Understanding Computer Networks Within an Hour: Evaluation of a Visual Simulator and Autonomous Supports Under Self-Learning Conditions

Yuki Kitamura
The University of Osaka
Osaka, Japan
u961432b@ecs.osaka-u.ac.jp

Hiroyuki Nagataki
The University of Osaka
Osaka, Japan

Tomonari Kishimoto
Otemon Gakuin University
Osaka, Japan

Susumu Kanemune
Osaka Electro-Communication University
Osaka, Japan

Shizuka Shirai
The University of Osaka
Osaka, Japan
shizuka.shirai.cmc@osaka-u.ac.jp

***Abstract*—This study evaluates "ProtoSim 2," an interactive learning environment that enables beginners to grasp fundamental network mechanisms, in a self-learning setting without teacher support. High school students now study how computer networks function, but, unlike in specialized university CS courses, they lack sufficient time to become proficient in operating general-purpose simulators. An educational simulator embedded in ProtoSim 2 visualizes data flow so that the key concepts of each TCP/IP layer stand out, helping learners focus on essential ideas. ProtoSim 2 also offers autonomous-learning support features, such as hints and automatic feedback on questions, which let students complete exercises independently both in class and at home. In an evaluation with 18 undergraduate and graduate students from non-CS majors, every participant finished the materials within one hour without external assistance. Pre- and post-test comparisons across all four TCP/IP layers showed significant gains, demonstrating that ProtoSim 2 enables efficient acquisition of a broad range of network concepts in a short time.**

***Index Terms*—Interactive learning environment, Computer science education, K-12, Self-learning, Visualization, Autonomous learning, Computer network.**

I. Introduction

In today's world, computers are ubiquitous and information technology has advanced rapidly, making it commonplace for children to engage with computers in everyday life. Consequently, novice learners, including high school students as well as computer science majors, must grasp the fundamental workings of computer networks so that they can use them effectively. Indeed, several national curricula in places like England, South Korea, New Zealand, and Japan have already integrated foundational network concepts into high school courses [1]–[5]. For instance, Japan's compulsory high-school subject "Informatics I" includes a unit on computer networks whose objective is to deepen students' understanding of real-world network mechanisms, from physical components such as routers to the multi-layered TCP/IP protocol suite [5], [6].

Despite this inclusion, learning networking remains challenging: they are technical, complex, dynamic, abstract, and invisible [7]–[12]. Hands-on practices that involve laboratory experiments or interactive simulations have been proposed at the university level to help students grasp these abstract mechanisms more concretely. They are generally effective for CS majors, who can devote ample time to such exercises, whereas high school students must cover many CS topics rapidly and therefore face severe time constraints. If learners could work on well-structured exercises at home without direct teacher support, the range of possible study methods would expand dramatically.

This paper focuses on a case in which "ProtoSim 2," an interactive learning environment designed to let high school students autonomously learn the mechanisms of computer networks, was used in an independent study setting without teacher supervision. Although ProtoSim receives continual updates, this study's evaluation experiment employed ProtoSim 2, the system's second major release. ProtoSim 2 contains a virtual network simulator that visualizes the flow of information along each layer of TCP/IP in an intuitive manner that even novices can understand. It is also bundled with an electronic textbook, and learners can cover the entire material by operating the simulator in line with that content. Missions to be accomplished through the simulator are also provided, and the system automatically delivers feedback based on both the learner's actions in the simulator and the text answers they enter. These features are expected to foster autonomous learning among high school students. Previous works have shown that students can quickly begin the exercises without any confusion about how to operate ProtoSim 2 when it is used in class under a teacher's supervision. This study investigates whether its autonomous learning support functions operate effectively and the learning objectives are fully achieved when participants use ProtoSim 2 independently outside the classroom context.

II. Related Works

Many researchers have introduced diverse teaching strategies for computer network mechanics [13]. A survey of American university syllabi shows that most networking courses incorporate laboratory work [14]. As technology advances, however, scholars argue that instructional methods must evolve [15]. Recent innovations include collaborative project based learning [16], blended learning formats [17], and IoT based exercises [18]. The online resource “Computer Science Field Guide” [19], which follows the tradition of CS Unplugged [20], also contains learning content of network.

Simulators used as interactive learning environments have been examined from several perspectives. Regarding motivation, Montagud et al. [21] observed that tools such as Cisco Packet Tracer [22] heightened enthusiasm among students. For conceptual understanding, Goldstein et al. [12] reported significant gains in pre- and post-tests when an active learning framework built around Packet Tracer was employed. Concerning problem solving, Zhang et al. [23] showed that the same tool fostered practical networking skills. Although some worry that virtualisation reduces authenticity, Chamberlin et al. [24] found no learning outcome differences between physical labs and virtual labs using Packet Tracer. Frezzo et al. [25] theoretically explained the benefits of features such as automatic scoring in Packet Tracer, and Smith et al. [26] extended the tool to support remote collaborative learning.

Some studies go further by enhancing existing simulators or creating new ones. Saha et al. [27] developed “ns2web,” a web interface that removes installation hurdles for Network Simulator 2. Pullen [10] conducted hands-on activities with the open source Java Network Workbench 2 (JNW2). Rahman et al. [7] recommended combined use of Delite (for wide area modelling), ns-3, and the topology generator Brite [28]. Ruiz-Martínez et al. [29] created VNUML-UM, enabling practice with concepts such as mobility and high availability. Wannous et al. [30] built NVLab, a web system for designing networks and managing virtual machines. Jevremovic et al. [31] introduced IPA-PBL, an e-learning tool for IP address configuration in problem-based settings.

