Giving the Head a Hand:
Constructing a Microworld to Build Relationships with Ideas in Balance Control

by

Arnan Sipitakiat

M.S. Media Arts and Sciences
Massachusetts Institute of Technology, 2001

SUBMITTED TO THE PROGRAM IN MEDIA ARTS AND SCIENCES, SCHOOL OF ARCHITECTURE AND PLANNING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

SEPTEMBER 2007

Signature of Author: ______________________________
Program in Media Arts and Sciences
Aug 10, 2007

Certified by: ______________________________
David P. Cavallo
Research Scientist, The Future of Learning Group
MIT Media Laboratory

Accepted by: ______________________________
Prof. Deb Roy
Chairperson
Departmental Committee on Graduate Students

Giving the Head a Hand:
Constructing a Microworld to Build Relationships with Ideas in Balance Control

by

Arnan Sipitakiat

Submitted to the Program in Media Arts and Sciences, School of Architecture and Planning on August 10, 2007 in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Abstract

The major promise of computational technology for learning is in making discovery and acquisition of knowledge accessible to a wider range of people. The protean expressive and constructive nature of computational technology facilitates more powerful and effective learning methodologies. Enabling multiple forms of representation through computational approaches to thinking about various phenomena not only potentially opens new domains of knowledge, but also permits a re-structuration of domains by re-thinking content and activity. This thesis provides an exemplar of this potential through children learning about Balance Control in Dynamic Systems (BCDS), which adds a particular value given that BCDS is considered too complex for young learners.

A Balance Control Microworld was created to help learners think about how to program physical robots to perform balancing acts, such as balancing an inverted pendulum, based on the observations of their own body motions. A Spatial Computing Paradigm (SCP) was developed to allow learners to carry out various control operations using familiar 2D properties of on-screen objects. The physical robots have a dual-mode ability that allowed learners to record and observe motions while controlling the robots manually by hand as well as under program control. The study involved two groups of learners, ages 13 to 15, over twelve months. BCDS concepts that emerged include the role of speed, creating predictions, managing system states, and analyzing system's stability. Moreover, powerful ideas in computational and mathematical thinking helped enable thinking and understanding in BCDS as well as reflection over the whole process.

The evolution of the Microworld was guided by a practice of applied epistemological anthropology. An iterative process was used to identify important themes as they emerged during the course of the fieldwork. The resulting themes, as reflected in the case studies, come in three flavors: One focuses on ideas in BCDS that were learned by youth and could lead to deeper understanding in that rich field; the second shows how the tools and approach evolved to better support the learner along with the role of the researcher in the learning process; the third discusses the learning implications of a technology-enhanced Microworld by demonstrating common learning assumptions that need careful reconsideration.

Thesis supervisor: David P. Cavallo
Title: Research Scientist, The Future of Learning Group, MIT Media Laboratory

Giving the Head a Hand:
Constructing a Microworld to Build Relationships with Ideas in Balance Control

by

Arnan Sipitakiat

Certified by: ____

Seymour Papert
Professor of Media Arts and Sciences: Emeritus
MIT Media Laboratory

Certified by: ____

Joseph Paradiso
Associate Professor of Media Arts and Sciences
MIT Media Laboratory

Certified by: ____

Edith Ackermann
Honorary Professor of Developmental Psychology,
University of Aix-Marseille I, France
Visiting Scientist, MIT School of Architecture

Acknowledgments

I arrived at MIT on Aug 17, 1999. That was roughly eight years ago. I have been so privileged to spend these years with remarkable mentors and peers that have influenced me in amazing ways. I arrived as a computer nerd. The lens that I used to view the world was really narrow, but yet so established. My brain cells would become vividly stimulated when faced with a technical problem. But when it came to ideas in epistemology and learning, all I got was a void. Although I was interested, I would not know what to think or say as if someone had pressed the mute button on my brain.

As the years went by, no one really "taught" me about epistemology. But I received plenty of opportunities to work with children in various places and cultures. I don't know how, but the more experience I gained (though fieldworks, writing the two theses, etc.) the louder the epistemological thinking in my brain became. In Marvin Minsky's terms, perhaps my brain needed time to build up enough resources, selectors, and critics to actually start functioning in this new domain. But whatever it was, I feel that I now begin to have access to ways of thinking that would have been far fetched to me back in 1999. I have no doubt that this rather mysterious transformation took place only because I am in a special place working with special people.

I truly need to thank my advisor David Cavallo for helping me since day one. I still can never quite predict what he would say or think of a particular situation but I have learned that he has always been on target and he has never steered off course (although the target is usually moving at all times). I truly believe this is a rare quality one can possess. When I get stuck with my work, I often go for a walk at the Charles River. There you can see Boston from a distance, which helps remind me how to step back and look at the big picture. This is truly a Cavallo influence. Also, thank you for letting me do what I want to do and supporting it through the end.

I am honored to have Seymour Papert on my committee. People have told me about the "Papert magic" where he can make a struggling student feels that his or her work is great, simple, and clear. I first experienced it a few days before my Master's thesis was due. Through one phone conversation and one e-mail, I suddenly realized what my work was all about. And I could not believe how I was not able to see it before. That is the kind of moment that makes you realize how special a person is. I am truly saddened by the tragic accident that happened to him, which is why his signature is left blank on page 5. I wish him the best for a full recovery.

Joe Paradiso has always been a role model to me. I am very proud to have him on the committee. He has given me countless insights throughout my thesis process. The thesis would have lacked many highlights without him.

Thanks to Edith Ackermann who have stepped in towards the end after Seymour's tragic accident. Every time we spoke, I learned so much. I have heard so many things about

Piaget throughout the years at MIT, but you are the one who really put them together in a coherent way for me. It is so valuable and I am grateful for your kind offer to help out.

I need to thank the students who participated in my work. They have been great and I really enjoyed our time together. Thanks also to their parents who have been so supportive. Thank you, Jacob, for helping me film the sessions. Those videos were really valuable.

I would like to thank Jacqueline for supporting me through the years. Whenever there is a problem, Jaco is always there for the rescue. I spent my first night in Boston at your place and that is something one does not forget. We have to make sure we get you over to Thailand one day. We'll do the long overdue elephant ride together!

I owe so much to Khun Paron Isarasena who had opened up countless opportunities for me to experience how learning innovations can be applied within the context of the Thai culture. He has done so much pioneering work on learning in Thailand, which makes me feel assured that I will have rich playing fields when I return. Prior to MIT, he had also provided me many opportunities to meet a number of MIT professors, which surely gave me the exposure that brought me to the Media Lab.

Michael Temple and the SEED foundation have been extremely supportive. My work with the GoGo Board has gone so far because of them. I am grateful for their financial support over the past year. They supported me by letting me do what I love. That is not easy to come by!

Thank you, Paulo, for being such a good friend. I am very grateful for your kindness and coming all the way from Chicago to be with me before and during the defense. That alone really sums up our friendship. It is beyond words. I am sure we still have many adventures waiting for us in the near future.

Thanks to Claudia and Laura, my two officemates, for making us feel like a group. It is really unfortunate we did not have the chance to travel and do work together very much. We will make it happen eventually!

I would also like to thank Cynthia Solomon for helping me with the thesis text. You have always been kind and supportive.

Thank you, Rinda, for your understanding and support. My life has truly been brighter than ever through your presence. You make me feel like there is a lot more to life than just MIT and my research. I consider myself very lucky.

Last but not least, I need to thank my parents for letting me be away for so many years. I hope I make you proud. I also want to thank my grandmother in Australia for caring so much about me. You have been a special person in my life since the day I was born. We don't meet that often but somehow I feel you have always been with me. Our relationship has made me stronger in ways that can never be expressed on paper.

Table of Contents

1. Introduction

This thesis is an exploration into how open, expressive computational environments can provide fertile ground for acquisition of knowledge by learners. Rather than looking at computers and digital media as just another platform for information delivery, the approach adopted here emphasizes the role of computational media in forming, expressing, testing, and debugging one's own hypotheses about phenomena at hand. This approach is applied not only in a domain previously considered too complex or challenging, but also as a means of developing a computational approach to thinking about phenomena that potentially extends beyond the particular subject matter.

This work presents the evolution of a Microworld for learners that provide access to ideas in Balance Control of Dynamic Systems (BCDS) through the construction of program models. The empirical basis for this thesis is a study with two groups of learners, ages 13 to 15 years old, over a twelve-month period. The activity was focused on programming physical robots to perform balancing acts (e.g. balancing an inverted pendulum). The development of the activities was driven by the learners' observations of their own body actions balancing the physical objects they wish to automate. The evolution of the Microworld was guided by a learning philosophy that puts an emphasis on building understanding from within the learner. Instead of passing down neatly organized concepts, a practice of applied epistemological anthropology was used to build new understandings from the learner's existing experiences.

The work with students has led to my development of a Spatial Computing Paradigm (SCP) which allows learners to carry out various operations using 2D properties of on-screen objects. The virtue of such approach is that it allows the learners to develop their understanding about balance based on their existing understanding of objects in space. The aim is not to deny the validity of other more formal approaches. Rather, SCP provides access (lower threshold) for children to the underlying principles while still being flexible in order to produce a working system (high ceiling).

The Microworld utilized physical robots that learners can both program or carry out the desired tasks manually by hand. This dual-mode ability has been augmented by applying ideas in Computerized Dynamic Posturography (CDP) where learners can record robot motions during both automatic and manual control for later comparison and analysis. The emphasis on SCP and developing understanding by reflecting on peoples' body motions balancing physical objects are the main components of a learning approach called "Physio-Syntonic." The Spatial Computing Paradigm, in this case combined with the physical apparatus and human activity, builds upon the concept of syntonicity, enabling people to project what they understand about the physical world and actions within it, into the on-screen Microworld and their descriptions of how to control events, thus facilitating the construction of knowledge about the principles of the underlying phenomena being studied.

I have observed further evidence that neither by being told nor even by observation of events that contradict learners' hypotheses, do learners necessarily change their minds. It was often the case that the best way to support the learner was by allowing them to act on the world as they see it and let them reflect upon the (often unexpected) results. Thus, rather than attempting to replace the learners' ideas with correct ones, the Microworld and the methodological approach values the integrity and idiosyncrasy of how learners' conceive phenomena, and seeks to provide means (which do not exclude expert advice) for the accumulation of self-realized refutations to the learners' theories, where appropriate, and the progressive construction of better understanding and more functional representations.

Because this work is a design project, the main theme in the case studies is about how the learners respond to (a) the situation being observed at hand and (b) the researcher's suggestions. The researcher did not play a role of an instructor who attempts to inject all the correct ideas to the learner. Instead, the decisions of what to do were always a negotiation process where the learners ultimately decide for themselves. Thus, the case studies consist of many situations where the students did not accept the ideas being suggested or altered the ideas by merging them with their own thinking. For the researcher, these interactions played an essential role in the refinement of the tools and development of new methodologies subsequently used.

