The goal of my research is to advance our ability to design, build, and study computational ecosystems that use intelligent systems to support a process of networked interactions among people and machines, resulting in new goods and services. My research bridges the fields of Social & Crowd Computing, Human-Computer Interaction (HCI), Artificial Intelligence (AI), and Decision Science.

Social and crowd computing systems are now ubiquitous. At home and on-the-go, millions of people use social media to share personal experiences, peer economies to provide and access physical services, online learning communities to develop new skills, and peer production sites to collaborate on tasks. Popular accounts highlight how these systems connect us with others to our mutual benefit, in ways that bring us closer, provide ease and convenience to our lives, help us learn, and increase our productivity. Yet despite recent advances, fundamental challenges persist in collaboration and coordination. Through four main areas of contribution, my work addresses four core challenges and provide novel socio-technical solutions:

1. **Facilitating community-based planning.** Community leaders regularly make complex planning decisions that affect the whole community, but lack ways to learn about people’s needs and tools for planning effectively. Despite best efforts, plans that result often fail to reflect the preferences of community members and can create suboptimal experiences for many. Addressing this challenge, I developed a novel community-based planning approach that engages many participants to communicate their needs and helps organizers produce outcomes that better serve the community.

2. **Coordinating opportunistic actions.** Physical crowdsourcing systems often rely on dedicated volunteers and paid workers who are provided with strong incentives to perform tasks outside of their existing routines. My work introduces new methods and tools that enable large numbers of qualified people to contribute through their existing routines, in ways that scale new services and advance individual and communal goals.

3. **Providing readily available learning experiences.** Online learning communities significantly expand the pool of advanced beginners, but lack support for addressing critical gaps in knowledge and experience that remain between advanced beginners and professionals. I introduce novel methods and tools that bridge these gaps to provide new opportunities for authentic learning online.

4. **Scaling cognitive apprenticeship.** Both online and offline, a small number of dedicated experts often have the onus of supporting many who need help. We lack ways of supporting many learners in complex domains for which coaching and mentoring would be highly effective. I address this by developing processes, models, and tools for scaling cognitive apprenticeship that greatly expand our ability to train large numbers of people in complex learning environments.

These fundamental challenges in social and crowd computing cannot be solved with technology alone, and neither can they be solved by simply asking people to do more for others. My work pioneers the design of self-supporting, computational ecosystems that interweave community process, social structures, and intelligent systems to bring forth people and machines to collectively take small parts to overcome larger challenges (see Figure 1). Computational ecosystems define new ways of structuring how we go about doing, present participants with new collaboration and interaction opportunities, and provide tools to support these opportunities. Computational ecosystems produce goods and services that help us achieve desired outcomes, and also promote a more participatory culture in which people become members of an emerging community that they helped support and create.

My team and I take an integrative approach to our design, technology, and research work. With each project, we conceptualize new ways of interacting, devise technical solutions that enable such interactions,
design and deploy systems that solve real-world problems using our novel concepts and solutions, and conduct formative and summative studies that help us build our understanding and generalize our solutions over time. To perform our work effectively and at scale, I pioneered Agile Research Studios, a new, sustainable model for structuring a research program and learning environment to greatly expand training and participation in academic research.

1 Community-based Planning of Large Events

The first track of my research aims to transform the way we plan large events. I demonstrate how community-informed planning (CIP) can significantly ease the load on organizers, engage many participants to communicate their needs and wants, and produce outcomes that better serve the community as a whole.

The task of planning large events involving thousands of people can be arduous, time-consuming, and shortsighted. Organizers often lack information about the diverse preferences, constraints and domain knowledge held by community members. Current tools for plan optimization focus primarily on satisfying easy-to-encode hard constraints, and lack support for managing the size, complexity, and latent considerations of the broader community. As an illustrative example, consider the task of scheduling presentations for a large academic conference. Traditionally, a handful of organizers produce the conference schedule with minimal computational support for resolving conflicts and little input from community members. Despite significant efforts from organizers, final schedules often contain incoherent sessions, similarly-themed sessions that run in parallel, and author- and attendee-specific conflicts.

