

The Virtual Breadboard: Helping Students to Learn Electrical Engineering at a Distance

Lori L. Scarlatos, Ahmad R. Pratama, Tatiana T. Tchoubar
Stony Brook University
Stony Brook, NY 11794-4404, USA
Lori.Scarlatos@stonybrook.edu

Abstract—This paper presents an interactive virtual breadboard system that provides automated guidance to electrical engineering students working on electronic circuits labs. The primary contribution of the paper is the unique invariant representation of the state of the breadboard, which enables instructors to develop their own lab assignments with a set of customized hints. The paper describes the invariant state representation and its implementation in the virtual breadboard system. It also presents results of a pilot test demonstrating that using this system achieves the goal of reducing the workload of teaching assistants.

Keywords—Virtual breadboard; circuit design; guide-on-the-side; distance learning; digital hints; user interface

I. INTRODUCTION

Colleges seeking to cut costs and boost enrollments are increasingly turning to distance education as a way to do both. Up until recently, many fields were considered unsuitable for online education primarily because of their requirements for hands-on lab work and close supervision from an instructor or teaching assistant. New online degree programs in electrical engineering overcome this problem by sending lab kits to the students' homes and providing web conferencing so that students can talk directly with the instructors and show them the work they've done. Yet this solution does not scale well; there are only so many students that an instructor can talk to at a time.

This paper presents an interactive system that provides automated guidance to students working on electronic circuit lab assignments. Although students must still show their completed lab work to the instructor, the guidance and hints provided by the system help students to complete their assignments with much less interaction with the instructor. This system has been tested with students in an electronic circuits lab, and we have found that students do indeed need to spend less time working directly with an instructor.

The primary contribution of this paper is a unique invariant representation of the state of the lab work. This representation frees students to apply many different, but valid, approaches to solving a creative problem. It has also enabled us to create a separate interface for instructors that lets them provide their own guidance that is customized to address the types of problems typically encountered when undertaking a specific lab assignment, based on what students have done with the lab. Although the system presented in this paper is focused

specifically on guiding students in electronic circuits labs, the approach may be extended to a variety of creative tasks where there are innumerable correct ways to solve a problem.

As proof of concept, we conducted a study of the developed virtual breadboard in an actual electrical engineering lab course. In particular, we were interested in what educational simulation design features were important for a positive learner experience with an eLearning application in the high need field of electrical engineering. The study is important in practice because it reduces the time and effort that must be spent by instructor helping each individual student.

II. RELATED WORK

A. Remote Laboratories

A number of institutions have implemented remote laboratories to give online students in electrical engineering the lab work they need [1], [4], [5], [7], [9], [10], [13], [21]. Basically, these remote laboratories use the approach of allowing students to remotely make measurements on lab equipment at the university. Evaluations of these programs have been generally positive, measured in terms of student's criteria and academic results [5]. A study conducted at Carnegie Mellon University compared the influence of physical and virtual instructional materials on elementary school students' ability to design experiments and found no difference [22]. All of this suggests that it is feasible to replace face-to-face labs with online ones.

Lab work is an important part of learning engineering because it gets students physically involved with the materials. The efficacy of this hands-on approach is supported by the literature on kinesthetic learning and embodied and grounded cognition. According to this research, having students make their own gestures as they follow instructional animations helps them to learn not only human movements such as knot tying [19], [23], but also to understand math concepts [6] and molecule structure rules in stereochemistry [16], and memorize concepts of physics [15]. Some researchers suggest that using gestures and movement in this manner helps students to alleviate cognitive load on other learning channels (i.e., visual and auditory) [8]. Other empirical studies have demonstrated that using direct manipulation animation, which incorporates the haptic channel in the learning process, helps students "to reason about structural causal interactions and functional relations in systems" [2]. A 2011 study of physical versus virtual manipulative experimentation, used in learning physics

This project was funded by a State University of New York (SUNY) High-Needs Grant, entitled *Scaling Up the SUNY Online Program in Electrical Engineering for Renewable Clean Energies*.

concepts, did not find significant difference between the learning outcomes [24].

A shortcoming in all of these studies is that the educational activities need to be designed and developed by engineers, programmers, videographers ... people typically not available to the average college instructor. Ideally, the instructor should be able to define his or her own problem set, and specify how guidance is to be given online, without having to rely on a team of experts. This approach has been taken before, to teach math to elementary school children [17], [18]. This proven approach therefore forms the basis for this project.

B. Significance of Completion Times

The research literature includes several examples which indicate that less time spent on a task correlates with greater understanding of how to complete that task. For example, in computer-based instruction, a greater number of trials of an animation results in longer completion times [14]. This is because the time of completion is an inverse measure of the number of errors, and can be used as a measure of learning as such, for each consecutive trial.

Time to completion was used by Marcus and Sweller as an indicator of the cognitive load associated with understanding different instructions. "If the cognitive load is reduced, understanding will be enhanced", according to this group of researchers [11].