The most dominant evolution that took place during the fieldwork resulted from a collision of two ways of thinking. It was observed that all the learners expressed the motions involved in balancing physical objects through discrete sets of states. "If this happens then do that" was a common way of defining what to do. The researcher's way of thinking, on the other hand, treated the attributes more as a continuous system, which, while provided better results, was alienating to the learners' thinking. This conflict led to the design of a new approach that could accommodate the benefits of both ways of thinking. The approach turned out to be similar to an orbital control system used on the space shuttle called a Phase-Plane Controller. Other domain knowledge that were encountered by the learners include state management, the role of rate and speed, stability analysis, prediction models, and multi-variable systems.

The next chapter describes the researcher's motivation and the origin of the research idea. It then describes the various components that are related. Chapter three depicts the idea of a learning Microworld in detail and gives an overview of the Microworld created for this work. It finishes by describing the research methodology employed. Chapter four portrays the events that took place during the case studies. Finally, Chapter five pulls together the important observations and discoveries that took place.

2. Motivation and Background

The inspiration for this work comes from my personal experience with robotics and balance control. But my interest in robotics had taken root long before. I have been working with children using robotics as a learning environment for many years. Given my computer engineering background, my personal work includes both the design and application of robotic tools. The over-arching theme governing the direction of my work has always been about "making connections" with people. For example:

- **Connection with the computational tool**: In the design of my tools, I focused on allowing people to build the robotics tools locally. The emphasis on "making your own tools" has resonated with students and teachers in many places, especially outside of the US where maintenance is more prominent in the culture. This connection through construction of one's own tools has allowed people to feel less alienated by the technology. It often leads to tools that cost less as well [Sipitakiat, 2004].

- **Connection with familiar materials**: My colleague, Paulo Blikstein, has expanded the theme above to include making use of locally available materials. Instead of only buying motors, sensors, and other building blocks, one can also obtain these components and others from existing sources (or found materials) such as broken electronics, obsolete computers, and craft materials. His idea was an extension of previous works that have tried to make use of crafts materials in robotic learning activities (see [Martin, 2000] for an example). The use of found materials allowed for the term "robotics" to become less foreign as one can make connections to everyday objects with which they are already familiar [Blikstein, 2003].

- **Connection with the learning activity**: Based on Paulo Freire's discussion about creating consciousness through engaging with one's environment [Freire, 1970], my advisor, David Cavallo, has put a strong emphasis on making the learning activities connected to a topic that has true meaning to the learner. A "generative theme" has commonly been used. For example, the theme "the city that we want" was used with students in poor neighborhoods in Brazil. The theme encouraged project ideas to emerge from issues that were real in the community [Cavallo, 2004]. Thus, the robotic projects were used as means to think about how to better understand a problem or how to improve the current situation in the community. The projects were not something created for the learners in isolation. Sensing water pollution levels, creating automatic trash separators, and automating an irrigation system are some examples.

When I started working with balancing robots, I felt that our ability to balance physical objects could lead to new ways for making connections between the learner and the learning activity.

2.1. Why Use Balance Control as a Project Theme?

I was initially inspired by Chris Hancock's doctorial thesis where the author briefly describes how he worked with students on a programmable balance beam where a ball was placed on the beam and children had to create a computer program that prevented the ball from falling off [Hancock, 2003]. I then spent time creating a programmable balance beam to try out the activity myself. Despite my engineering background, I did not know much about balance control at the time. Because it was easy to balance the ball when moving the beam by hand, I assumed that creating an autonomous system should not be much harder. But it turned out to be quite difficult. Although I finally made it work, it was only after I had done some research and learned a few basic control principles. I, thus, came to realize how experiential knowledge alone is not enough to develop a deeper understanding of the underlying principles.

Yet, I was intrigued by the fact that the task could be easily performed with our body. With minimal practice, a person can control the ball on the balance beam by hand. To me, this was potentially a new kind of connectedness that could be established between a learner and robotic learning activities. Thus, I decided to develop the following project theme: Engage children in thinking about their own balancing actions while trying to create an autonomous system that performs the same task.

This idea was especially challenging because balance control is typically considered too difficult for children. Nevertheless, it seemed to be a viable domain to investigate how a computational approach could open up new pathways towards learning that is extremely hard otherwise.

2.2. Physio-Syntonicity: Building Understanding from Within

The reason for my passion about the possible connection between body motion and ideas in balance was based on Papert's notion of syntonicity [Papert, 1980]. The most well know use of this term has been in describing a special relationship that takes place when a child programs an on-screen turtle object in the Logo programming language. Programming the turtle to move around the screen allows a child to draw and learn about various geometrical shapes, hence the name "turtle geometry." But programming the turtle is more than just learning about squares, triangles, and circles. Papert has shown that as a child exercises the utility of the turtle, one can identify the movement of this on-screen object with one's own body. The child can imagine how one can move around the room to create, say, a circle in the same way as the on-screen turtle. Papert calls this connectedness "body syntonic."

The key point about syntonicity is that it intentionally builds upon familiar, deeply-connected knowledge of the person and enables a "syntonic" extension into the new world. In the case of turtle geometry, the syntonicity is the anthropomorphisizing of the turtle and using one's knowledge of one's own movement in space to extend onto moving the turtle in the Microworld. The programming language, in this case Logo, provides the formal bridge for mapping the familiar onto the new, enabling learning about geometry, shapes, and properties, by extending upon what one understands. Syntonicity allows the child to become at ease and intellectually comfortable with the operations at hand. Syntonicity contributes significantly to what makes knowledge learnable.

My previous work has focused on other aspects of syntonicity. As mentioned earlier, my colleagues and I have provided ways for learners to relate to the robotic activities, e.g. by building their own electronics, extracting parts from familiar everyday objects, and working on real problems in their lives. Thus, the development of understanding (e.g. learning about mechanical design, ideas in sensing and control) was built on familiar grounds. The new ideas did not come to them from the abstract but rather from recognizable sources in their lives. This connectedness allowed for the learning activity to be personally meaningful (ego syntonic) and has relevance to the students' lives (cultural syntonic).

Given this framework, the idea of learning about balance control from our existing abilities to balance physical objects initially appeared to resonate with Logo's body syntonicity. I became passionate because I believed it could give me an opportunity to make a new contribution to this well-known discussion. But as my initial experiment with the balance beam showed, the resemblance was only skin deep. Unlike the turtle where it is clear to a child how their body movement can be translated into the turtle's terms, there is a large intellectual void between our body actions balancing a physical object and the operations of the computer program that accomplishes the same task. In fact, humans are actually not aware of most of the actions they are doing to accomplish a balancing act.

Despite this disappointment, I have discovered that people typically produce many ideas by observing body motions and then mapping them onto an unknown. An important point to note here is that most of these initial ideas and observations tended to be inaccurate. For example, while balancing a ball on a balance beam, learners often came up with descriptions that contradict their own actions. And they do not seem to notice it! This was rather fascinating. On one hand, this experience validated criticisms of basing understanding of this type of phenomenon on intuition. However, the fact that people were engaged, generated many ideas, and could become involved in constructing models to test their ideas led me to believe that observing body motions could provide a rich playground with ample opportunities for learners to encounter and perhaps overcome the shortcomings of their ideas.

As a result, there are now two aspects of connectedness to emphasize. One is the connection with body motion balancing physical objects. The second lies in the design of an environment that allows one to capture and implement their, often inaccurate, interpretations of the phenomena. If the learning environment is rich enough to support the development of ways of thinking that are more sophisticated and accurate, it could yield a complete picture of a new trajectory for balance control to become a valid and engaging topic for children while at the same time being able to create an appreciation of the underlying principles. The connectedness in this new environment is what I call "physio-syntonic."

2.3. Experiential Knowledge and the Role of Being Wrong in the Learning Process

An important assumption of my emphasis on connecting learning to our experiential understanding of balance is that learners will often be wrong. In the case of this study, the assumption is based on the fact that we often perceive our actions differently from what is actually going on. In many traditional educational settings, this mismatch often leads to an allergic reaction. Students' reasoning is something to be thrown away and replaced with the true and proven laws that the teacher gives them. As a result, any kind of experiential knowledge is treated as detrimental and there is no longer room for alternative reasoning other than the proven and formal ones.

This work suggests a different point of view provided by Piaget. Consider the following example of Piaget's conversation with a child [Papert, 1999].

> **Piaget:** What makes the wind?
>
> > **Julia (age 5):** The trees.
>
> **Piaget:** How do you know?
>
> > **Julia:** I saw them waving their arms.
>
> **Piaget:** How does that make the wind?
>
> > **Julia:** Like this (waving her hand in front of Piaget's face). Only they are bigger. And there are lots of trees.
>
> **Piaget:** What makes the wind on the ocean?
>
> > **Julia:** It blows there from the land. No, it's the waves.

While Julia's answer does not correspond to the kind of explanation an adult would produce, Piaget recognizes that it is not incorrect either. Given the level of understanding and the knowledge the child has of the world, the reasoning behind the answer is extremely creative and coherent. Moreover, the underlying thinking (e.g. causation, coherent model, exemplar-based, magnifying with scale from the wind an arm can generate to the wind millions and millions of trees could generate, etc.) is the type of scientific thinking we want children to develop and continue to enhance. To respond by telling her she is wrong and inject her with the "proven" adult explanation (e.g. air mass and pressure) would most likely fail. She would not be capable of assimilating the information because she does not have the framework. But worst of all, it would be disrespectful to the child's thinking. It would be discouraging for the child to continue the type of thinking we want to encourage and develop.

Piaget has shown that children are not incomplete adults. Their type of thinking is actually coherent for them to function in their world view. Moreover, both scientific progress and the development of human thinking build upon themselves and do not just replace one "incorrect" concept with the "correct" one. The construction of knowledge is a process most often incomplete, inaccurate, and inconsistent. But these are the building blocks we have and use.

Columbus did not have to wait for the invention of modern ideas of triangulation to set sail! Columbus and other ancient sailors have been navigating the oceans using stars and crude measurements for thousands of years. We cannot say that ancient sailors were incomplete sailors. Despite the arguable "inferiority" of their approaches and methods, they did have their own sophisticated thinking about sailing that well supported their needs and the possibilities at the time. In fact, the "inferiorities" were the force that drove the progress of modern navigational techniques. This process reflects Piaget's idea of "optimizing equilibration" in human learning and development [Piaget, 1985]. A learner may devise a method that appears inferior or primitive to a teacher, but to the students their ideas are complete and coherent to their world view. Thus, allowing the learner to carry out their ideas is a needed starting point for the progression of their understanding.

This does not mean, however, that we should leave the learners alone with no form of expert interaction. The knowledge of the teacher can provide many advantages for the learner. The teacher can play a key role in providing guidance through deeds and words. This can be particularly effective when this is a negotiation process that allows the learners to decide for themselves what to make out of the guidance being given to them.