These related works primarily target specialized, university-level courses, where learners can devote ample time and draw on support from instructors and teaching assistants. Within that environment, students are able to master sophisticated network simulators designed for expert use. However, such generous time allocations and expert guidance are unrealistic for the high school setting. Furthermore, some studies employ general-purpose simulators yet do not clearly specify the sequence in which learners should engage with the exercises. Consequently, high school students struggle to learn about networks within limited classroom time and, when attempting to study at home, cannot realistically do so on their own.

ProtoSim 2, by contrast, is designed for novices who must grasp network fundamentals in a limited timeframe. It is equipped with an educational simulator that emphasizes and visualizes key points essential for learning rather than precisely simulating networks. Coupled with a problem-based electronic textbook, the system is intended to guide students through the material smoothly and independently, even when teacher support is limited or unavailable. However, despite these expectations, its effectiveness in self-study settings has never been empirically assessed. Accordingly, this paper empirically investigates how well ProtoSim 2 supports learners under self-study conditions.

III. ProtoSim 2

A. Objectives and Design Principles

In response to the need for high school students to comprehensively learn network mechanisms within a short time, ProtoSim 2 was developed with two primary objectives:

O1. **Conceptual understanding of all four TCP/IP layers.** Students should be able to grasp both the overall structure and the specific roles of each layer, namely:

- Application Layer: recognising that communication involves a server as the peer entity
- Transport Layer: understanding segmentation of data into packets and retransmission control
- Internet Layer: understanding routing performed by routers
- Network-Interface Layer: learning how data are converted to physical carriers and how noise is detected

O2. **Efficient and independent completion in limited time.** Students should be able to finish all learning content on their own, in under one hour, without significant confusion about how to operate the teaching materials.

The necessity of covering every layer specified in O1 has received little attention so far. Some conventional simulators target only a narrow topic such as routing. When different tools are employed for each layer, instructors must assemble multiple tools to address every learning item, and students need extra time to master each tool’s operation. Such fragmentation also hinders students from relating concepts across layers and forming an integrated understanding of network behaviour.

Similarly, only limited effort has been devoted to achieving O2, supporting independent learning. Most existing simulators are general-purpose tools designed for expert users. Because they assume no specific instructional sequence, they provide little guidance for novices’ independent learning.

Taking the above objectives and their associated challenges into account, design principles of ProtoSim 2 were formulated so that each principle aligns with, and directly addresses, a specific objective, as follows:

DP1. Provide a comprehensive simulator that visualizes network mechanisms while highlighting the key properties that students must understand at each TCP/IP layer.

DP2. Supply an electronic textbook aligned with the simulator that offers autonomous learning supports, including contextual hints and automatic feedback.