Correlations between cognitive test scores and individual maze trial time-to-completion are also highly consistent. To obtain a reliable measure of overall maze performance in one study, an average score was computed for each subject across all five trials of the mazes for both time and spatial memory errors [12]. Such performance characteristics as completion time and number of steps were considered as indicators of understanding in the works of Skinner and Kaufman [20]. For nature-inspired algorithms, time of completion is also an important measure of cooperative learning [3].

III. DESIGN GOALS AND APPROACHES

Before designing the virtual breadboard, we interviewed electrical engineering professors and reviewed existing software, such as CadSoft EAGLE and Proteus PCB Design. While this proprietary software works well in terms of simulating circuit design, it has its own workflow and business process which is often not compatible with the curriculum or the learning process designed by the instructor within the course. Instead of modifying their current course design to accommodate existing software, it is better to custom build software that can work well within the course.

Furthermore, for each and every problem set, there are innumerable correct ways for students to build the circuit on the breadboard. While students need to use the correct components and have correct connections between them, there are virtually unlimited ways to make such connections. Where the components are placed and the lengths of wire used to connect those components is irrelevant.

We used this characteristic as the rule of what students can and can't do when we created the virtual breadboard app. That way, we can give as much freedom to the students as they need

to build the circuit while still keeping them in the right track at the same time. Whenever they make some mistakes with their circuit design, we can tell them what is wrong in a form of hint instead of the actual solution so that they can better understand the principles and learn from their mistakes using this feedback as a formative assessment mechanism.

A. Breadboard Representation

The electrical engineering professors we spoke to use Breadboard MB-106 (Fig. 1) in their labs. Breadboard MB-106 has six (6) columns of strips with sixty three (63) strips in each column in addition to four (4) columns of power pins. The breadboard itself is represented as a collection of strips, each consisted of five connected pins ("a-b-c-d-e" or "f-g-h-i-j"). A pin is the smallest unit of the breadboard and therefore we build our breadboard representation as a collection of pins. We used object-oriented design in our approach and we used a multilayered software architecture to separate the presentation layer from the application layer of this virtual breadboard.

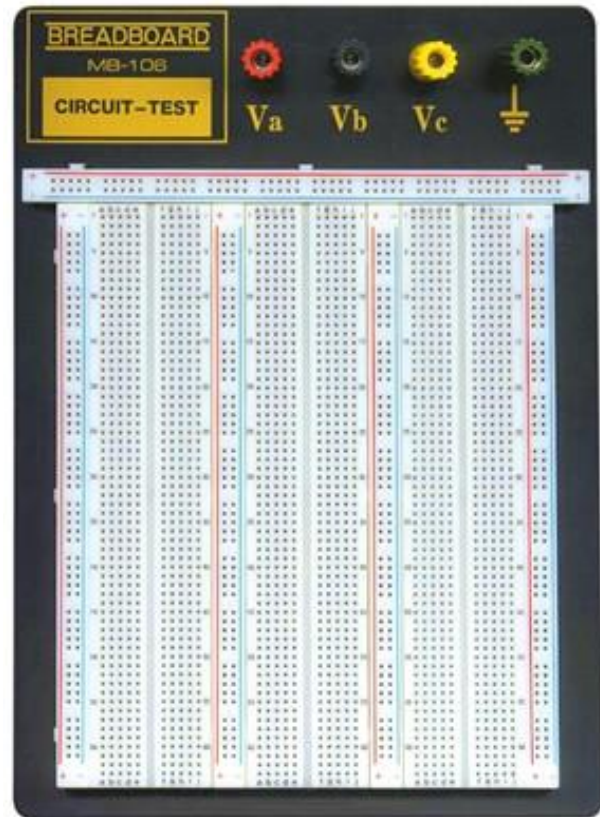


Fig. 1. Breadboard MB-106.



Fig. 2. Power Pin (in both left and right sides) and Regular Pin (in the middle) representation in the presentation layer of the virtual breadboard.

In the presentation layer, each pin is represented by a square area, arranged in such a way that it resembles the actual pin setup in Breadboard MB-106 (Fig. 2). In the application layer, this pin configuration is defined as a vector as follows:

```
pin[i] = pinType, x1,y1,x2,y2 3
```

This Pin representation consists of three important parameters as the following:

- 1) Pin Index; unique id for each pin on the breadboard.
- 2) Pin Type; power pin (designed for connecting power source to the breadboard) or regular pin (where most of the components sit on the breadboard).
- 3) Pin Coordinates in the presentation layer.

In addition, we also specify some other important parameters for each pin in the breadboard as the following:

4) Strip Index; five pins of the same strip share the same strip index in the breadboard.

5) Occupation Status, a Boolean variable that will be set as true whenever a component is placed on that particular pin.

Each electronic component (e.g. resistor, capacitor, or transistor) is represented by a vector image in the presentation layer and a class of objects in the application layer. The class includes properties regarding the relation to its placement to the virtual breadboard (i.e. width and height of the component) as well as its value and its connection with other components (i.e. netlist¹ string to be generated).

