INDIVIDUAL AND GROUP LEARNING IN PHYSICS EDUCATION

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the
Graduate School of The Ohio State University

By

Florin Bocaneala, B.Sc., M.S.

* * * * *

The Ohio State University

2005

Dissertation Committee:
Professor Lei Bao, Adviser
Professor Richard Furnsthal
Professor Gordon Aubrecht
Professor Linn van Woerkom

Approved by

___________________________
Adviser
Physics Graduate Program
ABSTRACT

Recently, a large variety of new instructional methods have come to complement or replace sometimes the traditional ways of teaching physics. Many of these methods bring along a very convincing array of evidence testifying for their improved performance in classrooms and outside. All this has been made possible by the efforts of the Physics Education Research community, and it is what one would expect as the first step for any developing field. However, we believe that the time has come for a shift of interest from “what might we do to improve our instruction?” to “why the methods that we already have seem to work well?” We believe that theoretical sophistication of the research methods could and will support experimental work. Formal theoretical models applied to educational systems could guide the way we build experiments and enlarge our understanding of the educational process. This thesis points out possible ways of constructing these models, their limitations as well as their power. To this avail, we build computational models that reproduce many of the students’ mistakes, that one may use to test different learning strategies, and that simulate aspects of the students interaction within a study group, such as the relation between communication and group and individual performance.
to Rebecca
καὶ ἓκαστον μὴν διορίσασθαι σαφὲς τί ποτέ ἦσθιν, οὐ σημαρόν οὐδὲ ῥάδιον ἔργον

- Πλάτων – Σοφιστής: Θεόδωρος
ACKNOWLEDGMENTS

This work would not have been possible without the continuous guidance and support given to me by my adviser Prof. Lei Bao. My heartfelt gratitude to him for putting up with me.

I am grateful to all my group mates throughout the years, Homeyra Sadaghiani for making sure I am still sane after all, Steve Stonebreaker, Dedra Demaree and Yuhfen Lin for reality checks, Pengfei Li and YeounSoo Kim for reminding me of the bare necessities.

Special thanks go to Prof. Neville W. “Bill” Reay for all his generous advice and honest view of the research community, and to Prof. E. Leonard Jossem for his unlimited willingness to help.

Thanks to our secretaries, Brenda Mellet and Beth Reisinger for not letting me worry about anything else but research.

I am also grateful to my committee, Prof. Richard Furnsthal, Prof. Gordon Aubrecht, and Prof. Linn van Woerkom for their time and guidance. Special thanks to Prof. Richard Furnsthal for his help and advice I sometimes failed to heed.

Finally, I would like to thank my family: my wife Rebecca for simply being perfect, my parents for all their support and help throughout the years, and my parents-in-law for the ice cream.
VITA

21 October 1974 ............... Born - Constanta, Romania
1998 .......................... B.Sc, Physics, University of Bucharest, Bucharest, Romania
2003 .......................... M.S., Physics, The Ohio State University, Columbus, Ohio
1999 - present ............... Graduate Teaching/Research Associate, Physics, The Ohio State University, Columbus, Ohio

PUBLICATIONS


FIELDS OF STUDY

Major Field: Physics - Physics Education
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapters</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Community meaning</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Collaborative learning</td>
<td>5</td>
</tr>
<tr>
<td>1.2.1 Practice and group learning</td>
<td>7</td>
</tr>
<tr>
<td>1.2.2 Guided practice and homogeneous versus heterogeneous groups</td>
<td>9</td>
</tr>
<tr>
<td>1.3 Structure of the dissertation</td>
<td>10</td>
</tr>
<tr>
<td>2. A neural network primer</td>
<td>14</td>
</tr>
<tr>
<td>2.1 Back-propagation neural network</td>
<td>16</td>
</tr>
<tr>
<td>2.2 Simple recurrent neural networks</td>
<td>22</td>
</tr>
</tbody>
</table>
3. A computational model for physics learning .......................... 24
   3.1 Introduction .................................................. 24
   3.2 The problem and the model .................................. 25
   3.3 Simulations and results ...................................... 30
       3.3.1 Cognitive performance ................................. 31
       3.3.2 Previous experience and training environment ....... 36
       3.3.3 Error function landscape .............................. 38
   3.4 Summary and conclusions .................................... 39

4. Active versus passive learning ................................. 42
   4.1 Introduction .................................................. 42
   4.2 Simulations ................................................... 44
       4.2.1 Interest based architecture ............................ 46
       4.2.2 Error based architecture ............................... 49
       4.2.3 Passive learning and random choice selection strategy ........................................... 51
   4.3 Results ....................................................... 52
   4.4 Conclusions .................................................. 58

5. Collaborative performance in group learning ................. 60
   5.1 Simulation design ............................................... 61
   5.2 Interpersonal interaction ..................................... 65

6. Semantic agreement in group learning ......................... 68
   6.1 The tentative nature of language usage .................... 69
   6.2 Finite state machines ......................................... 74
   6.3 Semantic agreement and group size ........................ 76
   6.4 Semantic agreement and cross group communication ....... 85

7. Conclusions and ending remarks ............................... 88
   7.1 Summary of Student Learning Behavior ...................... 90
   7.2 Computational models ....................................... 92

Appendices:

A. Questions ...................................................... 101
B. Conversation transcript ................................. 107

C. Supporting work: Protagoras web testing interface ........................ 112

Bibliography .................................................. 118
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Artificial neuron</td>
<td>18</td>
</tr>
<tr>
<td>2.2 Feed-forward neural network</td>
<td>19</td>
</tr>
<tr>
<td>2.3 Simple recurrent neural network</td>
<td>22</td>
</tr>
<tr>
<td>3.1 Question-22</td>
<td>26</td>
</tr>
<tr>
<td>3.2 All the possible a priori different configurations of external charge and grounding connection</td>
<td>27</td>
</tr>
<tr>
<td>3.3 Feed-forward neural network</td>
<td>28</td>
</tr>
<tr>
<td>3.4 Polarization perceptron</td>
<td>32</td>
</tr>
<tr>
<td>3.5 The sum of errors vs. epoch number</td>
<td>33</td>
</tr>
<tr>
<td>3.6 Activation patterns vs. epoch number</td>
<td>34</td>
</tr>
<tr>
<td>3.7 Sum of errors vs. epoch number : tabula rasa</td>
<td>36</td>
</tr>
<tr>
<td>3.8 Sum of errors function landscape throughout the configuration space</td>
<td>39</td>
</tr>
<tr>
<td>4.1 Balance with identical weights</td>
<td>45</td>
</tr>
<tr>
<td>4.2 Pseudo-active training student model architecture</td>
<td>46</td>
</tr>
<tr>
<td>4.3 Student module architecture for active choice training</td>
<td>50</td>
</tr>
</tbody>
</table>
4.4 Sum of errors versus time for active question selection based on error ranking ............................................. 51
4.5 Sum of errors versus running time for passive and random choice strategies ................................................. 54
4.6 Sum of errors versus running time for passive and “boredom” strategies ...................................................... 55
4.7 Sum of errors versus running time for passive and maximum error strategies .............................................. 57
5.1 Student design diagram .................................................. 62
5.2 Decision module architecture ........................................ 63
5.3 Distribution of the number of successful students for simulations with decision modules and random choices .... 66
6.1 Utterance periods distribution - normalized distribution versus speech times in seconds ............................... 73
6.2 Silence periods distribution - normalized distribution versus silence times in seconds ................................. 74
6.3 A finite state machine flow chart ...................................... 76
6.4 Finite state machines around a common posting board ................................................................. 77
6.5 The singularity of the denotational matrix vs. epoch number ................................................................... 80
6.6 The singularity of denotational matrix vs. number of students .................................................................. 82
6.7 The singularity of denotational matrix vs. number of students for different reconstruction algorithms ........... 85
6.8 The singularity of denotational matrix versus cross-group communication probability .............................. 86
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Training times measured in number of epochs when training alone, and with the other group members</td>
<td>65</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

The main theme of this thesis revolves around group learning and instruction in college level physics courses. Starting in early 1960s, much effort has gone into understanding and implementing group-centered instructional methods at all educational levels. Some of these efforts have proved fruitless, but most of them have led to increased efficiency and quality of the instruction as well as to improved overall student performance. Here, we initiate an avenue of research that we believe has been poorly represented in Physics Education Research: collaborative and group learning.

Generally speaking, there are many features that might characterize a study group ([43]): each of the group members displays a sense of collective belonging; the group members share a common set of needs and goals; they show some amount of interdependence inasmuch as they react to things that affect any of the group members; there is some form of social organization among the group members as they might take on special roles or acquire status; the group is held together by relatively successful communication; the group members tend to stick together
and develop an emotional connection. However, most of the time, one will find a subset of these features rather than all. One may find the members of a group debating some open-ended tasks, pondering issues in small consensus groups, tutoring one another, helping one another through problem solving, reading aloud to each other, analyzing and evaluating the statements made by others, carrying out long-term research projects.

Due to their collaborative experience, one expects the students to develop skills they may employ later in real-life group work situations, learn to rely on each other rather than making constant appeal to an outside authority, and be able to carry on knowledge-constructing tasks typical of academic and research work. The traditional educational formula emphasizes the independent and solitary work, while any collaborative work is punished. As a result, most students leaving the university have poor collaborative skills and have a difficult time fitting in the highly group-centered work environment characteristic of most jobs. Unfortunately, most college work focuses shortsightedly on the final product of learning, a student’s individual ability to perform well during testing, and sees little advantage in group work.

We believe this situation is due to a fundamental assumption about the nature of knowledge. Most of us see knowledge as static, as something that is to be transferred from someone who has it to someone who does not. The group-centered paradigm proposes a rather different view: knowledge is a socially sanctioned and discovered phenomenon.
A piece of knowledge needs to be agreed upon by the members of a community, needs a certain degree of consensus in order to be accepted as legitimate ([48]). From this perspective, learning becomes the process by which the students get inducted into knowledge communities. And in order to accomplish this, one needs to acquire a certain degree of fluency in the technical jargon and word usage that the community employs to construct, organize, and describe its knowledge.

Retaining the metaphor, learning groups become transitional environments, language laboratories where students explore and experiment with the structure of the discourse they are trying to master along with the knowledge that the respective community has to share. To support this view, early educational research has already dismantled the traditional faith in the role of the tutor-pupil interaction as being the main factor facilitating the intellectual development: there is no proof that an increased frequency of the contacts between instructor and student nurtures increased performance; on the contrary, the academic world influences the student in the biggest extent indirectly by setting up an environment, a community propitious for the pupil ([61]).

1.1 Community meaning

Community meaning is generated by and belongs to each individual member of that community in ways that make sense to that individual, and it is generated by and belongs to the community in ways that are meaningful to that collection of learners. Therefore, community
meaning depends on both the community and the individual as part of that community. In large part, community meanings involve the same processes required of personal knowing: inquiry, imagination, and creativity.

Bakhtin ([5]) redefines the relationship of individuals to themselves and to the others. For Bakhtin, self and others cannot be separated, they can only be distinguished or pointed out, and even then only with great difficulty, because they are so intertwined within the community. Bakhtin refers to the personal as the ”inside” voice and the social as the ”outside” voice. Both exist because and at the expense of each other. Each exists within the other. It is the tension between the personal and the social that stimulates the discussion necessary for intellectual growth.

Much of the research about collaborative learning focuses either on cognitive or social processes; it is rather rare for bi-directional influences to be considered. We wish to argue that a further issue needs to be considered, that of how verbalization may be a key process in students’ cognitive development. There is still uncertainty about which dimensions are sufficient and which dimensions are necessary for the facilitation of learning. It also remains an open question why some children benefit from collaborative learning while others do not.

Vygotsky ([96]) proposed as an essential feature of learning: the zone of proximal development, in which a stimulating environment awakens a variety of development processes within the student, leading to higher
levels of cognition. Such a stimulating environment can be organized by way of cooperative learning in relation to a complex and intriguing open task (a very good example for this kind of open-ended tasks is provided by the activities and experiments making up the Physics by Inquiry class, experiments that the students have to set up, analyze results, and use them to discover on their own the laws governing their systems without knowing a priori what these laws are). Once the processes involved in carrying out the given tasks are internalized, they become part of the student’s independent development achievement.

1.2 Collaborative learning

Group work and collaborative learning has been lately hailed as one of the main ways to improve the quality of collegiate education. However, cognitive and educational literature still reflects conflicting views over whether it is beneficial to learning. Moreover, it is not conclusive whether groups perform better than individuals on typical educational tasks during the performance stage and even less certain whether group members transfer the knowledge gain properly during subsequent individual testing. One may therefore ask whether group learning hinders or facilitates learning.

When one looks at the large body of literature documenting and supporting the effectiveness and ineffectiveness of group learning ([22], [17], [58]) one realizes that rather than raising the questions of whether collaborative learning works, one should ask instead why it works or
does not work. Additionally, we should ask what the features of group learning are, the characteristics of group members and the structure of the task are, that determine the efficiency of collaborative work.

Group work is judged to be successful when it accomplishes at least one of the following goals: the group as a whole performs better and issues a product superior to that of an individual, and the group experience teaches the group members something they can internalize and employ successfully at a later time. The former alternative alone is not generally considered satisfying in an educational setting — each individual member of the group is expected to perform better on a similar task after the group work, when compared to an individual who did not experience group learning. Ideally, one may desire a combination of the two alternatives, since that would entail both the group learning and the subsequent transfer, and examine whether they have been enriched and nurtured by the group work.