The difference between this work and Julia's situation is that when learners come up with inaccurate ideas, it hits them in the face when the robot fails. The emphasis on construction as means to express the learners' ideas allows them to reflect on the resulting consequences. They may choose to maintain their reasoning and look for mistakes in the implementation. Or they may come to realize the flaws of their ideas and feel the need for other ways of thinking that can better fit what was observed. The learning process then takes place through multiple iterations of this cycle of externalization and re-internalization of ideas [Papert, 1980].

In this work, misconceptions are expected and are treated as a normal part of the learning process.

2.4. Existing Work on Learning from Human's Ability to Balance

There have been examples of using people's experiential knowledge as a basis for the learning and teaching of ideas in balance control. In the 1980s, balancing a bicycle was used as part of the control course at the Mechanical Engineering Department at the University of Illinois [Klein, 1989]. Students were challenged to explain how a bicycle works. Different aspects of the bicycle were analyzed by custom building bicycles with rearranged characteristics. Rear steering and zero gyroscopic bicycles are some examples. Some of these bicycles are extremely difficult to ride and students were challenged to explain the difficulty. This approach was adopted by many other universities and newer iterations incorporated sensing devices that allow for computerized analysis [Astrom, 2005].

Similar to the emphasis in this work, it was reported that reflecting on one's ability to balance on a bicycle was motivating to students as they would often start off thinking that the actions should be easily explained. As they gradually discover the complexities of this problem, they become engaged in exploring the different aspects of balance which allowed them to eventually become more fluent and informed about the principles involved.

The idea of learning from a functional model is nothing new and is in fact a common approach in the development and discovery of ideas in almost all scientific disciplines. Understanding human postural control that maintains balance during perturbation or gait cycle has been a research topic widely studied in the area of kinesiology, biomechanics, and robotics. A typical approach is to put participants through a carefully designed situation that induces body motion and measure postural movements using various electronic sensors such as strain gauges, EMG probes, and vision systems. This methodology defines the study of Computerized Dynamic Posturography (CDP), which was pioneered by Nashner in the 1980s [Nashner, Black et al., 1982]. CDP has led to a knowledge domain influencing works in areas such as patient rehabilitation (e.g. [Wall, Weinberg et al., 2001; Bonato, 2003]), biomemitic robots (e.g. [Hirai, Hirose et al., 1998; Popovic, Englehart et al., 2004]), and biomechanical prosthesis (e.g. [Herr, Wilkenfeld et al., 2002; Herr and Wilkenfeld, 2003]).

The work in CDP has inspired me to explore how recordings of human motions could provide a basis for learning about the underlying principles. This method is explained in detail in the next chapter.

Figure 2-1: An example of CDP data collection device. Human gait control under various perturbations is recorded through electronic sensors for later analysis.

2.5. Balance Control as a Learnable Domain for Children?

The idea of learning concepts in balance control from experiential knowledge would be an overstatement without further elaboration and refinement. To many, I am suggesting the impossible. Balance Control involves many concepts that are part of Control Engineering which is a well-known university-level engineering domain that requires a significant amount of expertise and experience. It is commonly perceived that one needs to take these formal engineering courses if one wants to become serious about control. It also has a reputation for being one of the harder engineering subjects [Bissell, 1999]. Classical control engineering relies heavily on linear algebra, differential equations, and many other formal mathematical representations. Only those few who are able to master these formal representations can become successful in control engineering courses. Thus, learning balance control from this perspective is typically considered far out of reach for most learners, let alone children. It is rare for pre-university students to be introduced to this domain and even when they are, the topic is either highly simplified or presented as an introduction for further studies. See [Kolberg, Reich et al., 2003] and [Miller, 2001] for examples.

In practice, however, there are pathways that people have taken to accomplish a control task without depending on the pre-requisites of control theory courses. Many hobbyists have been able to create autonomous balancing structures using methods such as a Proportional-Integral-Derivative (PID) controller. PID is widely used due to its simplicity and practicality. This is reflected in articles like “PID Without a PhD” that provides implementation steps with quick results [Wescott, 2000]. Making this kind of quick-and-dirty control to work usually involves trial-and-error (e.g. in the tuning process) and other forms of reasoning which sets itself apart from the formal and analytical methods expected from a school-trained control engineer. Thus, the validity and acceptability of this kind of approach towards learning control is often disputed and debated among educators and engineers.

Discussions like the above typically lead to a dichotomy where people become unnecessarily divided in their beliefs about what counts as a valid and acceptable form of learning control. The split becomes exaggerated when people identify themselves on one side of the debate and then create an assumption that their oppositions reject everything and believe purely in the inverse. This leads to the “either-or” phenomena described by Dewey [Dewey, 1963] where people in one or both sides of a debate believe that the outcome has to be “***either*** one ***or*** the other.”

Examples of this phenomenon can be seen in science and education. A relevant example is perhaps the “qualitative vs. quantitative” learning approaches. Although ultimately nobody denies that both qualitative and quantitative ways of thinking plays a role in professional work, the dichotomy has led to situations where the focus is only on one side while eliminating the necessity of the other. For example, the term “qualitative reasoning” often leads to a study of reasoning techniques that does not require quantitative information (e.g. not dealing with any precise numbers).

This work is not situated in this starkly divided perspective. While I focus on using children's natural ability to balance physical objects in the learning process, I do not see it as being part of the "intuitive vs. formal" or "naïve vs. expert" debates. Rather, I focus more on stepping back and thinking about how a computational medium can allow for new and rich learning approaches for people to start developing their understanding about balance control. My goal is not to develop a control engineering curriculum for university students. Rather, it is about how to introduce children to ideas from a domain that has been previously considered off-limits. Thus, it is imperative for this work to have to re-think the assumptions of the current approaches that define what ideas are valid, how they are learned, and in what sequence. In some ways, the only way to become successful is by incorporating methodologies that are new and unconventional. But as this thesis will later show, many ideas that become central to the children's thinking still originate from existing engineering practices. The difference, though, was in the way the computational medium has played a central role in representing the ideas and making them accessible. As the next section will show, the essence of such an approach is in showing that the computational medium can rearrange the intellectual terrain in ways that make learning both personally meaningful (e.g. the ideas can make something work) and useful in other aspects of the learners' lives.

2.6. The Role of Computational Technology

Computational technology has rapidly become an essential part of real-world control applications in the past few decades. In the classical Control Engineering education sector, many courses have demonstrated how concepts can be exercised and learned through active engagement. Sensing and control of physical objects typical play a central role. Students are engaged in balancing inverted pendulums, balance beams, and other setups which allow them to apply the theories taught in class to real situations. Mathematical definitions are expressed in computational forms through computer environments such as MATLAB and packages such as Simulink allow interaction with physical objects. See [Teng, 2000] and [Lieberman, 2004] for examples.

Besides the efforts to incorporate technology into learning, there have also been more fundamental discussions about how technology is being applied in ways that reinforces the same existing abstract mathematical treatment of the field and how it may not reflect the new potentials that technology could offer. Bissell has described the situation as follows:

> "*… numerous textbook authors have recently incorporated MATLAB into their books—but certainly not by rethinking the contents! Rather, MATLAB and similar tools tend to be used to support the traditional teaching by simply automating tried and trusted exercises in partial fraction expansion, Routh array generation, Laplace- or z-transformation and inversion, compensator design, and so on.*"
> [Bissell, 1999]

He goes on to describe how the real challenge for the teacher using MATLAB, Simulink, or other teaching tools is in using the tools in ways sensible to a new and perhaps radically modified curriculum.

Although Bissell's discussion is situated in the context of university-level Control Engineering courses, the argument resonates with the tone of this work. The true value of a computational medium lies not in how it may enhance existing ways of teaching. Rather, it is more in the realization of how the intellectual terrain can be rearranged and in ways that could be more accessible to a broader audience.

In the spirit of Papert's work with computational technology, the implications of the discussion above go far beyond merely rethinking "what" should be taught. It covers "how" learners become in contact with the ideas, and doing so in ways that are relevant to them as well. Thus, the connection between the learner, the activity, and the ideas cannot be discussed in isolation.

Consider the following hypothetical case. It uses an example of how "triangulation" could become a relevant idea to learners through a digital medium.

Imagine yourself visiting a group of students who have been working on various projects using a programming language such as Logo. At the time of your visit, a student was working on a 2D simulation of a rocket launch. The goal is to make sure the rocket follows a straight path perpendicular to the earth. It must correct any diversions that are caused by the environment. Wind was selected as the source of disturbance for the initial program. The rocket can adjust its trajectory by steering its heading.

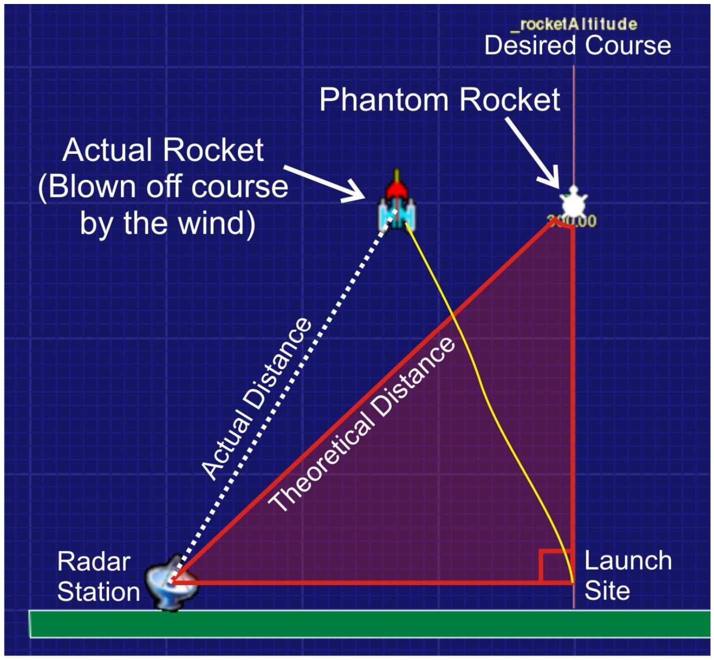


Figure 2-1: A rocket launch simulation. The goal is to guide the rocket to travel on a vertical course perpendicular to the earth. The rocket in the photo has veered off course due to wind blow. A phantom rocket has been created as a reference. This phantom rocket would always stay on course. Its distance from the radar station is the theoretical distance and it is compared against the actual distance between the radar and the rocket. This comparison can help detect when the rocket is off-course and how to correct it.

The student has decided that a reference point was needed, and the teacher suggested placing a radar station on the ground. Because it is a radar station, the distance between the station and the rocket can be obtained. In addition, the rocket has an altimeter. So, the altitude is known. Notice that the configuration of this problem is naturally a right triangle. Given the information that is available, trigonometry becomes naturally relevant in tackling the problem.

Many approaches were tried, but the one eventually selected was to compare the distance between the actual rocket and the radar station to a theoretical distance of a rocket that is on course at the given altitude. Depending on which value is larger, we can determine the direction to which the rocket has veered off. The rocket's heading can then be adjusted accordingly.