B. Connection and Series Representation

For every component placed on the breadboard, the respective pins where the component sits are set as occupied and the strip indices of the respective pins are used to identify connection with other components, if any. The connection with other component can be made directly by placing two or more components on the same strip or with the help of a wire connecting two different strips, where each strip has at least one component sitting. A chain of connected components in the virtual breadboard is called a series. There can be many series in the breadboard at a given time but a complete circuit will only have one series. Fig. 3 shows an example of two disconnected series in the presentation layer of the virtual breadboard with two or more connected components in each series.

Meanwhile in the application and data layers, we maintain a multidimensional array containing the component id, its type, its netlist parameter string, and the strip indices where the component has been placed. Based on this, we derive the series and the connection between components in the virtual breadboard. Fig. 4 shows the respective application and data layer representation of what we have in Fig. 3. There are two disconnected series in the breadboard; Series1 has two connected components, i.e. resistor *r,3.3k*(occupying pin158 and pin178) and resistor *r,10k*(occupying pin177 & pin197) connected directly on strip15. As for Series2,

there are three components: a diode *d, mod1* (pin471 & pin 481) connected to a wire (pin482 & pin 497) on strip76, which other leg is also connected to a resistor *r, 470* (pin496 & pin516) on strip79.

Series 1

Component 1: R | *r,3.3k* | 11,15 | true

Component 2: R | *r,10k* | 15,19 | true

Series 2	Comp Type	Netlist Parameters	Strip Indices	Commutative
Component 1:	D	<i>d,mod1</i>	74,76	false
Component 2:	W	-	76,79	true
Component 3:	R	<i>r,470</i>	79,83	true

Occupied pins:

158,178,177,197,471,481,482,497,496,516

r,3.3k *r,10k* *d,mod1* wire *r,470*

Fig. 3. Example of connection and series representation in the application and data layers.

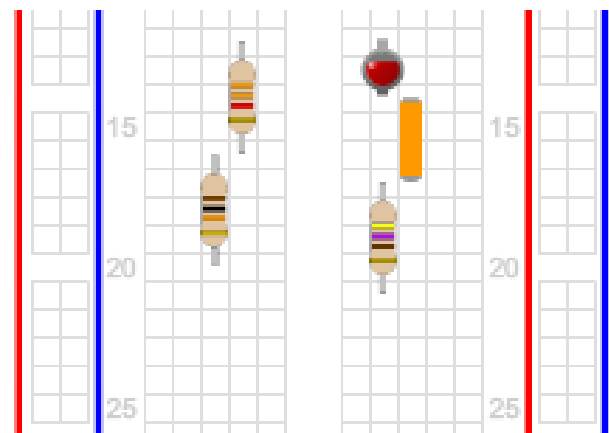


Fig. 4. Example of connection and series representation in the presentation layer.

For some components, such as resistor and non-polar capacitor, it does not matter which leg of the component is connected to another component. For other components, such as a diode and transistor, each leg is different and serves different purposes (i.e. Anode and Cathode in diode; Gate, Drain, and Source in MOSFET transistor; or Base, Collector, and Emitter in NPN/PNP transistor). In other words, legs in resistor and some other non-polar components are commutative while the legs in diode, transistor, and other polarized components are not. To accommodate this difference, we used additional *is Commutative* parameter for each connected component in the series, which is of Boolean type. It is set as true if it doesn't matter which leg of that component is connected to the other component and false when it does matter (Fig. 4).

C. Checking for Completed Circuit

The problem to be solved in a circuit design lab can typically be represented by a circuit diagram (Fig. 5). Regardless of the innumerable ways to build it, there is only one logical solution. In electronic design, a Netlist is commonly used to describe the connectivity of an electronic circuit. However, this is not a sufficient representation of the

¹ A netlist is a text representation of the connections among electrical components in an electronic circuit. We use the SPICE (Simulation Program with Integrated Circuit Emphasis) netlist syntax because it is widely used in university electrical engineering labs.

solution, as it is possible that one problem set has more than one correct Netlist, especially when the solution requires using multiple components of the same type (e.g. resistor). In order to recognize all correct solutions, we have the instructor specify all the components used in the problem set as well as all the connection points with the connected components on each one of them.

Select a problem set:

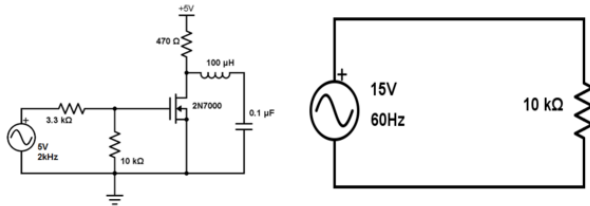


Fig. 5. Electronic circuit design problem presented to the students.

