Networking Unplugged: Hands-On, Inquiry-Based Learning for Undergraduate Networking Courses

Aaron Gember-Jacobson
agemberjacobson@colgate.edu
Colgate University
Hamilton, New York, USA

Abstract

This paper presents a collection of hands-on, inquiry-based learning activities for introducing undergraduate students to important concepts in networking without using computers. Our *networking unplugged* activities—inspired by the Computer Science (CS) Unplugged approach of using kinesthetic-based puzzles and games to introduce CS concepts—frame important concepts in networking as physical communication tasks between humans. By brainstorming and experimenting with different ways of completing the tasks, students "discover" and compare important networking algorithms/protocols. Over the past decade, we have used these activities multiple times at different levels of the curriculum at a primarily undergraduate institution. Based on our informal observations, the activities have a positive impact on student engagement and understanding.

1 Introduction

Hands-on learning in undergraduate networking courses often involves simulating, measuring, or implementing networking concepts [11, 16]. However, many such labs/projects can now be completed with (the assistance of) GenAI, which reduces students' learning, creativity, and critical thinking. Across higher education, instructors are questioning the role/value of out-of-class assignments, and looking for new ways to engage students in the classroom.

This paper presents a collection of hands-on, inquiry-based learning activities for introducing undergraduate students to important concepts in networking without using computers. The activities are inspired by the Computer Science (CS) Unplugged approach of using kinesthetic-based puzzles and games to introduce CS concepts to students as early as primary school [1, 9]. Each of our *networking unplugged* activities frames an important concept in networking as a physical communication task between humans—e.g., you can only communicate by passing written messages to people sitting next to you. Students must brainstorm and experiment with different ways of completing the communication task. The task/constraints are designed such that students' solutions can be mapped to actual algorithms/protocols used in networking—e.g., flooding, source routing, and link state routing are different ways of determining the path a packet should traverse to reach its destination. Each activity is designed to be completed in small groups in 20–40 minutes during a class session.

Over the past decade, we have used these activities multiple times at different levels of the curriculum at a primarily undergraduate institution with class sizes of 20-30 students. Based on our informal observations, students are more engaged during these activities and can better articulate the advantages and disadvantages of different approaches to networking challenges compared to when other forms of in-class learning (e.g., lectures, worksheet problems [14], Q&A) are used. Additionally, students demonstrate creativity and critical thinking during these activities, which are important skills for computer scientists [12]. While the activities are suitable for *introducing* important problems and solutions in networking, the activities do not teach students specific algorithms or protocols in detail. Hence, the activities should typically be followed by other learning activities (e.g., textbook readings, lectures, worked examples, labs/projects focused on implementation or measurement).

2 Background & Related Work

Inquiry-based learning: This form of active learning compels students to discover concepts themselves as they seek to answer a question or solve a problem. Inquiry-based learning has repeatedly been shown to be more effective at improving academic achievement and developing problem-solving skills compared to the traditional method of teaching science through lecture, worked examples, and individual problem solving [10]. Additionally, inquiry-based learning "is related to a student's perceived gains in science and technology understanding, intellectual development, and vocational preparation" [8]. Our networking unplugged activities are best categorized as *information-oriented, structured inquiry* learning, because they focus on existing disciplinary knowledge, and the question is defined by the instructor [17].

CS Unplugged: This pedagogical technique was developed to introduce foundational CS concepts to primary school and older students through hands-on activities that don't use computers [1, 9]. The activities are designed to be entertaining (e.g., a "magic trick" based on parity bits), kinesthetic (using physical objects and movement), and encourage students to "discover" concepts themselves. The collection of activities published in 2015 [3] includes activities for several topics typically covered in an introductory computer networks course: protocols and reliability, minimal spanning trees, routing, and asymmetric-key encryption. The CS Unplugged activities on protocols and reliability ("Tablets of Stone") and routing ("The Orange Game") inspired the intra-domain routing unplugged activity discussed in Section 3.4, whose success led us to develop the other networking unplugged activities discussed in Section 3.