Gabbert, Johnson, and Johnson ([29]) claim that group work not only yields a product superior to that of individuals, but that the group members are more likely to reproduce this superior performance and transfer it to tasks completed individually. However, other sources refute these claims ([87]). In general, it seems that an explicit incentive structure influences the group and subsequent individual performance. Slavin ([87]) shows that “there is no evidence that studying in groups *per se* is more or less effective than studying individually. The effects of group study depend entirely on the incentive structure used.”
In a standard classroom, the incentive structure is unfortunately unclear, if it is stated at all. Nevertheless, the instructors expect that the group work not only produces a superior product than what individual work could, but also that the members of the group undergo a more beneficial learning experience than what could be accomplished individually. Therefore, one might reformulate the central question: do members of study groups exhibit a superior ability to transfer what has been learned during the group activities than their peers who underwent the traditional individual study? Following Reder and Klatzky ([75]) “the most critical issue in any type of learning is how well the learning transfers from one situation to another.” Therefore, our guiding questions are: under which conditions, for which tasks, and for whom is group learning beneficial? What are the factors influencing the transfer of skills or concepts from the work performed in a group setting to individual activities?

1.2.1 Practice and group learning

The success shown by group learning may be due to the increased amount of practice and exposure the members of the group get to the different hypotheses and perspectives that each member brings into the group. The group members must continuously evaluate the proposed solutions to the task with its objectives, and by producing feedback to their group mates; they repeatedly collaborate and practice the given task. Being confronted with all the different alternative views and even
incorrect avenues that are proposed during their group explorations, the group members get additional training that might be subsequently responsible for increased individual performance.

There is evidence that practice does hone performance. Ericsson and Charness ([26]) show that sustained practice spanning a minimum of 10 years and averaging around 4 hours per day is a mandatory condition to attain expertise in domains such as music, chess, and athletics. Innate abilities apparently have a rather circumstantial effect, usually influencing and motivating the subjects’ decisions at the time when they choose a particular area of activity. However, it is sustained practice, and not innate ability, that determines whether they attain any level of expertise.

Returning to our basic question, one may justifiably ask whether expertise grants superior transfer abilities. Novick ([67]) has found that experts were more likely to transfer their knowledge and skills to solving problems outside their domain of expertise when a clue in the shape of a similar problem found in their domain was provided. Stringing these two findings (sustained practice grants expertise, and expertise enables transfer abilities) together, one may infer that practice might grant increased performance and transfer abilities. At this point in the argument, a question presents itself: is practice sufficient?
1.2.2 Guided practice and homogeneous versus heterogeneous groups

In order to address this last question, we point to some research done by Campione and Brown ([10]). The authors have shown that it is not sufficient to have subjects practice the same tasks repeatedly, but one must encourage the subjects to evaluate their solving strategies and correct them if it is needed. It is unlikely that, without such evaluation and assistance, the simple exposure to and practice of the same tasks will yield superior performance and facilitate transfer (some of their subjects kept approaching the given tasks the same way, using the same strategies, even though these strategies proved inappropriate). Once again, it might be that the group environment provides exactly this methodological assistance to its members and therefore improved performance as well as subsequent transfer.

One other aspect of group work is brought up by the dispute between homogeneous versus heterogeneous groups. The first way one might understand study group homogeneity is by looking at the abilities of its members. In this case it seems that the education research literature refutes the general held belief that heterogeneous groups would perform better than the homogeneous ones. The latter view springs out of the assumption that somehow the more advanced students would bring the rest of the students along with them. However as Ceci ([11]), and Tudge and Winteroff showed ([92]), more advanced students are
no more likely to lead peers to higher levels of development than someone of the same ability. In fact, more advanced students may hinder their peers’ efforts by confusing them with phrasing or opinions too sophisticated for students of lower ability.

The emphasis above is placed on the assumption that the kind of ability that is innate or fundamental determines the outcome of the training. For when one takes a look at the study made by Dunbar ([23]), one finds that a heterogeneous group made up of experts in different domains performs much better than a homogeneous group. In the latter case, each member of the group was not different from the others based on their abilities but based on their specialization within a limited body of knowledge and therefore bringing to the group a different perspective on the given task. Members could share their expert knowledge within the group and this way increase the chances that the right path might emerge from this potpourri of different hypotheses. It appears that group members’ having a certain expertise, which he or she is able to transfer to the the given task, is much more important to the good performance of the group than the different intellectual abilities the group mates might possess.

### 1.3 Structure of the dissertation

In the second chapter we introduce the reader to artificial neural networks: models that originate at the confluence of computer science and neural science, models that have been applied very successfully to
cognitive modeling of lower level cognitive processes. We present a short
description of neural networks in general with an emphasis on feed-
forward networks with back-propagation algorithm. Then we extend
the considerations to include “Elman networks” or simple recurrent
neural networks that will be employed in the fourth chapter to model
an adaptive decision block.

The third chapter, extending upon the work of McClelland and the
PDP group ([53]), introduces neural network modeling to physics edu-
cation, and suggests a larger usage of these models in the educational
research. It chooses a problem from standard Electricity and Magnetism curriculum and models the learning of this problem, using a
neural network to analyze and interpret the nature of the typical mis-
takes students make based on the features apparent in the simulation.
It then proceeds to look at the impact of the instruction structure on
the student subsequent performance. The third chapter ends with a
short discussion of the limitations and advantages of applying neural
network modeling to physics education research, and recommendations
on what one can expect and should not expect from neural network
models.

Active learning and its comparison with passive learning is the sub-
ject of the fourth chapter. We experiment with different motivational
strategies and compare their relative performance. We look in succes-
sion at passive learning (or random choice, as we have coded it), active
learning negatively motivated by increased lack of interest with repeatedly picked questions, active learning that selects questions to learn based on different errors the network reports to them. The strategies are reviewed and their relative advantages and disadvantages are considered.

Chapter five continues from where chapter three left off, extending the neural network with back-propagation to a simple recurrent network with a decision block, which we place in a four-member group. During the simulation, four artificial students try to understand a given problem while communicating with the other group members. We investigate the impact of communication on the individual and group performance as well as the role of the decision with respect to whom one listens to. The distribution profiles of the number of group members that complete the learning task for random and informed decisions are compared and checked against experimental findings. We then switch the emphasis from learning to communication in order to further our investigation of the role of the latter, meaning and its impact upon the former. The new simulation starts from the assumption that learning and understanding is a social phenomenon that may be construed as a processes of approaching and reaching a semantic agreement. This time, the study group is made up from a variable number of finite state machines communicating through a common posting board, where they can ask “quizzing” questions that the rest of the group may try answer.
As the simulation runs, the group members start employing their vocabulary consistently as they start understanding the problem at hand. We measure the semantic agreement with the singularity of the denotational matrix and show that a group size of four seems to be the most propitious configuration for reaching semantic agreement.

The last chapter reviews all the results and evaluates the application of the computational models as applied to physics education research. The thesis closes by pointing to further avenues for research.
CHAPTER 2

A NEURAL NETWORK PRIMER

Since two of the following chapters deal with computational simulations for physics education that employ neural networks, there is a need for a short introduction to the subject. A neural network is essentially a computational system whose structure and mechanisms attempt to simulate the operation of the human brain. Neural networks attempt to isolate certain key features of biological neurons, then they use these features to design artificial simulations of a very much simplified biological neural system. However, these are not simulations of the real biological neurons, for they do not attempt to reproduce the physiology, physics, or chemistry of the biological neuron. They approximate the computational abilities of the real neural systems. As such, one employs them in connectionist cognitive modeling of pattern recognition and information combining that take place in the real brain without raising any claim that a neural network could simulate the extremely complex system of the biological brain. In order to set the scale, we mention here that the largest artificial neural networks ever built contained around 1000 units (artificial neurons) while the human brain is
estimated to have around $10^{11}$ neurons. Nevertheless, artificial neural networks are known to learn to infer rules from given examples successfully, to generalize and predict behaviors from provided samples, and to match different sets of data.

The traditional artificial intelligence solution to the problem of understanding cognition has been the symbolic processing of data, in the way a standard computer program works. For example, we could mention here the relatively successful production systems. In contrast, connectionist models have proved themselves able to learn from examples without use of any prior assumptions that were implicit in the production system layout. If one provides a neural network with enough examples, the network will identify the patterns within the data and infer the rules governing it.

Let’s start by examining the main concepts artificial neural networks have borrowed from neural science. One of the salient neuroscience theories supports the idea that information in the biological brain is coded in the connection strengths between neurons, i.e., how much a neuron may influence the other neurons connected to it. Sustained exposure to similar events, that is, repeated undertaking of related tasks, seems to strengthen neural connections already present, weaken or break others, or even create new connections among the neurons making up the neural tissue. The next important concept is the dichotomy between excitation and inhibition: given a pair of connected neurons, firing the first one may either increase or decrease the probability that the second
will fire. In this way, a neuron within a pair is either inhibitory or excitatory. The strength of the connection between the two neurons regulates the amount in which the inhibition or the excitation takes place. The last important feature that artificial neural networks borrowed from the real neural networks is the transfer function. A biological neuron receives an input signal in the shape of pulses of electrochemical nature from neurons to which it is connected, and processes this input to decide its response or output: the processing function it employs is modeled in an artificial neural network as the transfer function. The transfer function is a mathematical algorithm that describes the way a given neuron reacts to a given input — it may sum the input and compare the sum with a given threshold to decide whether to fire, or use a more complicated sigmoid function over the input average, and so on.

2.1 Back-propagation neural network

In Figure 2.1, we have represented an artificial neuron as having \( n \) (in our example \( n = 4 \)) input lines and one output line. Every incoming signal to the neuron is weighted with the corresponding line weight \( W_i \), then the the resulting weighted inputs are summed up, and the sum is compared to a threshold value \( \theta \). If the total input passes the threshold test, the transfer function of the neuron computes the output. We may write all this as:
\[ s = \sum_{i=1}^{n} W_i x_i \quad \text{(2.1)} \]

and the transfer function process,

\[ o = F(s - \theta), \quad \text{(2.2)} \]

where \( x_i, i = 1, n \) are the input signals, \( W_i \) are the connection strengths, \( \theta \) is the neuron activation threshold, \( o \) is the neuron output signal, and \( F \) is the activation function. Most of the time, for notation and implementation reasons, one may denote \( \theta = W_0 \) and take the corresponding \( x_0 = -1 \), referring to the fixed input and threshold as bias input. Therefore,

\[ o = F(\sum_{i=0}^{n} W_i x_i). \quad \text{(2.3)} \]

The transfer function \( F \) may take a variety of shapes, but one we used — also one of the most commonly employed functions — is the logistic function:

\[ F(s) = \frac{1}{1 + \exp(-\alpha s)}, \quad \text{(2.4)} \]

where \( \alpha \) is the slope parameter of the logistic function. The logistic function is a smoother version of the threshold function. The logistic function becomes the step function when the slope parameter \( \alpha \) approaches infinity. The logistic function is differentiable.
Frank Rosenblatt ([78]) proved the perceptron convergence theorem (a perceptron is a simple feed-forward neural network with two fully connected layers — an input and an output layer), according with which there is a finite step convergent learning algorithm for any linearly separable learning task (the learned categories may be separated by straight lines in the input space). Moreover, he shows that a perceptron is capable of handling any linearly separable problem. However, nonlinearly separable problems need a multilayer network. A couple of decades later, Rumelhart and McClelland ([80]) came up with a back-propagation algorithm that can be successfully used to train multilayer networks.

The back-propagation algorithm attempts to minimize the error function, which one computes starting from the difference between the
network output and the desired output. This error function depends on
the connection weights and threshold, so therefore, one tries to adjust
these parameters to optimize the function. An iterative optimization
algorithm such as the conjugate gradient can be used. At each step in
the iteration, the conjugate gradient technique finds the direction with
the steepest descent and takes it. In this way the algorithm will even-
tually localize the minimum of the error function. There is one caveat
to this algorithm though: the algorithm is proved to converge and find
a minimum, but there is no guarantee that this is anything more than
a local minimum.

A multilayer network (see Figure 2.2) contains an input layer, in our
case identified by index \( k \), an output layer, identified by index \( i \), and a
number of hidden layers, in our case only one hidden layer identified by

Figure 2.2: Feed-forward neural network
index $j$. In this way, all the units in the input layer are connected to all the units in the hidden layer but they do not have any connections to the output layer. Similarly, the hidden layer units may activate only the units of the output layer. Therefore, the signal propagates from the input layer through the hidden layer to the output layer. Properly speaking, the input layer is not a full-fledged layer since it accepts only the input from the outside world and sends it to the hidden layer with no transfer function.

During the training of the network with the back-propagation algorithm, one sweeps the network twice. In the forward stage, the input from the outside world, or training environment, is given to the input layer, which passes it to the hidden layer after it gets weighted by the connection strengths, which remain unchanged during this stage, and so the process continues up to the output layer. Here the error function is computed and the back-propagation stage starts, moving the error backward from the output layer, through the hidden layer, to the input layer, adjusting the link weights accordingly.