We check for this circuit completion every time a new component is placed on or removed from the breadboard. For the sake of efficiency, we only check for a completed circuit if only one series exists on the breadboard; if more than one series exists at the same time, it means that the circuit is still disconnected somewhere in between. Yet the breadboard status is always updated to provide feedback for every user action, so if more than one series exists on the breadboard, a message will let users know about that problem so that they can deal directly with it.

D. Providing Hints

When we check for a completed circuit, we also look for any mistake in the value of the components and/or the connection between components. If this is the case, the virtual breadboard app will provide a hint based on the mistakes to help user learn what the mistakes are so that to make it easier to fix those. The hint itself is generated based on the following rules:

- Number of Components: students can use different amounts of wire for their circuit design but they have to use the exact number of components to build a correct circuit.
- Component Value: students must specify the correct value for each component used in their circuit.

- Number of Connection Points: students can use different wiring configurations to connect the components, but the number of connection points should remain the same.
- Components Connection: only the correct component, or in some cases the correct leg of the correct component, should be connected in each connection point.

The virtual breadboard app can use the information provided by the instructor when adding a new problem set to compare with the current circuit design made by the students to provide these dynamic hints.

IV. PROTOTYPE IMPLEMENTATION

We built the prototype for the virtual breadboard app using Adobe Animate CC with Action Script 3.0 hosted in a web server with PHP and a MySQL database. The prototype works in any modern desktop-based web browser but is not supported by any mobile version yet. While this works well as a proof of concept, plan to port the Flash application to HTML 5 in a future version.

A. Student User Interface

Students can choose from many problem sets, which have been added by the instructor previously (Fig. 5). Selecting any problem will bring them to the main student interface, which is where they will work on building their circuit for that problem set (Fig. 6).

For the main student interface, apart from the breadboard visualization itself, there are some other important parts for the student user interface of the virtual breadboard app. They are:

- Problem Set: showing the circuit diagram students need to build based on their selection.
- Components: showing one component of each different type that students can choose from to build the circuit on the breadboard.
- Component Value: specifying values for the component a student has selected and is about to drag and drop to the breadboard.
- Breadboard Status: giving instant feedback every time students do something (i.e. adding a component or removing the existing one) on the breadboard. It also indicates whether that action is valid or not (Fig. 7 and 8).

