-specific tuning for hybrid message passing/threaded architectures, and finally processing and visualizing the results, which included experience



(a) Threads are computationally expensive



(b) Threads benefit data output



(c) Final results

Figure 3: (a), (b), and (c) provide the performance analysis of MPI+OpenMP hybrid (resolution is 256^3). For the case of no OpenMP used for 40960 and 81920 cores, checkpoint output exceeded the maximum wall-times provided. Hence they can be noted as surpassing the limit of 1200s.

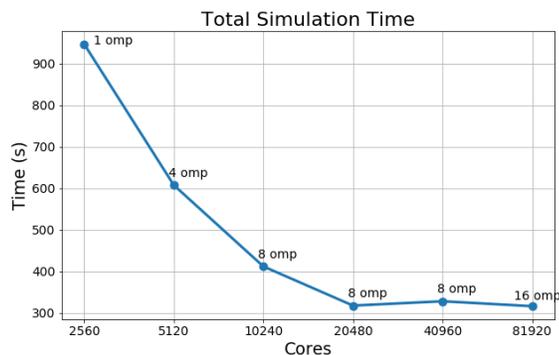


Figure 4: Higher number of cores demands higher number of threads to achieve superior performance. The data points shown are of the lowest total simulation time for a fixed core count, but varying OpenMP threads. All of the simulations were conducted at resolution of 256^3 .

with developing scripts. In addition, the intern learned valuable lessons concerning the science of BWPs and numerical methods for radiation transfer. Finally, the intern gained experience with the arduous process of writing a scientific paper with collaborators.

The graduate student (Barrios Sazo), despite serving as a teaching assistant during most of the course of this work, was able to spend considerable time on developing the code for the BWP problem. Her contributions included working with the intern on constructing the routines for problem setup, which includes understanding the physics and numerical approaches and thereby constructing physically meaningful initial conditions.

Lessons learned by the graduate student principally follow from her experience in taking the lead on constructing the hydrostatic star for the initial conditions. She began by exploring examples of problem setups provided in the code distribution, and a lesson here was in the importance of good software practices to make code readily re-usable. Then she learned about how to probe the simulation for problems—in this case, when a crash occurred, was it due to the radiation module or the hydrostatic star? This process gave her valuable experience with a multi-physics application. Similarly, she began by constructing two-dimensional initial conditions and then addressed the three-dimensional case. The lesson here was how to start with a simpler case in order to gain insight into the parameters of the problem and then take on the full three-dimensional case. Also, the graduate student gained valuable experience in working with the undergraduate and faculty as part of a team on the project. Lessons learned here include the importance of communication, particularly discussing ideas, and good software practices, e.g. version control.

The collaborating member of the research group (Zingale) brought expertise with *Castro* to the project as he is one of the principal developers. He guided the intern and graduate student with use of the code and the development of the BWP problem setup. He also provided guidance in assessing the performance and scaling, a good example being the suggestion of exploring how the choice of solver in the *hypr* package affects the performance. Such insight was invaluable to the intern’s experience. Similarly, guidance from

the *Castro* developer (Zhang) was critical to the success of this project. Finally, the mentor, Calder, assisted as best he could with all aspects of this study and gained experience both the the science of BWP systems and *Castro*.

3. CONCLUSIONS

We conclude that the Blue Waters Internship was an invaluable opportunity that greatly enriched the intern’s undergraduate experience. As of this writing, he is beginning a graduate program in Astrophysics and plans on pursuing a large-scale computational astrophysics problem for his dissertation research. The internship offered experience in all aspects of running a modern code at scale on one of World’s largest platforms, from the rote details of job submission and data storage to strategies for parallelism and meaningfully testing scaling.

We also conclude that while this project began with just the intern, the formation of a team made it far more successful than it would have been otherwise, to the benefit of all parties involved. What started as a project for one student wound up benefiting a research group.

4. ACKNOWLEDGMENTS

The authors gratefully acknowledge the National Computational Science Institute Blue Waters Student Internship Program for its support, training, and inspiration. This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications. The authors also gratefully acknowledge the contributions to this work from James Lattimer at Stony Brook. His expertise in the physics of neutron stars and interest in BWPs led to this study. This work was supported in part by the Department of Energy under grant DE-FG02-87ER40317. The research described in this paper also has made use of the high-performance computing system at the Institute for Advanced Computational Science at Stony Brook University. This research has made use of NASA’s Astrophysics Data System Bibliographic Services.

5. REFERENCES

- [1] A. Almgren, J. Bell, D. Kasen, M. Lijewski, A. Nonaka, P. Nugent, C. Rendleman, R. Thomas, and M. Zingale. MAESTRO, CASTRO, and SEDONA – Petascale Codes for Astrophysical Applications. *ArXiv:1008.2801*, Aug. 2010. Proceedings of the 2010 Scientific Discovery through Advanced Computing (SciDAC) Conference. Chattanooga, Tennessee, July 11-15, 2010. Oak Ridge National Laboratory. <http://computing.ornl.gov/workshops/scidac2010/>.
- [2] A. S. Almgren, V. E. Beckner, J. B. Bell, M. S. Day, L. H. Howell, C. C. Jogerst, M. J. Lijewski, A. Nonaka, M. Singer, and M. Zingale. CASTRO: A New Compressible Astrophysical Solver. I. Hydrodynamics and Self-gravity. *APJ*, 715:1221–1238, June 2010.
- [3] W. Bednarek and J. Sitarek. High-energy emission from the nebula around the Black Widow binary system containing millisecond pulsar B1957+20. *AAP*, 550:A39, Feb. 2013.

- [4] D. de Martino, J. Casares, E. Mason, D. A. H. Buckley, M. M. Kotze, J.-M. Bonnet-Bidaud, M. Mouchet, R. Coppejans, and A. A. S. Gulbis. Unveiling the redback nature of the low-mass X-ray binary XSS J1227.0-4859 through optical observations. *MNRAS*, 444:3004–3014, Nov. 2014.
- [5] R. Falgout and U. Yang. *hypre*: a library of high performance preconditioners. In P. M. A. Sloot, C. J. K. Tan, J. J. Dongarra, and A. G. Hoekstra, editors, *Computational Science—ICCS 2002*, volume 2331 of *Lecture Notes in Computer Science*, pages 632–641, Berlin, 2002. Springer-Verlag.
- [6] A. S. Fruchter, D. R. Stinebring, and J. H. Taylor. A millisecond pulsar in an eclipsing binary. *Nature*, 333:237–239, May 1988.
- [7] M. González, E. Audit, and P. Huynh. HERACLES: a three-dimensional radiation hydrodynamics code. *AAP*, 464:429–435, Mar. 2007.
- [8] A. R. King, M. B. Davies, and M. E. Beer. Black widow pulsars: the price of promiscuity. *MNRAS*, 345:678–682, Oct. 2003.
- [9] R. W. Romani, A. V. Filippenko, and S. B. Cenko. 2FGL J1653.6-0159: A New Low in Evaporating Pulsar Binary Periods. *APJL*, 793:L20, Sept. 2014.
- [10] R. W. Romani, A. V. Filippenko, J. M. Silverman, S. B. Cenko, J. Greiner, A. Rau, J. Elliott, and H. J. Pletsch. PSR J1311-3430: A Heavyweight Neutron Star with a Flyweight Helium Companion. *APJL*, 760:L36, Dec. 2012.
- [11] M. Ruderman, J. Shaham, and M. Tavani. Accretion turnoff and rapid evaporation of very light secondaries in low-mass X-ray binaries. *APJ*, 336:507–518, Jan. 1989.
- [12] B. W. Stappers, B. M. Gaensler, V. M. Kaspi, M. van der Klis, and W. H. G. Lewin. An X-ray nebula associated with the millisecond pulsar B1957+20. *SCI*, 299:1372–1374, 2003.
- [13] S. Tang, D. L. Kaplan, E. S. Phinney, T. A. Prince, R. P. Breton, E. Bellm, L. Bildsten, Y. Cao, A. K. H. Kong, D. A. Perley, B. Sesar, W. M. Wolf, and T.-C. Yen. Identification of the Optical Counterpart of Fermi Black Widow Millisecond Pulsar PSR J1544+4937. *APJL*, 791:L5, Aug. 2014.
- [14] M. H. van Kerkwijk, R. P. Breton, and S. R. Kulkarni. Evidence for a Massive Neutron Star from a Radial-velocity Study of the Companion to the Black-widow Pulsar PSR B1957+20. *APJ*, 728:95, Feb. 2011.
- [15] W. Zhang, L. Howell, A. Almgren, A. Burrows, J. Dolence, and J. Bell. CASTRO: A New Compressible Astrophysical Solver. III. Multigroup Radiation Hydrodynamics. *APJS*, 204:7, Jan. 2013.

Using Blue Waters To Assess Non-Tornadic Outbreak Forecast Capability by Lead Time

Taylor Prislowsky

Department of Geosciences
108 Hilbun Hall, P.O. Box 5448
Mississippi State, MS, 39762-5448
1-662-325-3915
tap211@msstate.edu

Andrew Mercer

Department of Geosciences
108 Hilbun Hall, P.O. Box 5448
Mississippi State, MS, 39762-5448
1-662-325-3915
a.mercer@msstate.edu

ABSTRACT

Derechos are a dangerous, primarily non-tornadic severe weather outbreak type responsible for a variety of atmospheric hazards. However, the exact predictability of these events by lead time is unknown, yet would likely be invaluable to forecasters responsible for predicting these events. As such, the predictability of nontornadic outbreaks by lead time was assessed. Five derecho events spanning 1979 to 2012 were selected and simulated using the Weather Research and Forecasting (WRF) model at 24, 48, 72, 96, and 120-hours lead time. Nine stochastically perturbed initial conditions were generated for each case and each lead time, yielding an ensemble of derecho simulations. Moment statistics of the derecho composite parameter (DCP), a good proxy for derecho environments, were used to assess variability in forecast quality and precision by lead time. Overall, results showed that 24 and 48 hour simulations had similar variability characteristics, as did 96 and 120 hours. This suggests the existence of a change point or statistically notable drop-off in forecast performance at 72-hours lead time that should be more fully explored in future work. These results are useful for forecasters as they give a first guess as to forecast skill and precision prior to initiating their predictions at lead times of out to 5 days.