Generally, one uses a set of input vectors counted by index $p$ to train the network:

$$x_j^{(p)} = \sum_k W_{jk} y_k^{(p)},$$

where $x_j^{(p)}$ is the input that the hidden layer receives and $W_{jk}$ are the weights of the connections between the input and the hidden layer.
This input is processed further by the transfer function on the hidden units:

\[ y_j^{(p)} = F(x_j^{(p)}) = F(\sum_k W_{jk} y_k^{(p)}). \quad (2.6) \]

Therefore, the signal produced by the output layer will be:

\[ y_i^{(p)} = F(\sum_j W_{ij} F(\sum_k W_{jk} y_k^{(p)})). \quad (2.7) \]

Now the back-propagation stage initiates with the computation of the error function. Most of the time, one uses a mean square expression,

\[ E = \frac{1}{2} \sum_p \sum_i (o_i^{(p)} - y_i^{(p)})^2, \quad (2.8) \]

where \( o_i^{(p)} \) is the desired output, and \( y_i^{(p)} \) is the output produced by the network in the forward stage of the iteration. The conjugate gradient method gives us a way to adjust the link weights in order to minimize the error function,

\[ \Delta W_{ij} = -\eta \frac{\partial E}{\partial W_{ij}} = \eta \sum_p (o_i^{(p)} - y_i^{(p)}) F'(x_i^{(p)}) y_j^{(p)}, \quad (2.9) \]

where \( \eta \) is the learning rate, and a similar expression holds for the update of the input to hidden link weights,

\[ \Delta W_{jk} = -\eta \frac{\partial E}{\partial W_{jk}} = \eta \sum_p \sum_i (o_i^{(p)} - y_i^{(p)}) F'(x_i^{(p)}) W_{ij} F'(x_j^{(p)}) y_k^{(p)}. \quad (2.10) \]

Updating the weights on all the connections following the above specification ends the back-propagation stage and the present iteration or
epoch. The algorithm continues to the next iteration until the network satisfies a certain pre-imposed condition for the error function. There are many technical issues to which one has to pay attention when designing and running a neural network back-propagation simulation that this primer cannot cover. Therefore, we will stop here, summarizing by noting that neural networks are essentially pattern processors and that their power resides in over-coding the problems at hand.

2.2 Simple recurrent neural networks

As opposed to feed-forward networks, recurrent networks memorize earlier states of the network and feed them back to network, in this way creating a two-way signal flow. Because of this feature, they are used for modeling systems whose behaviors depend on their previous states in a non-trivial way.

![Simple recurrent neural network](image)

Figure 2.3: Simple recurrent neural network
A simple recurrent network is a recurrent variation of the standard multi-layer network. A three-layer network is used, but the input layer is extended to contain a set of “context units” (see Figure 2.3), which are to store the internal state of the hidden layer: the hidden layer is to be copied unaltered into this context units after each sweep of the network. At each iteration, the input is propagated in a standard feed-forward way, and then a learning algorithm (usually back-propagation) is applied on the entire network. The fixed back connections into the context units always keep a copy of the previous values of the hidden units; thus, the network can maintain a sort of short-term memory of its own internal state, allowing it to perform such tasks as sequence-prediction that are beyond the power of a standard multi-layer perceptron.
In this chapter, we test the versatility of the neural network approach to modeling the dynamics of student learning. We choose a problem from an introductory physics class and we construct a neural network model for it. Based on this simulation, we argue that future use of neural network models in physics education research will be useful and that this approach can teach us some things about how physics learning takes place.

3.1 Introduction

Due to the nature of the model we are advocating in this chapter, we look at the learning process from a perspective that hasn’t been explored so far in physics education research. Whether some of the assumptions inherent to the simulation design are satisfactory remains to be determined based on the results of the simulation and its internal articulation. In the beginning, we need to make sure that our assumptions are clear in order that a dialog between standard experimental educational research and this computational modeling may take place.
In this chapter, we consider that the ability to answer all possible questions related to a concept or a situation correctly is the final goal of the learning task associated to that concept or situation. In the case of the problem modeled here, we assume for the purpose of this simulation that a student has mastered the situation represented by the problem when the student is able to answer correctly all possible combinations of external charges and ground connections (see below for a more detailed discussion).

The model that we propose here might not make any prediction about one particular student, but we believe that it may be useful in understanding the behavior of an ensemble of students. Therefore, when we speak here about student learning, we envision learning by a statistical entity (an ensemble of “students”) rather than a particular student.

### 3.2 The problem and the model

We have chosen an electricity and magnetism problem given to students in algebra and calculus-based classes during Spring Quarter 2003 (this includes students majoring in both engineering and non-technical fields). We reproduce here the text of the problem from [98].

**Question-22** A negative charge is brought and kept fixed in a location close to a neutral conducting rod. The end closer to the charge is then connected to the ground by a conducting wire as shown in Figure 3.1. What is the charge on the conducting rod after the ground connection is removed?

1. Positive charge.
2. Negative charge.
3. No charge (Neutral).

This question (3.1) was followed by a different version in which the ground connection and the external charge were on opposite sides of the conducting rod. After having answered the two questions, the participating students were asked to provide justifications for their answers. Based on their justifications, their answers were categorized and fell into 4 classes ([98]).

During our model design, we first identify the essential features and all the context features that seem to determine the students’ approach to the problem. Perusing their responses and arguments, we isolated the following salient features:

1. the presence or the absence of an external charge in proximity to the conducting rod
2. students generally treated the situations in which the external charge was brought close to one end or the other of the rod differently if an external charge was present
3. the presence or absence of the connection to the ground
4. students treat differently the situations when the ground connection is on one side or the other of the conducting rod

Regarding the last item in the above list, even though an expert looks upon the position of the ground connection with respect to the conducting rod as irrelevant for the given problem, the students seem to treat these situation as *a priori* different, so the model had to include this feature of student a priori understanding.

![Diagram of external charge and grounding connection configurations](image)

**Figure 3.2:** All the possible a priori different configurations of external charge and grounding connection

Therefore, we identified 15 possible configurations of external charge and ground connection that a student might consider independent cases
worthy of separate treatment (see Figure 3.2). Consequently, the mastering criterion mentioned in the introduction to this chapter becomes: a student has successfully mastered the given problem when the student is able to answer correctly all 15 related questions.

![Feed-forward neural network](image)

Figure 3.3: Feed-forward neural network.

We have built a neural network model for this problem, using feed-forward networks (see Figure 3.3) with 2 and 3 layers. The network receives the input on the first layer and produces an output on the last layer. During training, the output of the network is compared against the expected output and an error is computed. This error is back-propagated from the output to the input layer adjusting the link weights according to their contribution to the error. Certain factors motivated choosing back-propagation over other neurologically more plausible algorithms [68]:

1. The problem is essentially a categorization problem. Most students’ difficulties revolve around deciding which of the configurations are equivalent and which are different.

2. The back-propagation algorithm through the conjugate gradient infrastructure is an exploration of the error function in the configuration space. When we constructed these simulations, we were interested in the mistakes the networks encounter along their learning trajectories, or, in other words, the relatively stable regions in configuration space.

3. The elegance of the algorithm and the simplicity of its implementation.

4. Also, previous success in using this method [53] to higher level cognition simulations was more than encouraging.

Before we move on to the actual simulations we need to consider one design decision that has a major impact on the outcome of the simulations. The landscape of the error function is dependent on the way one represents (at network level) the examples used during training and the answers the network is to give ([81]). Two of the four input units for the network (see Figure 3.3) will represent the external charge: the presence and the nature of the external charge on one side or the other of the conducting rod (+1 for positive external charge; −1, for negative; 0 for no charge present on the respective side of the conducting rod). Since the side of the rod the external charge is brought near
seems to be a determining factor for the way the students approach the problem before they even start working on the problem, we chose to represent the position of the external charge with respect to the rod on separate input units. Moreover, for similar reasons, the presence of the external charge is represented on the remaining two input units (1 for a grounding connection on the respective side, 0 for its absence). We point out here that even though the network is by design symmetrical with respect to charge conjugation, it is not symmetrical with respect to the position of the ground connection. This does not imply that the network cannot end up treating the ground connection the same way, no matter which side it is on, (actually, the network accomplishes that when it learns the problem correctly); it only means that no such symmetry is present by design.

3.3 Simulations and results

In order to train or test a network, one needs examples (i.e., a tuple of input and output vectors) grouped together into environments (sets of examples). Considering the way our problem is presented in major textbooks ([31]), we have constructed three main environments: a polarization environment (including examples for an ungrounded conducting rod near an electric charge; in Figure 3.2, they are listed within the first column), a textbook environment (including examples for a grounded conducting rod near an electric charge such that the ground
connection and the external charge are on opposite sides of the conduct-ducting rod; first and second column of questions from Figure 3.2) and a general environment (including all the possible examples; all three columns in Figure 3.2).

A complete loop, which picks one question in the environment at a time, propagates the input signal forward through the network, computes the network output and computes the error with respect to the expected answer, sums all the errors and back-propagates the total error adjusting the link weights, through all the examples contained within the training environment is called an epoch or iteration. Every 30 epochs, we checked how far the network’s answers were from the correct answers. Since we batch together the examples contained in the environment, we employ the sum of squared errors to evaluate how far the network’s answers are from the correct ones. Whenever the sum of errors during an epoch became less than a critical error chosen before the simulation started, the network would have fulfilled its internal learning criterion and training was halted.

3.3.1 Cognitive performance

Because the standard curriculum teaches the students about polarization before potential, we trained the network using a modified polarization environment. We isolated the first two layers of the network (see Figure 3.3) from the rest of the network, obtaining in this way a two layer perceptron, and we kept the grounding units of the input
layer fixed to 0 throughout this stage. Since the polarization problem is separable in the input space, a perceptron such as this one can learn it well. Also, the hidden layer of the complete network becomes the output layer of the polarization perceptron. In this way, the future internal representation of the problem that the hidden layer stores will be biased by the polarization training completed at this stage.

Figure 3.4 shows the architecture of this polarization perceptron, where we have omitted the two units of the ground connection corresponding to the larger network since they do not contribute during this stage. The modified polarization environment is a simple copy of the polarization environment, containing the same situations, but it chooses to represent the charge distribution on the conducting rod as target answers for the polarization perceptron. Therefore, the presence and the position of the external charge is represented within this
modified polarization environment the same way as in the case of the complete network (−1, +1, and 0 for the external charge on the unit corresponding to its position with respect to the rod), while the expected perceptron answers extend this convention to the conducting rod (±1 represents the sign of the charge concentrated at one end or the other of the conducting rod, while the two units of the perceptron output layer correspond to two ends of the rod).

When the polarization training stage was completed and the polarization perceptron was giving correct answers to all the questions contained in the modified polarization environment, we started training the entire network using the textbook environment, and testing the network during this training using the general environment.

![Figure 3.5: The sum of errors vs. epoch number.](image)
Figure 3.5 shows the sum of errors as a function of epoch number during training with the textbook environment. You may notice three plateaus on the graph where the sum of errors function remains the same for some number of epochs: between 400 and 650 epochs, between 700 and 900 epochs and after 1300 epochs. Since the training algorithm looks for minima of the sum of errors function in the space of the connection weights, we recognize these plateaus as local minima. Because the function moves abruptly between these plateaus when we test the network during training, we are likely to find it in one of the states corresponding to one of these plateaus.

![Graph showing activation patterns vs. epoch number.](image)

Figure 3.6: Activation patterns vs. epoch number.

In order to see what these plateaus correspond to, we have plotted the activation pattern for each of the samples belonging to the textbook environment during the same training (Figure 3.6 shows only the
answers provided by the network to nine of the questions in the environment). As you can see, during the first plateau of the sum of errors function, the network’s response to any of the samples is 0, which corresponds to a neutral conducting rod. Let’s compare this answer to a student’s answer [98] (34% of the student answers fell in this category):

**Question 22 3.** the negative charge may cause charges within the rod to realign, but the overall charge on the rod remains neutral.

During the second plateau, the network answered correctly all the questions from the textbook environment except questions corresponding to a rod that was not grounded. Specifically, when we asked the network what the net charge was on a rod without a ground connection near a positive (negative) charge, the network answered “negative” (“positive”). We include here a student’s view [98] (14% have given similar answers):

**Q-21 1.** Positive charge because without the ground the conducting rod becomes negative but with the ground it becomes positive.

Comparing Figure 3.5 and Figure 3.6, we realize that the third plateau corresponds to the final stage in the training when the network has learned all the examples contained in the textbook environment. But what happens when we test the network on questions not included in the training environment? Well, the network gives incorrect answers. We remind the reader that neural networks are very robust in dealing with noisy data and as such they are quite capable of generalizing beyond their training [76]. However, in our case generalization failed.
We think that how it failed is very interesting: when we asked for the net charge of a conducting rod near a positive (negative) charge placed on the same side of the rod as the ground connection, the network answered “positive” (“negative”). Here is a student’s answer to the same question [98] (29% of the tested students have answered similarly):

**Q-22 2.** Negative charge. The negative fixed charge that is closer to the end that is connected to the ground attracts the protons in the rod and pulls them toward it and into the ground leaving the rod with an induced negative charge when the ground is removed.

3.3.2 Previous experience and training environment

![Figure 3.7: Sum of errors vs. epoch number: tabula rasa.](image)

It is often suspected that certain student difficulties are due to the way the instruction is structured. Let’s see what happens if we don’t
train the network with the polarization environment before running the simulation. Figure 3.7 shows the sum of errors if we start training the network from a *tabula rasa* (i.e., a random configuration of link weights). However, we remind the reader that the trajectory in the configuration space of the network during training depends strongly on the choice of the initial point. For example, if one starts the training from a point very close to the region that minimizes the error function, the network will learn the problem very fast and it will not visit all the possible local minima. For the sake of exposition, we have chosen to concentrate in this section on an initial point and its corresponding trajectory that is far enough from the minimum of the error function so that we can see all the meaningful plateaus along the learning trajectory (see, for example, the next subsection for a discussion of different starting points). Therefore, here we choose a random configuration of link weights but we make sure it is the same one that the previous subsection has employed, in order that one may compare the shapes of the error function along the two trajectories. The fact that the first plateau has disappeared provides strong evidence for the relation between cognitive behavior and pre-training experience.