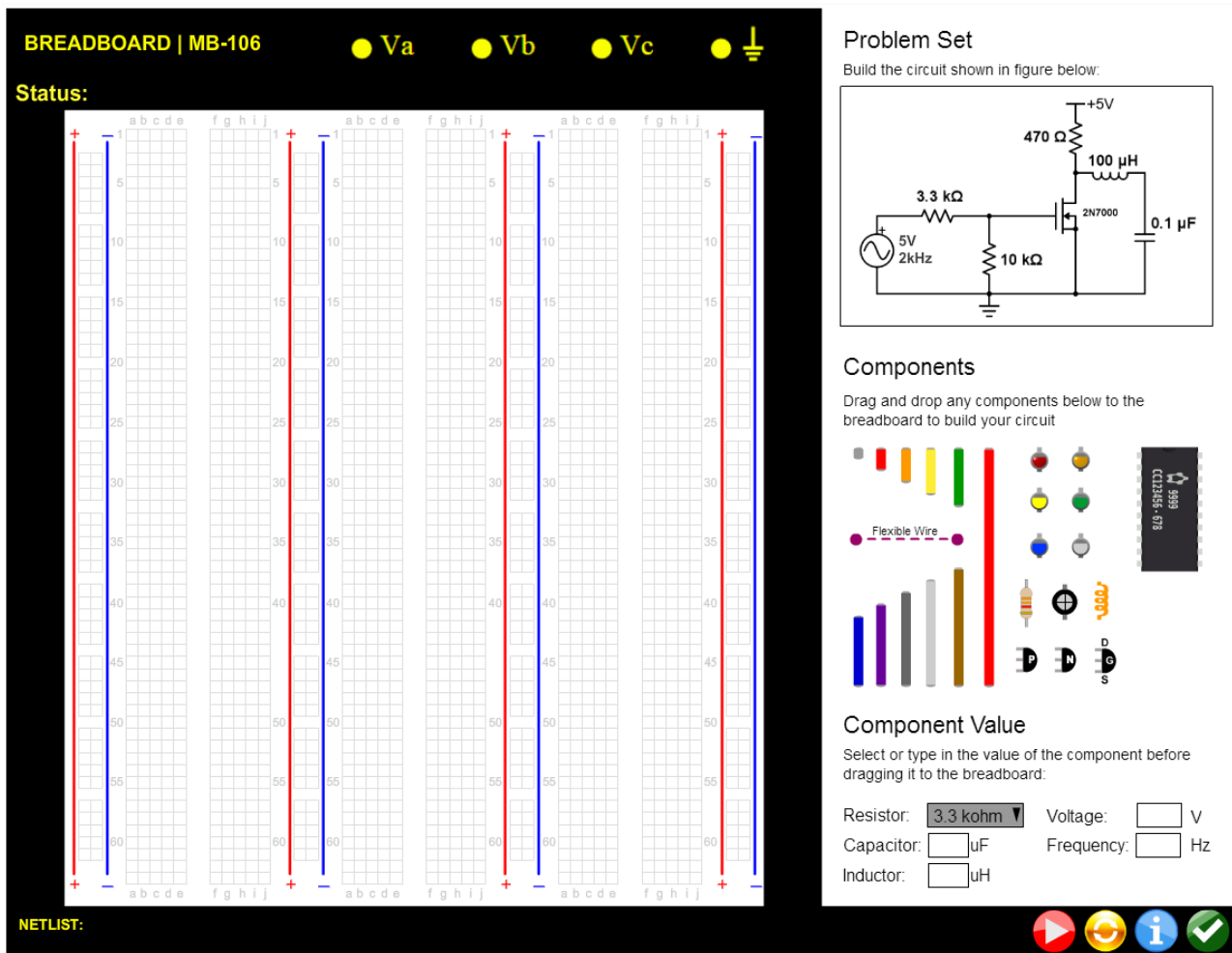


Fig. 6. Main student interface of Virtual Breadboard app.

- Status:** One series is present in the breadboard.
- Status:** 2 disconnected series are present in the breadboard.
- Status:** 3 disconnected series are present in the breadboard.

Fig. 7. Examples of breadboard status when a new component is successfully added to or removed from the breadboard.

- Status:** Component is placed out of the breadboard area!
- Status:** Component can't be mounted because pin is already occupied!
- Status:** Please specify voltage of the power supply!
- Status:** Please specify capacitance value of the capacitor!
- Status:** Please specify inductance value of the inductor!

Fig. 8. Examples of breadboard status when user failed to add a new component to or remove the existing one from the breadboard.

NETLIST: v1 1 0 dc 5 r1 1 0 3.3k

Fig. 9. Example of netlist generated by the virtual breadboard app.

- **Current Netlist:** generating a netlist string, which describes the connectivity of the electronic circuit based on the current state of the breadboard (Fig. 9).

- **Control Buttons:** allowing students to start the application, reset the breadboard, ask for hints, and save or download the image of their completed circuit design.
- **Hints:** giving a clue based on common circuit design mistakes made by the students in accordance with the basic principles of circuit design and the correct solution of the current problem set (Fig. 10).

When the students complete the circuit correctly, regardless of whatever circuit design they come with, the virtual breadboard will automatically detect it and congratulate them for the completion.

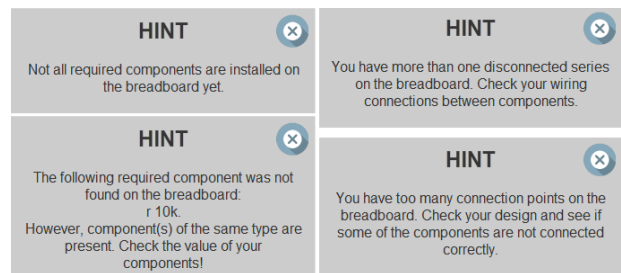


Fig. 10. Some of the hints provided by the virtual breadboard app.

B. Instructor User Interface

A separate user interface for instructors is available so that they can define new problem sets for the students to solve. On this interface, the instructor needs to specify all components used to build the circuit with their respective values as well as all connection points where some components are connected to each other. In addition, the instructor needs to provide a circuit diagram for the respective problem set (Fig. 11). All problem sets are stored on the server as a MySQL database record. Every time the instructor adds a new problem set to the database, it will be available for the students to try building their circuit design at the same time. The more problem sets added by the instructor, the more options students will have on their user interface to choose from. Of course, the instructor can also edit and remove any existing problem set if needed from this instructor user interface.

Add another problem set:

Problem Set Name:

Problem Set Diagram:

How many AC power sources?

How many DC power sources?

How many components?

How many connection points?

Fig. 11. Instructor interface to add new problem set.

V. USER STUDY

The purpose of our study was to determine whether the virtual breadboard was a useful learning tool for students. In particular, because we plan to use this in a distance education setting, we were looking for factors that will allow for a reduction in time and effort spend by instructor per student.

Our hypothesis is this: Using interaction with the simulation as priming for a lab reduces the time spent on the task, compared to the prelab with paper instructions.

A. Method

To solicit reactions and to gain feedback to refine our design and implementation of the virtual breadboard, we conducted a pilot test study with 56 undergraduate students in the Electrical Engineering program at Stony Brook University taking a course of Electronics Laboratory.

A random number generator was used to assign these students into two different groups. Both groups were given a text-based pre-lab assignment, which is typical for all lab assignments in this class. The experimental group was additionally asked to design their circuit using our virtual breadboard app, as a supplemental pre-lab assignment. Both groups were then expected to design and build the actual circuit in the lab. We then observed students from both groups

in their actual lab sessions and measured the time teaching assistants spent with the students. We used an independent-samples t-test to confirm our hypothesis that students in the experiment group take less time with the teaching assistants than students in the control group. Furthermore, we also asked students in the experiment group about their impression in the virtual breadboard app they used as their pre-lab assignment as a feedback for us as we go to the next stage of development of this app.