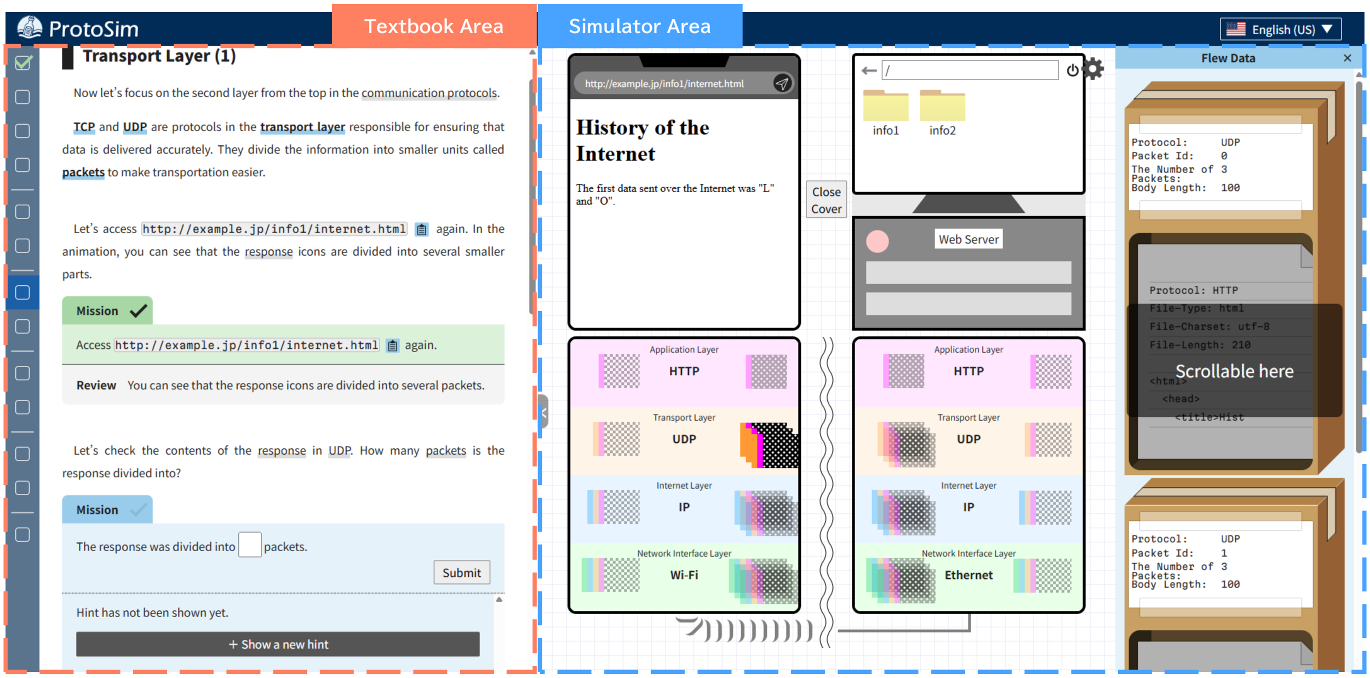

Fig. 1. The main screen of ProtoSim 2 composed of a simulator area and textbook area.

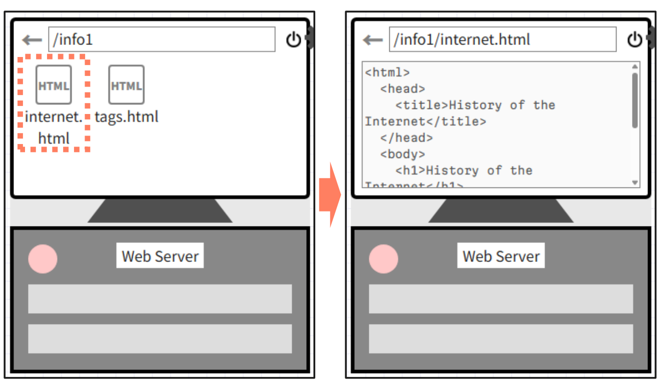

Fig. 2. Browsing files on the web server, illustrating how URL paths correspond to the directory structure.

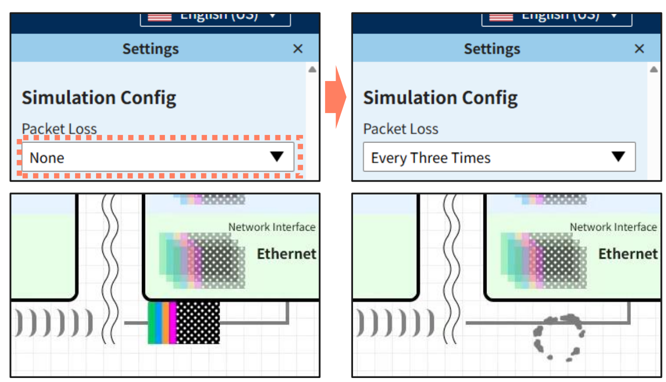

Fig. 3. Animation showing packets exploding when packet loss is enabled.

B. Implementation

Guided by these design principles, ProtoSim 2 was built as a web-based interactive learning environment where students can explore the workings of the TCP/IP protocol suite firsthand [1]. As the dashed boxes in figure 1 show, the interface is divided into a simulator area that animates data moving through the TCP/IP layers and a textbook area that supports autonomous learning with problem-based exercises.

The simulator area shows an animation of icons which represent data flowing through the TCP/IP layers when a user enters a URL into the virtual smartphone. The left edge of each icon is colored to indicate the header, making it clear that only the header increases or decreases while the data itself remains unchanged. After the animation finishes, clicking the traces of icons in each layer allows users to inspect the data contents in a panel on the right side of the screen. There, the characteristics of the data at that layer are illustrated in a metaphorical way, such as depicting packets as cardboard boxes.

Regarding the application layer, users can browse the files stored on the Web server, as shown in figure 2. This function helps them understand that website data is transferred from a specific communication partner, namely the Web server, and it also lets them confirm how a URL path corresponds to the directory structure on the server.

Regarding the transport layer, users open the settings screen from the gear icon in the upper right corner and enable packet-loss simulation. When this setting is active and a URL is entered, the page does not appear and an animation of exploding packets plays, as illustrated in figure 3. Switching

[1]ProtoSim 2 is accessible at https://protosim.csle-lab.org/.

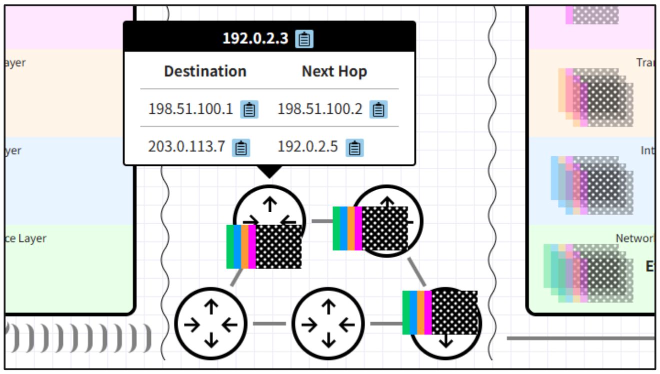

Fig. 4. Routers revealed between the smartphone and the web server, with their routing tables displayed when the cursor hovers over each router.