The question then was how to figure out this theoretical distance? Pythagoras's formula ($c^2 = a^2 + b^2$) is one possible choice. But because the student has learned from an earlier project that the computer can report the distance between two objects, the following strategy was developed instead. A phantom rocket was created and it would always stay on course. The altitude of this phantom craft is set to be the same as the actual rocket. The distance between this phantom craft and the radar station is then the desired theoretical distance. In retrospect, the strategy used was essentially a simple case of triangulation. Although it is not complete in any expert standard, it was creative and sufficient to solve the problem.

The idea of triangulation was conceptualized in a form that does not resemble any canonical form (e.g. Pythagoras's formula shown above). It was one that is both accessible and useful to solve the problem at hand. Such conceptualization was possible because of the dynamics of the digital media. It gave life to ideas. Triangulation has become more than just something learned and exercised on paper. In Papert's terms, triangulation is a *powerful idea* and the learner is *appropriating* it for personal meaning and use [Papert, 1980].

This example also shows how learning took place within a context that has meaning to the learner. They learned about triangulation because they needed the idea to make something work. And the environment was rich enough for them to pursue the task in ways that made sense to them and that can lead to a functioning system. It was an environment that allowed them to be "wrong" as much as it allowed them to be "right." This is the basis of learning that takes place in a Microworld, which is the topic of the next chapter.

3. Moving Towards a Microworld for Exploring Balance Control of Dynamic Systems

This chapter builds upon Papert's idea of using a Microworld as an incubator of knowledge. It then describes two new methodologies that have been created particularly to support a BCDS Microworld. It later describes the implementation and the research methodology employed in this work.

3.1. Learning with Microworlds

A Microworld is a computationally-rich environment that is designed to highlight important ideas by restricting the world while maintaining the conceptual integrity with regard to the domain under investigation. It makes key activities predominant in that world, providing a class of useful operations that are rich enough for learners to explore the ideas. In the construction of a Microworld, the idea of expressing ideas via programming is essential. The power of a Microworld is in the high degree of expression available to the learner. It is a fertile environment for the development of human understanding about a domain and about exploration of domains.

The learner does not merely give values to pre-set attributes in a domain and observe an interaction. Rather, the learner is compelled to create models attempting to accomplish something of interest. A key underlying principle is: the more open the environment is for the learner to express his or her ideas in the manner the learner thinks, the better the results will be. There is an active dimension where the models can be run according to the laws of the domain and the operation and feedback observed. Then through an iterative process of designing, constructing, running, observing, reflecting, discussing, and debugging, the learner modifies the models and the thinking. The programming aspect of expression enables learners to concretize their ideas. The recursive process of externalizing one's thinking, critical observation, and re-internalization is intrinsic to learning. The power in a Microworld is in its power of expression.

When looking at a Microworld from an epistemological perspective, it is an environment that can open up new ways for learners to experience and think about the phenomenon under investigation. This work has focused on syntonicity which allows the learner to relate the activity to other aspects of their life that are familiar to them. Thus, a Microworld is not only about giving access to new ideas, it is also about creating new relationships between the learner, the domain, and the learning of the learning process.

3.1.1. Examples of Microworlds

A few Microworlds have been developed. Logo is the best known platform where a "turtle geometry" Microworld gives learners access to mathematical ideas, geometry in particular, and, as emphasized in Papert's book Mindstorms, other fundamental concepts that exceed the traditional definition of mathematics [Papert, 1980]. A later variety of Logo, exemplified by StarLogo and NetLogo, created a Microworld where a massive number of objects can execute a set of rules in parallel. Learners experience how many real-life phenomena, such as traffic jams; the spread of forest fire, can be exhibited through a decentralized system [Resnick, 1994; Wilensky, 1999]. Lego/Logo and the programmable brick was an expansion of the on-screen Microworld to the physical world where learners encounter engineering and design ideas through the construction of machines with programmable behaviors [Resnick, 1988; Martin, 1994].

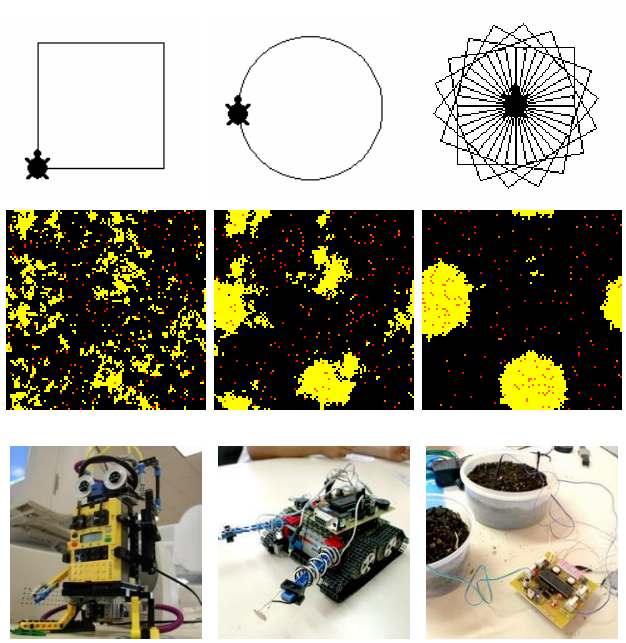


Figure 3-1: Examples of existing Microworlds. Turtle geometry in Logo (top), decentralized systems in StarLogo and NetLogo (middle), and mechanics and robot control that comes with programmable bricks (bottom).

3.1.2. Active Exploration

An easy trivialization can take place when project-oriented learning is transformed into a problem-based instruction where a situation is constructed to teach a particular idea in the curriculum [Hoyles, 1993; Papert, 2002]. For example, consider a lunar-lander game where learners are challenged to control various parameters to prevent a crash. Although this presents a situation that is similar to the topic of this thesis, many simulation games expect learners to learn about gravity and other ideas in physics simply by observation. Learners might be able to adjust some parameters via sliders and knobs, but, as will become evident through the case studies in this thesis, seeing is not always believing! Simulation in a Microworld, on the other hand, allows learners to not only change parameters but to also define them and program their relationships. Learners define the rules of the system based on their existing understanding. Technology helps carryout these ideas and reveal how well they perform. Modified or entirely new strategies are then put in place until a satisfactory behavior is accomplished. Instead of attempting to force a particular idea or a particular worldview on a learner, a Microworld provides a rich enough environment for learners to make their own explorations and develop a deeper understanding of the underlying principles along the way.

Personally meaningful projects have a number of aspects that are different from problem-based approaches. Papert has articulated how projects are potentially richer when carried out over a long time period. Learners should have time to try different approaches, work on sub-problems, establish a common language with collaborators, and make connections to other problems. A structure of how progress is made can emerge. For example, learners can start to recognize activity cycles that consists of designing, experimenting, debugging, and reflecting [Papert, 1971]. These are the qualities that allow the learner to truly internalize their actions in the world.

While Piaget's focus is on how internalization allows for the progressive development of knowledge structures, Papert has added that this process takes place especially well when the learner is engaged in constructing a "public entity" [Harel and Papert, 1991]. As described by Harel and Papert, this public entity can be an article that can be shared among others. It can be tangible (e.g. a sand castle, a robot) or not (e.g. a poem, a computer program). Essentially, while Piaget is more focused on the process of internalization and abstraction of ideas, Papert adds a focus on externalization through "individual's conversation with their own representations, artifacts, or objects-to-think with" [Ackermann, 2004]. This externalization process allows learners to transform their internal world view into concrete forms, and expose them to the world to be manipulated and shared. When the world reacts back in surprising ways (e.g. something doesn't work, a teacher or friend gives an unforeseen insight), Piaget's negotiation and internalization process kicks in. But for any idea to "hold in one's mind," it is best if we can make it tangible or externalize it. And when we do this, by the same token, we sharpen and shape inner feelings and ideas within the constraints proper to the given media, or tools. Thus, the learning process becomes a cycle with an emphasis on both externalization of ideas and re-internalization of actions.

3.1.3. Idea Power

Papert's work on learning has always stressed the role of digital technology to facilitate and amplify the learning process. It is important to note that he has also put a special emphasis on a special kind of idea. It is the kind that is not only useful for a particular problem the learner is working on but that can also be generalized and applied to a large class of phenomena. It is an idea that one can apply in multiple situations, and it is also an idea that can resonate and take root in the person. This is the notion of "idea power" which can be amplified through computational technology [Papert, 2000]. An example of idea power that Papert provides is how: "simple local rules can create a much larger global effect." In the Logo programming language, a child commonly encounters this idea through programming turtles (a common object in Logo) to draw geometrical shapes. For example, a circle can be drawn by telling the turtle to simply repeat "move forward a little and turn a little." The commands themselves do not hint any conventional property of a circle, but the result is spectacular.

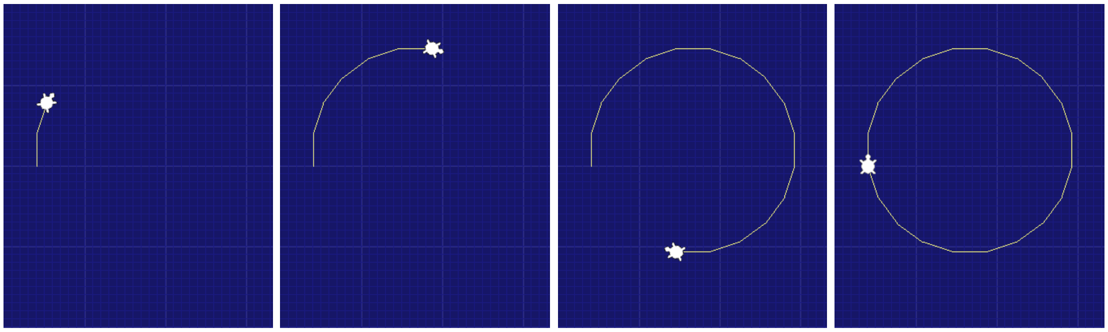


Figure 3-1: A circle is drawn in Logo. It exemplifies the power of how "simple local rules can create a much larger global effect." The turtle is told to "move forward a little and turn a little." The circular shape gradually emerges after the instruction is repeated over and over.