As seen above, when we tested the network using the general environment after we had trained it using the textbook environment, the network answered the questions not included in the textbook environment incorrectly. However, when we trained the network using the general environment, which includes even the questions on which the
previous simulations had failed (external charges and ground connections on the same side of the conducting rod), the network answered to all the questions correctly.

### 3.3.3 Error function landscape

In all the previous simulations, we started the training from the same initial point (randomly generated), and therefore we analyzed the same trajectory with or without polarization training preconditioning. This time we set up a simulation where the initial point changes; therefore, we look at different trajectories in the configuration space of the neural network and try to sample the landscape of sum of errors function.

We chose to train the network once again with the general training environment. We randomly generated 100 initial points for the training and we followed their trajectories recording the sum of errors functions for every 100 epochs. We labeled the four rules that we discovered previously in order: Rule 0, Rule 1, Rule 2, Rule 3. Then we plotted in Figure 3.8 the normalized distributions of trajectories passing by each of the four rules versus the epoch number. As one can see, all the rules are well represented in this profile and, given a fortunate initial placement, a trajectory may reach the minimum of the sum of errors function much sooner than the rest. Also, the 3000 epochs the simulations ran proved not enough for some of the trajectories, the latter never finding the global minimum.
In order to understand this situation, one has to remember that the errors are back-propagated through the network in proportion to the values of the weights. If the final solution requires different weights among the network connections but the initial point has some of its weights equal, the conjugate gradient algorithm could never find it. Moreover, it appears that internal symmetries of this species are responsible for the sum of errors function landscape periodicities, flat valleys and temporary minima ([90]).

![Figure 3.8: Sum of errors function landscape throughout the configuration space](image)

3.4 Summary and conclusions

Let’s review what we have given to the network:
• The network architecture: the number and types of layers, units, and connections among units.

• How to represent the questions and interpret the answers.

• Feedback regarding how far its answers were from correct answers.

We did not code rules into the network by which the network would associate the input directly with the output: it was its task to find these “rules.”

After training, the network was able to answer our questions correctly, but during training it committed many of the same mistakes students make. This point needs further explanation. The model’s performance has been checked throughout training for all the 15 questions. Because each question had 3 possible answers, the number of patterns the network could produce was $3^{15}$. However, both the student population and the network seemed to select a limited number of patterns (in our case, 4 categories of argumentation). Moreover, including the correct answers, these four groups account for $83 \pm 4\%$ of the tested student population ([98]). This feature of neural network models has been noted previously in other settings ([53]) but, since we find it particularly interesting for the PER community, we emphasise it here, too.

One may use these models in lecture, laboratory, and testing design (see, for example, the questions included in the Appendices). In general, when it comes to structuring the presented material, a neural
network simulation might help. Our network failed to generalize in a case we didn’t have any reason to believe it would. Hence, the network model may provide a reliability check for our assumptions of similarity. Due to time and resource constraints, we often find ourselves in the position of selecting the material to be presented to the students based on our internal criteria of similarity. Alas, it turns out that our choices frequently were not the best ones even though they seemed so logical when we made them. Therefore, we find this feature of the model helpful for making some educational decisions.

The process of training (teaching, in the real world) can significantly affect how the system (students) develops its knowledge. In particular, previously acquired knowledge about polarization influences the system’s cognitive behavior. Moreover, our decision to leave out certain examples from the textbook environment biased the training and the answers of the network. We believe this to be a feature of the model that has direct applications in physics education research.
CHAPTER 4

ACTIVE VERSUS PASSIVE LEARNING

4.1 Introduction

Active and self-directed learning are proven strategies for improving learning performance and quality ([77], [85]). A vast literature (see [100] for a review) documents and supports their beneficial impact. Unfortunately, due to reasons such as dysfunctional interfaces among teachers and learners, institutional resistance, and the assumption of radical change that student-centered instruction allegedly requires ([40]), the implementation of these strategies is not common practice ([18]).

Moreover, even where a shift toward the implementation of active learning has been attempted, an inadequate understanding of active learning mechanisms and requisites has hindered its success. Herein, we argue that, besides the common emphasis on the need for students to understand the learning goals, to understand the assessment criteria, and to have the opportunity to reflect on their work ([27]), the instructor has to make sure the students employ a sound meta-cognitive strategy of self-organizing and self-selecting their learning resources.
Consequently, the author believes that only a careful investigation of the mechanisms behind active learning can provide the solid understanding necessary to an efficient implementation of student-centered instruction that eliminates redundancies and common myths that might shroud our comprehension of the active learning method. Therefore, in the following, we address the old issue of active versus passive learning from a different perspective: instead of asking “What do I need to do in order to encourage active learning among my students?” we ask, “How is it possible that active learning performs better than passive learning?” We are looking for a possible scenario that accounts for the success of active learning, and, inherently, the minimum requirements for enacting such a scenario. To approach this desideratum, we start with computational models living in artificial environments, models that exhibit behavior statistically similar to that of real students, taking care to check our simulation setup against the real world at every step.

Within this chapter we understand “active” learning to mean that students wield some measure of control over what question subsumed to a given problem is to be studied next. This is in contrast with “passive” learning, in which the order of study is predetermined and out of the students’ control. This understanding might not be the one that the physics education research community favors presently. Many curricula that encourage active learning (such as UW Tutorials, Physics by Inquiry, Interactive Lecture Demonstrations, etc.) do so within a very
structured and regimented format. The “activeness” in students’ learning comes about because of the instructional environment, not because of any explicit choices the students make while learning. However, a student studying alone may exhibit active learning by monitoring his or her own progress through a series of exercises. Moreover, one may find this form of “activeness” incumbent at a smaller scale as part of popular “active” educational methods employed and supported by the physics education research community. For example, when a group reaches an impasse or an unexpected outcome of an experiment in a standard PBI class, they are encouraged to tamper with the standard setup they built based on detailed textbook instructions, to alter and explore variations of it, to go back and forth between situations they understand and situations they find surprising. Therefore, although the author does recognize the limited form of “active learning” with which this chapter concerns itself, investigating even such a simplified definition of “active learning” may yield results that provide insight in more elaborate understanding of “active learning”.

4.2 Simulations

The simulations of this chapter extend upon those of the previous chapter. In particular, we have modeled our artificial students around three problems. The first was the problem of the previous chapter (external charge near a grounded conducting rod) while the second was an old, relatively famous problem, used extensively in psychological and
cognitive research, starting with Piaget and his seminal studies in child development:

Please predict what will happen to the balance in Figure 4.1, where the weights are identical, and the holes are equally distant from each other and the fulcrum.

![Figure 4.1: Balance with identical weights](image)

- Left side descends.
- The balance remains in equilibrium.
- Right side descends.

Since we have been interested during these simulations in aspects that were not problem dependent (all these simulations showed similar behavior with respect to the targeted features), we will concentrate during this chapter on a simpler problem that shows at a smaller scale all the features discussed within this chapter: the XOR problem.

Two twins having equal masses play with a teeter-totter. If both children sit at opposite ends, is the teeter-totter balanced? What if only one child sits at one end? What if both children stand by the teeter-totter without touching it?

Because the learning block of our artificial students is modeled as a neural network, we need to represent the last problem as a set of
four questions corresponding to all the possible configurations of children on or off the teeter-totter. This will be our training and testing environment in the following.

4.2.1 Interest based architecture

Being motivated by Parisi and Ceconi’s work ([70]), we have started our investigations with a similar architecture (see Figure 4.2). The network has one input layer, one output layer, and one hidden layer. The input receives a binary representation of the question the artificial student chose in the previous iteration and the correct answer. The output layer gives us back the predicted answer and the next question. In this way the network may try to predict the correct answer to a given question and choose a different question to study.

Figure 4.2: Pseudo-active training student model architecture
Since our XOR problem spans four questions, two units in the input layer, which may take binary values, will represent a question in a fashion similar to the previous chapter. The rest of the input layer represents the correct answer. In addition, the XOR problem is not linearly separable; therefore, one needs to employ a multi-layer network to model it. Moving to the output layer, one unit will store the network answer to the asked question, and the other two units will provide the choice the network makes for the next question.

In contrast to traditional architectures, where each layer is made up of one unitary block, the input and output layers are made up from two disjointed pieces: this will cause the network to run in two different modes. In one mode, the network is given a question (the correct answer units do not contribute to this mode), and, after it produces an attempted answer, the network is trained using positive back-propagation considering on the error computed between the network’s answer and the correct answer to this question only — the part of the network connected to next question output units does not play any role. The latter side of the network would be swept and trained independently in the next mode.

In the other mode, the network chooses a new question to study, receiving as input the question and the correct answer. Since the network has only one memory placeholder for the previously employed training question, only the last question has an impact on the choice of a next question to train with. After the network has made its choice,
the part of network involved in the decision is trained with a negative back-propagation algorithm, making the present choice less likely to be made again next time: whatever the network’s choice was, the “correct” choice is computed by flipping all the units of the next question output units, and then computing the error against this result.

Negative back-propagation tries to mimic in our simulation the increasing lack of interest that a student exposed to the same question shows in real life. We were trying to see how a simple mechanism that models boredom as a motivational factor for learning performs when compared to other alternatives. Our “student” alternately chooses a question from its training environment and gets trained with it, then it makes a new choice for a new training question, which becomes increasingly “boring” for as long as it is used, and the training is resumed anew. During the decision step, it is both the present question and the answer-category (correct answer) it belongs to that influence the choice. In this way, the student will more likely choose a question that does not have the same answer, or if it chooses the latter, it will be even less probable that it chooses the same question. This design decision has been motivated by the experimental observation that real students, during tutoring sessions and especially during periods of cognitive conflict, tend to move back and forth between questions or situations they have previously considered as belonging to the same answer-category (having the same answer) ([42]).
4.2.2 Error based architecture

Considering the results we found with the first architecture, the next step we took was to build a slightly different architecture. In this case, instead of increasing lack of interest about a given question in order to motivate self-directed learning, we opted for a more elaborate scheme: the simulated student chooses the next training question based on the results it gets to a test administrated at each iteration.

The architecture (see Figure 4.3) now contains two separate blocks: one is responsible for mastering the problem encompassing the training environment questions, and is, therefore, made up from the same learning block as before, a 2 : 2 : 1 architecture neural network; the other, with testing the learning block and with choosing a new question in order to continue the training. At each iteration, during the next-question choice stage, the decision block uses the testing results to rank all the four questions from the training environment, based on how far off the answers provided by the learning block are from the correct answers, then it makes a decision based on a predetermined strategy. For a given problem, our student may choose at each iteration to train itself with any question between the worst (the question that the learning block’s answer is the most far, in the output space metric, from the expected correct answer among all the questions in the training environment) or the best question (the question with the best predicted answer).
Figure 4.3: Student module architecture for active choice training

Figure 4.4 shows the simulated student’s performance during training for the XOR problem. Given the fact that our training environment contains four questions, the artificial student may use three different strategies to guide the choice of the next question to study. Always choosing the question it answers best will simply end up always choosing the same question. Therefore, the only valid error-based strategies remaining are choosing the second, third, or fourth ranked question (see Figure 4.4). We have run simulations embodying each of these strategies: choice every time of the same position in the error-based ranking of the training environment (the second, the third, or the fourth question).
Figure 4.4: Sum of errors versus time for active question selection based on error ranking

4.2.3 Passive learning and random choice selection strategy

For comparison reasons, we extended the number of learning strategies to include passive learning and random choice (we kept the same learning block made by 2 : 2 : 1 feed-forward network with back-propagation algorithm). Passive learning corresponds to the lack of any input from the student in the training sequence choice. The student is just trained with an *a priori* organized sequence made from the questions included in the training environment. As pointed out elsewhere
The organization of this passive learning training sequence has an important impact on the simulated student’s performance during training, but here our main concern lies with comparing self-organized training sequences made up during training, based on internal criteria, and not imposed from outside.

Random choice selection strategy represents for our purposes the null strategy. The student selects with equal probability any of the questions in the training environment. Therefore, random choice provides the standard background for any performance comparison among different strategies. Passive learning and random choice are both represented in Figure 4.5.

4.3 Results

Before we start analyzing our results, we have to assure the reader all the features we will discuss here have been found to be shared by all our simulations — independent of the simulated problem, starting point or network parameters (for the purpose of the comparison, we started from the same point in the configuration space of the learning network). Therefore, as mentioned above, we have chosen our simplest case, the XOR problem, to illustrate our findings in this presentation. Moreover, with a small caveat for “boredom” based architecture, the learning block has been kept the same, independent of the next-question choice strategy employed.
In general, for a given problem, the training profile for any model or strategy is strongly sensitive to the starting point: as already known for back-propagation (see, for example, [68]), the training of a neural network architecture starts from a point in the network configuration space and progresses toward mastering the given problem by minimizing the sum of errors between the predicted answers and the correct answers using a variant of the conjugate gradient algorithm. This starting point corresponds in reality to the previous knowledge and biases a student brings into the training. Hence, the starting point sensitivity of the neural network models would have been expected.