VI. RESULTS

As expected, the results of the study show that all students came up with several different circuit designs for the same problem set, some of which can be seen in Fig. 12. In fact, none of the 45 students in the experiment group has exactly the same circuit design. This variety was also found in the netlist strings generated by the app, some of which can be seen in Fig. 13. However, unlike the circuit designs that happen to be unique for each student, we found many students getting the same netlist string, which is understandable given the difference in the netlist is more about the order of component placed on the breadboard instead of the layout on the breadboard itself. Hence, it is possible for different circuit designs to have the same netlist string.

The rich variety in circuit design and netlist string generated by students in the experiment group shows that our virtual breadboard app satisfies the goal to give as much freedom to the students in building their own circuit while teaching them how to build a circuit correctly.

As for their impression to the virtual breadboard app they used as the pre-lab assignment, many students said they liked and enjoyed using the app. They said it was good and helpful for them in term of circuit building and telling them about their mistakes. Some other students said that they wanted more examples and problem sets for practice before they do the actual lab. This is where the instructor interface will come handy. In addition, we also received some suggestions to improve the user experience. Many were useful, and will be incorporated into the next version of the software.

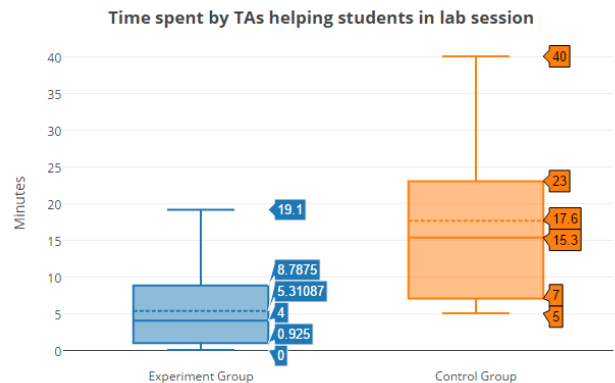


Fig. 12. The difference in time spent by the Teaching Assistants helping students building their circuit in the actual lab.