Analogies: Analogies are often used to teach computer science [2, 4, 7, 13, 15]. Our networking unplugged activities use physical communication tasks between humans as analogies for commonly used approaches to addressing specific problems in networking.

Methods for teaching networking: Prvan and Ožegović's survey of methods used to teach computer networks [11] includes: problem-based learning, which is a form of inquiry-based learning [10]; games, which is one of the key characteristics of CS Unplugged activities [9]; and analogies. Many problems and games used to teach networking focus on rectifying specific failure scenarios or designing networks/protocols to meet specific requirements. In contrast, our networking unplugged activities are designed to introduce students to important networking algorithms/protocols at a high-level through the lens of communication between humans without computers. Iyer and Murthy's use of analogical problem solving to teach networking to non-majors [6] is most similar to our work.

3 Activities

Our *networking unplugged* activities frame important concepts in networking as communication tasks between humans in a physical setting with no computers. Each activity is designed to be completed during a class session in 20–40 minutes and includes three phases: (1) the instructor explains the problem/scenario; (2) small groups (2–6 students) are formed, and each group works independently to develop and evaluate multiple possible "solutions;" and (3) the instructor facilities a discussion in which students share their ideas and the instructor relates these to algorithms/protocols used in networking. In this section, we present activities for four different topic areas: link layer encoding, link layer multiplexing and access control, addressing, and routing. Detailed guides for each activity are available at [5].

3.1 Encoding Unplugged

This activity focuses on the problem of transmitting bits of information from one device to another using electromagnetic waves. Through this activity, students "discover" a variety of wired and wireless encoding techniques. Before completing this activity, students should know that information is represented using bits and transmitted over cables or free space using electricity, light, or radio waves.

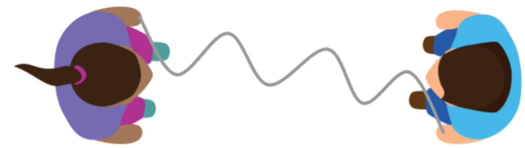

Figure 1: Students doing encoding unplugged

Each group (2+ students) is given a piece of rope ≈3m long and must determine how a person can manipulate the rope on one end to transmit a short sequence of bits to a person holding the rope on the other end (Figure 1). Students' ideas, and their networking equivalents, typically include:

- keeping the rope still to transmit a 0 and moving the rope to transmit a 1 ≈ non-return to zero (NRZ)
- moving the rope side-to-side slowly for a 0 and quickly for a 1 ≈ binary frequency shift keying (2-FSK)
- moving the rope side-to-side slightly for a 0 and significantly for a 1 ≈ binary amplitude shift keying (2-ASK)
- moving the rope to the left for a 0 and to the right for a 1 ≈ binary phase shift keying (2-PSK)

When students try out these different forms of movement, they discover the importance of agreeing on: (1) which pattern represents a zero versus a one, and (2) how long the rope should be manipulated for each bit so the receiver doesn't interpret a sequence of two of the same bit as a single bit.

To encourage further exploration, students can be asked to transmit two bits with a single manipulation pattern. This leads to students "discovering" approaches like 4-FSK, 4-ASK, and quadrature amplitude modulation (QAM).

3.2 Multiplexing Unplugged

This activity focuses on the problem of multiple senders and receivers communicating over the same cable or free space. Through this activity, students "discover" a variety of multiplexing and access control techniques for wired and wireless networks. Students should be introduced to circuit- and packet-switching before completing this activity.

Each group (6+ students) must determine how multiple pairs of people can successfully engage in separate conversations while sharing the same physical space. Group members should be seated or standing in a circle, and each person should be paired with the person across from them (Figure 2). Each conversation should involve a back-and-forth exchange of questions and answers, and pairs can only communicate using sound—no hand signals, written messages, etc.