Keywords

Non-tornadic severe weather, stochastic initial condition perturbation, numerical weather prediction, ensemble forecasting, derecho forecasting

1. INTRODUCTION

Predicting severe weather occurrence continues to be a difficult forecasting challenge, despite many advances in this research area and the importance of the research problem. Many severe

weather studies have considered tornadoes the primary hazard

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.

DOI: <https://doi.org/10.22369/issn.2153-4136/8/3/5>

associated with major severe weather outbreaks owing to the catastrophic damage associated with tornado impacts. Receiving less attention are non-tornadic outbreaks of severe weather, which still have tremendous impacts from their known hazards, particularly derechos. Derechos are defined as a widespread, convectively induced windstorm, which can contain tornadoes but has a primary hazard of straight line wind damage (Johns et al. 1986). Derecho hazards are often as costly as many tornado and hurricane events that affect the United States (Ashley et al. 2005). For instance, on 4 April 2011, there was a severe derecho outbreak that impacted over twenty states and caused over \$16.5 million dollars in property damage, \$320,000 in crop damages, 3 deaths, and 13 injuries (Storm Database). Despite their importance, derecho predictability remains difficult in many instances, particularly as it relates to the timing of the event (Gallus et al. 2005).

Numerous studies have assessed the climatological aspects of non-tornadic severe weather events (including derechos). Coniglio et al. (2003) suggested that warm-season (summer) derechos tended to be confined to northern latitudes, while cool season derechos primarily impacted southern states. Ashley et al. (2005) noted an elevated occurrence probability for derechos when a previous derecho had impacted a region recently. These efforts gave insight into the basic characteristics of derechos, but offered little in terms of predictability.

Initial efforts at predicting derecho extent and timing have centered around the use of localized sounding observations (Cohn et al. 2007, Coniglio et al. 2004, others). These efforts have centered around predictability of a single event (i.e. Cohn et al. 2007) or identification of parameters useful in identifying and predicting derecho environments (Coniglio et al. 2004). Doswell et al. (2003) also noted that the initial mechanism by which convection begins is likely a major contributing factor to a mesoscale system evolving into a derecho. These studies supported initial work that identified a typical environment conducive for derecho formation, which requires a 1-2 km surface based stable layer, an elevated mixed layer of 2-4 km, and an upper tropospheric layer of intermediate stability extending up to the tropopause (Schmidt et al. 1991). These characteristics typically result in a storm system known referred to as a "bow echo" as the wind stress behind the line causes the line to bow outward (Przybylinski et al. 1995). These advances are certainly important to explain the current state of knowledge and fundamental characteristics of derecho events, but their applications in forecasting are limited, owing to data constraints and the impracticality of launching soundings into every derecho event. The availability of an accurate forecast model would

certainly help meteorologists improve predictions of future derecho events.

Recent work in numerical modeling of derecho events (e.g. Kain et al. 2004) has found good predictability for the strongest events. Mercer et al. (2009) explored the discrimination capability of machine learning methods in identifying tornadic and non-tornadic environments within numerical weather prediction simulations, the first effort in diagnosing outbreak mode outside of a forecast office. Their results showed overall good discrimination capability (with forecast skill scores exceeding 0.7). They also noted only a slight degradation in classification performance by lead time (out to 72-hours), motivating a research question regarding non-tornadic outbreak predictability by lead time.

Forecasters have long assumed that outbreak forecasting limitations exist in the short-term without properly quantifying that time period (though the 72-hour results in Mercer et al. 2009 are a first guess). This lack of specificity, combined with the results of Mercer et al. (2009), motivate the current research objective. The primary objective of this project is to identify model forecast uncertainty within non-tornadic severe weather outbreaks as it relates to outbreak lead time. It is hypothesized that variability patterns within outbreak lead times of 3 days and shorter will be statistically significantly different than lead times of 4 – 5 days. To demonstrate this, a set of 5 major non-tornadic outbreaks will be simulated with the Weather Research and Forecasting model using 9 stochastically varied initial conditions at lead times of 24, 48, 72, 96, and 120 hours. Shifts in the variability associated with the 9 simulations per lead time will help assess forecast precision by lead time, which will be useful for forecasters to identify the maximum skill within their forecasts.

This research is part of the Blue Waters Undergraduate Internship Experience. As such, the paper contains not only information about the resulting research, but aspects of the internship including lessons learned and reflections. Section 2 contains a summary of data and methods used in this research, while section 3 shows the results from multiple non-tornadic outbreak simulations at varying lead times. Section 4 contains discussion regarding important results and lessons learned from the internship, while section 5 contains reflection information on the internship experience. Section 6 summarizes the results and provides important conclusions from the research.

2. DATA AND METHODS

2a. Data

As the primary objective of this project was the diagnosis of forecast variability by lead time for major non-tornadic outbreaks, a set of outbreak events was required. For this study, five major derecho events from the Storm Event database (Storm Data) recorded by the National Climatic Data Center were selected, all of which spanned multiple states over a multi-hour period. The 19 June 1979 event included 137 severe thunderstorm wind reports (those in excess of 58 mph) occurring over 9 states, with a peak wind speed of 90 mph. The 30 May 2004 affected 19 states, resulting in 578 individual severe wind reports, with a peak wind speed of 97 mph. The major derecho of 4 April 2011 had 1318 wind reports across 18 states with a maximum wind speed reported at 90 mph. On 21 June 2011, a derecho impacted 21 states and resulted in 604 wind reports with a peak speed of 81 mph. Finally, the 29 June 2012 derecho

event (e.g. Fig. 1) affected 15 states with 1195 wind reports and a maximum observed wind speed of 93 mph.

Once a case set was established, continuous atmospheric data for each event was required for input into the WRF model. Since many of the predictors used for convective forecasting are mesoscale, a mesoscale analysis dataset, the North American Regional Reanalysis (NARR) was used to initialize the WRF. NARR data are provided on a 32-km Lambert conformal North American grid with 29 vertical levels and 3-hourly temporal resolution from 1979 to present. NARR data valid at 24, 48, 72, 96, and 120 hours prior to conclusion of the outbreak were retained.

2b. Model Configuration and Simulations

The proper simulation of a non-tornadic severe weather outbreak requires a gridded, convection allowing non-hydrostatic atmospheric model. The Weather Research and Forecasting (WRF – Skamarock et al 2008) version 3.8 was used to simulate the 5 outbreaks mentioned previously. Since the primary objective of this project was the diagnosis of variability of outbreak forecasts by lead time, each event was simulated at 24, 48, 72, 96, and 120 hours prior the end of a given event (as described previously). This timing ensured the peak outbreak time, which typically occurred on or after 0000 UTC on the event day, was sufficiently captured. Traditional model parameterizations for severe weather events were selected for the WRF simulations, including:

- The Yonsei University Planetary Boundary Layer scheme [YSU] for all five cases (Hong et al. 2005)
- The WRF Single-moment 6-class micro physics scheme [WSM6] (Hong et al. 2006)
- No cumulus parameterization
- The Dudhia Shortwave Radiation Scheme (Dudhia et al. 1989)
- The RRTM Longwave Radiation Scheme (Mlawer et al. 1997)
- The 5-layer Thermal Diffusion Land Surface Scheme (Dudhia et al. 1996)

The simulation domain was centered on a kernel density estimated outbreak center provided by the results from Shafer et al. (2012) and formulated on a 250 x 150 12-km grid-spacing grid with 45 vertical levels (e.g. Fig. 1). While the domain size was the same for each event, the geographic location of each simulation varied based on the storm report estimated outbreak center.

While event simulations were useful to depict overall environmental characteristics associated with each outbreak, a measure of variability was required to assess forecast precision by lead time. Variability was introduced into the WRF simulations using the Stochastic Kinetic Energy Backscattering Scheme (SKEBS, Berner et al. 2009) built into WRF 3.8. SKEBS adds random noise to potential temperature and stream function fields within the NARR input data, introducing perturbations and adding simulation variability via generation of an initial condition ensemble. At model initialization, only NARR are used, but SKEBS introduces random noise throughout the rest of the simulation, ensuring maximum spread in ensemble output and providing a direct measure of model variability. The SKEBS routine was used to generate nine initial condition ensemble members for each of the 5 lead times for each case, for a total of 225 individual model simulations.

The resulting simulations provided multiple diagnostic variables which are useful for addressing general weather variability, but derived severe weather parameters were required to assess environmental proneness to non-tornadic severe weather. One well known parameter, the derecho composite parameter (DCP – Evans and Doswell 2001) was computed on all gridpoints within each simulation domain to identify those locations which had elevated risk for derecho impacts (e.g. Fig. 1). The DCP is based on the following equation (from Evans and Doswell 2001):

$$DCP = \left(\frac{DCAPE}{980 \frac{J}{kg}} \right) \left(\frac{MUCAPE}{2000 \frac{J}{kg}} \right) \left(\frac{\Delta \vec{V}_{0-6}}{20 \text{ kt}} \right) \left(\frac{\vec{V}_{0-6 \text{ km}}}{16 \text{ kt}} \right)$$

Here, DCAPE refers to downdraft CAPE (a measure of positive stability associated with strong downdrafts and potential for extreme straight-line winds), MUCAPE is a maximum measure of instability, $\Delta \vec{V}_{0-6}$ refers to the vertical wind shear over the 0-6 km layer, and $\vec{V}_{0-6 \text{ km}}$ is the mean wind vector over the 0-6 km vector. Evans and Doswell (2001) defined this formula based on a large database of derecho proximity sounding data. They showed that the DCP was attuned at identifying atmospheric environments that were favorable for cold pool wind events through four mechanisms:

1. Cold pool production [DCAPE]
2. Ability for strong storms to be sustained along the leading edge of a gust front, the strongest section of a gust front [MUCAPE]
3. The potential for organization for any possible ensuing convection [0-6 km shear]
4. Enough flow in the ambient environment to favor development along a downstream portion of the gust front [0-6 km mean wind].