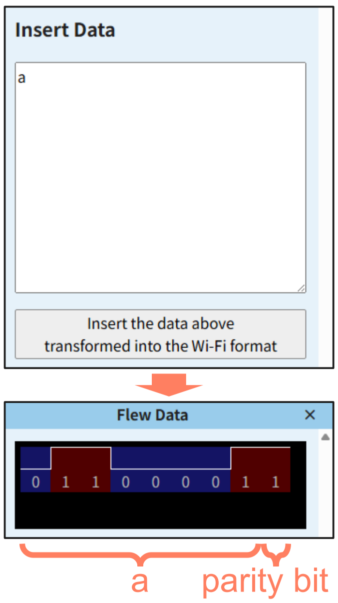

Fig. 5. Experiment on the network interface layer where a single character of data triggers the addition of a parity bit for error detection.

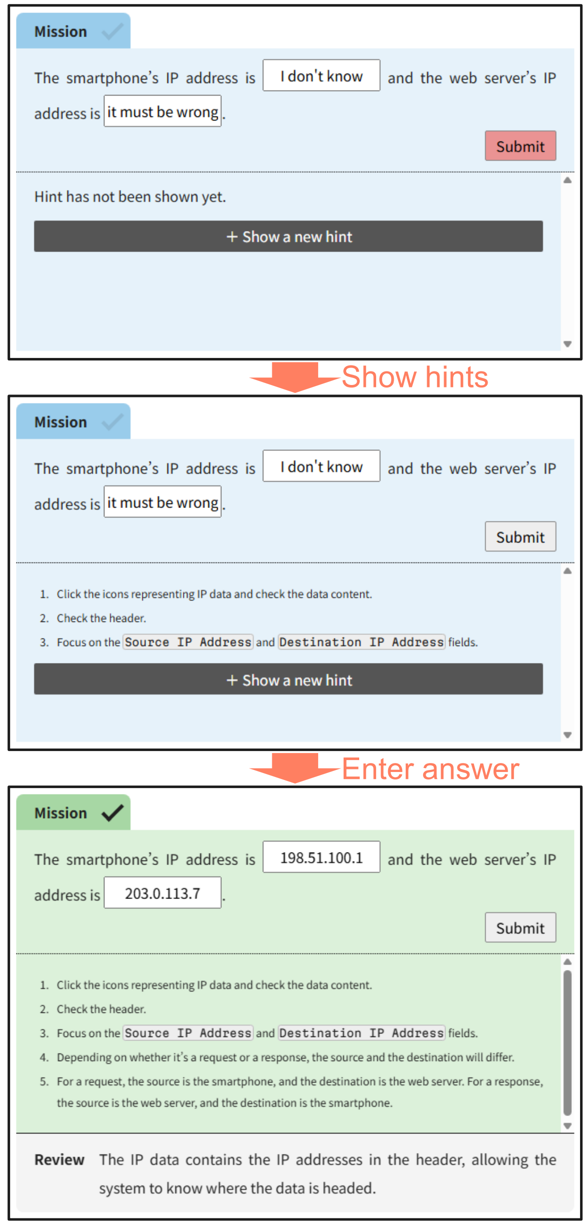

Fig. 6. Mission where hints can be revealed step by step and automatic correctness feedback is provided.

the protocol from UDP to TCP brings up a clock icon that measures the time elapsed after packet transmission and after a set interval the lost packets are retransmitted and the page finally loads. Through this sequence, users experience the reliability mechanism provided by TCP.

Regarding the internet layer, clicking the wavy line between the smartphone and the Web server reveals routers that had been hidden from view. Entering a URL in this state triggers an animation in which icons flow sequentially through each router, and hovering the cursor over a router displays its routing table, as shown in figure 4.

Regarding the network-interface layer, ProtoSim 2 appends a parity bit. Users can inject very small data such as a single character directly into this layer, as demonstrated in figure 5, which makes it easy to experiment with the parity-bit specification.

The textbook area consists of several pages that learners switch by using the table of contents. Each page contains learning content aligned with the simulator. On every page several problems called missions are placed and the system automatically returns feedback when learners enter an answer or perform the correct operation in the simulator, as illustrated in figure 6. Some missions include hints and on average about five hints can be revealed in stages.

In addition to the standard learning module, there is an exercise module called additional missions. This module provides no support features such as explanations or hints about network mechanisms and simply presents missions whose expected answers have been changed. After completing the standard module learners can tackle the additional mission module as a form of assessment.

TABLE I
CORRECT ANSWER RATES FOR THE MULTIPLE-CHOICE QUESTIONS ASKED IN THE PRE- AND POST-TESTS ($\mathrm{N} = 18$).

#	Question Overview	Pre-test	Post-test	χ^2	p	$\frac{b-c}{b+c}$
1	Select the term, such as “request” or “response” to fill each blank in the client-server diagram.	94.4%	100.0%	0.000	1.000	1.000
2	Select what each component of a URL represents.	61.1%	94.4%	4.167	0.041*	1.000
3	Select the action that the transport layer must perform when it receives a packet from its peer.	61.1%	55.6%	0.000	1.000	-0.333
4	Select the reason data is exchanged in packets, so that only lost packet can be retransmitted.	77.8%	100.0%	2.250	0.134	1.000
5	Select the reason domain names are used instead of IP addresses.	55.6%	94.4%	5.143	0.023*	1.000
6	Examine a diagram showing router cabling and routing tables and select the correct route.	77.8%	94.4%	0.800	0.371	0.600
7	Examine a waveform diagram with a charset table and select the word being represented.	100.0%	100.0%	0.000	1.000	-
8	Select the resulting bit sequence after a parity bit has been added.	33.3%	88.9%	8.100	0.004*	1.000

* $p < 0.05$

TABLE II
PROPORTION OF LEARNERS WHO USED EACH TECHNICAL TERM IN THEIR WRITTEN ANSWERS ON THE PRE- AND POST-TESTS ($\mathrm{N} = 18$).