To address this challenge, I created a computational ecosystem for community-informed planning that integrates: (a) interfaces that facilitate the large-scale distributed collection of knowledge, preferences, and constraints from various community members, (b) methods to encode and present community-provided information as actionable feedback to organizers, and (c) mixed-initiative interfaces—systems that support humans while running machine optimizers behind the scene—to facilitate effective planning activities. As an illustrative success story, my work with Rob Miller, Steven Dow, Juho Kim, and others applied this CIP approach to reinvent how we plan large conferences. With Cobi, we engage various stakeholders in the planning process: program committee members group papers in their area of expertise; authors relate their papers to other papers; and attendees identify talks of interests. These community data are encoded into a mixed-initiative scheduling system used by conference chairs to iteratively refine the schedule based on system recommendations and their own knowledge and expertise (see Figure 2). In deployments at premier conferences in human computer interaction (CHI 2013-14) and social computing (CSCW 2014-
Cobi transformed conference scheduling and produced three key outcomes: instead of just a handful of organizers, our process engaged 1,500+ community members throughout various phases of planning; our mixed-initiative solution reduced organizers’ planning time from 100 hours to 5 hours; and organizers produced better schedules by resolving hundreds of previously undetected conflicts. This research [9, 2, 6, 3] on community-informed conference planning received a best paper honorable mention at ACM CHI 2014 and a notable paper award at HCOMP 2013. Given the success of our deployments, Cobi is now part of ACM CHI’s standard process for planning the schedule for its 3000+ attendees.

My research demonstrated the many benefits of community-informed planning and addressed a number of core challenges in realizing these benefits. The core data problem is the organizers’ lack of information about community members’ preferences, constraints, and domain knowledge, and their inability to collect such information in a timely and actionable manner so as to affect the plan. To promote participation throughout multiple phases of planning, I designed a socio-technical process that I call incentive chaining, which promotes broad participation by gathering data that supports planning while also using gathered data to promote further contributions from others. For example, the affinities generated by program committee members produces a short list of similar papers for authors; this list helps authors focus their attention rather than scan through hundreds of accepted papers [2]; these data provide the seeds for our recommendation engine for attendees, avoiding the “cold-start problem” faced by other collaborative filtering systems [3].

Without appropriate tools, event planners often find it difficult to make changes to a preliminary plan to resolve conflicts without creating new ones. The core technical problem is thus the lack of tool support to flexibly leverage community input, machine intelligence, and organizer expertise to resolve conflicts and address community preferences. In order to effectively leverage the wealth of community input from a CIP process, I developed community-informed, mixed-initiative interfaces that put organizers in the driver’s seat and provide a holistic view of all constraints and preferences. Behind the scene, constraint solvers suggest

---

**Figure 2:** Cobi’s scheduling interface recommends moves and swaps based on community-sourced data.
planning actions to help resolve conflicts and address preferences. For example, to account for noisy and incomplete community-sourced data, Cobi’s scheduling tool [9] empowers organizers to make informed scheduling decisions based on optimizer-suggested moves and their own tacit knowledge (see Figure 2).

Our work also explored opportunities for community members to be directly involved in the planning process. With Frenzy [6], program committee members collaboratively form sessions while guided by actionable feedback that highlights needed contributions. Frenzy addresses both the need for drawing on the diverse expertise of many community members early, and for more careful deliberation and discussion to resolve conflicts later. This is accomplished by presenting different actionable feedback at different phases of problem solving, so that the system encourages independent crowd work early when constraints are ‘loose,’ and smaller group work later when constraints are ‘tight’ and conflicts are likely to arise. In this manner, Frenzy demonstrates an effective way to flexibly move between crowdware and groupware models of collaboration to support changing needs over the course of a community-based problem solving process.

2 Coordinating Opportunistic Actions to Produce Globally Effective Behaviors for Physical Crowdsourcing

The second track of my research aims to enable physical crowdsourcing systems that can tap into the daily physical routines of billions of people on-the-go to map the world in exquisite new detail, accomplish small tasks for one another, and provide just-in-time support when people need it the most.

The growth of mobile devices in recent years has helped to bring about physical crowdsourcing systems that connect people to tasks and lower barriers to participation. Citizen science and communitysensing projects recruit volunteers to track migration patterns, report city problems, and refine maps. Peer economies such as TaskRabbit and Uber coordinate the provision of physical resources on-demand, and in doing so, are transforming entire service sectors.