The DCP was utilized for this project owing to its global depiction of derecho-prone environments. Tremendous variability in DCP values is likely associated with uncertainty in the DCP forecast, which can be directly assessed by lead time using the above described methodology.

2c. Simulation Analysis

Once the simulations were completed, the resulting model runs were analyzed by assessing gridpoint variability along the 9 initial conditions. That is, moment statistics (mean, variance, skewness, and kurtosis) of DCP were computed at each gridpoint using the 9 stochastic perturbations for each case and each lead time. However, many points which yielded zero DCP values were excluded, as their moment statistics did not provide meaningful insight into variability structures within the simulations. Once non-zero DCP gridpoint variability was computed, 1000 bootstrap-resampled moment statistics on those non-zero points were formulated, allowing for the generation of confidence intervals for each moment statistic by lead time. These confidence intervals allowed the primary research hypothesis regarding lead time and forecast variability to be assessed. Results from these analyses are provided below.

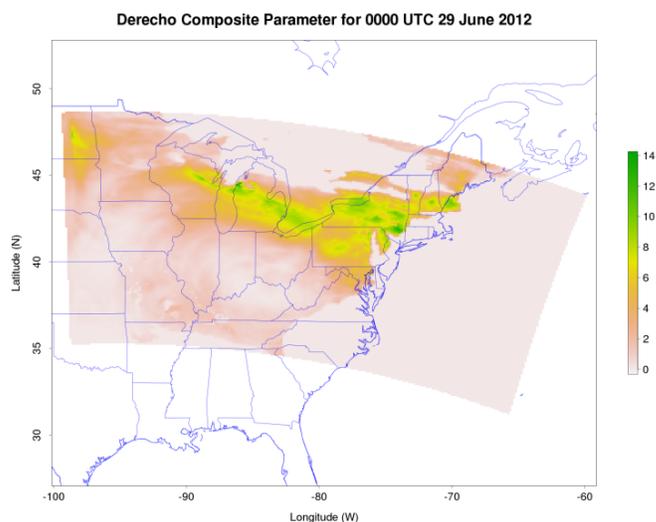


Fig. 1 Derecho Composite Parameter (DCP) for 0000 UTC 29 June 2012. Note the area shaded in gray represents the outbreak domain used for all simulations of the 29 June 2012 derecho event. Higher DCP values support derecho formation maintenance.

All cases showed considerable increases in DCP variability as the event valid time approached, and their resulting variability measures were widely dispersed, with an interesting pattern emerging. In general, 24-48 hour simulations tended to cluster fairly closely with all cases, while 72 to 120 hour simulations tended to cluster with each other and away from the 24-48 hour simulation groups. As an example, the 29 June 2012 derecho event is provided in Fig. 2. In this case, variance statistics at 24 and 48 hours (second panel orange and green lines) tended to cluster very closely together, while the remaining observations tended to group together and away from the 24-48 hour pairing. The gridpoint distribution tended to become more positively skewed as the outbreak progressed, and the skewness values were particularly enhanced at longer lead times, suggesting the tendency for larger DCP outliers with longer lead times. Kurtosis results were similar, as all distributions were platykurtic but longer-lead time simulations revealed more peaked results. Note that similar results were present for the other 4 cases as well (not shown here). Interestingly, 96 and 120 hour results tended to correlate strongly, suggesting that DCP forecasts at 96 and 120 hours offer similar performance, a previously undocumented result.

While differences among moment statistic distributions were a useful component of this research, the primary research objectives dealt with output variability by lead time over all cases, not just individual example case studies. To demonstrate overall performance, individual lead time data for all 5 cases were averaged, yielding average moment statistics by lead time for the selected events. These analyses revealed several interesting patterns (Fig. 3). First, as expected, variance by lead time increased with increasing lead time, with a notable jump observed at 96 and 120-hour lead times. Also, note that the 96 and 120-hour results are not significantly different based on the confidence intervals, supporting the previous conclusion regarding 96-120 hour day predictability essentially remaining equal. Another notable result was the sharp increase in positive skewness at 96 and 120-hour lead times in contrast with the relatively unskewed 72 and earlier lead time forecasts. This result supports the previous conclusions regarding the

similarities among the 24-48 hour forecasts and their stark contrasts with 96 to 120 hour predictions. Additionally, kurtosis values showed an unusual drop off in mean kurtosis value at the 72-hour forecast, likely owing to outlier results due to periods of relatively low DCP values (e.g. the green dip in Fig. 2's bottom panel). Outside of that individual outlier, the kurtosis behavior was in line with skewness behavior, with a relatively platykurtic distribution observed at 24-48 hours lead time and a more peaked (but still non-Gaussian) distribution observed at lead times in excess of 72-hours. These results further support the

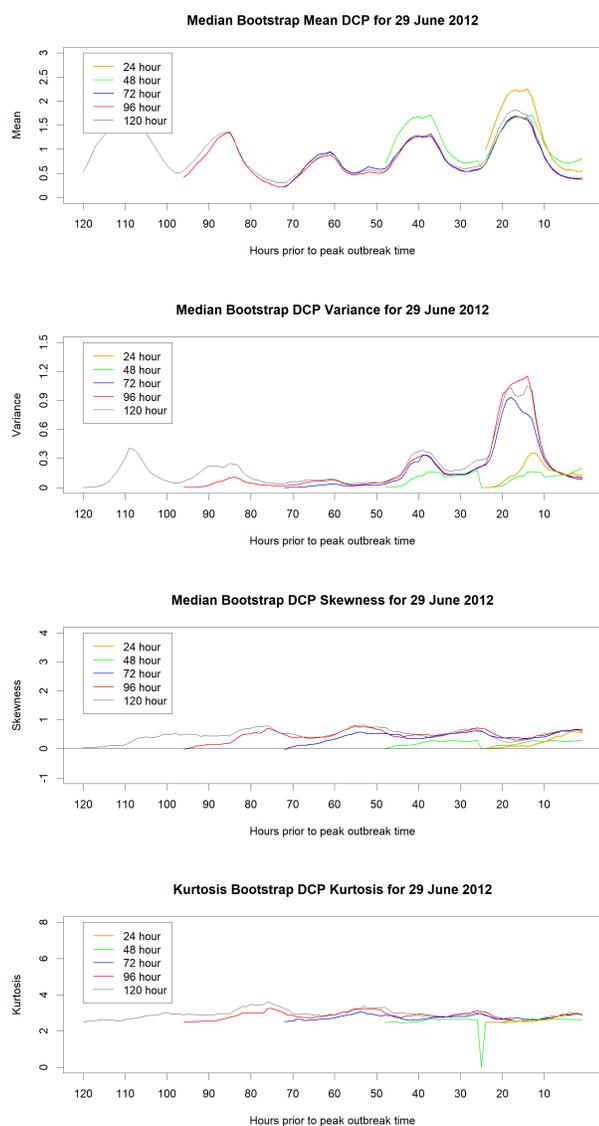


Fig 2. Moment statistics for the 29 June 2012 derecho. The top panel shows median bootstrap mean DCP values, while the second panel shows variance statistics, the third panel skewness, and the fourth kurtosis. Note that for this event outbreak valid time is roughly 12 hours prior to the end of the simulation.

existence of a few outlier points that are driving up the kurtosis values and increasing the skew of the distributions.

3. DISCUSSION AND LESSONS LEARNED (EDUCATIONAL IMPACT)

The primary objective of this research was to ascertain WRF model uncertainty by lead time for five major non-tornadic severe weather events. Outbreak severity was assessed using DCP as a proxy measure for the derecho environment. Overall, the major findings included the consistency among 96 and 120-hour lead time runs, the outlier and relatively unpredictable nature of 72-hour simulations, and the similarities among 24-48 hour runs. These results are very useful to forecasters for a variety of reasons. First, forecast confidence in a 120-hour forecast is unlikely to change for a 96-hour forecast, a result that has not been quantified previously. Second, similar behavior exists at 24 and 48-hours lead time as their DCP mean and variability structure was quite similar. While this study does not measure accuracy of the DCP forecasts, all events selected were major derecho events, and as such higher values of DCP (e.g. the 24-72 hour runs for 29 June 2012 – Fig. 2) are more supportive of an environment conducive for derecho formation. Mean DCP values were generally higher in the shorter lead-time runs, which should increase forecaster confidence in derecho occurrence as well.

This project was completed using Blue Waters supercomputing resources as a part of the Blue Waters Undergraduate Internship program. The educational component of the research for the undergraduate student fell in two key areas. First, the student was exposed to the challenges of dynamic atmospheric modeling within a high-performance computing environment, including the temporal and physical constraints of simulations and configuring parallel processing jobs. The student also gained valuable experience working with big datasets (the project generated nearly 1 TB of data) and the computational challenges associated with such big data interactions.

In addition to the general education experiences for the undergraduate student, they learned key lessons regarding supercomputing research. These are listed below.

1. Simulation data were lost due to typical file system cleaning and the undergraduate student not storing the data properly. The student learned the importance of file backups as the cases were rerun.
2. The student gained valuable insight into the challenges of forecasting non-tornadic severe weather events, including the forecasting metrics that are used to evaluate the likelihood of these outbreaks.
3. The student learned data organization and the challenges of large data transfers, as the 225 simulations needed to be moved between machines prior to running the model.
4. The student discovered an issue with an initial condition, which forced the original 10 initial condition ensemble members to be reduced to 9. This helped the student learn the importance of data quality and close interaction with the project to limit the risks of future issues.