All the next-question choice strategies considered in this chapter are general in the sense that one may apply them to any given problem produce similar performance for learning times. For example, assigning probability weights, such that a higher probability makes that question more likely to be chosen for study, to each of the questions in the training environment in order to shorten the time to learn for a particular given problem, although it could be construed as a passive learning strategy, is not a general strategy: it might show an improved learning time but that would be dependent on the problem at hand.

The most conspicuous feature of Figure 4.5 is that one may discern almost only 1 plot. This is due to the fact that the passive learning graph matches almost perfectly the random choice graph. As long as the random choice is made with uniform probability, the network does not care whether it randomly selects the next question or it receives it
from a list in a prearranged order — it performs in the same way. To interpret this accurately, we need to remember that random choice is white noise: it is equivalent to scrambling the order of the questions in a test we intend to give to a group of students to average out all the effects related to the order of the questions on the test. It is exactly what we need for a background when we look at the effects of the different selection strategies that students might employ during learning. From this perspective, the fact that the passive learning training profile and random choice profile coincide agrees with reality: passive learning
corresponds to the case when the student does not influence the organization of learning material — the student only follows an already prescribed order.

Figure 4.6: Sum of errors versus running time for passive and “boredom” strategies

The most erratic graph in Figure 4.6, which corresponds to interest-based selection strategy, has a story that goes beyond the scope of this presentation. However, for our purposes here, we mention that, in terms of the convergence time (the time it takes for the model to learn its problem), the model with a interest-based selection strategy
performs slightly, but statistically meaningfully and consistently, better than passive learning. It starts by lagging behind passive learning but eventually picks up the pace to finish for all the starting points, all problems, slightly ahead of passive learning. That suggests that a simple increasing lack of interest mechanism informing the next training question selection is enough to upset the balance in favor of a self-directed strategy as compared to passive learning. However, it comes as no surprise that simply getting bored with what we already know does not motivate us enough to effect a powerful change in our learning performance.

The error-informed selection strategy is represented in Figure 4.7 by the green plot. The convergence time for the error-based selection is clearly superior to passive learning in all the runs (for the data plotted, the difference is 2 orders of magnitude). However, as one could see in Figure 4.7, it started slower than passive learning: from the point of view of the search in configuration space corresponding to the back-propagation algorithm, the error-based selection is more elaborate, but also more accurate in finding the right direction. This feature corresponds in reality to the fact that the students employing self-directed learning strategies take more time to show any change of belief, because they have more internal criteria they evaluate the change against, but, once the change takes place, it is qualitatively superior to passive learning and eventually makes up for the extra time used in actively
organizing their learning; in contrast, the passive learners are more eager to embrace new opinions, but often run the risk of settling too soon into a misconception (a local minimum).

Figure 4.4 shows a comparison of different error-based selection strategies. In this case, only one strategy, namely the one consistently choosing the question with the biggest error, is successful. In general, for a problem with more questions in the training environment, the maximum error strategy is not the only successful one, although it always remains the most successful one in terms of convergence times. We point
out that higher error strategies are consistently better than lower error strategies. However, one should remember that this study is limited to questions belonging to the same problem, and therefore subsumed to the same topic or central concept. One may not extend this result to questions across concepts.

4.4 Conclusions

During our simulations and experiments with different active learning strategies, we have found several ways to construct theoretical computational models for study of active learning. We have seen that, while passive learning does reach its goal eventually, learning efficiency can be improved greatly by active involvement of the learner in the organization of the learning process. However, the author believes that one should resist the temptation to embrace active learning indiscriminately, and make the assumption that the students are naturally inclined to employ efficient active learning strategies. The reality is that we receive most information from outside, simply as spectators without active involvement. A program trying to implement active learning using motivational techniques only and neglecting to train the learners to perform efficiently as active learners, does not have much chance of success. Our model, based on negative motivation to explore a problem task, performs slightly better than passive/random training, but not much better.
Moreover, our simulations suggest that one has to pay additional attention to the choice of strategy, since the various strategies perform differently. Among the strategies we experimented with, the maximum error selection strategy performed the best. Contrary to the common view, choosing to train with questions that are too close to correct answers does not work too well (this conclusion refers to questions subsumed to the same concept or problem only — simulations encompassing different problems may produce quite different results). We believe that, for as long as a student tries to cope with a new concept exemplified by a set of examples, the sooner the student tackles the most confusing situations, the sooner the student will comprehend the whole concept appropriately. Our simulations with different error rank strategies have illustrated this point.

The similar performance of the random choice and passive learning strategies did not come as a surprise considering the nature of these strategies and their correspondence to reality. This shows once again the accuracy and robustness these models may manifest in modeling educational data, making themselves useful tools for future physics education research.
CHAPTER 5

COLLABORATIVE PERFORMANCE IN GROUP LEARNING

Recently, group study has become an educational strategy commonly employed by educators throughout the country. In this chapter and the next, we attempt to construct theoretical models to support and help us understand the reputed efficiency of group learning over traditional instruction. Specifically, we concentrate on one aspect of group dynamics: the role of the decision process inherent to the relation between speaker and listener.

An extensive literature in psychology and education research analyzes and documents the productivity differences between individual, competitive, and cooperative approaches to classroom instruction and learning ([51],[99]). It is generally accepted that cooperative learning produces greater student achievement than the traditional learning methodologies. According to one of the authors ([88]) more than 63% of the cooperative learning groups included in a metastudy showed increased productivity as compared with individual learners. In this chapter, we construct a formal model of group dynamics that one may employ to study the mechanisms of the group interaction.
Physical proximity alone does not guarantee interpersonal interaction: a group of students may sit together, but, as long as successful communication does not take place, the group remains dysfunctional. Deutsch theory ([19]) states that interpersonal interaction stems from positive cathexis. Freudian literature uses cathexis to denote psychological energy invested in outside objects, such as colleagues, friends, or hobbies. If a member $X$ of a social group performs actions that facilitate another member $Y$ approaching $Y$’s goals, then $Y$ is likely to reward $X$’s actions and $X$ as a person. In the case of a study group, if a student issues statements that are perceived by another student as helpful or insightful — within their own understandings —, the receiving student rewards the issuer a vote of confidence; from now on, the listener is more likely to accept any statement the issuer will make.

5.1 Simulation design

Motivated by the standard physics laboratory instructional setup at Ohio State University and the results of the simulation presented in the next chapter, we designed a system including four “students”. Every student is made up from two modules: the learning and decision modules (see Fig. 5.1). The learning modules were feed-forward neural networks with a back-propagation algorithm ([53]), similar to the models in the previous chapters; we have experimented with different neural
architectures suitable for learning different physics problems (see previous chapter). For each simulation, all students had identical learning network architectures meant to master the same problem.

Figure 5.1: Student design diagram

In the present simulation, the learning block learns its task through two channels: it may either learn by itself with the standard backpropagation algorithm, or from a chosen “teacher.” Consequently, the links of the learning network were updated by the individual learning algorithm in conjunction with the training samples, then they would get jolted in the direction of the state of their “teacher.” Particularly, during “teaching,” each student increments its links proportionally with the teaching rate and the distance between its link values and the teacher’s. Some of the links communicated from teacher to student
were tempered according to a standard communication mutation rate in order to model communication failures:

\[ W_{ij}^{(s)} = W_{ij}^{(s)} + \tau(W_{ij}^{(t)} - W_{ij}^{(s)}) + \mu(W_{ij}^{(t)} - W_{ij}^{(s)})P_{ij}^m, \]  

(5.1)

where \( W_{ij}^{(s)} \) and \( W_{ij}^{(t)} \) are the link weights of the “student” and the “teacher,” \( \tau \) is the teaching rate, \( \mu \) the communication mutation rate, and \( P_{ij}^m \) is 0 with a certain standard mutation probability or a random number between 0 and 1. For our purposes in this chapter of examining the dynamics of the group interaction and its impact on group and individual performance, we believe these greatly simplified communication and teaching mechanisms are appropriate.

![Decision module architecture](image)

Figure 5.2: Decision module architecture

During each iteration, the decision module has to decide whether the corresponding student will keep listening to its present teacher. The
decision module is a simple recurrent neural network with a version of the back-propagation algorithm ([24]). The input layer contains three units (see Figure 5.2) organized into two blocks. One unit making up the first block represents the behavior of the learning function of the learning module: the error function may increase, decrease, or stay the same within a certain tolerance, and we choose to represent these alternatives as $+1$, $-1$, or $0$. The other two units are the context units of the simple recurrent neural network that store the previous state of the hidden layer. In the end, the network decides whether to change the present teacher or not; therefore, one single output unit suffices.

If the learning block got closer to minimizing the error function during the previous iteration, the decision block links would be reinforced through a positive feedback; otherwise they would be inhibited. At the end of each iteration, when the network is called to decide between choosing another teacher or keeping the present one, the previous state of the hidden layer of the decision network is used as input to the network to inform that decision. In this way, previous experience contributes to the “student”’s present decision of whether to change the “teacher” or not. This decision block design provides a flexible way to model the reward mechanism of Deutsch theory: gains in learning (getting closer to minimizing the error function of the learning module) acquired by listening to a group member will encourage the decision module to keep that “teacher”; also, previous decision module experience, or previous “student” experience, does influence the present
Table 5.1: Training times measured in number of epochs when training alone, and with the other group members

<table>
<thead>
<tr>
<th>student</th>
<th>learning rate</th>
<th>alone</th>
<th>together</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1</td>
<td>738 ± 96</td>
<td>230 ± 134</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>256 ± 74</td>
<td>380 ± 262</td>
</tr>
<tr>
<td>3</td>
<td>0.6</td>
<td>134 ± 44</td>
<td>216 ± 103</td>
</tr>
<tr>
<td>4</td>
<td>0.8</td>
<td>110 ± 23</td>
<td>221 ± 109</td>
</tr>
</tbody>
</table>

decision — even if this time the “student” did not record an increase in learning, there is still an emotional bond that might make the “student” keep its present teacher.

5.2 Interpersonal interaction

We have run the simulation with different learning rates for the student learning blocks and teaching on or off. In Table 5.1, we have recorded the training times for each student. When the members learn the task individually (teaching is turned off), the training times are scattered due to differences in the learning rates. However, when the students may learn from each other, the training times for all the students cluster together. This clustering of the times to learn does not mean that group members with a higher learning rate pull along the group members with a lower learning rate. To see this, one has to pay attention to the way knowledge transfer between “teacher” and “student” is coded within the simulation. It is the state (the set of all
the link weights in the learning module) of the “teacher” that jolts the “student”’s state and not its learning rate directly. A superior learning rate might bring the group member closer to a state that minimizes the learning error function sooner, and only in this way this “teacher” is able to help a “student” when the latter chooses to listen to the former.

Figure 5.3: Distribution of the number of successful students for simulations with decision modules and random choices

We investigated the role of the decision module by running the simulation alternatively with a decision module and a random choice process, when the “student” decides randomly whether to keep the present
“teacher” or choose another “teacher” from the rest of the group members. In Figure 5.3, we plotted the distribution of the number of students who mastered the task in 3000 epochs for students with a decision module and with a random decision, respectively. One may notice the difference between the steep profile for random choice and smooth profile for the decision module: for random choice, either the entire group mastered the given problem or no member of the group managed to master it; for the simulation with decision block, there were groups where only a subset of them mastered the problem, the highest point of the distribution corresponds to the case where the entire group mastered the problem. Psychology literature results (see, for example [9]) seem to agree on the shape of the distribution for the simulation with the decision module.

The meaning of this result is twofold. On one hand, it supports the claim that a reward mechanism might be responsible for the decision process governing transfer among group members in a study group, since the distribution profile corroborates to the psychological research findings. On the other hand, contrasting the distribution profiles corresponding to random choice and decision module reinforces the essential role that interpersonal interaction plays in the well-functioning of a group and the subsequent individual performance.
CHAPTER 6

SEMANTIC AGREEMENT IN GROUP LEARNING

As seen in the previous chapter, an optimization process of a goal function governs the dynamics of the group: every member of the group apprises the progress made by the group based on an internal goal function. However, as we’ve seen in the preceding chapter, the members of the group come to rely on one another in order to satisfy their goals. Consequently, acting within the group places constraints on the process due to communication requirements: successful communication assumes and compels a semantic agreement. Therefore, the interpersonal interaction has to negotiate between individual goal optimization and semantic agreement. One might expect that ontological agreement emerges from this tension.

To investigate the emergent semantic agreement within a group, we designed a system with a variable number of students modeled as finite state machines ([47]). All the students have access to a common area, a posting board, which they can either read or change the state of (6.4). This acts as the communication medium that the group members may manipulate ([64]). The finite state machine contains one slot of internal
memory state that stores the previous state of the posting board, and one slot that stores one question, which represents the question the student may ask the rest of the group at a time through the posting board. In this way, any action the student might take when its turn comes is determined by the present state of the posting board and the known question. No student may share the content of the internal question slot directly with any other student: they can communicate only through the posting board, using the limited set of symbols that makes up their vocabulary. However, the way in which each group member uses this vocabulary is dependent on the semantic conventions specified by its own behavioral matrix. This situation reproduces the real case in which a student doesn’t know what another student is thinking; he just receives an encoded message in a sound sequence.