Despite successes, existing systems are limited by the fact that contributions are gathered via one of two approaches: opportunistic or directed. Opportunistic approaches receive input from users when and where they decide to contribute. This can lower participation barriers, but has the downside that systems cannot make demands for particular data to be collected or particular tasks to be completed. Directed approaches prompt users for specific input to advance system goals, which often require contributions that are largely outside people’s existing routines and as a result strong incentives are required. The benefit is that contributions more readily achieve system goals; the challenge is recruiting large enough numbers of motivated users to go out of their way to contribute.

With the support of a Microsoft FUSE Labs research award, my work in this area advances a hybrid approach to physical crowdsourcing that brings about the best elements of both opportunistic and directed approaches. Our approach develops computational ecosystems that permits access to a large number of people who when directed to make small, convenient contributions through their existing routines, contribute actions that jointly achieve system goals.

In one direction, I am developing interaction techniques that support people reporting objects and events of interest in physical environments while they are on-the-go. In order to tap into the mobility routines of people in transit, my students Zak Allen, Stephen Chen, Nicole Zhu, and I developed a set of low-effort sensing interactions that allow people to report event locations and provide small pieces of additional information in just seconds, without stopping, and in some cases, without having to take out their phone [1, 4]. Compared to existing approaches that rely on more dedicated volunteers to fill out complete reports, our approach seeks to broaden participation by breaking sensing goals into smaller tasks to reduce the burden on individual contributors. Zak presented our work on the Gaze system at the CHI 2015 Student Research Competition, placing 3rd in the undergraduate student category.
Continuing in this direction, my recent work on habitsourcing explores opportunities to leverage the habit-building activities of millions of people to collect sensing data about the physical environment they encounter while going for a run or a walk [12]. My students Katherine Lin, Henry Spindell, Yongsung Kim, and I developed sensing through actuation, a new interaction technique for adding immersion and enjoyment into habit-building apps in ways that elicit useful environmental data. Figure 3 shows an example interaction from Zombies’ Interactive, an exergame based on the popular running app Zombies, Run!. In this example, an audio cue prompts a runner to escape zombie hordes by sprinting to the next fire hydrant. The desired interaction result in sharp changes in accelerometer readings; inferring object locations is easy with simple pattern detection and does not require advanced machine learning algorithms. Our studies show that our two habitsourcing apps, Zombies’ Interactive and ZenWalk, are as preferred or more preferred than their non-interactive counterparts, and further, that the sensing data inferred from user actions accurately represents the environment.

Beyond sensing, my student Yongsung Kim and I are designing interactions for on-the-go physical tasking, whereby people are asked to complete small tasks for others that are conveniently along their way. Our pilot study of Libero [10]—a system for peer-to-peer package delivery—showed that sending just-in-time notifications helped promote delivery, but other factors, such as reciprocity, community building, and social obligation were also important drivers for promoting participation. To investigate the effect of task notification strategy on participation, we conducted a follow-up study [11] with 16 participants and found that small increases in notification radius led to order of magnitude increases in the number of people reached, but at the same time, affected other factors such as interest in ways that lead to order of magnitude decreases in the likelihood of task pickup. This finding reveals a core challenge in on-the-go crowdsourcing to effectively manage the tradeoffs imposed by being too aggressive in recruitment—which can be overly disruptive and result in a low task pickup rate—and being too restrictive in recruitment—which can involve too few helpers, overburden the ones that are involved, and leave a disproportionately large number of task demands unfulfilled. Yongsung presented our work at the CHI 2015 Student Research Competition, placing 1st in the graduate student category.

In addition to on-the-go crowdsourcing, we are also exploring opportunities to engage ad-hoc crowds that gather at large events. As one thread of research, my student Leesha Maliakal and I are working on CrowdCheer, a system for providing motivational support for marathon runners by engaging a crowd of spectators who provide just-in-time cheers to runners. Typically, supporters tend to cheer at the beginning and end of a race, but not during pain points where the runners need motivation most. To effectively coordinate the provision of live, crowdsourced support, we designed a socio-technical architecture for supporting continual crowdsourcing ecosystems that monitor user state, trigger on state changes, identify potential helpers, present helping context, provide help, and monitor outcomes. Leesha presented our early work on CrowdCheer at the Grace Hopper ACM Student Research Competition, placing 2nd across a field of 118 students.