One important computational challenge was encountered as well, which required the use of an external machine to post-process the results. The Blue Waters Cray system was not compatible with the Unified Post Processor software used to post-process the WRF simulations due to compilation issues,

which the student struggled with for a long period in the internship. Despite this small setback, the student felt the experience was largely successful and the mentor was satisfied that the student gained important supercomputing skills that are essential for successful research meteorologists.

The work would not have been possible without access to the resources offered by the Blue Waters Supercomputing Center, particularly the quantity of simulations in the required 1-year study period. The computing time utilized by the project exceeded 1000 computational hours, which is difficult for an undergraduate student to finish in a traditional computing environment, particularly given the limited timeline. Additionally, the quantity of data produced by the project (nearly 1 TB) and quantity of forecast hour files (16,425) are unwieldy for even a modest supercomputing center, requiring the robust resources offered by Blue Waters.

4. REFLECTIONS

Undergraduate research projects are typically fraught with challenges simply owing to the student's inexperience working in research. While this internship had its share of challenges, the student gained valuable experience working in supercomputing, which is becoming more important in operational meteorology as National Weather Service offices begin to maintain their own small supercomputing clusters for regional modeling. Additionally, with the introduction of high resolution imaging provided by new data platforms such as the GOES-R satellite, big data experience is an essential part of any successful research meteorologist's repertoire. Finally, the student's participation in the project prepared them for graduate study, which they are now engaged in, and that experience, combined with the Blue Waters Summer Internship Program experience, will help set the student apart from their peers when they begin searching for jobs.

5. SUMMARY AND CONCLUSIONS

The primary objectives of this research were to obtain measures of forecast precision and variability at lead times from 1 to 5 days. It is well established in meteorology that short-term forecasts are more precise and accurate than longer-term predictions, but few studies have formally quantified these differences. The research objectives herein addressed these concerns in the context of derecho forecasts, with interesting results.

To address this variability, five derecho events spanning 1979 to 2012 were selected. Outbreak-centric domains were retained from WRF simulations of each event, where WRF simulations were run with 24, 48, 72, 96, and 120 hours lead time. Each lead time's simulations were perturbed stochastically nine times, introducing forecast uncertainty from which lead time precision could be obtained. Resulting gridpoint precision for non-zero gridpoints of DCP were retained using bootstrap-resample moment statistics for each case and global values for all cases.

Overall, several key findings resulted from this analysis. First, model precision (and predictions) tended to remain very similar with both 24 and 48-hour lead times and with 96 and 120-hour lead times. This suggests only minimal drop-offs in forecast skill between 24 and 48 hours lead time and between 96 and 120 hours.

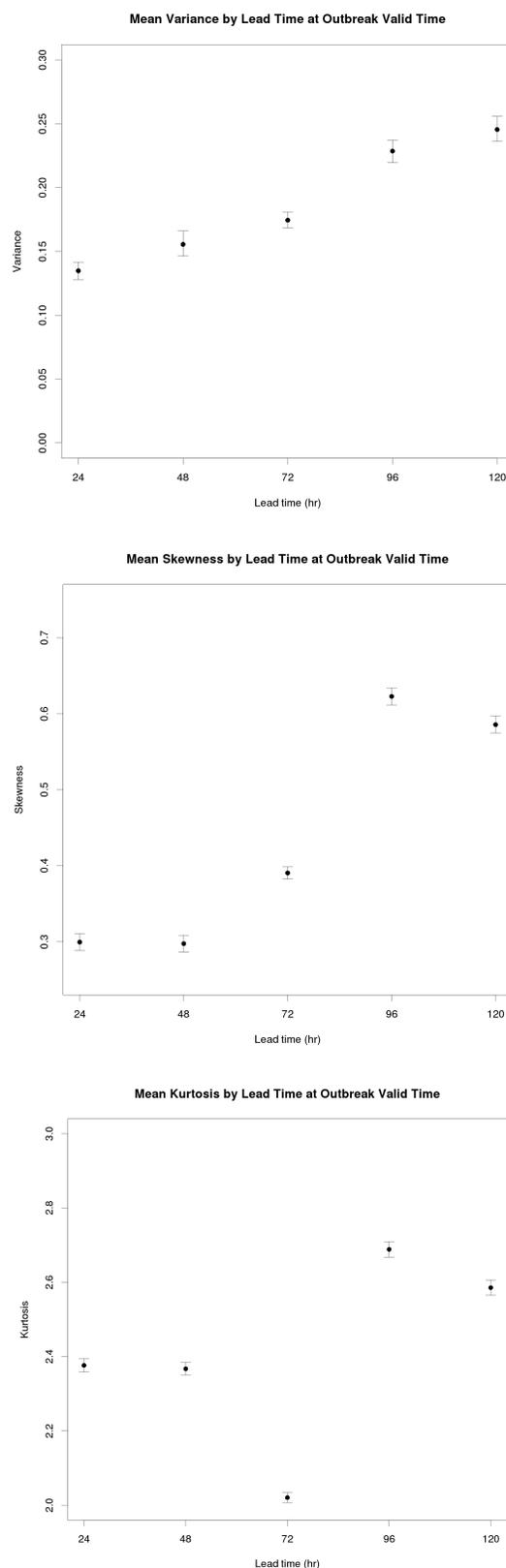


Fig. 3. Average moment statistics by lead time for all derecho events at all forecast times. The top panel represents variance by lead time, while the middle panel is skewness and the bottom panel is kurtosis.

The 72-hour simulations were inconsistent, with some producing lower variability than either 24 or 48 hours lead time and others higher. This suggests the existence of a forecast confidence change point around 72 hours lead time, which is a likely good demarcation between short-term high skill forecasts and medium-term modest skill forecasts. These results are in line with Mercer et al. (2009) who noticed some skill drop-off at 72 hours but did not consider longer lead times, which would have likely revealed these patterns as well.

These results have significant forecast implications, as they allow forecasters to have prior knowledge of anticipated WRF forecast skill, which is useful for prediction purposes. The results also help reveal a cutoff point in terms of lead times; that is, what defines a “short term” and a “medium term” derecho forecast. Future work will address this issue with additional non-tornadic derecho events and add tornado outbreaks as well. It is expected that similar behavior in tornado outbreaks will exist, though tornado outbreaks are less predictable than their non-tornadic counterparts. Overall, this study reveals important insight into non-tornadic outbreak predictability, which will be useful for future outbreak forecasts.

6. ACKNOWLEDGEMENTS

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. This work resulted from a Blue Waters undergraduate internship opportunity, and we are thankful for their financial and technical support throughout the project. We are also grateful for feedback during the peer-review process.

7. REFERENCES

- Ashley, W. et al. (2005), On the episodic nature of derecho-producing convective systems in the United States. *Journal of Climatology*, Vol. 25, pp. 1915-1932.
- Ashley, W. S. and T. L. Mote (2005), Derecho Hazards in the United States. *Bulletin of the American Meteorological Society*, Vol. 86, pp. 1577-1592.
- Berner, J., et al. (2008), A Spectral Stochastic Kinetic Energy Backscatter Scheme and Its Impact on Flow-Dependent Predictability in the ECMWF Ensemble Prediction System. *Journal of the Atmospheric Sciences*, Vol. 66, pp. 603-626.
- Cohen, A., et al. (2007), Discrimination of Mesoscale Convective System Environments Using Sounding Observations. *Weathering and Forecasting*, Vol. 22, pp. 1045-1062.
- Coniglio, M., and D. Stensrud, (2003), Interpreting the Climatology of Derechos. *Weather and Forecasting*, Vol. 19, pp. 595-605.
- Coniglio, M., et al. (2004), An Observational Study of Derecho-Producing Convective Systems. *Weather and Forecasting*, Vol. 19, 320-337.
- Doswell III, C., and J. Evans, (2003), Proximity sounding analysis for derechos and supercells: an assessment of similarities and differences. *Atmospheric Research*, Vol. 67-68., pp. 117-133.
- Dudhia, J., (1996), A Multi-Layer Soil Temperature Model For MM5. The Sixth PSU/NCAR Mesoscale Model Users' Workshop. National Center for Atmospheric Research, Boulder, Colorado, pp. 1-3.
- Dudhia, J., (1989), Numerical Study of Convection Observed during the Winter Monsoon Experiment Using a Mesoscale Two-Dimensional Model. *Journal of Atmospheric Sciences*, Vol. 46, pp. 3077-3107.
- Evans, J., and C. Doswell III, (2001), Examination of Derecho Environments Using Proximity Soundings. *Weather and Forecasting*, Vol. 16, pp. 329-342.
- Gallus, W., et al., (2005), 4 June 1999 Derecho Event: A Particularly Difficult Challenge for Numerical Weather Prediction. *Weather and Forecasting*, Vol. 20, pp. 705-728.
- Hong, S., et al, (2005), A new vertical diffusion package with an explicit treatment of entrainment processes. *Monthly Weather Review*, Vol. 134, pp. 2318-2341.
- Hong, S. and J. Lim, (2006), The WRF Single-Moment 6-Class Microphysics Scheme (WSM6). *Journal of the Korean Meteorological Society*, Vol. 42, pp. 129-151.
- Johns, R., and W. Hirt, (1986), Derechos: Widespread Convectively Induced Windstorms. *Weather and Forecasting*, Vol. 2, pp. 32-49.
- Kain, J., et al., (2005), Examination of Convection-Allowing Configurations of the WRF model for the Prediction of Severe Convective Weather: The SPC/NSSL Spring Program 2004. *Weather and Forecasting*, Vol. 21, pp. 167-181.
- Mercer, A., et al., (2009), Objective Classification of Tornadic and Nontornadic Severe Weather Outbreaks. *Monthly Weather Review*, Vol. 137, pp. 4355-4368.
- Mesinger, F., et al. (2003), The North American Regional Reanalysis. *Bulletin of the American Meteorological Society*, Vol. 87, 343-360.
- Mlawer, E., et al., (1997), Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave. *Journal of Geophysical Research*, Vol. 102.D14, pp. 16663-16682.
- Przybylinski, R., (1995), The Bow Echo: Observations, Numerical Simulations, and Severe Weather Detection Methods. *Weather and Forecasting*, Vol. 10, pp. 203-218.
- Schmidt, J., (1991), Numerical and Observational Investigations of Long-Lived MCS-Induced Severe Surface Wind Events: the Derecho. *Thesis-Dissertation Abstracts International*, Vol: 52-09.
- Shafer, C., et al, (2012), An Assessment of Areal Coverage of Severe Weather Parameters for Severe Weather Outbreak Diagnosis. *Weather and Forecasting*, Vol. 27 pp. 809-831.
- Skamarock, W., et al, (2008), A Description of the Advanced Research WRF Version 3. *NCAR/TN-475+STR NCAR Technical Note*.
- Storm Prediction Center (2017). *Derecho Composite Parameter (DCP)*. <http://www.spc.noaa.gov/misc/AbtDerechos/derechofaq.htm>. Web.
- Storm Data (2017). *National Climatic Data Center*, March 2017. **59**, 588 pp.