The idea is not taught to the learner in the abstract. Rather it is a concept that is used to accomplish something. The leaner feels its immediate power though the concrete utility. But the idea is "powerful" in the fact that it has an application in many situations. For example, it can be used to make a robotic vehicle move towards the direction with the brightest light (e.g. to make it follow a flashlight). Assuming that we have two light sensors on the vehicle, one on each side, a simple instruction to give is to repeat "turn a little towards the brighter side and move forward a little"

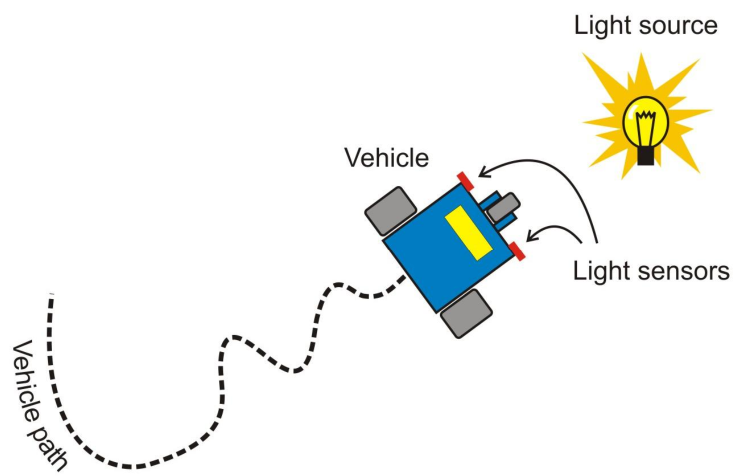


Figure 3-2: Another situation where thinking locally can yield a global behavior. The vehicle appears to move towards the light. This behavior is exhibited by telling the robot to repeat "turn a little towards the brighter side and move forward a little."

As Papert has pointed out, this idea can also be referred to as "differential calculus." But instead of being presented in a form like "D of x to the n is n times x to the n minus 1," it has been conceptualized in a digital form that is both accessible and useful to the learner [Papert, 2002].

3.2. Microworlds for BCDS?

The BCDS Microworld presented in this work is based on the Logo programming environment. It adds the following components on top of Logo to provide better support for learners to start working with the basic operations in control.

3.2.1. Observing Body and Robot Motions

One of the main reasons I first became interested in balance control is that humans can perform amazing balancing actions while having very little conscious understanding of what we are doing. People can walk, run, and bike without falling flat to the ground. Some can balance a ball on their heads. Others are good at maneuvering cars through tricky terrains. However, despite our impressive abilities, people are not usually able to give a good description of how they are actually performing those tasks. In fact, as shown in this work and many others, people's formulations of their actions are often different from what is actually going on.

This work sees this mismatch as a rich arena that, if well organized, can facilitate the progressive development of learners understanding of the observed phenomena. A negotiation of "what is happening" and "what the learner thinks is happening" is an essential ingredient of a constructivist learning environment. Moreover, observing body motions can provide a good basis for this process.

3.2.1.1. Designed Activities

In this work I have applied the methodology from Computerized Dynamic Posturography (CDP) to record human motions. However, there are three differences. First, because human's body motions are extremely complex, the use of CDP methodologies in this work has been simplified by focusing on motion of mechanisms that can be controlled by the body and easily sensed. Second, the general tone of CDP is to gather data about body movement so that experts can analyze and make conclusions. CDP in my research is more for the learners themselves to interpret and make meaning out of the recorded information. Finally, the recordings have not been limited to human motions. The process was used to record motions of autonomous robots as well.

I have developed two initial activities for the learners. A special characteristic of both activities is that the challenges can be performed either by using one's hand to physically manipulate the device or by creating a computer program to automatically perform the task using sensors and motors. These actions can be recorded on video and onto the computer via sensors. Learners can use this recorded data to observe a working model and reflect upon how an autonomous system can be created.

The Inverted Pendulum (The Cart-Pole Challenge)

The setup is an inverted pendulum mounted onto a robotic vehicle (Figure 3-3). The challenge is to prevent the pendulum from falling.

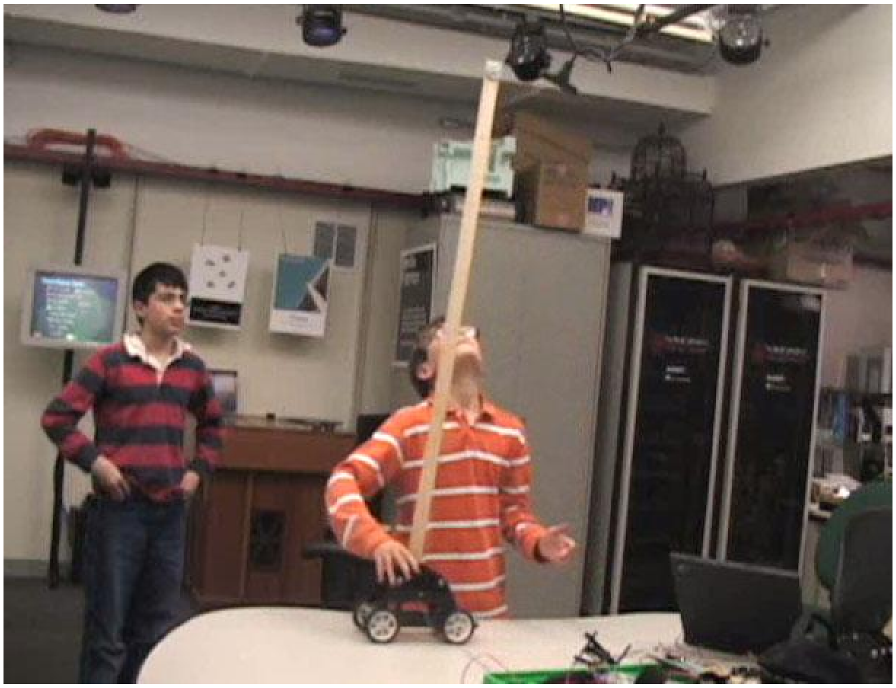


Figure 3-3: Students playing with the inverted pendulum. The pendulum is connected to a car at the base. This restricts (and simplifies) the controlling gesture to a one-dimensional motion on a linear path.

The Balance Beam

The goal of this task is to balance a robotic vehicle that runs back and forth on a beam (Figure 3-4). When the car moves away from the center, the learner needs to prevent the vehicle from falling off of the beam.

Figure 3-4: The balance beam as it is being manipulated by a student. The goal of this challenge is to prevent the rolling car from falling off the beam.

3.2.2. The Spatial Computing Paradigm: Computation via Familiar Properties of 2D Objects

Although many computer environments for control applications already exist (e.g. Matlab/Simulink, Labview), they are unfit for this work mainly due to the built-in assumptions about control operations that are based on the canonical formal approaches. Despite being powerful within its paradigm, the required understandings are represented in forms that are far removed from a novice's experience (e.g. putting a signal through a low-pass filter, adjusting the gain, frequency domain analysis, etc.). Simply put, they were designed for experts to carry out tasks; not for learners to develop initial understanding of the domain principles and concepts.

Spatial computing is a form of computation designed to connect various operations to familiar properties of 2D physical objects. Some of the operations include value manipulation (e.g. using a set of on-screen linear sliders for amplification, range conversion, range limit, non-liner transformation), state definition (e.g. using rectangular areas in a 1/2D graph to define states), and data tracking (e.g. tracking object motion, creating graphs).

SCP is syntonic by allowing learners to relate common control operations to familiar properties of 2D objects. SCP has supported many operations that would have taken a lot of time and effort to become familiar through the formal approach. Learners were able to carry them out comfortably. As Papert puts it, “the prerequisites are rooted in personal knowledge” [Papert, 1980].

The appendix provides a complete overview of the operations supported by SCP in this work.

3.3. Tools

This research utilizes a combination of custom-made hardware and software systems. The hardware primarily refers to the physical structure that allows learners to control the motion via direct manipulation with their hands and via computer programs. This presents a two-way operation that requires careful design of the mechanical and electrical components.

The software side consists of a programming environment that could handle the sensor inputs from the physical structure and calculate the desired motor actions. It was designed based on the Logo programming language. But in addition to the basic Logo language, it adds the ability to make use of spatial properties of objects to support the kinds of signal manipulation required in this work.

Data recording of learners’ actions were done using a traditional video camera and sensor information that allowed learners to see on-screen objects mirroring the actions of the physical structure.

3.3.1. Construction of the Physical Structures

The Inverted Pendulum

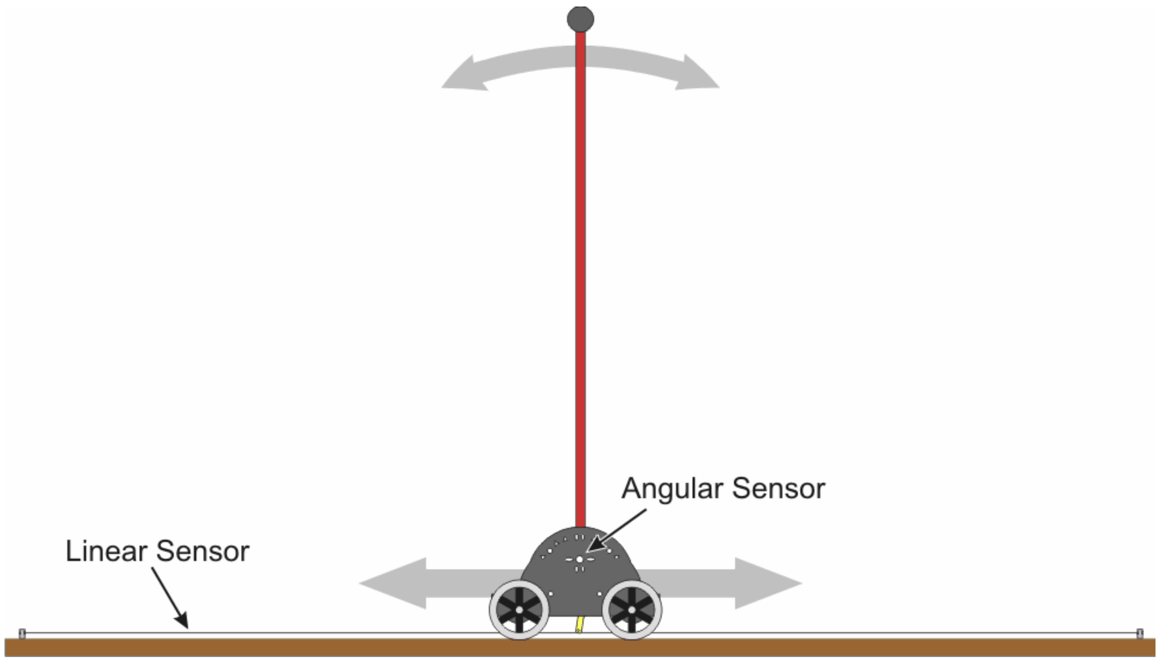


Figure 3-5: Construction of the inverted pendulum task

The pendulum is four feet tall with weight added to the top tip. Two cars were created, one for the autonomous mode and the other for manual control. The reason for the second car was that the motorized car had too much resistance from the motors when physically moved by hand. Detaching the motors was also a tedious process. Both cars were made of acrylic and reinforced with metal rods. Two 7.2V gear-head DC motors were used in the motorized car. An angular sensor was mounted on the pendulum's rotating axel to measure the pendulum's motion.