Common ideas, and their networking equivalents, include:

- take turns in a circle ≈ time division multiplexing (TDM)

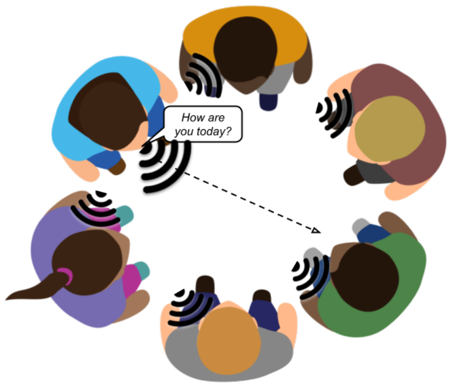

Figure 2: Students doing multiplexing unplugged

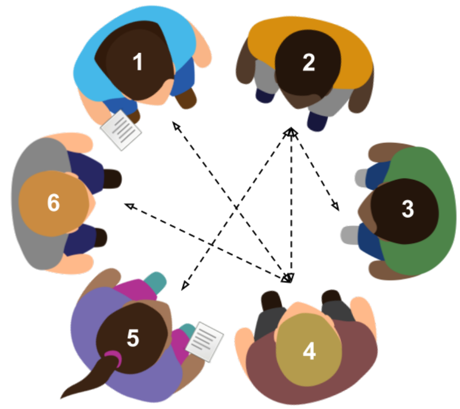

Figure 3: Students doing routing unplugged

- establish an order of speakers before starting to communicate ≈ scheduling (in cellular networks)
- use different pitches or types of sound (e.g., talking, clapping, tapping) ≈ frequency division multiplexing (FDM)
- start talking if no one else is talking; if two people start talking at the same time, both people stop talking and wait before trying again ≈ carrier sense multiple access with collision detection (CSMA/CD)

Ideas that violate the restriction of only using sound but adhere to the problem in principle include:

- raise your hand and wait to be called on ≈ request-to-send/clear-to-send (RTS/CTS)
- pass an object around the circle; a person can only talk when they are holding the object ≈ token passing

To encourage further exploration, students can be asked to consider: (1) what if someone has nothing to say when it is their "turn" to speak? (2) what if someone wants to talk for a long time before giving someone else a turn to speak? (3) what if there were more pairs of people that wanted to communicate? These can lead to discussions about TDM versus statistical TDM, having a limited number of frequencies (i.e., channels), getting less "air time" and/or needing to wait longer before being able to speak again, etc. The activity can also be extended by having a group member introduce "noise" (i.e., interference) that prevents a person from hearing their partner and necessitates asking their partner to repeat their message.

3.3 Addressing Unplugged

This activity focuses on the problem of specifying a packet's intended destination. Through this activity, students "discover" the benefits, limitations, and relationships between different network addressing systems. Students should know about packet-switching before completing this activity.

The following question is posed to students: if you wanted a taxi driver to take you to a particular restaurant, how would you specify your destination? Students' ideas, and their network equivalents, typically include:

- a street address ≈ Internet Protocol (IP) addresses
- the restaurant's name ≈ domain names
- a restaurant type ≈ top-level domains (TLDs) or anycast
- step-by-step directions ≈ source routing
- latitude and longitude coordinates ≈ MAC addresses

Students also often suggest providing a nearby landmark, for which we have not identified a networking equivalent.

These analogies can continue to be used when explaining addressing and routing in detail, such as:

- class-based addressing—e.g., a street address without a house number is like a class C address
- DNS records—e.g., a restaurant with multiple locations is like a domain name with multiple A records
- Address Resolution Protocol (ARP)—e.g., a street address can be converted to latitude and longitude coordinates like an IP address can be resolved to a MAC address
- port numbers—e.g., apartments or suites within a building are like processes within a host
- autonomous systems (ASes)—e.g., major cities each have their own public transit system like each AS manages its own intra-domain routing
- routing entries—e.g., highway signs indicate where exits lead to like routing entries map subnets to next hops

3.4 Intra-domain Routing Unplugged

This activity focuses on the problem of determining the sequence of routers or switches a packet should traverse to reach its destination. Through this activity, students "discover" a variety of intra-domain routing algorithms.