Term	Pre-test	Post-test
Application layer	0.0%	50.0%
Client	0.0%	11.1%
Server	38.9%	77.8%
Request	16.7%	38.9%
Response	11.1%	22.2%
URL	50.0%	44.4%
File	0.0%	27.8%
(at least one application layer related term)	61.1%	94.4%
Transport layer	0.0%	50.0%
Packet	0.0%	55.6%
Segmentation	0.0%	38.9%
Reassembly	0.0%	27.8%
Retransmission	0.0%	0.0%
(at least one transport layer related term)	0.0%	88.9%
Internet layer	0.0%	44.4%
IP address	22.2%	11.1%
Domain	11.1%	0.0%
Router	5.6%	38.9%
Routing table	0.0%	0.0%
(at least one internet layer related term)	27.8%	72.2%
Network interface layer	0.0%	5.6%
Wave	0.0%	5.6%
Bit	0.0%	5.6%
Parity	0.0%	5.6%
(at least one network interface layer related term)	0.0%	22.2%

IV. EVALUATION

A. Overview of the Experiment

An experiment was conducted to evaluate whether the two design principles functioned effectively and the two objectives were sufficiently achieved. This study received ethical approval from an IRB committee (code number 2022-27).

The participants were eighteen undergraduate and graduate students recruited through a public call. None specialized in computer science, so they were presumed to have limited prior knowledge of the network mechanisms and to share characteristics relatively close to those of high school students.

Each participant brought its own laptop and completed every procedure individually while reading printed instructions that explained how to use each tool. Although the authors provided no verbal instructions or assistance, this implementation method was designed to simulate a home-study environment. The instructions asked them to finish the following tasks in no more than one hour and thirty minutes:

1) Access Moodle and take a pre-test (time limit fifteen minutes).
2) Launch ProtoSim 2 from Moodle via LTI and study with the standard module until all missions on every page are completed.
3) Try the additional mission module in ProtoSim 2 until all missions on every page are completed.
4) Return to Moodle and complete a questionnaire.
5) Take a post-test on Moodle (the same content and limitation as the pre-test).

The authors assumed maximum completion times of 40 minutes for 2), 10 minutes for item 3), and 10 minutes for item 4), and also expected that some would finish even sooner.

B. Methods

To evaluate whether DP1 functioned effectively and O1 was sufficiently achieved, both questionnaire-based subjective ratings and pre- and post-tests results were examined. The questionnaire asked, on a four point scale, whether each visualization method for each TCP/IP layer helped their understanding. The pre- and post-tests each contained two multiple-choice questions per layer and open-ended question about what mechanisms are working in the comuputer networks. All items were identical in both tests. For the multiple-choice items, the proportion of correct answers was calculated, and McNemar's test was applied to detect significant differences at the 0.05 level. Open-ended responses were coded to identify which technical terms were appeared.

To evaluate whether DP2 functioned effectively and O2 was sufficiently achieved, subjective ratings from the questionnaire together was used with operation logs from both the standard module and the additional mission module. The questionnaire asked, again on a four point scale, whether the mission and hint functions helped participants' understanding, in the same manner described above. In analyzing the operation logs, mission completions, elapsed time, and the total number of hints displayed by each participant were examined to evaluate both whether learners could proceed smoothly without confusion about the operations or underlying network concepts and whether the hint function was used effectively.

TABLE III
QUESTIONNAIRE RESULTS ON EACH PROTOSIM 2 FEATURE (N = 18).

#	Question Overview	Mean	SD
1	Animation of moving icons after URL input aided overall data flow understanding	3.94	0.24
2	Data content visualization such as cardboard boxes at right on icon-click aided overall flow understanding	3.56	0.70
3	Viewing folders and files on web server display aided application layer understanding	3.61	0.61
4	Protocol switch and clock icon for retransmission aided transport layer understanding	3.61	0.70
5	Icons travelling via routers and routing table view aided internet layer understanding	3.67	0.49
6	Parity-bit experiment with small data aided network interface layer understanding	3.67	0.49
7	Viewing hints at own pace helped exercise progress	3.72	0.57
8	Mission-based problem format helped exercise progress	3.89	0.32
9	Auto judging mission correctness helped exercise progress	3.78	0.55
10	Solving missions myself was interesting	3.89	0.32
11	Solving missions myself boosted confidence	3.78	0.43

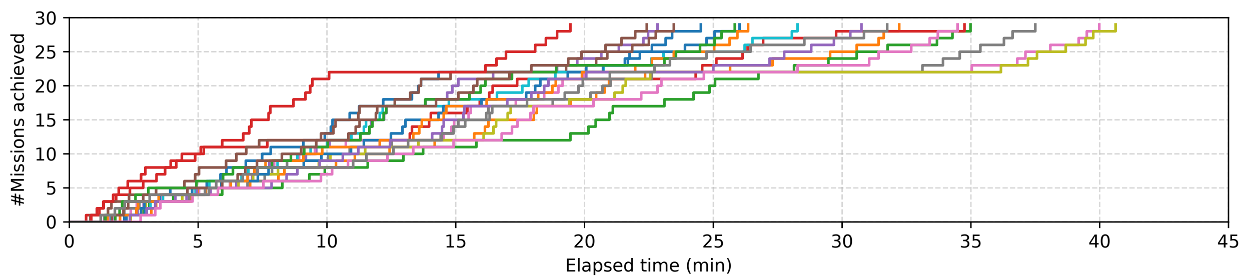

Fig. 7. Mission achievements over time on standard module (N = 18).