Together, my work in this area introduced a variety of interaction techniques that effectively direct people to make small, convenient contributions through their existing activities. As elaborated in my NSF
Cyber-Human Systems grant currently under review, to fully realize the benefits of our hybrid approach, my upcoming work aims to complement our interaction designs with decision-theoretic frameworks that opportunistically recruit a well-motivated and minimally disrupted crowd to efficiently achieve complex system goals. Specifically, we proposed to develop (1) a 4X framework for scaffolding data collection through phases that continually expand both resolution and spatial coverage; (2) a decision-theoretic, hit-or-wait framework for deciding whether to send someone a task now or to wait for the person to potentially approach a more important task later; and (3) a supply management framework for optimizing and adjusting community-level policies that manage available helper resources to meet system goals and adhere to desired community outcomes.

Sidebar: Remote Paper Prototype Testing
Our work in physical crowdsourcing requires us to design and test mobile interactions in realistic use cases. In order to learn about effective designs quickly, my student Kevin Chen and I created new methods and tools for remote paper prototype testing (RPPT) [5]. With RPPT, designers facilitate and wizard a testing session from the lab while a user tests the lo-fi prototype remotely. This makes it possible for mobile app designers to observe users interacting with paper prototypes of mobile apps in realistic scenarios. Compared to current practices in which testing in realistic scenarios tend to occur after developing a high-fidelity prototype—which may take weeks or months to build—RPPT allows mobile app designers to gather valuable user feedback on lo-fi prototypes that can be built in just a few hours.

This work is supported by an NSF CISE Research Initiation Initiative award. Building on the success of our initial prototype and pilot, our latest work significantly improved the usability of RPPT and enabled designers to wizard a wide range of common mobile interactions. We also recently open sourced our code and made publicly available the latest version of the RPPT tool. To further expand the use cases for RPPT, our next steps will incorporate and support the remote testing of mixed-fidelity prototypes in which functioning mobile interface components that are hard to wizard can be readily mixed into a paper prototype.

3 Readily Available Learning Experiences for Budding Web Developers
The third track of my research aims to provide authentic learning experiences for budding web developers. Online learning platforms such as Codecademy attract millions of novice learners and significantly expand the pool of advanced beginners. Yet, critical gaps in knowledge and experience remains between advanced beginners and professionals. Current online platforms provide few resources for progressing from these first steps to writing production-quality software. Professional training programs exist but cost tens of thousands of dollars and are thus not accessible to most.

My student Josh Hibschman and I are developing models and tools that provide new opportunities for authentic learning for millions of learners on the Web. Specifically, we aim to support advanced beginners learning from professional websites of personal interest. Professional examples offer the following learning advantages: they contain rich details that are missing from training examples, their content relates to the real world, and their richness provide opportunities to think in the models of the discipline. But despite the abundant availability of front-end code, most professional code examples are complex; they may be difficult for learners to understand as is and contain superfluous details that distract learning core concepts.

As a first step toward helping developers learn from professional examples, we introduced Unravel, a tool to help developers find entry points into the code for understanding UI features on professional websites [7]. Built as an extension of the Chrome Developer Tools, Unravel provides affordances for quickly tracking and visualizing HTML changes, JavaScript method calls, and JavaScript libraries. To handle large observations of events, Unravel’s UI provides affordances to reduce, sort, and scope observations. Testing Unravel with 13 web developers on 5 large-scale websites, we found a 53% decrease in time to discovering
At 65,000 lines of code, it’s difficult for a developer to understand how features on Tumblr’s homepage are implemented. But with Telescope, a developer can focus in precisely on relevant portions of code that relate to a feature, and see clearly the connections between front-end interactions and the HTML/CSS/Javascript code that produced that interaction.

the first key source behind a UI feature and a 32% decrease in time to understanding how to fully recreate the feature.

While Unravel and other tools allow for inspection and line-by-line analysis of source code, the magnitude of source code on professional websites can easily overwhelm the curious user. Our recent work moves beyond finding entry points into code to helping learners see an accurate picture of a UI feature’s underlying implementation. We introduce Telescope, a platform for discovering how HTML/CSS/Javascript work together to support a UI interaction [8]. Figure 4 shows the Telescope interface for helping a developer understand the scrolling feature on Tumblr’s homepage. With a few simple interactions, a Telescope user can quickly focus in on the less than 100 lines of relevant code out of the 65,000 lines of code on Tumblr’s homepage that produces the feature. To support fine-tuned discovery and exploration of relevant code, Telescope provides affordances that help users scope by Javascript execution time and detail. Further, it draws visual links to help users see clear connections between front-end interactions and the HTML/CSS/Javascript code that produced that interaction.