6.1 The tentative nature of language usage

Before we go into simulation details, we need to discuss some of the assumptions and implications of the design sketched above. This simulation goes beyond the one of the previous chapter, in the sense that it is not concerned with the learning *per se*, or to put it differently, it regards learning from a different perspective. The finite state machines are quite rudimentary and lack the nuances that human cognition entails. Our choice was motivated by the assumption that learning could be modeled as a language acquisition process (from some point of view,
learning physics or any science could be even more problematic, because there are no decent dictionaries: one learns through an elaborate process of corroborating behavior and discourse; see for example [48]) and in as far as the acquisition process is concerned, even a rudimentary one would work. The emphasis of this simulation falls on communication and how it emerges along with semantic agreement — and learning itself — during the group performance of an educational task.

The language employed during this simulation was built around a limited vocabulary of symbols that group members could place on the common posting area. Generally speaking, real languages involve more complicated phenomena than what may be reproduced using such a limited vocabulary. However, technical jargon involves a limited number of terms. Moreover, a group of students working within a learning task usually experiments and explores the given problem, struggling to describe it using a finite set of words.

During their explorations, utterances may be tentative, searching, and often fragmented because students are uncertain of the technical language usage, although they may have expectations (regarding the actions generated by their words since most of the time ordinary vocabulary does overlap with the technical vocabulary) based on responses to previous discourses, and memories of previous experiences. This remembrance of past experiences is part of each learner’s personal knowledge. Exploratory talk is an acceptable, even necessary, way for speakers to bring in their tacit knowledge, or “personal knowledge”
([72]), such that through socialization they construct new meanings that all the members of the community will employ and be able to recognize henceforth.

To further support this assumption, we have recorded groups of students in Physics by Inquiry during Winter 2005 at Ohio State University, during class activities. We measured the noise level for each recording and we considered all the remaining signal to constitute students’ utterances. Further, we segmented the signal in non-overlapping frames of 20 ms. One technical challenge was the separation of the speakers on the recording. In order to overcome this difficulty, we assembled sets of microphones that were spatially separated and we recorded two-channel soundtracks for each of the monitored groups.

In order to identify the speakers on the two-channel recordings we used a DSP algorithm for source location ([74]). We made the assumption that, most of the time, the group members will not overlap each other on the recording. Therefore, a single source of sound will produce in each microphone a delayed signal

\[ s_i(t) = \alpha_i x(t - t_i) + n_i(t), \]  

(6.1)

where \( s_i \) is the signal as perceived at the microphone \( i \), \( \alpha_i \) is a sensitivity factor, \( t_i \) is the time it takes for the signal to travel from the source to the microphone, and \( n_i \) is the noise level at microphone \( i \). If we define the time delay of arrival by
\[ d_{ik} = t_i - t_k, \quad (6.2) \]

the cross-power spectrum phase (cpsp) function,

\[ cpsp_{ik}(\tau) = F^{-1}\left\{ \frac{S_i(\omega)S_k^*(\omega)}{|S_i(\omega)||S_k(\omega)|} \right\}, \quad (6.3) \]

where \( S \) is the Fourier transform of the signal \( s \), will reach its maximum at \( \tau = d_{ik} \) ([74]). Because the students do not change locations during the time segments we chose for this analysis, the time delay of arrival contains information about the location of the signal source. In this way, we managed to separate the frames into tracks corresponding to each group member’s speaking periods with an accuracy estimated to be 83\% (we selected randomly three 5-minutes recordings, separated the one speaker channels manually, and compared the results against the algorithm).

We filtered the resulting streams into a silence-speech signal with a time tolerance of 1/4 of a second (see, for example, [97] for related studies), adjusted to correspond to the average topic-sentence pause for our speakers. We plotted the distribution of speech and silence periods that we found in these recordings (see Figures 6.1, 6.2). These distributions remain similar for a study group from one session to another, and similarities are found even across groups. As the figures show, the students favor short-time utterances on the order of 1 second (most of the utterances did not take more than 10 seconds) during their discussions, which we find indicative of the exploratory and tentative nature of their conversations.
of their discourse. One should not give too much credit to the 1 second peak: because there is no perfect algorithm that could take out the entire noise from a recording, the noise along with interjections and other meaningless sounds do contribute to the 1 second peak. What we can state from the graph is that speakers tend to favor short utterances over complex discursive constructions. To further illustrate and support this point, we have included a transcript of a conversation in the appendices.
6.2 Finite state machines

Finite State Machines (FSM) are models of behavior for a system or a complex object. The essential features exhibited by these behavioral models are that they can model systems encompassing a limited number of conditions or modes, and that they are built with a specified way of learning or changing their behavior in response to changes environmental circumstances. Finite state machines are typically used as a type of control system in which knowledge is represented by states, and actions are constrained by rules. They are extremely simplified.
rule-based behavioral systems, which is exactly their strength. One may employ them to code behavior in a way that is not excessively demanding of resources.

There are four main elements that go into the design and building of a finite state machine:

1. A finite state machine encodes behavioral states that may generate actions, provided that a certain set of conditions are met.

2. There is a clearly specified way for the finite state machine to change its internal state, switching from one to another.

3. A finite state machine contains rules or conditions that must be met to allow a state transition.

4. A finite state machine accepts input events that are either externally or internally generated, which may possibly trigger rules and lead to state transitions.

One may visualize a finite state machine as a matrix having the dimension of the input space, keeping in mind that the input space contains all possible events generated from outside or inside the finite state machine. The simulation must start from an initial state. During the simulation this state will subsequently be modified by the interaction with the input events, but the finite state machine must be able to remember the product of the last state transition. In particular, received input events act as triggers that cause an evaluation of the rules
that govern the transitions from the current state to other states. In Figure 6.3, we represent the flow chart of a finite state machine having three internal states.

6.3 Semantic agreement and group size

We return now to our simulation. Before the simulation starts, we initiate the FSM behavioral matrices by use of random values in order that the group members begin with different semantic rules: none of them knows the semantic rules used by its neighbors when posting on the common board. Then, we allocate random questions from the training environment to the students to store in their internal states. The training environment of this simulation is made up from a set of symbols, each symbol corresponding to one question a student may ask or be asked with respect to a given task or problem.
The simulation starts by sweeping the entire group a sufficient number of times (this number depends on the size of the vocabulary used and was chosen such that the denotational matrix — see below — has enough time to reach a stable region), prompting each FSM to either post its internal question as a question (symbol) for everyone else to see, or try to answer to the posted question (symbol). If the student who posted the question finds the answer incorrect, the student who answered updates the behavioral matrix rule employed accordingly in order to increase the chance of answering appropriately next time. Thus, the student who attempted to answer is able to learn. If the
answer is, however, found appropriate, the student who answered receives a reward. In this way the simulation keeps track of the students’ performance.

After these sweeps of the group, the students receive fresh questions from the training environment and the process is repeated. When the group members have renewed their internal questions another number of times, the simulation decides which group member has the smallest number of rewards; in other words, the student who most consistently did not agree with the rest of the group. This student will simply discard its behavioral matrix and make up a new one composed from the two members of the group who performed consistently the best:

\[
c_{ij} = \theta(in+j-r_1)\theta(r_2-in-j)b_{ij} + [\theta(r_1-in-j)+\theta(in+j-r_2)]a_{ij},
\]

where \(a_{ij}\) and \(b_{ij}\) are elements of the behavioral matrices corresponding to the best and next best group members (for simplicity, we have given a formula for square matrices of size \(n\); the actual simulation used a generalization of this formula to three-dimensional matrices), \(\theta\) is the step function, and \(r_1\) and \(r_2\) are two random integer numbers \(r_1 < r_2\). The parent matrices are unwound and the resulting arrays are cut at the positions given by the two random numbers \(r_1\) and \(r_2\). The central piece from one parent is combined with the ends from the other parent to form the reconstructed matrix. The reconstruction formula makes sure that the reconstructed behavioral matrix preserves the symbol-action relations (semantic rules) that the parent behavioral matrices are
using. Also, the reconstruction formula is general enough to assure a combination between the semantic rules of the parent matrices without becoming implausible: the reconstructed matrix inherits one compact subset of semantic rules from one parent and one compact subset from the other.

On top of this combination between the best two behavioral matrices, there is a certain probability that some elements of the matrix are compromised during copying. In this way, the simulation makes the assumption that in real life, when one of the group members reconstructs the semantic rules or the understanding of a problem domain, as long as the group is well-balanced in terms of expertise, the worst student will employ mainly the two best choices. We also need to mention that this is the only point in the algorithm at which we introduce a scale. This fact will become important in the following sections.

Forty iterations of the sort described above make up a training epoch. During each epoch we compute the denotational matrix ([37]). This matrix records all the communication events associating a given answer with an expected answer: each time one group member gives an answer for the posted question, the element of the denotational matrix corresponding to the given answer and the answer that the student who posted the question was expecting is incremented. Therefore, the denotational matrix is a correlation matrix between expected and provided answers.
Initially, this matrix is uniform because we initialized all the FSM behavioral matrices randomly: in the beginning there is basically no correlation between expected and attempted answers because the group members employ arbitrary semantic rules. As the students begin to understand one another, the denotational matrix becomes singular. As we see in Figure 6.5, after a number of epochs, the singularity of the denotational matrix (see 6.5) reaches a plateau: for the given group size and structure, the denotational matrix has become as diagonal as it can become. In order to evaluate the singularity of the denotational matrix, we start by identifying the peak elements in the matrix, and
then reorganizing the columns and the lines of the matrix such that the peak elements line up on the diagonal. The average distance between the peak diagonal elements and the off-diagonal elements will characterize the singularity of the matrix:

\[
\sigma = \frac{1}{2(n-1)n} \sum_{i=1}^{n} \sum_{j \neq i} 2m_{ii} - m_{ij} - m_{ji},
\]

(6.5)

where \(m_{ij}\) is the denotational matrix, \(\sigma\) is its singularity, and \(n\) the dimension of the matrix or the cardinality of the vocabulary set.

One observes that this average distance peaks for the four-student group in Figure 6.6, a result that agrees with experimental results: starting from data describing academic achievement and active engagement in small study group, J. D. Hagman and J. F. Hayes recommend group sizes of four to five students ([33]; see also [44]). In general, the psychology and education literature agrees that groups of four or five members work best. Larger groups decrease each member’s opportunity to participate actively and have a negative impact on the academic performance, especially in the case of lower skill students.

There are other factors that have also been shown to contribute to the optimum size of the study groups. The amount of time available for instruction seems to relate proportionally to the group size: less available instruction time suggests smaller group sizes such that each group member receives a larger amount of individual instruction time. The skill of the students making up the group also varies proportionally with the group size: some authors suggest that less skillful students
need more instruction time, which may be increased by reducing the number of students in a group that share the total instruction time (see, for example, [16], [45], and [89]).

Figure 6.6: The singularity of denotational matrix vs. number of students

Two large-scale studies investigating the impact of instructional group size and academic achievement drew essentially the same conclusion: Larger group sizes correlated inversely with academic achievement for special education students. Gottlieb and Alter ([32]) based this conclusion on their evaluation of mandated increases from five to eight students in New York City resource and speech language classrooms.
Results from statewide reading achievement tests revealed that only 16% of sixth graders met state reading criteria after group size increases, compared with 29% before increases (1994-95).

Summing up the phenomenological results, there are two main factors that seem to influence the relation between group size and academic performance. First, the instruction time share for each individual member of the group: more students in a group decrease the amount of individual instruction time per group member. Following this line of argumentation one has to favor smaller groups over bigger groups, and even expect that single student groups should be the optimum instructional formula. However, the experimental findings show that larger groups (4-5 student groups) have a more beneficial impact on group and individual performance.

Second, the ability of the group to emulate and support active engagement of its members has a better chance to explain the phenomenological results. On one hand, a single member group does not engage the student much because there is no peer to support and entice the student. On the other hand, being part of a large group encourages isolationism because the group members wouldn’t have many opportunities to contribute actively to the group discussions. Our simulation, starting from different assumptions and monitoring a factor different from the two phenomenological factors discussed above, corroborates the experimental results and proposes a different way to look at them.
The peak in Figure 6.6 has been obtained starting from the assumption that one may judge the group performance based on the quality of the semantic agreement the group reaches. There are two essential ingredients responsible for this result. In the first place, one has to count the ability of the group members to learn and adjust their behavioral rules based on the group interaction. Although necessary, this feature of the simulation is not sufficient: the denotational matrix does not become singular if one turns off the reconstruction step (the singularity of the denotational matrix remains to lower values). Therefore, one has to extend the simulation to include the other essential ingredient: the reconstruction step.

It turns out that the reconstruction step in the simulation is essential for two reasons. First, the denotational matrix does not become singular without this step even after very long simulation times. Apparently, the simple learning and local adjustment of the behavioral matrix is not enough for reaching a semantic agreement. From this point of view, the “worst” students in the group initiate the semantic compromise necessary for a subsequent agreement.

Second, the reconstruction step introduces a scale in the simulation that relates to the position of the peak in Figure 6.6. In order to see this, we generalized the reconstruction algorithm to involve more than two students and plotted the singularity of the denotational matrix versus the group size for scenarios when 2, 3, 4 and 5 group members contribute to the reconstruction of the “worst” student. Figure 6.7
Figure 6.7: The singularity of denotational matrix vs. number of students for different reconstruction algorithms

shows that the peak in figure 6.6 does shift with the scale change. We see this finding as an experimental challenge: one may go and set up an experiment analyzing relatively homogeneous study groups of varying sizes for the optimal number of group members that seem to influence the group outlook in a larger extent.