Blue Waters Supercomputing Applications in Climate Modeling with the WRF Model

Morgan Smith

Department of Geosciences
108 Hilbun Hall, P.O. Box 5448
Mississippi State, MS, 39762-5448
1-662-325-3915
m1s561@msstate.edu

Andrew Mercer

Department of Geosciences
108 Hilbun Hall, P.O. Box 5448
Mississippi State, MS, 39762-5448
1-662-325-3915
a.mercer@msstate.edu

ABSTRACT

Long-term atmospheric forecasting remains a significant challenge that in the field of operational meteorology. These long-term forecasts are typically completed through the use of climatological variability patterns in the geopotential height fields, known in the field of meteorology as teleconnections. Despite heavy reliance on teleconnections for long-term forecasts, the characterization of these patterns in operational weather models remains inadequate.

The purpose of this study is to diagnose the ability of an operational forecast model to render well-known teleconnection patterns. The Weather Research and Forecasting (WRF) model, a commonly employed regional operational forecast model, was used in the simulation of the major 500 mb Northern Hemisphere midlatitude teleconnection patterns. These patterns were formulated using rotated principal component analysis on the 500 mb geopotential height fields. The resulting simulated teleconnection patterns were directly compared to observed teleconnection fields derived from the National Center for Environmental Prediction/National Center for Atmospheric Research (NCEP/NCAR) reanalysis 500 mb geopotential height database, a commonly utilized observational dataset in climate research. Results were quite poor, as the resulting teleconnection patterns only somewhat resembled those constructed on the observed dataset, suggesting a limited capability of the WRF in resolving the underlying variability structure of the hemispheric midlatitude atmosphere. Additionally, configuring the regional model to complete this simulation was met with a series of computational challenges, some of which were not successfully overcome. These results suggest future needs for improvement of the WRF model in reconstructing teleconnection fields and for use in climate modeling.

Keywords

Teleconnections, interannual variability, climate modeling, rotated principal component analysis, Weather Research and Forecasting model

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.

DOI: <https://doi.org/10.22369/issn.2153-4136/8/3/6>

1. INTRODUCTION

Long-term forecast ability in meteorology remains heavily linked with climate-scale interannual variability patterns within the mid-latitude geopotential height fields. These interannual variability patterns, known in the field of meteorology as teleconnections, are a result of cyclical internal atmospheric dynamics (in particular synoptic-scale waves – Feldstein 2003, Woolings et al. 2008) coupled with ocean circulation patterns and sea-surface (Franzke et al. 2011). The importance of these fields in long-term forecasting is evident (Wagner 1989) in the numerous studies that have applied them for this purpose. For example, Johansson (2007) diagnosed prediction skill from two of the most commonly cited teleconnections, the North Atlantic Oscillation (NAO) and the Pacific North American Oscillation (PNA), noting that the forecast skill of the PNA exceeds the NAO and that both are improved by larger values of the PNA indices. Luo and Cha (2012) diagnosed the relationship between the NAO and the variability within the North Atlantic polar jet, which is directly linked to long-term predictability in that region. Lin et al. (2009) even noted a relationship between the NAO and a shorter-temporal scale tropical teleconnection, the Madden-Julian Oscillation (MJO – Madden and Julian 1971). Additionally, smaller-scale studies such as Pablo and Soriano (2007), which defined the relationship between the phase of the NAO and winter lightning in western Europe and DiNezio et al. (2009), which looked at Florida wind patterns as they relate to the NAO, are two of countless examples of applications of teleconnection patterns used in medium and long-term predictability research. Clearly, if improvements of long-term operational forecasts are desired, it is imperative that the predictability of these teleconnection patterns be ascertained.

Teleconnections were initially proposed in the literature by Wallace and Gutzler (1981) through correlation analysis of 500 mb geopotential height fields over the Northern Hemisphere mid-latitudes. The teleconnection patterns were formally identified and named by Barnston and Livezey (1987), which utilized rotated principal component analysis (RPCA) to identify the common modes of variability within 700 mb Northern Hemisphere geopotential height data. The findings of Barnston and Livezey (1987) were updated by Chen et al. (2003) and van den Dool et al. (2000), which utilized empirical orthogonal functions in lieu of RPCA. More recent work conducted by Richman and Mercer (2012) concluded that these previously derived patterns were limited owing to the orthogonality constraint in their methods, as well as a limited sample size and an insufficient number of retained RPCs. Their work relaxed the

orthogonality constraint (via Promax rotation), and updated the previous results using the National Center for Environmental Prediction (NCEP)/National Center for Atmospheric Research (NCAR) Reanalysis Project dataset (hereafter referred to as the NNRP - Kalnay et al. 1996), which provide gridded geopotential height observations on a denser grid than previous studies. Richman and Mercer (2012) also identified new teleconnection patterns and updated the spatial configurations of already established teleconnections. Outside of the most commonly cited of these patterns (e.g. the NAO, the PNA), these new observed modes have had limited consideration in climate simulations, which is noted by Richman and Mercer (2012) in their study limitations.

Some efforts have been undertaken at assessing predictability of these patterns within climate models. Papadimas (2012) used a statistical climate prediction model for monthly Northern Hemisphere geopotential heights using the general concept of teleconnections (without testing for the modes from Richman and Mercer 2012) in order to find relationships between air temperature and sea-level pressure during winter. The resulting SLP oscillations were shown to correlate strongly with the air temperatures of neighboring areas with a lead time of up to two months. Phase five of the Coupled Model Intercomparison Project (CMIP5 – Sheffield 2013) used intraseasonal to multidecadal time scales to assess North American geopotential height variability in a dynamic climate simulation. Their simulations reproduced certain aspects of the geopotential height variability well, while other aspects were rendered poorly by most models. Errors were attributed to unexplained natural variability within the climate model.

While several studies (noted above, as well as many others) have quantified forecast error of teleconnections within climate models, no study has attempted to quantify such errors using an operational forecast model. However, operational models are more familiar and commonly implemented in daily and weekly weather forecasts. This lack of consideration of longer-term variability patterns in operational model verification likely contributes heavily to the poor medium to long-term forecasting ability of meteorologists. As such, the goal of this project is the formal quantification of teleconnection index forecast error from an operational weather model that can be used to educate forecasters about the limited predictability of these features at medium and long time scales.

2. DATA AND METHODS

As the primary goal of this work was quantifying an operational weather model's performance at rendering the major teleconnection modes in the 500 mb height fields, a database of observed teleconnection patterns and simulated teleconnection patterns was required. Each dataset is described in detail below.

2a. Observational Teleconnection Dataset

The derived teleconnection patterns and associated indices from Richman and Mercer (2012) were utilized as the observed verification dataset. In their study, monthly mean January 500 mb geopotential heights from the NNRP were used to derive the primary variability modes. While the Richman and Mercer (2012) study period extended from 1948-2009, issues with the weather model (as described below) limited the simulation period to 1948-1964. To maintain consistency with the simulated study period, the work of Richman and Mercer (2012) was redone for

the shorter period of record, which resulted in slight deviations from what was observed in their original study (not shown).

2b. Model-derived Teleconnection Dataset

In line with the primary objectives of this research, a commonly utilized operational forecast model was required for creation of the model-derived teleconnection dataset. The Weather Research and Forecasting (WRF – Skamarock et al. 2008) model version 3.6 was used as a global climate model for this work. The WRF is a non-hydrostatic mesoscale model that can be configured for global simulations with appropriate global input data. As such, the NNRP data, a global meteorological reanalysis dataset with 2.5° latitude-longitude grid spacing and 17 vertical levels, were used as initialization data for the forecast model. The NNRP contain many important meteorological fields, including base-state fields (such as temperature, humidity, wind velocity components, geopotential heights) and derived fields (heat fluxes, radiative fluxes, precipitation, surface pressure, many others). Additionally, the NNRP have a long period of record that extends from 1 January 1948 to present with new observations provided at 6 hourly intervals. Further, the base-state fields (which are required by the WRF for the global simulation) have high reliability according to Kalnay et al. (1996), ensuring their heavy dependence on observations as opposed to simulated information, supporting using the NNRP as boundary conditions for the WRF.

The forecast model was run for the full Northern Hemisphere at 3° latitude-longitude resolution from 1948 to 2009 (the period of record of Richman and Mercer 2012). However, model instability caused the simulation to prematurely end after 1964, so the final simulation only consisted of 17 years of the full 63-year period. Attempts were made to correct this issue but were unsuccessful (as described in section 4 below). Default model physics parameterizations were selected for the forecast simulation. The global simulation, even at its coarse spatial resolution, required parallel-processed supercomputing resources owing to the duration of the simulation (17 years). This simulation was completed using Blue Waters supercomputing resources, and the project required roughly 500 node hours of simulation time.