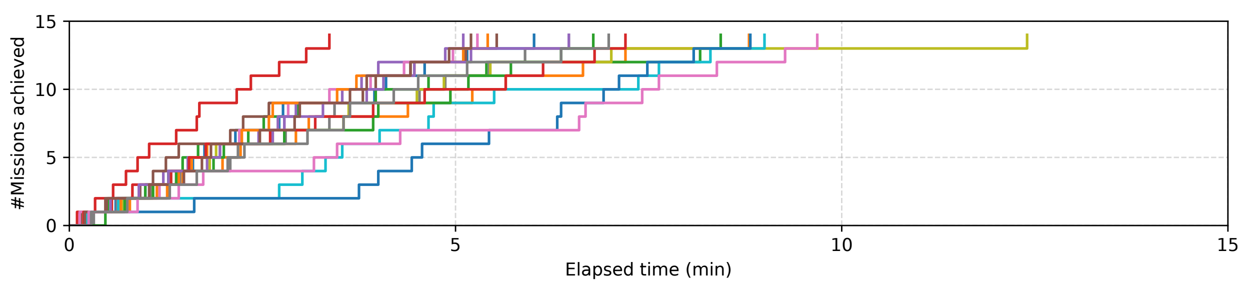

Fig. 8. Mission achievements over time on additional missions module (N = 18).

C. Results

The multiple-choice correct answer rates for the pre- and post-tests are shown in table I. Questions that exhibited a significant difference at the 0.05 level in the McNemar test are marked with an asterisk. Significant improvements were observed for question 2 on the application layer, question 6 on the internet layer, and question 8 on the network interface layer. Several items already had high correct answer rates in the pre test, so their increases were not statistically significant.

Table II shows the distribution of technical terms that appeared in the answers to the open-ended question asking what mechanisms operate in a network. In the pre-test, the descriptions were overwhelmingly biased toward the application layer. There were no references at all to the transport layer or the network interface layer. In the post-test, a wide range of descriptions was observed covering the application, transport, and internet layers. Although not many, some learners also mentioned the network interface layer.

The user' subjective evaluations of each ProtoSim 2 feature, as asked in the questionnaire, are presented in table III. All items recorded very high mean scores. In particular, ratings related to autonomous learning support, such as automatic mission feedback and hints, remained consistently high.

The standard module, which enables learners to study network mechanisms from the very beginning without prior knowledge, produced the mission completion trajectories shown in figure 7. Every user completed all 29 missions, and the time required ranged from approximately 20 minutes to 40 minutes depending on the individual.

Figure 8 presents the mission completion trajectories for each user in the additional mission module, which lacks autonomous learning support features other than automatic

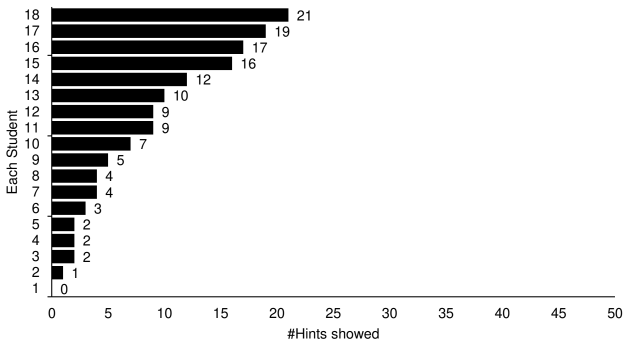

Fig. 9. The number of hints showed by each learner (N = 18).

correctness feedback. All users completed all 14 missions, and in most cases this took less than ten minutes.

Figure 9 presents, for each participant, the total number of times hints were displayed across all missions in the standard module. While one participant completed the missions without viewing any hints, another displayed 21 times. Because 49 hints were available for the 29 missions, it indicates that no learner indiscriminately displayed every hint.

V. Discussion

A. O1 and DP1: Understanding of all four layers

O1 was to achieve a comprehensive understanding of all four layers of TCP/IP and to grasp the key roles played by each individual layer. To that end, DP1 employed a simulator that visualizes and highlights the essential topics.

In the multiple-choice questions of the pre- and post-tests, significant gains in correct responses were observed for question 2 on the application layer, question 5 on the internet layer, and question 8 on the network interface layer. Only question 3, which concerned the transport layer, showed a slight decline. This item asked learners to choose the processing step that occurs when a packet arrives from a peer. The correct choice was "reassembly," yet many learners selected "fragmentation." After learning that packets are fragments of data, they may have chosen the distractor without reading the stem carefully. Emphasizing in the problem statement that the relevant timing is reception rather than transmission may improve the correct answer rate. These shifts in accuracy indicate that understanding of key mechanisms deepened across the layers.