Each group (6+ students) must determine how to pass written messages to their intended recipient in a setting where each group member is only allowed to pass messages to some other group members. Each group member is assigned a number and provided with a list of group members (identified by number) to whom they can pass messages. We recommend the permitted message passing between group members is initially defined to form a loop-free topology (Figure 3). Individuals are not allowed to verbally share this information, but they may create a written message with this information. Additionally, any group member can modify, and/or duplicate written messages they receive from other group members. Each message includes the number of the individual to whom it is destined.

Common ideas, and their networking equivalents, include:

- a person adds their number to a message before passing it along to one of the individuals to whom they can pass messages and whose number is not yet listed in the message ≈ self-avoiding random walk
- a person passes along a message to all individuals to whom they can pass messages, duplicating the message as needed ≈ flooding
- a person creates a message with a list of individuals to whom they can pass messages, and gives a copy of this to each of these individuals; when a person receives such a message, they write down this information in their notebook and pass the message to all individuals to whom they can pass messages, duplicating the message as necessary; when a person receives a "real" message, they use the information written in their notebook to determine whom to pass the message to ≈ link state routing

Students are often not creative enough to "discover" learning switches or distance vector routing, but these can be discussed after students have come up with their own ideas.

To encourage further exploration, the instructor can give groups messages with specific sources/destinations that "stress test" their ideas—e.g., a message that must be passed along many times. Additionally, the permitted message passing between group members can be updated to form a topology that can result in messages being passed in a loop and demonstrates the limitations of naive flooding.

4 Classes & Observations

We have used these activities multiple times at different levels of the curriculum at a primarily undergraduate institution with class sizes of 20-30 students. We have used the activities most frequently in an upper-level elective on computer networks, which we have taught six times over the past decade. In this course, we have used routing unplugged (the first activity we developed) five times, multiplexing unplugged four times, encoding unplugged three times, and addressing unplugged (our newest activity) once. We have also used routing unplugged and addressing unplugged a few times in other courses: an introductory course for non-majors called "The Unreliable Internet"—which focuses on how the internet works and the ways it is susceptible to failures, cyberattacks, and censorship—and a mid-level Introduction to Computer Systems course required for computer science majors. In all instances, we have followed the activities with lectures, worked examples, and/or worksheet problems that cover the relevant networking approaches in more detail.

During the pandemic, we used routing and addressing unplugged in courses that were taught synchronous online. We adapted routing unplugged to have students send chat messages to each other instead of passing messages on paper. The online version of this activity actually worked better than the in-person version, because students couldn't visually observe which group members were exchanging messages. No adaptations were required to use addressing unplugged in a synchronous online course. Multiplexing unplugged could also be used in this setting without any adaptations.

Based on our informal observations, these activities are engaging and fun for students. The kinesthetic nature of these activities—especially encoding unplugged—seems to improve students' emotional engagement, and the use of physical analogies seems to improve students' cognitive engagement. Moreover, students seem to better explain and compare the relevant networking concepts after completing these activities versus engaging in other forms of in-class learning (e.g., lectures, worksheet problems [14], Q&A). Additionally, the activities foster creativity and critical thinking, which are important skills for computer scientists [12]. In the future, we hope to formally assess the activities' impact on student engagement and understanding.

5 Conclusion

Inquiry-based learning has been shown to positively impact students' problem-solving skills and understanding of concepts in science and technology [10]. Our four *networking unplugged* activities frame important concepts in networking—encoding, multiplexing, addressing, and routing—as physical communication tasks between humans. By brainstorming and experimenting with different ways of completing the tasks, students "discover" and compare important networking algorithms/protocols. Based on our informal observations of the activities' positive impact on student engagement and understanding, we hope instructors at other institutions will leverage our publicly available facilitation guides [5] to integrate one or more of these activities in their courses.

References

[1] [n. d.]. CS Unplugged. https://www.csunplugged.org.