6.4 Semantic agreement and cross group communication

The next question we asked was, What will happen if we extend the above simulation to more groups? How will communication across
groups influence the previous results? Therefore, we have settled for two groups of four but we extended the options a “student” has by giving the student an opportunity to ask, or listen, to a “teacher” in an adjacent group. Specifically, this time our “classroom” contains two four-student groups and each group member may, when its turn comes, post a question, answer a posted question, or learn (with a certain standard probability) the answer to the posted question from the best student in the adjacent group. This latter learning was implemented mechanically, the student simply updating the internal state in the FSM matrix with the corresponding state of the “teacher”’s.

Figure 6.8: The singularity of denotional matrix versus cross-group communication probability
For relatively large probability the singularity becomes smaller, which was expected given that asking students from the adjacent group is equivalent with extending the numbers of group members, but as we have seen already the singularity in that case peaks at around four group members. What is interesting, though, is the fact that for relatively small probabilities the singularity for this simulation is superior to a close group singularity for the same group size. Therefore, the simulation seems to suggest that cross-group communication is beneficial to reaching a semantic agreement as long as the amount of extra group communication is kept to lower values. This result corroborates experimental results ([33], [44]). Also, Robert J. Beichner ([8], [36]) has implemented a modified version of PBI tutorials developed by the PER group at the University of Washington: instead of organizing students in separate work groups, he placed three groups of three students each around a 6-foot round table in order to encourage cross-group interaction. However, every three student group was evaluated as an independent unit and students were encouraged to collaborate mainly within their group; interaction with the students in the adjacent groups was accepted only when the entire group reached an impasse, and before the group would request the assistance of an instructor. He reports significantly improved performance in problem solving and increased conceptual understanding: 43% and 50% average normalized gain on FCI as compared with only 23% normalized gain for traditional classes.
In the area of physics education research, many studies are concerned with developing descriptive and phenomenological cognitive models of student learning processes. However, there has been little research that attempts to systematically construct computation-based theoretical models for student learning in physics. We believe that developing such models will have a significant impact upon existing education research and teaching practice. Through such studies, researchers will gain deeper understanding of the dynamics of student learning. Given a well-developed model, educators will be equipped with additional means to evaluate curriculum and pedagogy. It will help researchers identify latent and implicit structure among student behavioral patterns provided by empirical studies.

Computational modeling of cognitive process has been around for a long time. The availability of powerful learning algorithms such as back-propagation has created a situation in which we now know how to teach neural networks many complex things in a wide range of task situations. For example, there are successful models of the acquisition
of word reading skill ([71], [83], [84]), of physical knowledge such as object permanence ([60]), and of conceptual knowledge ([41], [79]) such as kinship relations and the natural kind hierarchy. A more complete collection of recent work in cognitive neuroscience is presented in [30].

Within the vast neural science literature, there are relatively few studies on modeling the developmental processes of student learning in highly specialized knowledge domains such as physics. There is some related work (e.g., [25] [54]); however, systematic studies on applying the methods of neuroscience and computational systems to model physics learning are largely absent. In this work, we have shown that it is possible to build computational models that are relevant to the educational process in physics. This shows that this field can contribute to the future of physics education research and should give an impetus to this fledgling field.

We have used neural networks to model the context-dependent knowledge acquisition of a student population. We have observed that, during training, our models visit the same sorts of mistakes that students make. The number of possible mistakes for a given problem is relatively large; however, the number of mistakes students commit is amazingly limited and this set of mistakes remains the same from one student generation to another. Our network models, starting from only a few assumptions about the nature of the learning process, have shown features similar to the learning behavior of the real student population.

We conducted research in three areas of studies:
1. Development of computational models to simulate students’ learning behavior for conceptual problems in introductory Mechanics and Electricity and Magnetism courses. The emphasis was on studying how the computational systems categorize the context cases and how differences in training trajectories may affect the categorization.

2. Study of the connections between computational models and phenomenological models; in particular, we tried to gain deeper understanding of issues underlying the context dependence of learning, the categorization of context cases, and the transfer of knowledge.

3. Development and testing of a simulation method that could evolve into an educational tool that instructors might use to evaluate courses, curricular material, and instructional strategies.

4. Enlargement and refinement of formal models borrowed from artificial intelligence and cognitive science in order to apply them to understanding the mechanisms of group dynamics and interpersonal interaction of collaborative groups in physics education.

7.1 Summary of Student Learning Behavior

Students come into the classroom with a knowledge corpus distilling their personal experiences and previous education. In recent decades, education research has documented a rich collection of student difficulties in learning science and mathematics. Many cognitive theories have
been proposed to interpret these findings. The research suggests that student learning behavior exhibits four general features:

1. Learning is affected by students’ previous knowledge. Students come to new learning experiences in possession of a system of knowledge developed from personal experience and previous learning. Many studies in physics education have shown that certain preconceptions can pose strong barriers to understanding physics ([14], [34], [39], [38], [52], [93]).

2. Students can create new alternative ideas that can be different from both their previous knowledge and the ones accepted by the scientific community. Both young children ([95]) and college students ([6]) have been reported to have created new alternative understanding. Students’ creations of alternative ideas are significantly affected by the instructional settings and methods, i.e., the created “wrong or semi-wrong” ideas are often directly related to certain instructional approaches.

3. Students’ knowledge structures can be different from that of experts in many ways. For example, experts categorize problems based on the principles used in their solutions, while novices tend to categorize problems using surface features of the problem ([13], [50], [49]). Clearly, a deeper conceptual understanding is necessary for students to go beyond the surface details to succeed in gaining knowledge.
4. There are commonly observed inconsistencies in students’ use of their knowledge in equivalent contexts ([7], [15], [69]). For example, when presented with a set of questions that are conceptually equivalent but are designed with different contextual features, a student who does not see the coherent deep structure may call on different (and often contradictory) types of knowledge. This implies that students may hold more than one type of understanding in their minds simultaneously. Which type they choose to apply is likely to depend on the context presented and on their experiences with similar contexts.

There is a large body of literature comprising both theoretical and empirical studies that addresses specific issues of students’ learning in physics and other subjects ([12], [20], [21], [35], [73], [59], [91], [94]). A detailed discussion of these learning behaviors and the development of assessment methods can be found in ([7]).

7.2 Computational models

There are two types of systems widely used in modeling student learning: the production system approach ([4], [63], [62]) and the connectionist approach ([57], [68] [80]). The production system approach is based on a fine-grained analysis of the sub-skills involved in a problem solving task, and on the representation of this hierarchical skill set as a collection of “productions” (logical if-then rules). There are also successful examples of the use of computational models in education
practice. For example, intelligent tutoring systems (ITS) have been constructed that employ very fine-grained production system models of both expert and novice behavior; these ITSs have shown major instructional impacts in the teaching of topics in programming and mathematics.

The features of the system we are trying to model make it difficult to construct as a production system. A production system usually codes what a subject is doing, but not why the subject ends up doing what he is doing. ACT-R is more finely tuned than that (e.g., stochastic decision making, schemes, loose probabilistic coding, etc.); however, its strength is in its descriptive power. The goal of our model is to find why our system behaves the way it does. In physics learning, there are a huge number of pre-existing, co-existing, and on-the-spot creations of knowledge pieces that are locally and or globally associated with an even larger set of contextual features. The complete association network, including its components, evolves constantly through non-linear pathways. Therefore, we need to use non-linear methods (similar to many of those used in physics) to treat the system with an adaptive and self-organizing structure.

The model we have developed emphasizes the one issue that we believe is fundamental to understanding learning behavior in physics: the relation between contextual features and the formation, cuing, and application of specific types of knowledge. A significant feature of this model has many similarities to “categorization,” which refers to the
process that specific contextual features became associated with specific types of knowledge and later serve as the cuing stimuli to activate the knowledge. Another important process in the model is how the system synthesizes the activated knowledge with the contexts to produce responses and to obtain feedback.

The complete process is adaptive and self-organizing. The challenge resides in the way we represent the physical contexts and the knowledge, and in the self-organizing structure of the system so that it can learn adaptively in ways similar to human learning.

There are two unique characteristics of this type of system.

- We do not pre-code any behavioral rules into the system.
- We do not pre-code any internal representations into the system.

The system learns its own rules and develops its own internal representations through teaching and learning interactions. These features reflect a different emphasis compared to many existing categorization algorithms such as EBRW exemplar model of Nosofsky and Palmeri ([65]), the RULEX rule-based model of Nosofsky, Palmeri, and McKinley ([66]), and the ACT-R implementation of a hybrid model based on variations of EBRW and RULEX ([3]). All these methods use pre-coded rules to perform categorization tasks.

The results contained in the previous chapters showed how such models can be useful to education research and practice. For example,
we have seen the “stage-like” developmental process similar to the results discovered by McClelland and Jenkins ([54]) with balance scale tasks. What is interesting is that our study is performed with a very different domain of knowledge, which is conceptually more complex and subtle. In addition, our population is also very different: college level engineering majors. Still, we see similar developmental characteristics that are well documented in children’s learning with simple tasks ([42], [86]). It will be important to identify whether such a developmental pattern is general among college-level students in their learning of science topics, or how such pattern may differ when the population and the content domain vary.

Another advantage we see from this study is that we have shown how to combine both empirical research and computational modeling in a single study, which can significantly enhance our ability to understand the problem. Basically, we can address a problem from two different approaches simultaneously. In chapter 3, we have already tasted the fruitfulness of this integrated approach. For example, the simple BPN model has aided us in interpreting student behaviors of “unlearning” and in identifying “hidden” categories, both of which are difficult to study using only empirical data and methods.

We have seen that some of the mistakes students make are integral parts of the learning process — not in the sense that a targeted instruction could not avoid them or that all the students necessarily make them, but in the sense that students, based on their initial state,
background, and training environment are likely to make. One should devote some effort to the assistance and guidance students need to troubleshoot their impasses and common mistakes they might end up making. Students do tend to reach a state of relative complacency once they settle into a plateau of the error function and are not willing to spend the effort needed to jump out of local minima. This is the moment they need all the attention, support, and motivation they can get from the instructors or their study group.

Phenomenological investigations in physics education research have already pointed out the existence of the student difficulties in understanding certain concepts contained in the standard physics curriculum and the need to dedicate special attention to them. Even if the conceptual view that the instruction is trying to enforce is plausible, the students tend to cling to their beliefs because having taken the effort to construct their private understanding, the students will not easily relinquish their view points ([46]). Only by directly confronting their views can instruction successfully trigger cognitive conflict and subsequent conceptual change. This corroborates Chapter 4 findings that the maximum-error selection strategy has the biggest success.

A well-known way to manage students difficulties has been developed by the Physics Education Group at the University of Washington (see [2] and [56]). At the heart of their tutorials are carefully structured worksheets that guide the students through the reasoning necessary to overcome specific difficulties identified by research. Students work
collaboratively through the tutorials in small groups (see chapters 5 and 6 for a theoretical view at the group dynamics and size), and the instructor teaches by questioning rather by telling. We believe that our simulations provide further support and insight into the widely documented success of these methods ([1], [55], [28], [82]).

Organizing the study material in an active way based on continuous progress checking during the learning process may decrease the learning times, as the simulations of Chapter 4 have shown. We believe that the most convenient way to implement this kind of testing is by encouraging self-testing and self-evaluation during learning. In this way, the students acquire the criteria and the strategy for becoming learners actively involved in organizing their learning activity. However, a certain degree of guidance in terms of extra and objective evaluation remains necessary because small changes in the next question selection strategy may produce dramatic effects on the performance of the artificial students: having an consistently inaccurate evaluation of their own understanding with respect to the target expert understanding can throw off the entire process — some of the error-based selection strategies have never managed to learn the given problem.

There is definitely no good substitute to self-guided learning, but Chapter 5 suggests one may obtain a lot of help from one’s peers. Working in a group has a positive impact on the individual performance, especially for learners who would have taken more time to learn on his or her own, without slowing down the faster learners. Moreover, rather
than listening to the “leader” of the group, the slower members seem to favor the advice from the intermediate level learners. Also, the comparison between the distributions of the number of group members who finished the learning task after a fixed amount of time when they either choose randomly whom they listen to or they use a decision module powered by emotional attachment in conjunction with previous experience seems to favor the decision module over the lack of a decision, and therefore support Deutch theory.

However, the internal dynamics of the study group is determined by the quality of the communication. Moreover, to a certain extent, one may see a learning process taking place within a group as similar to a process of reaching a semantic consensus within that group. Chapter 6 proposes a scenario for a group semantic play that seems to lead to the emergence of the semantic agreement within the group. As a side result of the simulation, making the assumption that two of the players of the semantic game influence the rest of the group, the simulations suggest that a group size of four is the optimum size for reaching the highest semantic agreement. This argument, seen in reverse, suggests an experimental test for the above theoretical assumption.

The results of this thesis can significantly impact the ways we conduct education research, assessment, curriculum development, and teaching practice. This research can help researchers gain a deeper understanding of the dynamics and the interactions between context and the
students’ learning processes. For example, comparisons between computational results and empirical results of students’ learning behaviors across a set of expert-equivalent contexts can help us understand how students categorize the context set. This provides valuable insights into the origin of many learning difficulties, such as inappropriate fragmentation of knowledge, and issues within the transfer of knowledge. Additionally, these computational simulations can provide new tools to help researchers interpret student learning behaviors, which are otherwise difficult to be addressed with only empirical studies (e.g., see the phenomenon of “unlearning” and “hidden” context categories discussed in the simulation of Chapter 3). In this research, we looked for commonalities and differences of learning patterns between both real student populations and simulations. Any common patterns that may surface, such as the “stage-like” patterns, have significantly values to development of cognitive theory of learning physics. We explored ways to develop simulation tools, which can conveniently perform simulations of learning under different contexts for instructors and researchers to test and explore learning hypotheses. This type of development can pave the ways for the birth of the future generation education environment.