Analysis of answers to the open-ended question likewise revealed changes in the technical terms that appeared. Both the total number and the variety of terms increased, showing that learners became able to use domain vocabulary to describe network mechanisms in prose. When the terms were classified by layer, pre-test responses were biased toward the application layer. This pattern matches results from our previous study and suggests that learners tend to write about content that is more visible and familiar to them. In the post-test, however, terms related to the application, transport, and internet layers appeared widely, indicating a broader understanding of network mechanisms. Although more learners mentioned the network interface layer, coverage remained limited, implying that it may need a more effective way to convey that the final delivery of data over a physical carrier is also part of communication.

Subjective ratings of each ProtoSim 2 feature in the questionnaire were uniformly high, with no noticeable bias. These findings suggest that the satisfactory attainment of the learning objective can be attributed to the combined contributions of its visualization features developed based of DP1.

B. O2 and DP2: Efficient and independent exercise

O2 aimed to enable high school students to complete the exercises on their own, within a short time, and in a home-study setting. To realize this goal, DP2 provided an electronic textbook integrated with the simulator and equipped it with self-regulated learning supports such as hints and automatic feedback on missions.

In both the learning session with the standard module and the practice session with the additional mission module, every learner finished all missions within an hour. This outcome indicates that there were no critical obstacles and that learning progressed smoothly overall.

The time required to finish each module differed by roughly a factor of two. Although classroom lessons often allot the same amount of time to every student, in reality the time needed for understanding varies from person to person. ProtoSim 2 appears to have allowed each learner to continue learning to the end at a pace that felt comfortable to them.

A closer inspection of the plots shows that some learners stalled after achieving 22 missions, and this stall contributed to the spread in total time of the standard module. Mission 23, which caused the delay, required learners to repair a routing table. In the virtual network, requests travel along the optimal route whereas responses follow a detour. Learners had to fill three blanks: which router, when receiving data for which destination, should forward it to which next-hop router. An item that demanded reasoning while comparing multiple routing tables proved exceptionally difficult. Although the lengthy time required imposed a burden on learners, they likely persevered because of the sense of accomplishment from completing all missions.

Hint usage varied across individual users, which can be interpreted positively. The hint feature seemed to effectively bridge gaps arising from differences in each learner's ease of understanding, thereby helping all participants make steady progress through the material.

In the questionnaire, the mission format and self-regulated learning supports such as hints and automatic answer checking received uniformly high ratings. These facts suggests that these ProtoSim 2 features were functioning effectively.

VI. Conclusion

This paper tackled the challenge that high school students must acquire a broad understanding of network mechanisms across all four layers of TCP/IP within a limited time, and reported an evaluation of the interactive learning environment, ProtoSim 2. In an experiment with undergraduate and graduate students, every participant completed the entire learning sequence in under one hour by following the simulator-integrated electronic textbook and making use of autonomous learning supports such as hints and automated feedback. Pre- and post-test scores improved, confirming that comprehension of the key topics emphasized and visualized by ProtoSim 2 increased.

However, several items already showed sufficiently high correct answer rates at the pre-test stage. Contrary to expectations, the university-level participants may have possessed prior knowledge of network mechanisms exceeding that of typical high school students. To overcome this limitation, future evaluation experiments will be conducted with high school students.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Numbers JP23H01019, JP23K11366, JP25KJ1792.

REFERENCES

[1] M. Oda, Y. Noborimoto, and T. Horita, "Implications for computer science curricula in primary school: A comparative study of sequences in england, south korea, and new zealand," in *Towards a Collaborative Society Through Creative Learning*. Springer Nature Switzerland, 2023, pp. 641–652.

[2] Department for Education (England), *Computing programmes of study: Key stages 3 and 4 National curriculum in England*, 2013.

[3] Ministry of Education (South Korea), *고등학교 교육과정 (I, II, III)*, 2015.

[4] Ministry of Education (New Zealand), *Technology in the New Zealand Curriculum*, 2017.

[5] S. Kanemune, S. Shirai, and S. Tani, "Informatics and programming education at primary and secondary schools in japan," *Olympiads in Informatics*, vol. 11, no. 1, pp. 143–150, 2017.

[6] Ministry of Education, Culture, Sports, Science and Technology (Japan), *National Curriculum Standards for High School (public notice of 2018)*, 2018.

[7] M. A. Rahman and A. Pakstas, "Tools and techniques for teaching and research in network design and simulation," *SN Computer Science*, vol. 4, no. 3, p. 269, 2023.

[8] M. Abdullah and A. Ehsan, "Teaching methodologies for computer networks lab," *International journal of advanced scientific and technical research*, vol. 2, no. 5, pp. 109–119, 2012.