A wooden track was built to measure the position of the car. Although this is not usually needed in typical inverted pendulum challenges, many operations designed and implemented in this work required this information. The sensor utilizes a simple resistive technique where two pieces of Nichrome wire were strung across the length of the beam in parallel leaving a small gap between them. The Nichrome wire used had a resistance of about 5 Ohms per foot. A copper stub was installed underneath the vehicle creating a contact point between the two wires at the position of the vehicle. The resistance of the wire was then measured giving a value proportional to the location of the vehicle. Conductive grease was applied to the wire and copper contact to reduce spikes in the sensor readings which was caused by momentary separation of the wires and the copper contact. This separation is inherent when the vehicle moves. The construction of the linear sensor is shown in the figure below.

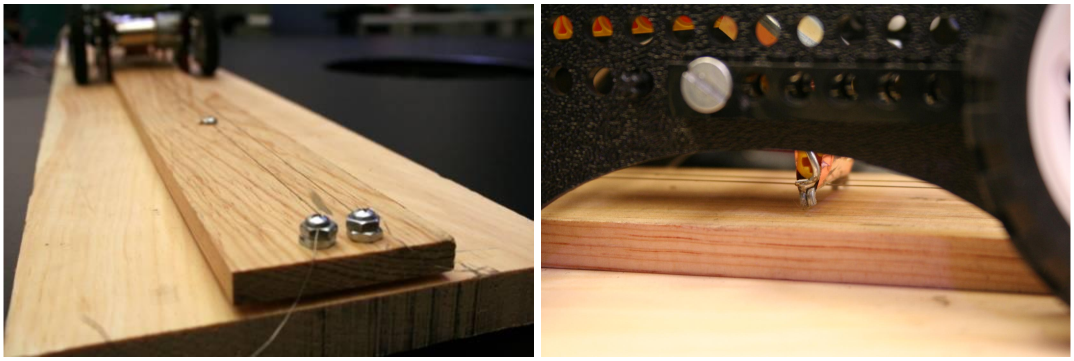


Figure 3-6: The position of the pendulum car was sensed using a pair of Nichrome wires, which was strung across the length of the wood platform (left). The car has a small conductive stub (right) connecting the wire at the car's current position.

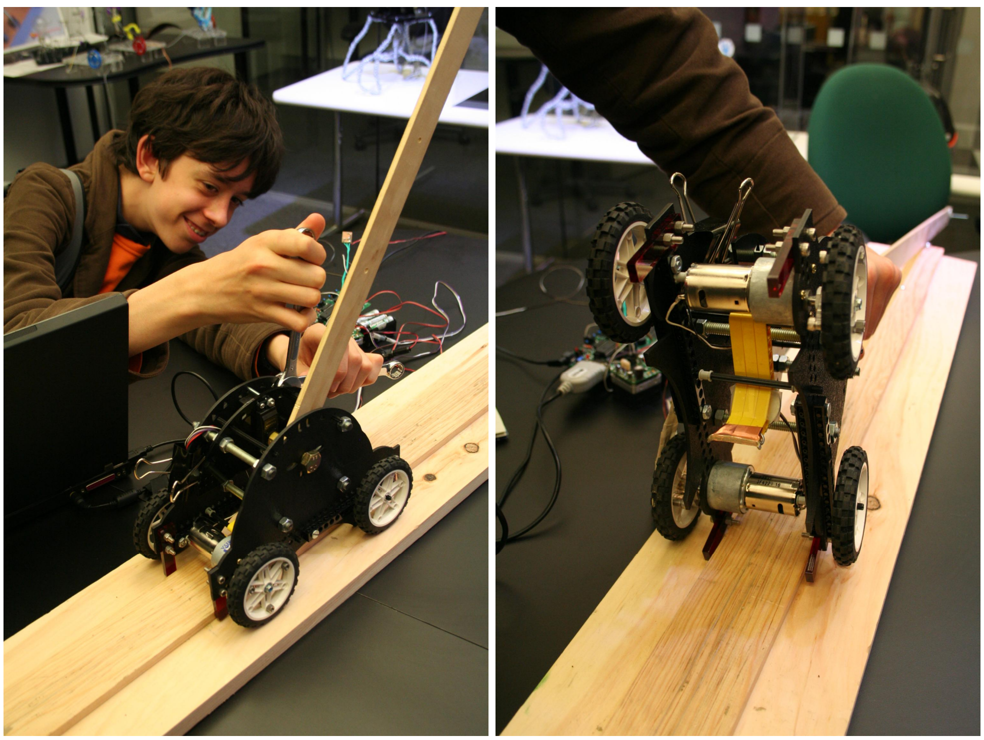


Figure 3-7: The pendulum car shown from the top (left) and from underneath (right)

The Balance Beam

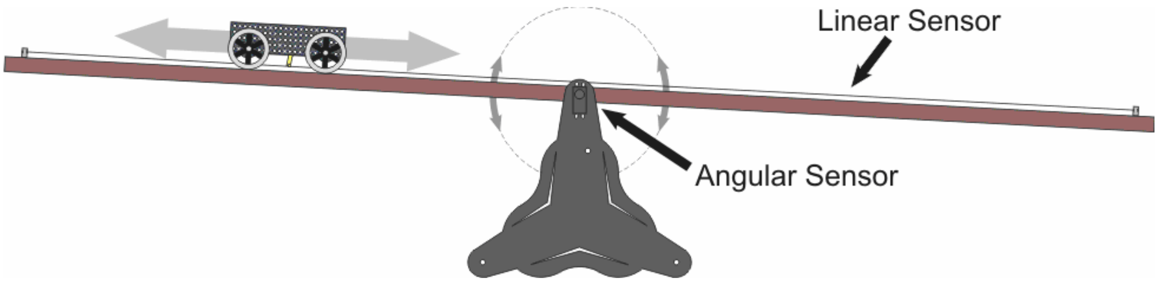


Figure 3-8: Construction of the balance beam task

The balance beam was constructed with acrylic and wood. An angular sensor (a potentiometer from a modified servo motor) was placed at the rotational axis to measure the beam's orientation. A linear sensor was mounted on the beam to determine the vehicle's position (and speed). It was built in the same way as the inverted pendulum activity using Nichrome wires and a copper contact on the car.

The vehicle was constructed mainly with Lego[1]. The copper stub that was added to make contact with the linear sensor added extra friction to the vehicle which made the motion less predictable in the early design versions. Larger wheels and added weight were satisfactory solutions for the vehicle design. However, the added weight required a stronger motor for the beam as well. A 12V DC gear head motor has proven to provide the required torque.

Because the beam was large, it was easy enough for the learners to physically move the beam even when the motor was attached. Thus, nothing had to be re-configured when the learners wanted to physically manipulate the beam. The motor's gear head was constructed with metal, which was strong enough to sustain the reverse torque.

[1] http://www.lego.com

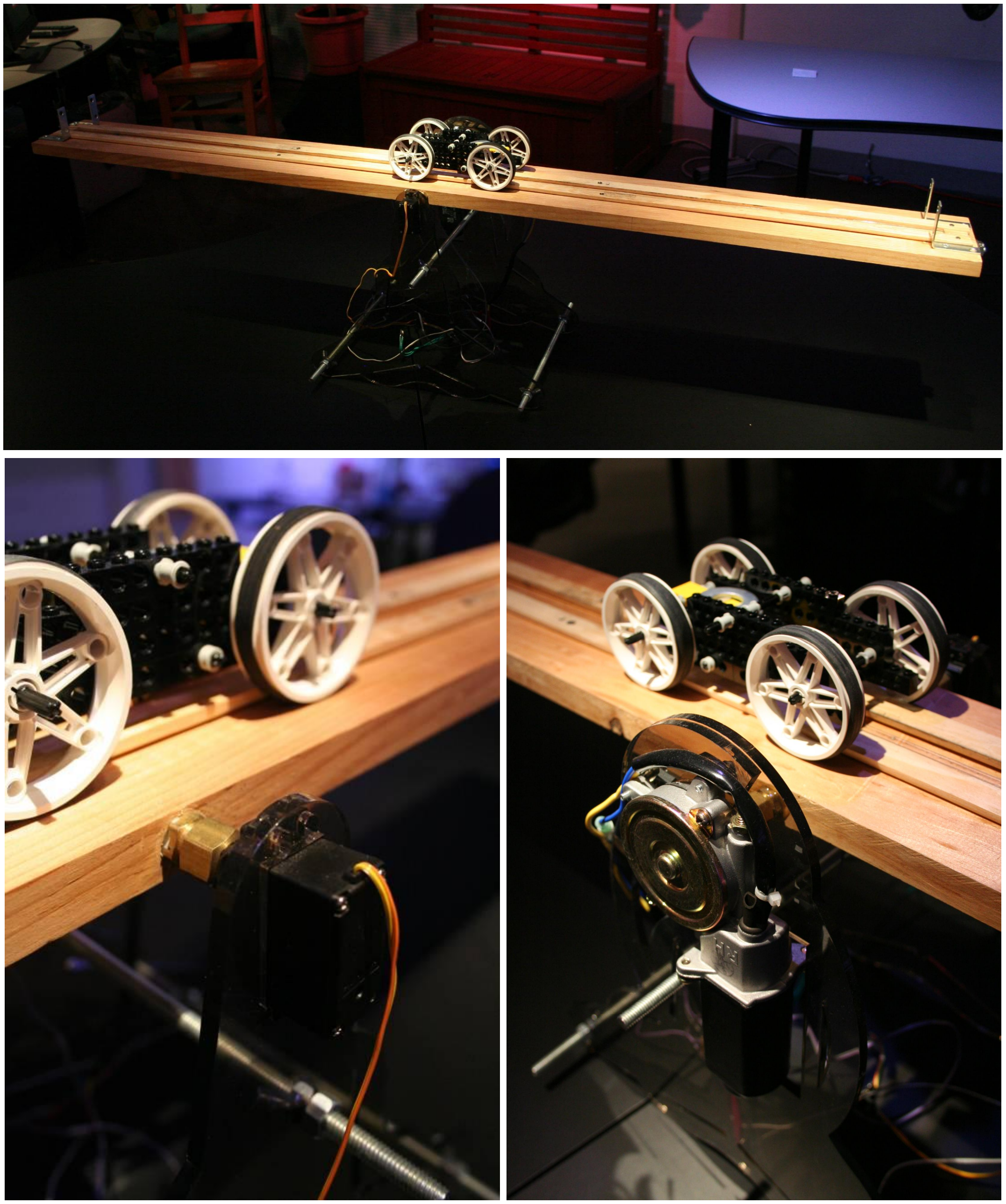


Figure 3-9: Photo of the balance beam (top), a servo motor hacked to function as an angular sensor (lower left), and the driving 12V DC motor (lower right)

3.3.2. Sensing and Control Electronics

The sensing and control hardware utilized in this work is based on the GoGo Board platform [Sipitakiat, 2004]. With a slightly modified firmware, the GoGo Board provides eight sensor ports sampling at 60Hz per channel. The data was transmitted to a computer via a serial port at 115.2Kbps. Since all of the sensors used in this work are resistive, a standard voltage divider circuit was used for the sensor interface. Over time, a passive low-pass filter was also added to reduce signal noise. Since learners often had to calculate rate-of-change of the sensor data, noise in the input signal had to be minimized to yield an accurate differential.