Combining theoretical modeling and experimental design, we will be able to advance research methodology through the development of a paradigm of how to integrate both methods effectively in our research to maximize their strengths. In addition, physics education research
as a field can benefit from the development of two mutually supportive components: an empirical research and a computation-based theoretical component. These two aspects could jointly investigate more successfully the complex system of educational research.
CHAPTER A

QUESTIONS

The following questions have been part of the checking protocol for the PBI class in Winter 2004 at the Ohio State University. The “x” sign in front of some of the multiple choices represents student’s’ answers and not necessarily the correct answers — the correct choice is highlighted in green. The testing has been administered through a web interface specially designed for this task.

5 - Please predict what happens with the balance in the following figure:

[ ] The left side descends.
[ ] The balance remains in equilibrium.
[ ] The right side descends.

Figure A.1: PBI 106 Section 2
4 - Knowing that 2 nuts balance 3 beans, what will happen when we place 4 nuts and 1 bean on the left pan and 3 nuts and 3 beans on the right pan of a balance?
[ ] The left pan descends.
[ ] The balance remains in equilibrium.
[X] The right pan descends.

Figure A.2: PBI 106 Section 1

3 - If the balance in the figure is even, what will happen with the balance in the following configuration?

[ ] The left pan descends.
[X] The balance remains even.
[ ] The right pan descends.

Figure A.3: PBI 106 Section 3
4 -

Please compare the shapes in the figure below:

[X] The left shape is bigger.
[ ] The shapes have the same size.
[ ] The right shape is bigger.

Figure A.4: PBI 106 Section 6

2 -

A rod of length l=5cm made from a material of density 2g/ml is immersed in a liquid of density 2g/ml up to a depth of h=4cm as in the figure. What happens to the rod once it is released?

[X] The rod descends.
[ ] The rod remains in the same position.
[ ] The rod ascends.

Figure A.5: PBI 106 Section 12
4 - Knowing that the density of the saturated salt water is 1.2 g/ml, please predict what will happen when one adds 3g of salt to a solution made out of 9g of salt and 19ml of water.
[X] The density of the liquid will increase.
[ ] The density of the liquid will remain the same.
[ ] The density of the liquid will decrease.

Figure A.6: PBI 106 Section 17

3 -
For the graph in the figure below, please compare the slopes of the graph at the points A and B.

[X] The slope at the point A is bigger
[ ] The slopes at the points A and B are the same.
[ ] The slope at the point B is bigger.

Figure A.7: PBI Section 13
3 - You are given two containers. The first one has 350ml of 6% solution and the second one has 432g of 7% solution. Please compare the amounts of solute in the containers.

1. The first container has more solute.
2. Both containers have the same amount of solute.
3. The second container has more solute.

Figure A.8: PBI Section 20
(1) In each of the following situations, there is an initially-neutral conducting rod (______) made of copper. In some cases, positive or negative external charges ( raided or ) are placed closed to the rod. The rod can also be grounded at one of the ends. For each of the following cases, please indicate the net charge on the rod. Please indicate your confidence of your answers: “0” for not sure/guessing, and “5” for very sure.

<table>
<thead>
<tr>
<th>Situations</th>
<th>Net charge on the rod</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Positive</td>
<td>Neural (zero)</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
</tbody>
</table>
This conversation took 3 minutes and 30 seconds.

Student A: Put the bulb in the middle of it, too.
Student B: You mean...
Student A: Like make it connected to it.
Student C: And we can turn the switch on or off.
Student B: Yeah.
Student C: Like put the bulb in between.
Student A: Yeah.
Student C: Right there.
[silence]
Student A: Move it easily outside.
[silence]
Student A: Let’s turn it off.
Student B: Like that?
Student A: And then put a bulb connected to this one and that one. I put a wire, I am sorry. With the bulb.
Student C: Right in here connected to... here.
Student A: Mhm.
[silence]
Student A: Will this work? Are this switches on or off?
Student B: So it will be like...
Student A: Er. Or maybe not.
[silence]
Student C: It says whether you turn the bulb on or off...
Student A: That does not work... hm.
[silence]
Student A: Will that bulb...?
Student C: This bulb just connects to...
Student A: Yeah.
Student C: Put the other one on.
Student A: And the connection from these two somehow...
Student B: This to this?
Student A: Yeah.
Student C: And now...
Student A: Wait a second...
Student B: Can I swipe these two?
Student C: Oh...
Student B: Or no.
Student C: Oh, wait. Keep that one down... Ok.
[silence]
Student C: Oh... see what this one...
Student B: [laughter]
Student A: Hm.
[silence]
Student C: They still have to be on this side unfortunately...
[silence]
Student C: Mmm. Uh? Hm. Here...
Student B: What?
Student C: This one has to be like this... You can...
Alright.
Student B: Let’s move this one.
Student C: This is like the controlling one right now...
Student A: And you hold this...
Student C: Hold one moment... wait... let me... This one is the controlling one right now.
Student A: Mhm.
Student C: So remove the connection... the other connection from this side... and then remake it here.
[silence]
Student B: Well, some of them...
Student C: Yeah. Put it here. Oh... Why don’t we do like this?
[silence]
Student C: Ok. You know what I’m saying?
Student B: Me too?
Student C: Yeah.
[silence]
Student C: This side... Ok.
[silence]
Student C: Oh, oh... Didn’t this just turn off like that?
[silence]
Student C: That one should.
[silence]
Student A: There could be not enough batteries...
Student C: Yeah. That is true.
Student A: Just push it down like this.
Student B: The switch can turn on or off independent of the position...
Student A: That should turn off. Close that now... Put it to the other side. If it’s open...
Student B: That’s closed.
Student A: It shuts it off.
[silence]
Student A: Yeah... Turn it on. Here’s...
Student B: How to connect on this side?
Student A: I don’t know: turns it on and off...
Student B: Oh... Yeah! We are right.
Student C: They didn’t have to go through... It is
supposed to be regardless of what the other’s doing.

Student B: Oh... Yeah.

Student C: Yeah... wait for a second. This is it.

Student B: Let me adjust this.

Student A: So... Should we draw that?

Student C: Yeah.

Student B: Yeah.
CHAPTER C

SUPPORTING WORK: PROTAGORAS WEB TESTING INTERFACE

Initially, we designed and wrote Protagoras to assist us with the checkpoint protocols for PBI classes: students work in groups following detailed textbook directions, but from time to time they are prompted by the textbooks to request an instructor’s assistance. The instructor will check their group’s progress and make sure they have appropriately assimilated the material covered in their experiments or exercises. Therefore, we needed an integrated testing interface that would keep track of the groups and administer different tests for each checkpoint in the textbook before the students need to ask for instructor’s presence. This way, we intended to have a way of recording the student’s status and understanding for ulterior educational analysis (the traditional way of handling the checkpoints does not include any way to record the group’s status, giving a free hand to the instructor to ask any questions she might find appropriate).

Meanwhile though, the design has been extended to cover most of our research group testing needs. Up til now, the system has been
used extensively by most of our research group members, its behavior remaining very robust as an increased number of students, questions, tests, and test results are included.

Since its main goal is acquisition of educational research data for easier further processing, the system implements an intricate hierarchy of tags and categories; it provides a multilayer architecture of rights and privileges to accommodate and encourage shared authorship on questions and tests; it exhibits a gradings system for three types of questions including open-ended questions and a way for the instructors to give feedback to students or tag student’s answers.

The system records, besides student answers, various things that a researcher might find useful, such as administration date and time, duration of the test, number of attempts, all the answers that the students

Figure C.1: A test question during test design — instructor view
gave even if the instructor may choose to ignore all the attempts except the last during grading; it may record the order of the questions in a test that can be enforced by the instructor at the test design stage or the questions may be batched in a test. All those data may be used subsequently to generate scripted reports and plot charts, run different statistics on any organized subset of questions, tests, or students. Since the system keeps track of students by organizing them in groups, the tests are given to students at the allocated time and to the groups of students it has been designed for, making the student interface to the system very straightforward.

The instructors have the option of pooling questions together in categories or subjects and generate the tests on the fly by having the
Figure C.3: A student group results frame — instructor view

system randomly select questions from the groups of questions indicated at the test design stage. In combination with sequencing the questions in a test and tagging the questions (the instructors may define their own tagging system), one may use the system to give test that automatically adjust the difficulty of their questions based on the student’s performance during the test. In order to discourage cheating, the system caches the generated tests, if they have dynamic content, and make sure that the clients may not reload the web page, and, in this way, request a new test, once the test has been assigned.

<table>
<thead>
<tr>
<th>User ID</th>
<th>Name</th>
<th>Test Name</th>
<th>Pass/Fail</th>
<th>Score</th>
<th>Date/Time</th>
<th>View Answers</th>
<th>View Answers (PDF)</th>
<th>Delete</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>239</td>
<td>ELIASLOPEAN</td>
<td>test1</td>
<td>Pass</td>
<td>100%</td>
<td>05/12/2020 9:00 am</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>235</td>
<td>CRISTOLAHAM</td>
<td>test2</td>
<td>Pass</td>
<td>80%</td>
<td>05/12/2020 10:00 am</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>227</td>
<td>DEEPAKDHILLI</td>
<td>test3</td>
<td>Pass</td>
<td>90%</td>
<td>05/12/2020 11:00 am</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>246</td>
<td>HARSHBAGASDA</td>
<td>test4</td>
<td>Pass</td>
<td>75%</td>
<td>05/12/2020 1:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>229</td>
<td>HARSHBAGASDA</td>
<td>test5</td>
<td>Pass</td>
<td>85%</td>
<td>05/12/2020 2:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>232</td>
<td>HARSHBAGASDA</td>
<td>test6</td>
<td>Pass</td>
<td>90%</td>
<td>05/12/2020 3:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>235</td>
<td>HARSHBAGASDA</td>
<td>test7</td>
<td>Pass</td>
<td>80%</td>
<td>05/12/2020 4:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>230</td>
<td>HARSHBAGASDA</td>
<td>test8</td>
<td>Pass</td>
<td>75%</td>
<td>05/12/2020 5:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>237</td>
<td>HARSHBAGASDA</td>
<td>test9</td>
<td>Pass</td>
<td>90%</td>
<td>05/12/2020 6:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>231</td>
<td>HARSHBAGASDA</td>
<td>test10</td>
<td>Pass</td>
<td>85%</td>
<td>05/12/2020 7:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>228</td>
<td>HARSHBAGASDA</td>
<td>test11</td>
<td>Pass</td>
<td>90%</td>
<td>05/12/2020 8:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>229</td>
<td>HARSHBAGASDA</td>
<td>test12</td>
<td>Pass</td>
<td>80%</td>
<td>05/12/2020 9:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>230</td>
<td>HARSHBAGASDA</td>
<td>test13</td>
<td>Pass</td>
<td>75%</td>
<td>05/12/2020 10:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>231</td>
<td>HARSHBAGASDA</td>
<td>test14</td>
<td>Pass</td>
<td>90%</td>
<td>05/12/2020 11:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
<tr>
<td>232</td>
<td>HARSHBAGASDA</td>
<td>test15</td>
<td>Pass</td>
<td>85%</td>
<td>05/12/2020 12:00 pm</td>
<td>Yes</td>
<td>Yes</td>
<td>Delete</td>
<td>Grade</td>
</tr>
</tbody>
</table>

This page shows test results for all tests that have been scored. To view the individual questions and answers for a test, click on the ‘view answers’ link. You can reorder the results to appear in order you are interested in by clicking on the test name.
Students have welcomed the new system, adapting very quickly. Among 121 students who have used the system during Spring Quarter 2003, only 2 students had questions regarding technical issues. We have asked students to express their preference between the traditional paper-based testing and the new web-based testing:

- 44 ± 6% prefered paper-based system
- 78 ± 8% favored web-based system

In particular, we reproduce here the opinion of one of the students:

I think the web journal is a much better idea than a paper-based one because it decreases the chances of losing your journal and the penmanship is much easier for the teachers to read. I also like it because after I submit my journal I am done with it until the next week. I think paper based journals are a waste of paper and too complicated.

In order to further analyze the way the web-based testing interface influences student’s responses, we have compared the word-count for student’s answers to open-ended questions (mainly PBI journal questions) between a class that used the web-based system and a control group that used the traditional paper-based system. The number of words in answering a particular question was:

- paper-based system: 97 ± 42
- web-based system: 64 ± 11

We quote here the opinion of one of the students that we find insightful in the above difference between the verbosity levels:
I think when writing a paper journal you can write bigger and use more spaces to take up a lot of the paper as opposed to a web where everyone has the same style and spacing. I also think that in a web journal you can concentrate more on the problem rather than taking up space.

As conclusion to this appendix, we review the most salient features of our web-based testing system: Protagoras manages and facilitates the authoring process, it handles different testing protocols, it contains an intricate PER tagging system, it provides a simple interface to a statistical package or an adaptive system engine, and, in the end but not the least, it has been well-received by students.
BIBLIOGRAPHY