[2] Robert M. Belcher, Wesley P. Yeatman, and Ryan E. Dougherty. 2026. Gamified Learning and Instructional Analogies for Theory of Computing Courses. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education (SIGCSE) V.1.* Association for Computing Machinery, New York, NY, USA, 107–113. doi:10.1145/3770762.3772502

[3] Tim Bell, Ian H. Witten, Mike Fellows, Robyn Adams, Jane McKenzie, Matt Powell, and Sam Jarman. 2015. CS Unplugged:

An enrichment and extension programme for primary-aged students. https://classic.csunplugged.org/documents/books/english/CSUnplugged_OS_2015_v3.1.pdf.

[4] Yingjun Cao, Leo Porter, and Daniel Zingaro. 2016. Examining the Value of Analogies in Introductory Computing. In *Proceedings of the 2016 ACM Conference on International Computing Education Research (ICER)*. Association for Computing Machinery, New York, NY, USA, 231–239. doi:10.1145/2960310.2960313

[5] Aaron Gember-Jacobson. 2026. Networking Unplugged Activities. https://github.com/agember/netunplugged.

[6] Sridhar Iyer and Sahana Murthy. 2013. Demystifying networking: teaching non-majors via analogical problem-solving. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE)*. Association for Computing Machinery, New York, NY, USA, 77–82. doi:10.1145/2445196.2445223

[7] Andrew Manches, Peter E. McKenna, Gnanathusharan Rajendran, and Judy Robertson. 2020. Identifying embodied metaphors for computing education. *Computers in Human Behavior* 105 (2020), 105859. doi:10.1016/j.chb.2018.12.037

[8] Linda B Nilson. 2016. *Teaching at its best: A research-based resource for college instructors*. John Wiley & Sons.

[9] Tomohiro Nishida, Susumu Kanemune, Yukio Idosaka, Mitaro Namiki, Tim Bell, and Yasushi Kuno. 2009. A CS unplugged design pattern. *SIGCSE Bulletin* 41, 1 (March 2009), 231–235. doi:10.1145/1539024.1508951

[10] Michael Prince and Richard Felder. 2007. The many faces of inductive teaching and learning. *Journal of college science teaching* 36, 5 (2007), 14.

[11] Marina Prvan and Julije Ožegović. 2020. Methods in Teaching Computer Networks: A Literature Review. *ACM Transactions on Computing Education* 20, 3, Article 19 (June 2020), 35 pages. doi:10.1145/3394963

[12] Junaid Qadir, Kok-Lim Alvin Yau, Muhammad Ali Imran, and Ala Al-Fuqaha. 2020. Engineering Education, Moving into 2020s: Essential Competencies for Effective 21st Century Electrical & Computer Engineers. In *2020 IEEE Frontiers in Education Conference (FIE)*. IEEE Computer Society, Los Alamitos, CA, USA, 1–9. doi:10.1109/FIE44824.2020.9274067

[13] Joseph P. Sanford, Aaron Tietz, Saad Farooq, Samuel Guyer, and R. Benjamin Shapiro. 2014. Metaphors we teach by. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE)*. Association for Computing Machinery, New York, NY, USA, 585–590. doi:10.1145/2538862.2538945

[14] Chandrika Satyavolu. 2020. Worksheets to Foster Active Learning in Computer Networks. In *Online Networking Education Community Discussion: Teaching and Learning Computer Networking During the Pandemic and Beyond*.

[15] Pawan Saxena, Sanjay Kumar Singh, and Gopal Gupta. 2023. Achieving Effective Learning Outcomes through the Use of Analogies in Teaching Computer Science. *Mathematics* 11, 15 (2023). doi:10.3390/math11153340

[16] Jürgen Schönwälder, Timur Friedman, and Aiko Pras. 2017. Using Networks to Teach About Networks (Report on Dagstuhl Seminar 17112). *SIGCOMM Computer Communication Review (CCR)* 47, 3 (Sept. 2017), 40–44. doi:10.1145/3138808.3138814

[17] Rachel Spronken-Smith, Rebecca Walker, Julie Batchelor, Billy O'Steen, and Tom Angelo. 2011. Enablers and constraints to the use of inquiry-based learning in undergraduate education. *Teaching in Higher Education* 16, 1 (2011), 15–28. doi:10.1080/13562517.2010.507300