Upon completion of the simulation, appropriate analysis was required to retain the simulated teleconnection fields. The simulated January mean 500 mb geopotential height data were used to compute teleconnection indices, maintaining consistency with Richman and Mercer (2012) which was used as the observational dataset. Note that all gridpoints south of 20°N latitude were removed prior to analysis (since geopotential heights are generally uniform in the tropical latitudes). These midlatitude monthly geopotential height fields were used to formulate teleconnection indices following the methodology of Richman and Mercer (2012), which included:

- Interpolation of the simulated geopotential height fields to a Fibonacci grid (Swinbank and Purser 2006 – Fig. 1) that ensured all gridpoints were evenly spaced. Since both observation and forecast fields are provided on latitude-longitude grids, spatial correlations will artificially increase along converging longitude lines unless interpolated to an evenly spaced grid.
- Formulation of an S-mode (Richman 1986) principal component analysis through formulating the correlation matrix on the spatial dimension of the data, eigenanalysis of the correlation matrix, and computation of the RPC loadings (scaling the eigenvectors to unit length).

- Rotation of the RPC-loading vectors using Promax (non-orthogonal) rotation (Richman 1986).
- Truncation of the number of WRF-derived RPC loading dimensions to 8 to maintain consistency with the results of Richman and Mercer (2012).
- Projection of the RPC loadings onto the original data, yielding RPC scores (the teleconnection indices).

This approach yielded forecasted RPC loading patterns, which are spatial representations of the teleconnection patterns, as well as RPC scores, which represent the teleconnection index time series for each map (the phase of the teleconnection).

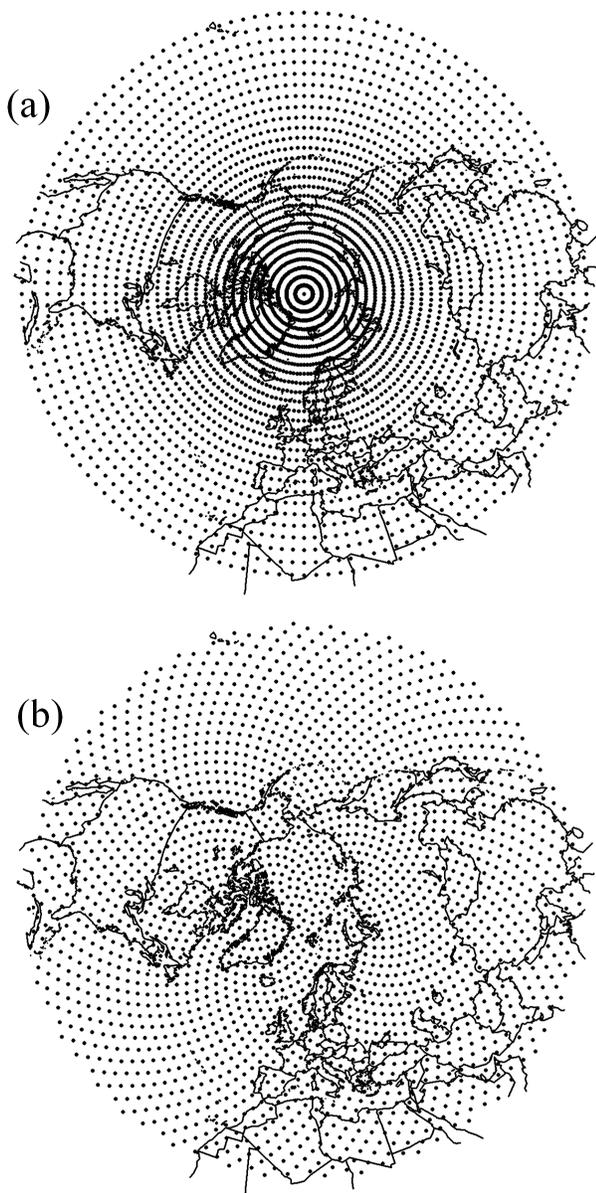


Fig. 1. The NNRP grid (panel a) and the Fibonacci grid from Swinbank and Purser (2006 – panel b) used in this study. Note that the Fibonacci grid retains equal grid spacing across the hemisphere, a requirement when computing spatial correlations such as those used in RPCA.

Once the simulated and observed teleconnection patterns and indices were obtained, the match between patterns was ascertained through a second spatial correlation analysis (for the RPC loading maps) and root-mean square error analysis (for the RPC score indices). The resulting verification statistics are presented below.

3. RESULTS

3.1 RPC loading map results

To ascertain pattern matching between observed and forecasted RPC loading maps, the fields were correlated against each other to identify which simulated RPC best matched the observed patterns. The correlation results (Table 1) revealed that despite retaining 8 RPCs, only four of the 8 (WRF-RPC1, WRF-RPC3, WRF-RPC4, and WRF-RPC5) had a modest correlation with the original 8 RPCs, and many of the observed RPCs correlated with the first and fifth simulated RPCs. These results are discouraging as they suggest that the forecast model is doing a poor job of distributing variance in the geopotential height fields, or that the model distribution of variance is not representative of observations. Additionally, RPC1 and RPC5 in the observed fields are the dipole height couplet over the northern Pacific (the West Pacific Oscillation, WP) and Atlantic (the North Atlantic Oscillation, NAO) Oceans, which have well-understood properties. The subtler variability characterized by the remaining 6 observed RPCs is poorly rendered by the WRF simulations, which is a discouraging result.

Table 1. List of observed (NNRP) RPCs with associated forecasted (WRF) RPC with the highest correlation (and that corresponding correlation value). Values near +/- 1 are ideal.

<u>NNRP RPC</u>	<u>WRF RPC</u>	<u>Correlation</u>
RPC1	RPC1	0.394
RPC2	RPC3	-0.405
RPC3	RPC1	-0.274
RPC4	RPC4	-0.270
RPC5	RPC1	0.421
RPC6	RPC5	-0.237
RPC7	RPC5	-0.351
RPC8	RPC5	-0.210

According to Richman and Mercer (2012), the three most commonly known (and most widely used in North American forecasting applications) patterns from the observed database were RPC1 (the WP), RPC3 (the Pacific North American Oscillation – PNA), and RPC5 (the NAO). Interestingly, the simulated RPC1 correlated most highly with all three of these patterns. Figure 1 shows each of these observed RPCs (panels a-c) and simulated RPC1 (panel d) to provide an example of the WRF model performance.

The observed WP (Fig. 2a) showed a strong positive center located throughout the Arctic Circle and extending over the Gulf of Alaska, with the expected negative region over the north-central Pacific, a key feature of the WP. However, the limited 17-year sample size left this feature a bit farther north than is typically observed for the WP. The simulated RPC1 (Fig. 2d) showed this same maximum over the North Pole, but the associated negative region over the north-central Pacific was notably absent from the pattern. It is likely this feature was

distributed to another simulated RPC or not simply not simulated, representing the variance distribution issue discussed previously.

The observed PNA (RPC3 - Fig. 2b) shows the expected tripole pattern that extends over the Gulf of Alaska, the southwestern United States, and the Northeast. However, traditional renderings of the PNA (Richman and Mercer 2012) show this pattern shifted a bit farther south than is represented by this limited sample size. The simulated RPC1 (Fig. 2d) shows two of the three poles in the PNA, the same maximum over the Gulf of Alaska and the Arctic and a local minimum over the Desert Southwest. However, as was the case with WP, the final piece of this pattern was missing from the simulation (the feature over the Northeast United States), supporting the conclusion that geopotential height variability is being improperly distributed by the WRF.

The observed rendering of the NAO (RPC5 - Fig. 2c) showed the expected maximum over Iceland (the Icelandic low) and the associated minimum over Bermuda and the central Atlantic (the Bermuda high). The smaller sample size revealed has a few other poles in addition to the two primary features which are typically not considered part of the NAO (Barnston and Livezey 1987, Richman and Mercer 2012). The simulated RPC1 (Fig. 2d) showed the maximum over the Bermuda high region and a region of the large positive area over the Gulf of Alaska that extended to cover the Icelandic low region. Many other patterns in the simulated RPC1 were not present in the observed NAO, but the important features of the NAO were sufficiently characterized, which is supportive of high correlation between the observed RPC5 and simulated RPC1 (0.421). Overall, the dipole patterns of the WP and NAO, as well as two of the three poles of the PNA, were represented as a single RPC in the WRF simulated loading field, suggesting variability is not being properly distributed among loadings when conducting global WRF simulations.

3.2 RPC score validation

In addition to a strong match between the simulated and observed RPC loading patterns, a good simulation of the teleconnections should portray the phasing of the patterns (by way of teleconnection indices) with the same success. By definition, the product of the RPC score and loading map is in units of standard anomalies, so that large errors in the RPC scores suggest phasing issues in the monthly mean fields used to derive the teleconnection indices. For example, if a root mean square error between the modeled and observed RPC score series was in excess of 1, it could suggest either a complete reversal of the pattern or an anomalously weak/strong forecast. Consider an RPC of -0.5. With an RMSE in excess of 1, this could mean that point may have a value of either 0.5 or larger (a complete phase reversal) or an RPC of -1.5, which is considerably stronger than the forecasted -0.5. Fig. 3 shows the RMSE of the RPC scores when compared to the map that contained the highest correlation (e.g. comparing the observed RPCs with their associated forecasted RPCs as per table 1). These results were quite poor, which was anticipated given the poor match between the observed and the simulated loading fields. All 8 RPCs had RMSE values in excess of 1, suggesting major phasing issues with the simulated RPC fields over time. Unfortunately, these results demonstrate several issues with using the WRF model for diagnosing teleconnections and suggest not using it for such an application.

4. DISCUSSION AND LESSONS LEARNED (EDUCATIONAL IMPACT)

The primary research objective of this study was the evaluation and validation of WRF-simulated teleconnection indices within Northern Hemisphere 500 mb geopotential height fields. A proper knowledge of these patterns is essential for medium and long-term forecasts, which are heavily based on the phases of these features. The results of the study suggest that alternatives to the WRF (such as global climate models like the NCAR Community Climate System Model – Gent et al. 2011) should be considered when forecasting teleconnection indices as climate-mode configured WRF simulations were inadequate.