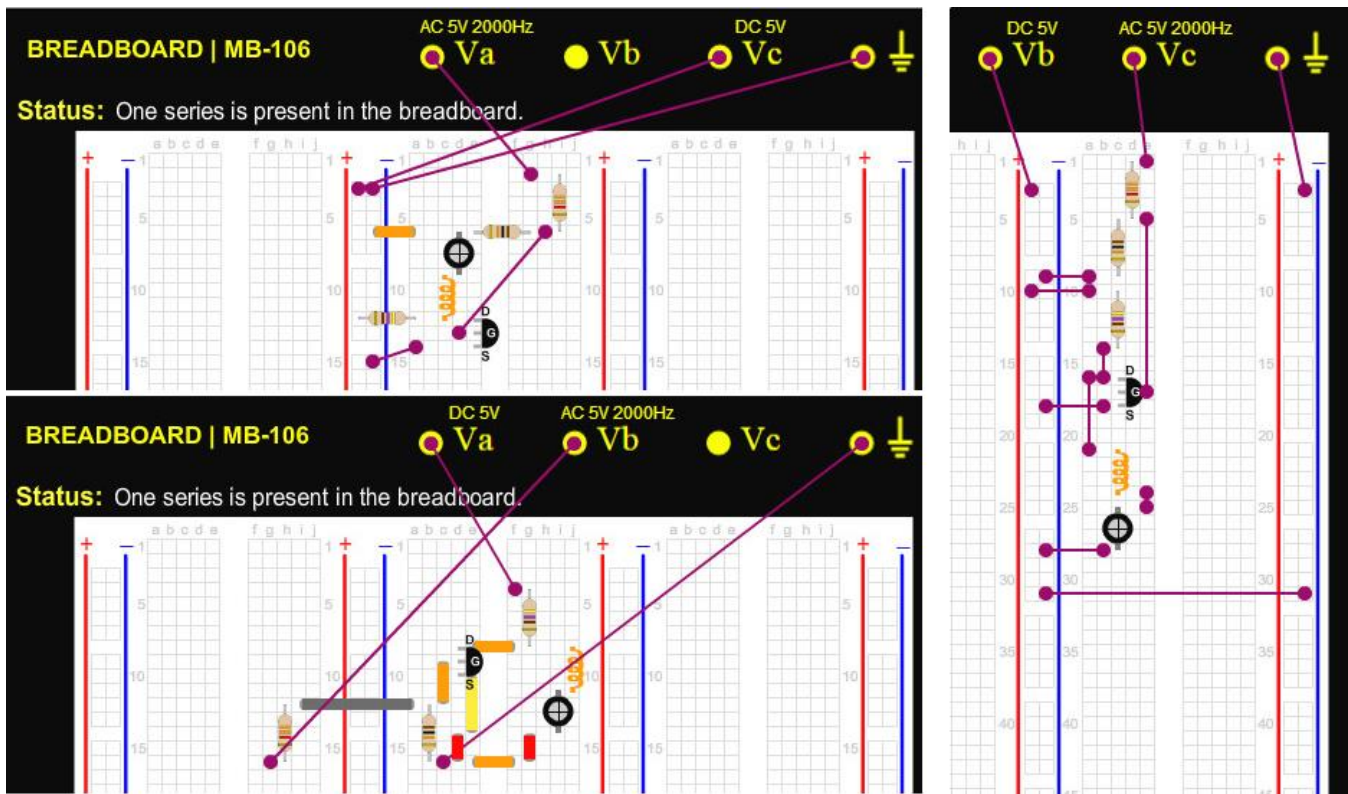


Fig. 13. Variety of correct circuit designs made by the students for the same problem set on the pre-lab assignment.

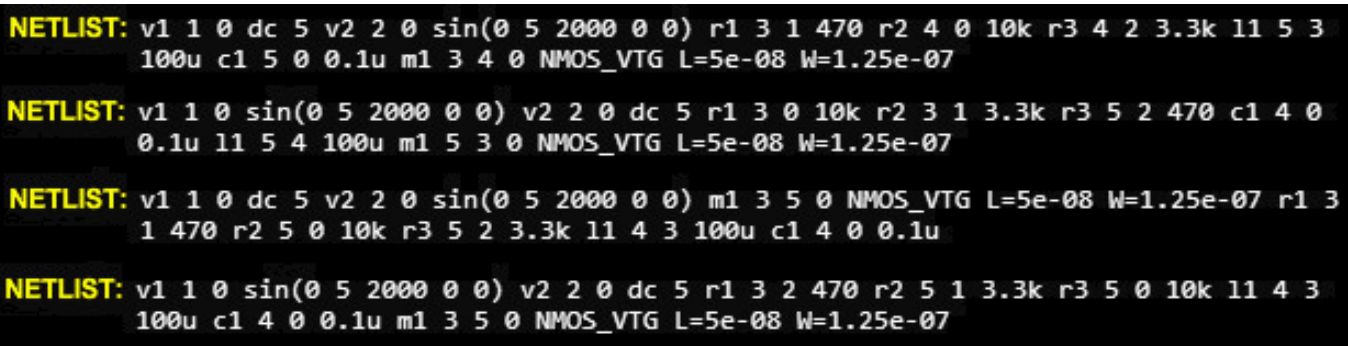


Fig. 14. Variety of correct netlists generated from students' circuit design for the same problem set on the pre-lab assignment.

Regarding the time spent by the teaching assistants while helping students building their circuit in the actual lab, we see that students in the experiment group indeed took significantly less amount of time than students in the control group (p value = 0.034). As shown in Fig. 14, each teaching assistant spent 17.6 minutes in average for helping each student in the control group building the circuit on the problem set. This is more than three times than that of the experiment group (5.31 minutes). Again, this result confirms that our virtual breadboard app can help reducing the workload of teaching assistants in the actual lab since the students have learned how to solve the problem set and to build the circuit correctly when they did it with our virtual breadboard app in the pre-lab assignment. This can give a huge impact on the learning activities, especially when we try to scale up the size of the class, which is usually one of the major problem in open and distance learning program.

VII. CONCLUSIONS

We developed a virtual breadboard application to be used as a pre-lab for introductory courses in electronics. The application has a drag-and-drop interface and unique state-based architecture, which enables the system to automatically check, and provide hints for, a wide range of creative circuit designs. It also has an instructor interface that gives teachers the freedom to design their own problem sets for their students.

Our pilot test indicated that the virtual breadboard's kinesthetic interface helped students to prepare for their labs, and significantly reduced the amount of time teaching assistants had to spend helping students in the actual lab. As part of this ongoing effort, we plan to test the system with students studying electrical engineering at a distance, to see if this reduces the amount of web conferencing that is required. We have also considered using the physical breadboard as a

tangible user interface that provides customized hints. Although the current system is focused on the problem of detecting the state of an electronic circuit board, we believe that the principles of this work may be extended to other student labs that require a measure of creativity and therefore have more than one way of correctly solving a problem.

ACKNOWLEDGMENT

We wish to thank Drs. Wendy Tang, and Mikhail Gouzman, who provided valuable guidance and insights into the art of electrical circuit design. We especially thank Dr. Emre Salman, who gave us insights into what is expected in his circuit design course, and worked with us to make the interfaces more useful and usable for both students and instructors. We also thank his teaching assistants who helped us to observe and collect data regarding use of the virtual breadboard in the lab: Andres Camargo, Zhihua Gan, Emerson Vernon and Chen Yan.