The Telescope platform serves as the core enabling technology for realizing the vast opportunities of learning from the entire corpora of websites on the Internet. Not only can we now design tools to address various inquiries about professional web sites, we can begin to envision the creation of Readily Available Learning Experiences (RALEs) that support the diverse learning goals of many individuals by making readily available resources for learning that are valuable to the individuals who want it. For example, can we provide personalized learning pathways that provide learners with deep structure for learning concepts embedded in professional websites, and provide scaffolds to help learners practice concepts by recreating examples or using them in their personal projects? To achieve these and other ambitious goals, we are using our understanding of computational ecosystem design to develop processes, interactions, and tools for growing a learning resource for individuals into a learning resource for a community of learners. Specifically, our upcoming work will explore (1) how traces from people’s interactions with Telescope can be used to create searchable indices into professional examples, (2) how micro-authoring and micro-curation efforts from one
The fourth track of my research aims to greatly expand our ability to train novice researchers. My goal is to significantly improve the quality of learning through effective mentorship and support, afford order of magnitude increases in the number of students who receive authentic research training, and lower the barrier to participation in authentic research experiences.

Cognitive apprenticeship provides a powerful model for training young researchers, but imposes an enormous orchestration burden on a single faculty researcher who can only mentor a handful of students at a time. Novice researchers lack regulation skills, i.e., cognitive, motivational, emotional, metacognitive, and strategic behaviors for reaching desired goals and outcomes; in a complex learning scenario such as learning to conduct academic research, they are challenged by demands to perform self-directed planning and to adopt effective help-seeking and collaboration strategies to overcome challenges. Training and support for developing regulation skills is necessary but time-consuming for mentors and difficult to support with software alone. Scaling cognitive apprenticeship for research training thus requires overcoming the “1:X” challenge of allowing one teacher to successfully respond to the learning needs of many students.

To overcome the 1:X challenge, my core innovation is the design of Agile Research Studios [15], that: (1) adapt agile methodologies from software development for the purposes of learning Socially Shared Regulation of Learning (SSRL) and conducting scientific research; (2) provide social structures and processes for SSRL training and support without reliance on a single teacher and that draw from professional best practices; and (3) use cyberlearning technologies to extend the research studio by supporting learners throughout the SSRL cycle using scaffolds, quicker feedback, and quicker connections (see Figure 5). In doing so, we solve challenges of 1:X by: (1) reducing the cognitive load on the teacher; (2) reconceptualizing the role of students as self directed learners; and (3) changing learning spaces and places to support learning both in
Agile research studios are computational ecosystems in which the research process, social structures, and tools work together to support community members learning to conduct research. For example, sprint planning promotes novice research teams to develop SSRL skills so that learners orchestrate their own learning; Special Interest Group (SIG) meetings and studio meetings further support communities of learners co-regulating their learning. In studio meetings, students use the pair research tool [13] to match with other students who can best help each other on their projects. Outside of in-person meetings, students ask and receive help from members across the studio via Slack.

The Design, Technology, Research (DTR) program, a research learning environment I created and have directed since Spring 2014, uses components of Agile Research Studios to support a research learning community centered around my research interests. Over the last two years I have hosted six studio sessions with 4 graduate students and 32 undergraduate students. Students iteratively designed, built, and tested 18 new socio-technical systems. 40% of DTR students are female; student retention beyond their first quarter is near 100%, and most students continue until they graduate. DTR student papers have been accepted for publication at premier conferences in HCI and won 1st, 2nd, and 3rd place at the ACM CHI and ACM Grace Hopper Student Research Competitions. DTR students have received 19 Undergraduate Research Grants and $30,000+ in outside funding for their projects. DTR received a Murphy Society Award in each of the last two years for its role in advancing undergraduate engineering education.

DTR illustrates some of the advantages of adopting Agile Research Studios over traditional methods of orchestrating research training. For my undergraduate students, participation in DTR led to significant gains in student learning about design, technology, and research, increases in innovation self-efficacy, and positive feelings of belonging in a supportive community. For my graduate students, DTR provides authentic practice in mentoring and gives them an opportunity to learn to lead their own research group. For me, DTR greatly expands the scope of my research program and empowers me work with large numbers of talented students despite having similar time and resource constraints as most other junior faculty.