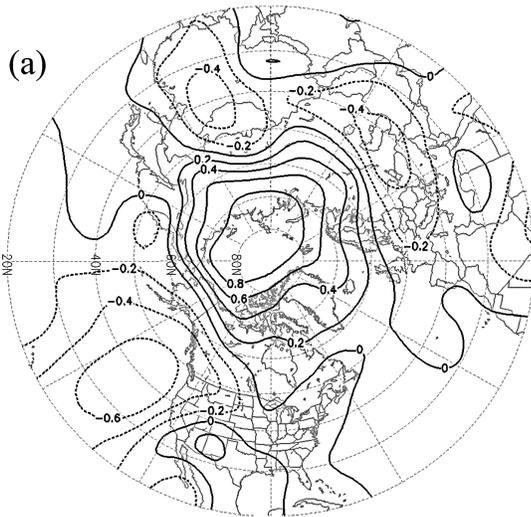
This project was completed as part of the Blue Waters Undergraduate Internship Program. The primary educational component of this research came from the undergraduate student's interaction with the high-performance computing environment, including learning how to work with big-data, the limitations of traditional computing environments with big-data problems, and learning to troubleshoot parallel processed software. Two major computational issues arose when completing the research project:

- 1) The WRF simulations were initially conducted without updating global sea surface temperature (SST) fields. However, the model parameterizations used within the WRF are not designed to update SST through the associated solar radiation model, and as such, global temperatures cooled to an infeasible value by roughly day 100 of the simulation. This required the undergraduate student to update the model with NNRP data as the simulation progressed to maintain realistic hemispheric mean temperature values. This was not ideal, since the NNRP boundary conditions could have artificially improved teleconnection depictions within the simulations (though this ended up not being the case).
- 2) Despite the claims that the WRF can be run in climate mode, the model simulations continued to fail past 1964 of the study period. This failure likely resulted from model instability with the upper-level wave patterns, which is a common problem within numerical weather prediction models with timesteps that are too large. However, lessening the timestep did not fix this issue, suggesting some larger underlying issue within the WRF itself for such an application. It is likely that the shorter sample size affected the results detrimentally, though a 17-year period is sufficient to render all phases of each of the resulting teleconnection patterns multiple times, so this impact is likely small when compared to the results of the study.

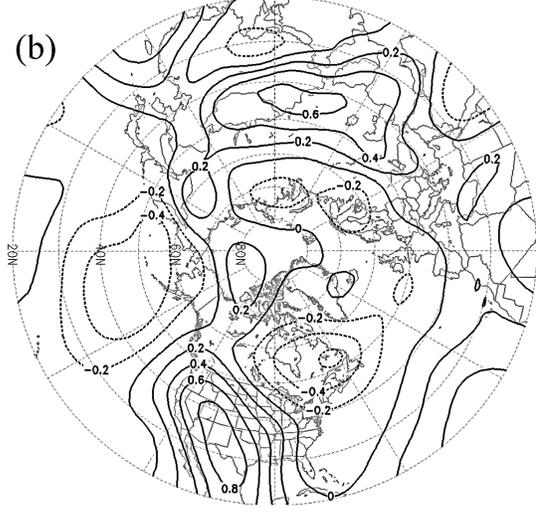
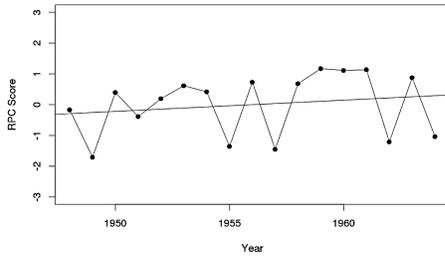
Despite these inherent issues with the implementation of the study, the undergraduate student learned many important lessons regarding climate-scale work in the field of atmospheric sciences. The student learned how to utilize RPCA on global hemispheric data and interpret the resulting variability modes, as well as common practices within operational meteorology for verifying numerical weather prediction models.

This work would have not been possible without the utilization of high-performance computing in the construction of the climate simulation. The use of the Blue Waters supercomputing center's resources helped support this climate simulation, which required a week of simulation time and tens of thousands of files. The resulting simulation produced daily Northern Hemisphere climate output for the full 17-year period, resulting in a dataset of roughly 6300 files and in excess of 70 GB, which cannot be fully analyzed utilizing traditional computing environments. Additionally, the

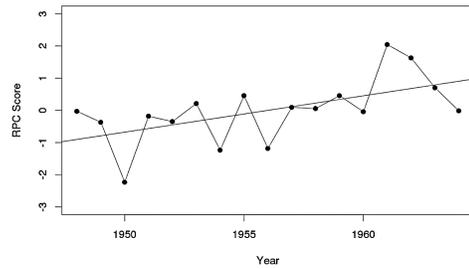
computing time required to complete the simulation exceeded 500 compute node hours, which was infeasible in a traditional computing setting.



RPC1 Time Series for NNRP



RPC3 Time Series for NNRP



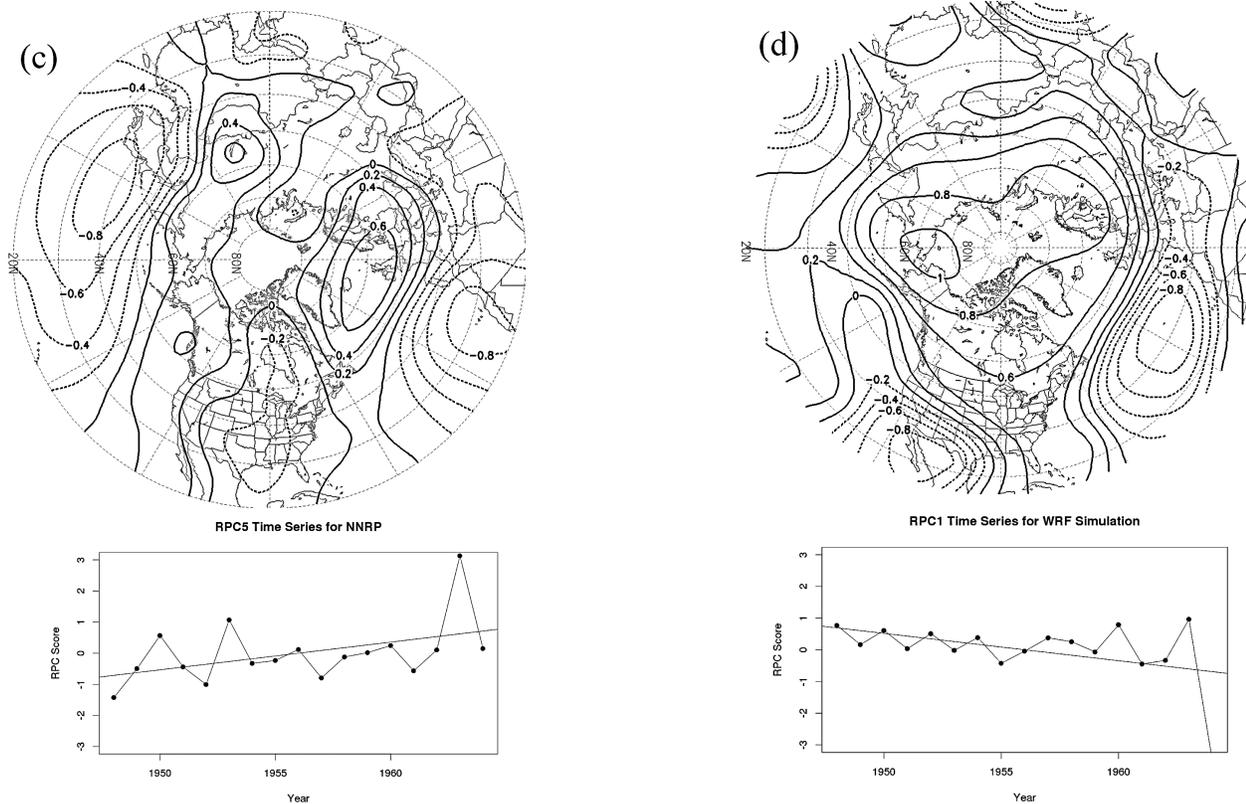


Fig. 2. NNRP loading maps for RPC1 (panel a), RPC3 (panel b), and RPC5 (panel c), and their associate RPC scores. Panel d shows the WRF simulated RPC1, which is most highly correlated with all three of these patterns.

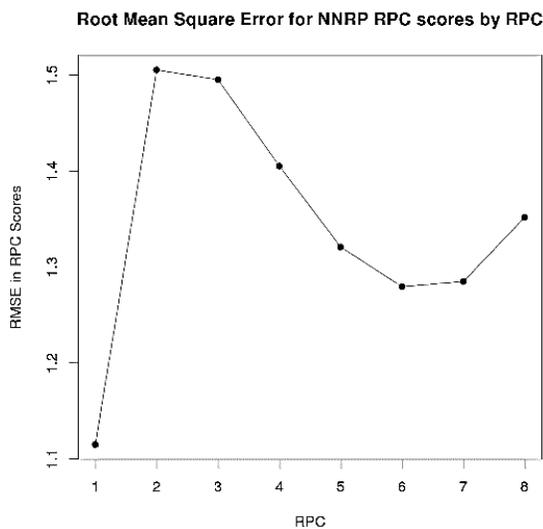


Fig. 3. RMSE for NNRP RPC scores vs. WRF RPC scores, paired based on the results in Table 1.

In addition to demonstrating the value of supercomputing, this work reveals a key issue associated with medium and long-term forecasting, that issue being the inability of a regional, operational forecast model with global capabilities to render teleconnections. Forecasts of teleconnection indices will remain based entirely on trends for the foreseeable future, and the results presented herein

support the need for further development of operational teleconnection forecasts from dynamic weather models.