REFERENCES

- [1] Barrios, A., Panche, S., Duque, M., Grisales, V. H., Prieto, F., Villa, J. L., and Canu, M. (2013). "A multi-user remote academic laboratory system", *Computers and Education*, vol. 62, pp. 111-122.
- [2] Chan, M. S., and Black, J. B. (2006). "Direct-manipulation animation: Incorporating the haptic channel in the learning process to support middle school students in science learning and mental model acquisition", in *Proceedings of the 7th international conference on learning sciences*, International Society of the Learning Sciences (pp. 64-70), June 2006.
- [3] Dorigo, M., and Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. *Evolutionary Computation*, IEEE Transactions on, 1(1), pp. 53-66.
- [4] Esche, S. K. (2006). "On the Integration of Remote Experimentation into Undergraduate Laboratories-Technical Implementation", *International Journal Of Instructional Media*, 33(1), pp. 43-53.
- [5] Fabregas, E., Farias, G., Dormido-Canto, S., Dormido, S., and Esquembre, F. (2011). Developing a remote laboratory for engineering education. *Computers and Education*, 57(2), pp. 1686-1697.
- [6] Goldin-Meadow, S., Cook, S. W., and Mitchell, Z. A. (2009). Gesturing gives children new ideas about math. *Psychological Science*, 20(3), pp. 267-272.
- [7] Joolingen, W. v., Jong, T. d., and Lazonder, A. W. (2005). Co-Lab: research and development of an online learning environment for collaborative scientific discovery learning. *Computers in Human Behavior*, 21(4), pp. 671-688.
- [8] Huang, S. C. D. (2013). Grounded learning experience: Helping students learn physics through visuo-haptic priming and instruction (Doctoral dissertation, Columbia University).
- [9] Koning, B. B., Tabbers, H. K., Rikers, R. M., and Paas, F. (2011). Attention cueing in an instructional animation: The role of presentation speed. *Computers in Human Behavior*, 27(1), pp. 41-45.
- [10] Maiti, A., and Tripathy, B. (2013). Remote Laboratories: Design of Experiments and Their Web Implementation. *Educational Technology and Society*, 16(3), pp. 220-233.
- [11] Marcus, N., Cooper, M., and Sweller, J. (1996). Understanding instructions. *Journal of educational psychology*, 88(1), pp. 49.
- [12] Moffat, S. D., Hampson, E., and Hatzipantelis, M. (1998). Navigation in a "virtual" maze: Sex differences and correlation with psychometric measures of spatial ability in humans. *Evolution and Human Behavior*, 19(2), pp. 73-87.
- [13] Orduña, P., Bailey, P. H., DeLong, K., López-de-Ipiña, D., and García-Zubia, J. (2014). Towards federated interoperable bridges for sharing educational remote laboratories. *Computers in Human Behavior*, 30, pp. 389-395.
- [14] Park, O. C., and Gittelman, S. S. (1992). Selective use of animation and feedback in computer-based instruction. *Educational Technology Research and Development*, 40(4), pp. 27-38.
- [15] Pine, K. J., Lufkin, N., and Messer, D. (2004). More gestures than answers: children learning about balance. *Developmental psychology*, 40(6), pp. 1059.
- [16] Ping R, Larson SW, Decatur M-A, Zinchenko E, Goldin-Meadow S. Unpacking the gestures of chemistry learners: what the hands tell us about correct and incorrect conceptions of stereochemistry, under review.
- [17] Scarlatos, L.L. (2006). Tangible Math, *International Journal of Interactive Technology and Smart Education*, vol 3, no. 4, November 2006, pp. 293-309.
- [18] Scarlatos, L.L. and Scarlatos, T. (2008). Teacher Directed Active Learning Games, *Journal of Educational Technology Systems*, 37(1), summer 2008, pp. 3-18.
- [19] Schwan, S., and Riempp, R. (2004). The cognitive benefits of interactive videos: Learning to tie nautical knots. *Learning and Instruction*, 14(3), pp. 293-305.
- [20] Skinner, B. F. (1954). *The science of learning and the art of teaching*. Cambridge, Mass, USA, pp. 99-113.
- [21] Tawfik, M., Sancristobal, E., Martin, S., Diaz, G., Peire, J., and Castro, M. (2013). Expanding the boundaries of the classroom: Implementation of remote laboratories for industrial electronics disciplines. *Industrial Electronics Magazine, IEEE*, 7(1), pp. 41-49.
- [22] Triona, L. M., and Klahr, D. (2003). Point and click or grab and heft: Comparing the influence of physical and virtual instructional materials on elementary school students' ability to design experiments. *Cognition and Instruction*, 21(2), pp. 149-173.
- [23] Wong, A., Marcus, N., Ayres, P., Smith, L., Cooper, G. A., Paas, F., and Sweller, J. (2009). Instructional animations can be superior to statics when learning human motor skills. *Computers in Human Behavior*, 25(2), pp. 339-347.
- [24] Zacharia, Z. C., and Olympiou, G. (2011). Physical versus virtual manipulative experimentation in physics learning. *Learning and Instruction*, 21(3), pp. 317-331.