To support my continuing efforts to advance the design and study of Agile Research Studios, I have applied for an NSF Cyberlearning grant with Matt Easterday and Liz Gerber that is currently under review.

Future Directions and Research Impact

The growth of mobile devices, wearable computing, AI, and Internet-based technologies will continue to change the ways we interact with one another and the world around us. But technology itself is not enough for realizing the opportunities that my work pursues, and likely won’t be for quite some time. Instead, we need principles and methods for designing sustainable ecosystems that use technology as a bridge to connect us to do what none of us nor technology can do alone, and to build connections that strength our communities beyond completing the immediate task at hand. My future work will continue to advance the breadth and depth of our understanding of how to design, build, and study computational ecosystems, with a particular focus on expanding access to a broader participant base.

Our interest in community-informed planning is expanding to more broadly focus on participatory, end-to-end collective action, in which a crowd or community identifies opportunities, formulates goals, brainstorm ideas, develops plans, mobilizes, and takes action. In early work [14] with Aaron Shaw, Andrés Monroy-Hernández, Sean Munson, Benjamin Mako Hill, Liz Gerber, and others, we described five stages of Computer Supported Collective Action and identified the intersections and barriers between stages that need to be addressed in order to support end-to-end collective action. In ongoing work, my students Kevin Chen, Shannon Nachreiner, Ryan Madden, and I are developing a Collective Experience API to power new

---

See my teaching statement for more details on DTR.
platforms that will allow us to create, share, and engage in collective experiences with others locally and around the world as readily as we would share and consume information via social media today.

We are expanding our physical crowdsourcing research to examine new ways of interacting with our physical environment. Our own work on on-the-go sensing and the rapid development of machine learning and computer vision technologies open the door for building new classes of applications that can transform the way we interact with the physical world on a daily basis. As one direction, my students Shawn Caeiro and Jennie Werner are exploring interactions, algorithms and architectures that will allow us to interact programmatically with objects in our physical environments in interesting ways via new forms of affordance aware computing.

Complementing our work on learning from professional examples, we are looking to support novice developers working on their personal projects. Currently, it is rare to find tutorials addressing learners’ exact projects and generating a full-fledged tutorial for a single learner is prohibitively expensive. Learners often receive insufficient guidance and struggle without the help of more experienced peers and mentors. Together with my students Katie George, Greg Kim, and Nikhil Pai, we are attempting to bridge this gap by providing beginners with personalized action plans that more experienced web developers can ideally create in just minutes. We are taking a design-based research approach and also exploring socio-technical solutions for effectively reusing and repurposing action plans.

Finally, we are continuing to advance Agile Research Studios as a scalable model for structuring research learning environments. To complement effective social structures for mentoring and coaching, I am particularly interested in developing tools that help students develop their regulation skills over time. In one direction, my students Leesha Maliakal, Bomani McClendon, and Sameer Srivastava are designing and evaluating a sprint planning tool for research learners that promote novice researchers to consider their larger research goals when planning and revising their weekly sprints. Longer term, I am broadly interested in bringing the benefits of Agile Research Studios to a larger community of researchers, and advancing our designs to support diverse needs and use cases.

My research, to date, has resulted in 30 publications with over 670 citations, leading to an h-index of 15 (using Google Scholar). My work has been published at premier venues in HCI (CHI, UIST, interactions), AI (AAAI, IJCAI), Social and Crowd Computing (CSCW, HCOMP), and the Web (WWW). Our current research activities are supported by an NSF CISE Research Initiation Initiative award, a Microsoft FUSE Labs research award, two Murphy Society Awards, an Office of the Provost Award for Digital Learning, and 19 undergraduate research grants. To support our upcoming research activities, I am the PI of two currently pending NSF grants – CHS: Small: Coordination of Opportunistic Actions to Produce Globally Effective Behaviors for Physical Crowdsourcing, 2016–2019; EXP: Agile Research Studios: Scaling Cognitive Apprenticeship to Advance Undergraduate and Graduate Research Training in STEM, 2016–2019. I also plan to submit an NSF Cyberlearning grant to support our work on Readily Available Learning Experiences, as well as an NSF CAREER grant on my larger research agenda in computational ecosystem design.

References