5. REFLECTIONS

As is the case with any student-led project, it is important to discuss the follow-up educational impact that the project has had on the student’s academic success. Students who wish to conduct research or work in the operational meteorology sector require some prior research experience, and this project provided a unique opportunity to learn weather research using with state-of-the-art technology. Operational weather offices are more frequently utilizing multi-core parallel processing computing environments to complete mesoscale weather simulations, and this experience has offered the student an edge over others competing for graduate degree positions. Additionally, this research experience has provided the student the necessary experience to be successful once beginning to work in graduate meteorology. The opportunity to complete undergraduate research is a rare one in meteorology, and the opportunity afforded by the Blue Waters Summer Internship Program will ultimately allow the student to stand out among her peers competing for jobs.

6. SUMMARY AND CONCLUSIONS

The study of teleconnections and their links to long-term weather forecasting has been an area of ongoing research over the past few decades. Research has shown that teleconnections have skill in forecasting longer-term (in excess of a week) processes (Johnson

et al. 2014, Jones et al. 2011, Hamill and Kiladis 2014, many others). To this point, most operational forecasts of teleconnection indices have been limited to simple trend analysis, with no study considering the value of using an operational forecast model in recreating teleconnection indices, despite their known importance for medium and long-range forecasting. As such, the primary goal of this project was to determine any limitations associated with deriving teleconnection indices from simulations from a well-known regional forecast model run in global climate mode. This performance is a strong indicator of possible utility of such a model in medium and long-range forecasting.

Model performance was determined through a 17 year (1948-1964) simulation of Northern Hemispheric midlatitudes flow patterns computed using the WRF model. Geopotential heights at 500 mb were retained following the methodology of Richman and Mercer (2012) to formulate teleconnection maps and indices on the simulated fields. These patterns were directly compared with observation patterns derived from 500 mb NNRP, again following the methodology of Richman and Mercer (2012). The resulting RPC loadings for both the simulated and observed teleconnections were compared to ascertain WRF model performance at recreating the most commonly observed teleconnections. Additionally, RMSE computed on the RPC score time series was computed to quantify timing issues with the onset of given phases of each teleconnection pattern.

The resulting WRF simulated teleconnection maps showed generally weak correlation with NNRP (observed) RPCs, distributing variance presented in the NNRP fields among the different WRF simulations (Table 1). An example of these issues was presented associated with the simulated RPC1 (Fig. 2), which was modestly correlated with three different observed RPCs and contained some of the features of those patterns while removing others. The lack of correlation with the observed fields and the missing poles within the simulated teleconnections suggested the WRF is not retaining the proper variability structure in the 500 mb height fields.

The RPC score RMSE plot (Fig. 3) showed several concerning issues regarding the simulation of the patterns as well. All values exceeded 1, meaning that it was likely phases of the patterns were completely flipped, which suggested the WRF simulations were out of phase with the NNRP patterns. This was particularly concerning for forecast purposes, since different phasing of the teleconnection pattern lead to completely different forecast conclusions (e.g. a positive NAO suggests one forecast outcome, a negative NAO a different outcome).

Overall, these results suggested that major deficiencies remain in long-term hemispheric-scale WRF simulations, as the 500 mb geopotential height fields are a primary representation of dominant atmospheric flow patterns. It is likely that the WRF was formulating Rossby wave patterns with an improper wave phase speed when conducting longer-term simulations, and future work could help reveal this issue by using pattern recognition techniques. Additional future work involving longer simulation periods (e.g. the full period of record of the NNRP) will help reveal if the model is more stable at longer climate time-scales. Despite the limited model performance, operational forecast models such as the WRF are the best opportunity at rendering operational teleconnection forecasts that in turn will reveal patterns in medium and long-term flow variability, and further improvements to these models will likely alleviate the issues presented herein.

7. ACKNOWLEDGEMENTS

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. This work was a direct result of an undergraduate internship opportunity from the Blue Waters organization, and we are grateful for their support throughout the completion of this research. We also wish to thank the anonymous reviewer whose review helped improve the quality and scope of the manuscript considerably.

8. REFERENCES

- Barnston, A. & Livezey, R. (1987), Classification, seasonality and persistence of low-frequency atmospheric circulation patterns. *Monthly Weather Review*, Vol. 115, No. 6, pp. 1083-1126.
- Chen, W., and H. van den Dool (2003), Sensitivity of Teleconnection Patterns to the Sign of Their Primary Action Center. *Monthly Weather Review*, Vol. 51, pp. 2885-2899.
- DiNezio, P., et al (2009), Observed Interannual Variability of the Florida Current: Wind Forcing and the North Atlantic Oscillation. *Journal of Physical Oceanography*, Vol. 39, 721-736.
- Feldstein, S. (2003), The dynamics of NAO teleconnection pattern growth and decay. *Quarterly Journal of the Royal Meteorological Society*, Vol. 129, pp. 901-924.
- Franzke, C., et al. (2011), Synoptic analysis of the Pacific-North American teleconnection pattern. *Quarterly Journal of the Royal Meteorological Society*, Vol. 137, pp. 329-346.
- Gent, P., et al. (2011), The Community Climate System Model Version 4. *Journal of Climate*, Vol. 24, pp. 4973 – 4991.
- Hamill, T., and G. Kiladis (2014), Skill of the MJO and Northern Hemisphere Blocking in GEFS Medium-Range Reforecasts. *Monthly Weather Review*, Vol. 142, pp. 868-885.
- Johansson, A. (2007), Prediction Skill of the NAO and PNA from Daily to Seasonal Time Scales. *Journal of Climate*, Vol. 20, pp. 1957-1975.
- Johnson, N., D. Collins, S. Feldstein, M. L'Hereaux, and E. Riddle (2014), Skillful Wintertime North American Temperature Forecasts out to 4 Weeks Based on the State of ENSO and the MJO. *Weather and Forecasting*, Vol. 29, 23-38.
- Jones, C., L. Carvalho, J. Gottschalck, and W. Higgins (2011), The Madden-Julian Oscillation and the Relative Value of Deterministic Forecasts of Extreme Precipitation in the Contiguous United States. *Journal of Climate*, Vol. 24, 2421-2428.
- Kalnay, E., and coauthors (1996). The NCEP/NCAR 40-year reanalysis project. *Bulletin of the American Meteorological Society*, Vol.77, No.3, pp. 437-471.

- Lin, H., et al. (2009), An Observed Connection between the North Atlantic Oscillation and the Madden-Julian Oscillation. *Journal of Climate*, Vol. 22, pp. 364-380.
- Luo, D., and J. Cha (2012), The North Atlantic Oscillation and the North Atlantic Jet Variability: Precursors to NAO Regimes and Transitions. *Journal of the Atmospheric Sciences*, Vol 69, pp. 3763-3787.
- Madden, R., and P. Julian (1971), Description of a 40-50 day oscillation in the zonal wind in the tropical Pacific. *Journal of Atmospheric Science*, Vol. 28, pp. 702-708.
- Pablo, F., and L. Soriano (2007), Winter Lightning and the North Atlantic Oscillation. *Monthly Weather Review*, Vol. 135, pp. 2810-2815.
- Papadimas, C.D., et al. (2012), "Sea-Level Pressure-Air Temperature Teleconnections during Northern Hemisphere Winter." *Theoretical and Applied Climatology*, Vol. 108, pp. 173-189.
- Richman, M. (1986), Review article: rotation of principal components. *International Journal of Climatology*, Vol.6, No.3, pp. 293-335.
- Richman, M. and A. Mercer (2012). Identification of Intraseasonal Modes of Variability Using Rotated Principal Components, *Atmospheric Model Applications*, Dr. Ismail Yucel (Ed.), pp. 273-296.
- Sheffield, Justin, et al. (2013), "North American Climate in CMIP5 Experiments. Part II: Evaluation of Historical Simulations of Intraseasonal to Decadal Variability." *Journal of Climate* 26.23, pp. 9247-9290.
- Skamarock, et al (2008). A Description of the Advanced Research WRF Version 3. NCAR Technical Note, 125 pp.
- Swinbank, R. & Purser, J. (2006). Fibonacci grids: a novel approach to global modelling. *Quarterly Journal of the Royal Meteorological Society*, Vol.132, No.619, pp. 1769-1793.
- van den Dool, H., et al. (2000), Empirical Orthogonal Teleconnections. *Journal of Climate*, Vol. 13, pp. 1421-1435.
- Wagner, A. (1989), Medium- and Long-Range Forecasting. *Weather and Forecasting*, Vol. 4, pp. 413-426.
- Wallace, J., and D. Gutzler (1981), Teleconnections in the Geopotential Height Field during the Northern Hemisphere Winter. *Monthly Weather Review*, Vol. 109, 784-812.
- Woolings, T., et al (2008), A New Rossby Wave-Breaking Interpretation of the North Atlantic Oscillation. *Journal of the Atmospheric Sciences*, Vol. 65, 609-626.

TABLE OF CONTENTS

Introduction to Volume 8 Issue 3	1
<i>Steven I. Gordon, Editor</i>	
Using Inexpensive Microclusters and Accessible Materials for Cost-Effective Parallel and Distributed Computing Education	2
<i>Joel Adams, Suzanne J. Matthews, Elizabeth Shoop, David Toth, and James Wolfer</i>	
Innovative Model, Tools, and Learning Environments to Promote Active Learning for Undergraduates in Computational Science and Engineering	11
<i>Hong Liu, Andrei Ludu, Jerry Klein, Michael Spector, and Matthew Ikle</i>	
Computational Approaches to Scattering by Microspheres	19
<i>Reed Hodges, Kelvin Rosado-Ayala, and Maxim Durach</i>	
Toward simulating Black Widow binaries with CASTRO	25
<i>Platon I. Karpov, Maria Barrios Sazo, Michael Zingale, Weiqun Zhang, and Alan C. Calder</i>	
Using Blue Waters To Assess Non-Tornadic Outbreak Forecast Capability by Lead Time	30
<i>Taylor Prislovsky and Andrew Mercer</i>	
Blue Waters Supercomputing Applications in Climate Modeling with the WRF Model	36
<i>Morgan Smith and Andrew Mercer</i>	