[9] D. A. Figueroa, L. Gimson, E. Sánchez, and A. Gamarra, "Experience on "networking i" course using simulation software and real devices for practices," *South Florida Journal of Development*, vol. 3, no. 6, p. 7152–7158, Dec. 2022. [Online]. Available: https://ojs.southfloridapublishing.com/ojs/index.php/jdev/article/view/1966

[10] J. M. Pullen, "Teaching network protocol concepts in an open-source simulation environment," in *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, ser. ITiCSE 2018. New York, NY, USA: Association for Computing Machinery, 2018, p. 165–169. [Online]. Available: https://doi.org/10.1145/3197091.3197137

[11] G. Woolcott and V. Bui, "Design and implementation of a smart learning environment for teaching computer networking," *European Journal of Open, Distance and E-Learning*, vol. 25, no. 1, pp. 162–179, 2023. [Online]. Available: https://doi.org/10.2478/eurodl-2023-0013

[12] C. Goldstein, S. Leisten, K. Stark, and A. Tickle, "Using a network simulation tool to engage students in active learning enhances their understanding of complex data communications concepts," in *Proceedings of the 7th Australasian Conference on Computing Education - Volume 42*, ser. ACE '05. AUS: Australian Computer Society, Inc., 2005, p. 223–228.

[13] M. Prvan and J. OžEGOVIć, "Methods in teaching computer networks: A literature review," *ACM Trans. Comput. Educ.*, vol. 20, no. 3, jun 2020. [Online]. Available: https://doi.org/10.1145/3394963

[14] G. Mokodean, X. Meng, and M. Aburdene, "Analysis of computer networks courses in undergraduate computer science, electrical engineering and information science programs," in *2004 Annual Conference*, no. 10.18260/1-2–13660. Salt Lake City, Utah: ASEE Conferences, June 2004, https://peer.asee.org/13660.

[15] M. Vinay and S. Rassak, "A technological framework for teaching-learning process of computer networks to increase the learning habit," *International Journal of Computer Applications*, vol. 117, pp. 1–4, May 2015.

[16] J. Dong and H. Guo, "Enhance computer network curriculum using collaborative project-based learning," in *2011 ASEE Annual Conference & Exposition*, Jun 2011, pp. 22.611.1–22.611.14.

[17] R. Saputra, N. Jalinus, and Krismadinata, "Development of blended learning model based on project in computer network design and management," *Journal of Physics: Conference Series*, vol. 1387, 2019.

[18] N. Nasir, V. Govinda Rajan, P. Pannuto, B. Ghena, and B. Campbell, "Experiences teaching a wireless for the internet of things course co-operatively at multiple universities," in *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, ser. SIGCSE 2024. New York, NY, USA: Association for Computing Machinery, 2024, p. 923–929. [Online]. Available: https://doi.org/10.1145/3626252.3630848

[19] T. Bell. Computer science field guide. Computer Science Education Research Group at the University of Canterbury. [Online]. Available: https://www.csfieldguide.org.nz/en/

[20] T. Bell, J. Alexander, I. Freeman, and M. Grimley, "Computer science unplugged: School students doing real computing without computers," *The New Zealand Journal of Applied Computing and Information Technology*, vol. 13, pp. 20–29, 2009.

[21] M. Montagud and F. Boronat, "Analysis, deployment, and evaluation of the use of network simulation as a learning resource," *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, vol. 9, no. 3, pp. 82–90, 2014.

[22] Cisco. Cisco packet tracer. [Online]. Available: https://www.netacad.com/ja/courses/packet-tracer

[23] Y. Zhang, R. Liang, and H. Ma, "Teaching innovation in computer network course for undergraduate students with packet tracer," *IERI Procedia*, vol. 2, pp. 504–510, 2012, international Conference on Future Computer Supported Education, August 22- 23, 2012,Fraser Place Central - Seoul. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212667812001323

[24] J. Chamberlin, J. Hussey, B. Klimkowski, W. Moody, and C. Morrell, "The impact of virtualized technology on undergraduate computer networking education," in *Proceedings of the 18th Annual Conference on Information Technology Education*, ser. SIGITE '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 109–114. [Online]. Available: https://doi.org/10.1145/3125659.3125693

[25] D. C. Frezzo, J. T. Behrens, R. J. Mislevy, P. West, and K. E. DiCerbo, "Psychometric and evidentiary approaches to simulation assessment in packet tracer software," in *2009 Fifth International Conference on Networking and Services*, 2009, pp. 555–560.

[26] A. Smith and C. Bluck, "Multiuser collaborative practical learning using packet tracer," in *2010 Sixth International Conference on Networking and Services*, 2010, pp. 356–362.

[27] B. K. Saha, S. Misra, and M. S. Obaidat, "A web-based integrated environment for simulation and analysis with ns-2," *IEEE Wireless Communications*, vol. 20, no. 4, pp. 109–115, 2013.

[28] A. Medina, A. Lakhina, I. Matta, and J. Byers, "Brite: an approach to universal topology generation," in *MASCOTS 2001, Proceedings Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2001, pp. 346–353.

[29] A. Ruiz-Martínez, F. Pereñíguez-García, R. Marín-López, P. M. Ruiz-Martínez, and A. F. Skarmeta-Gómez, "Teaching advanced concepts in computer networks: Vnuml-um virtualization tool," *IEEE Transactions on Learning Technologies*, vol. 6, no. 1, pp. 85–96, 2013.

[30] M. Wannous and H. Nakano, "Nvlab, a networking virtual web-based laboratory that implements virtualization and virtual network computing technologies," *IEEE Transactions on Learning Technologies*, vol. 3, no. 2, pp. 129–138, 2010.

[31] A. Jevremovic, G. Shimic, M. Veinovic, and N. Ristic, "Ip addressing: Problem-based learning approach on computer networks," *IEEE Transactions on Learning Technologies*, vol. 10, no. 3, pp. 367–378, 2017.