Motors used in this work are also controlled through the GoGo Board. The control signal from the GoGo Board is sent to a HB-25 heavy-duty motor controller from Parallax[2], which then sends out the controlling voltage to the motor.

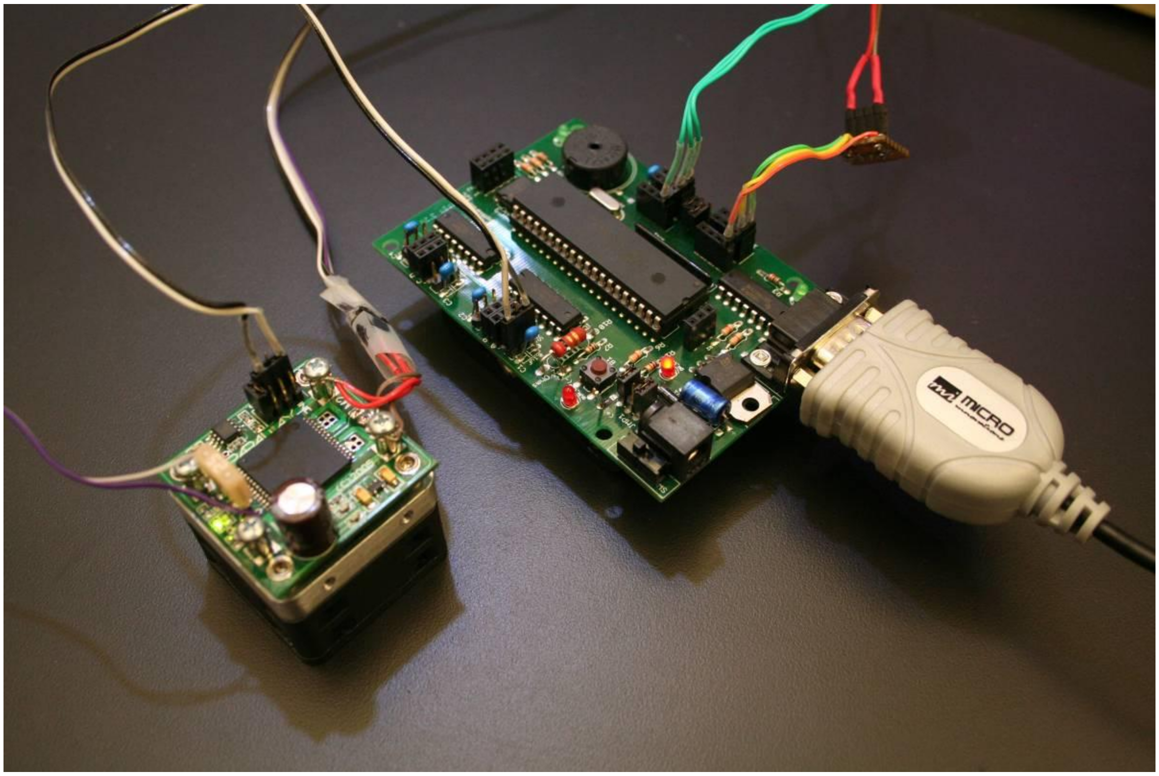


Figure 3-10: The GoGo Board (right) was used to connect the sensors to the computer. It also controlled the motor driver unit (left), which took care of the torque and spin direction of the motor.

Since the entire processing takes place on the computer, handling latency is important. The average latency of the current system is between 20-30ms, which is not optimal but has proven to be acceptable. A somewhat more serious problem is the timing irregularity of the computer software. When many things are happening on the screen, the messaging system often gets queued up and the processing time becomes irregular. This problem has been minimized by isolating the time-consuming screen update procedures to a separate low-priority thread. The user code thread was able to produce an acceptable regular rate down to 15ms. The down side of this approach is that it creates a longer delay between the physical and on-screen events. Thus, the program appears on the screen to be running slower than it really is.

---

[2] See http://www.parallax.com

3.3.3. The PyoLogo Programming Environment

In this work, a special version of Logo called PyoLogo (Python-Open-Logo) was created. Logo was picked as the desired programming language because of its well-known learning philosophy underlying the language design. The primitives, syntax, and objects are designed with human comprehension in mind. When reading a Logo program, the commands often resemble human statements, which is beneficial especially when the activity focuses on expressing ideas through a computer program more than learning how to program in and of itself. This is the idea of syntonicity that Seymour Papert has been discussing since the early days of using programming as a learning tool for children [Papert, 1980].

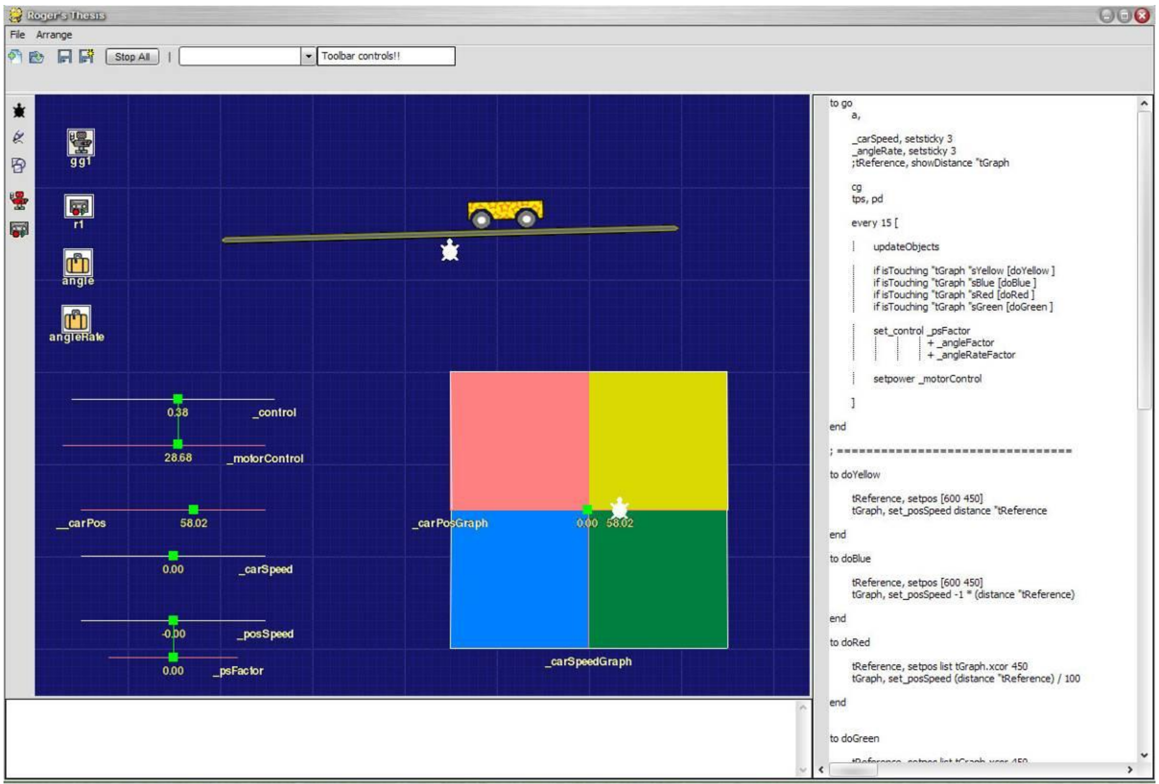


Figure 3-11: A screen shot of the PyoLogo programming environment

Although Logo was used as the basic programming language, many new components were also added to support the activities in this thesis. The main addition is the “Spatial Computing Paradigm” where physical properties of on-screen objects can be used as part of a calculation. This component has proven to be useful in many operations performed in this thesis. A detailed description of this component and its use are described in greater detail in the appendix.

A second main addition is the ability to record sensor information and play it back at a later time. This feature allows learners to make recordings of their actions manipulating the physical structures and play it back later for further review. It allows playback at slower or faster speeds. Processing of the recorded data (e.g. plotting a graph) is also possible. Recordings can also be made of the execution of learners programs. This provides a way to assess and quantify the performance of their code and use this information to debug or further improve their strategies.

PyoLogo was implemented entirely in Python. Python is a programming language that gives a high-level interface to various modules that allows for a rapidly developed and high-quality program. This was an important feature for this thesis. I had anticipated that the programming environment would have to be changed and refined as my work with the students progressed. Being able to do so in a timely manner was crucial. Although rapid development often comes at a cost in terms of performance, Python's seamless integration with C libraries allowed time sensitive routines to be optimized. Most of the mathematical routines used in the program are optimized C libraries accessed via a Python wrapper[3]. The end result is also cross-platform. The Logo environment will be open to the public so that it could be further developed and perhaps become a general purpose, open, and free version of Logo.

3.4. Research Methodology

This thesis is a design research. It is not about testing a finalized pre-constructed learning environment with children. Quite the contrary, I started my field work with a rough implementation of the tools. The ways the activities were to proceed were not fixed. It was the experience of working with students that allowed me to further advance my thinking and refine the tools to better support the activities. Thus, the research methodology needs to reflect this design nature where innovations come not only from the researcher but also from the learner and the interactions between the two.

3.4.1. Focus on Construction and Reflection

The students are engaged in creating operational systems that can control robots to perform balancing actions. Although learners spend most of their time constructing computer programs and testing them, a special emphasis was also put on engaging the learners in reflecting on their body motions and generating ideas that would later be used as part of their model. Learning then takes place through multiple iterations of the following cycle:

[3] See the NumArray module at http://www.stsci.edu/resources/software_hardware/numarray

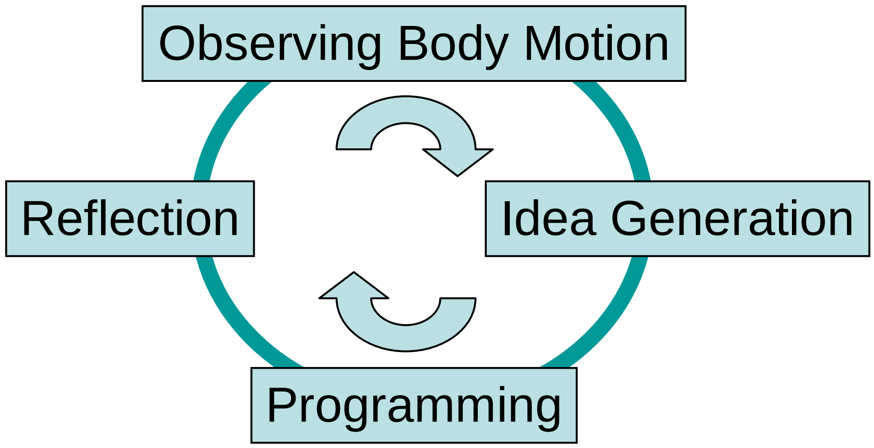


Figure 3-12: This diagram depicts the iterative cycle that governs the learning process throughout the research fieldwork.

Observing Body Motion: Learners perform balancing actions by controlling the robot by hand. That is, they would balance an inverted pendulum by pushing the base back and forth. This action can be recorded on video and on the computer through sensors. The learners can playback the recordings and reflect on the strategies they believe was being used. This information can also be used to prove a particular assumption made by the learner (e.g. is a lighter pendulum is easier?).

The recording can also be performed on the automated robots as well. This information could be used to aid in the debugging process.

Idea Generation: The learners could generate ideas and strategies to balance the robot from discussions and reflections of their body motions. The researcher plays a significant role in helping the learners establish their thinking and translate them into forms that can be experimented and programmed.

Programming: Learners externalize their ideas and strategies through programming in Logo and through the Spatial Computing Paradigm.

Reflection: When problems occur in their program or strategy, learners are encouraged to explain what they believe are the root causes. The researcher identifies these explanations in ways that facilitates further investigation. These ideas are then connected back to the body motions and this cycle continues.

3.4.2. Emergent Design

Much of the process in emergent design resembles the well known participatory design approach [Schuler, 1993] where the researcher's inquiries with students were highly contextualized and the students (or users) played a significant role in determining what aspects of the tools worked and what did not. But it also puts an emphasis on applied epistemological anthropology [Cavallo, 2000] as a means to identify learner's common ways of thinking about the phenomenon. It then directs the design of the environment to make use of these ways of thinking in the learning process.

Since learning is not merely a matter of the linear accumulation of facts, Applied Epistemological Anthropology intends to unearth (anthropological investigation) how learners are making meaning (epistemology) in order to better design a learning environment for their development. Emergent Design builds upon this as the process is fraught with uncertainty as one cannot know beforehand how learners will construct meaning, what mistakes they will make and what serendipitous events will occur to take advantage of. This will emerge with engagement and so the design process iteratively builds upon this. An advantage to a constructionist approach is that among the artifacts the learners create, their expectations for the artifacts' behaviors, and how they talk about their underlying ideas, the designer has a much more explicit window upon which to base epistemological judgments [Cavallo, 2000].

3.4.3. Data Collection

There are three types of data that have been collected throughout the duration of the fieldwork. First, I collected recordings of sensor information. This includes learners' code that is produced during the course of the project. Video recordings of students working on projects have also been collected. This has provided me with the raw data of the whole developmental process.

Second, I wrote my observations of case studies that were significant to the research goals. These case studies have served as a researcher's insight of what was going on during the course of the projects.

Lastly, I conducted interviews with the students involved in the fieldwork. The interview questions were not strictly structured. The goal was to provide an opportunity for the interviewees to become engaged in a conversational two-way communication. The topics were initiated from what is most relevant to the interviewees (such as their engagement, use of the tools in their projects). As a conversation was established, more specific questions were then asked. Although I have prepared a set of common questions, the actual questions asked were not necessarily prepared in advanced.

3.4.4. Analysis

I employed an iterative process to supplement the emergent design framework. I started the analysis process during the time of the experiment and treated it as a continuous procedure. This approach allowed me to iteratively collect more information and direct my attention on any specific aspects to conform, clarity, and elaborate the emerging thesis.

Iterative process is often used in qualitative studies [Miles and Huberman, 1994]. It involved:

- Transcribing the data.
- Building a thematic framework.
- Applying the thematic framework to the transcribed data.
- Charting the data according to the thematic framework. This allows the researcher to easily see the overall picture of the data and their relations.
- Identifying patterns in the data.
- Interpreting the data. This step leads to further questions and directs the development of the tools and methodology that were applied in the next data collection iteration.

The data analysis and methodology refinement took place in parallel with the learning cycle. As illustrated in the figure below, the learning activity is influenced not only by the learner's reflection on the current situation but also by this on-going development of the researcher's analysis as well.

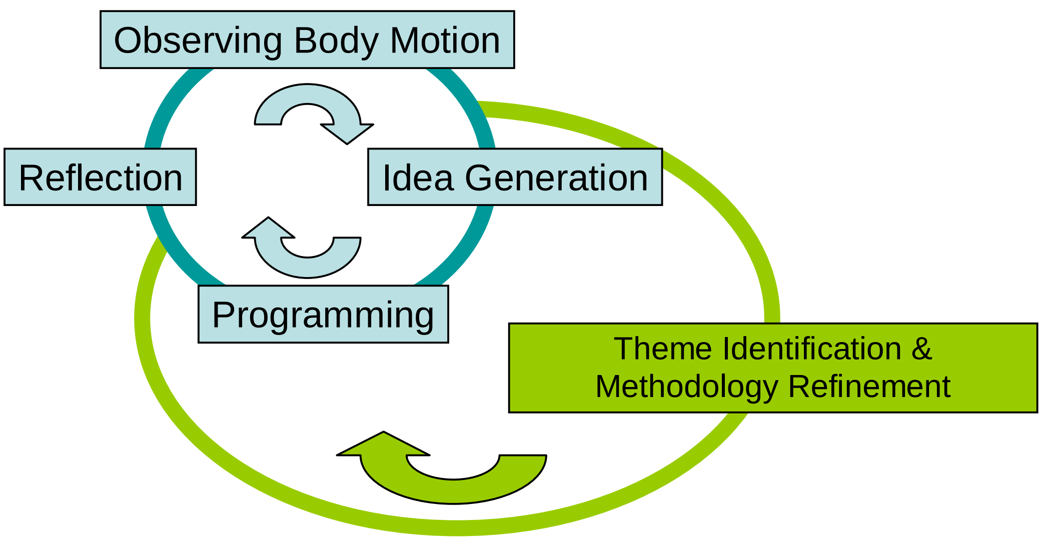


Figure 3-13: The diagram illustrates how the research theme and methodology constantly influences the learning cycle.

The thematic framework used for this analysis includes:

- **Students' explanation of their creation**. Record the students' perspective on the goals, approach used, the issues that emerge, and the language used to explain a situation or an idea.
- **Development of sophistication**. Compare the initial approach and its evolution as problems and new ideas are perceived.
- **Connection to learner's existing knowledge of the physical world**. Investigate the origin of strategies and their connection to learners existing knowledge. Observe how learners transform the purposed strategy into a computational form.

- **Connection to new ideas**. Identify cases where there are opportunities to introduce new ideas that could facilitate the existing process or help move the ideas forward. Observe learners’ reaction to the new ideas.
- **Role of the digital form**. Observe the significance of the provided environment. Identify aspects that are supportive and those that are less significant.

4. Case Studies

Each of the two case studies has been divided into phases, corresponding to different micro-developmental stages in the researcher's thinking and the students' approach to the problem. Each group of students is described separately, in chronological order. As a way to preserve the micro-genesis of both students and my thinking, each session will be presented in the form of a conversation among us—as well as with—and through—the artifact(s) (physical and programming components).

4.1. Participants

Two groups of students were involved in this research. The first group consisted of students I will refer to as Albert, who is 12 years old, and Anderson, who is 13 years old. Both students participated in eight sessions throughout a seven week period. The second group consisted of students I will refer to as Greg, who is 15 years old, and Joe, who is 14 years old. Joe participated for the first six weeks (seven sessions) before he had to move to another country. Greg was the longest participant with 24 sessions throughout a twelve-month period.

<table>
<tr><th colspan="2" rowspan="2"></th><th colspan="2">Group 1</th><th colspan="2">Group 2</th></tr>
<tr><th>Albert</th><th>Anderson</th><th>Greg</th><th>Joe</th></tr>
<tr><td rowspan="9">2006</td><td>April</td><td></td><td></td><td colspan="2">1</td></tr>
<tr><td>May</td><td colspan="2">6</td><td colspan="2">6</td></tr>
<tr><td>June</td><td colspan="2">3</td><td>4</td><td></td></tr>
<tr><td>July</td><td></td><td></td><td>3</td><td></td></tr>
<tr><td>August</td><td></td><td></td><td></td><td></td></tr>
<tr><td>September</td><td></td><td></td><td>1</td><td></td></tr>
<tr><td>October</td><td></td><td></td><td>1</td><td></td></tr>
<tr><td>November</td><td></td><td></td><td>1</td><td></td></tr>
<tr><td>December</td><td></td><td></td><td>1</td><td></td></tr>
<tr><td rowspan="3">2007</td><td>January</td><td></td><td></td><td>2</td><td></td></tr>
<tr><td>February</td><td></td><td></td><td>2</td><td></td></tr>
<tr><td>March</td><td></td><td></td><td>2</td><td></td></tr>
</table>

Figure 4-1: A record of the number of sessions each student participated over the course of the thesis fieldwork. Each session is between 1.5 to 2 hours long.

All the students were recruited from a local school in Cambridge. Posters were distributed to the school and interested students would signup for the sessions. The students would commit for at least six sessions. But most of them continued for much longer.

Four out of the five students had not had prior programming experience. However, all of them use a computer on a daily basis for general purposes such as e-mail, instant messaging, web browsing, gaming, etc. In addition, they all have computer classes at school, which focuses on basic operations such as typing and word processing. Therefore, all the students have good facility operating a computer. Greg was the only student with prior Logo programming experience so he was very comfortable getting started with the Logo programming environment used in this work. However, all the other students picked up programming very quickly. They learned the basic programming primitives such as control flow and procedures within the first two or three sessions. In general, I never had trouble with the mechanics (or syntax) of programming with any of the students.

4.2. Group 1: Albert and Anderson

4.2.1. Phase I: Early Interactions and Strategies

This phase describes my first interaction with the students. As they begin to engage in the problem, I noticed that the students tended to think discretely about the motions involved in balancing the objects. “If the pendulum falls to the left then move the car to the left to correct it” is a simple example. How much the pendulum is falling or how fast the car should move did not seem to matter. My immediate response at the time was that these discrete rule-based descriptions of motion will not be sufficient to produce a working system. Driven by my prior exposure to conventional control systems, I believed the balancing motion needs to be a system of continuously changing analog variables summed together to produce an output.

This section describes the collision between two vastly different approaches in thinking about control systems. On one hand, the students were following their intuitions by creating discrete rules describing the robot. On the other hand, I was driven by a more conventional approach within the discipline, and was spending a great deal of time trying to “convert” the students into adopting my approach. After some time and difficulty, I started to wonder whether this was an unnecessary dichotomy and thought that perhaps allowing both ways of thinking to intertwine could yield a system that would both function and be more comprehensible to the students.

4.2.1.1. Student's First Thoughts on the Inverted Pendulum

In the first session with Albert and Anderson, I introduced them to the inverted pendulum challenge. We spent some time trying to balance an inverted pendulum placed on their hands. One of the goals was to let the students start to think about the balancing action they are doing and to come up with initial strategies for the program.

Another goal was to establish an approach for creating ideas and proving them. I asked both students to think of a beam that would be the easiest to balance. The two main variables considered were the length of the pendulum and the position of a weight on the pendulum. When first presented with the questions, both students would come up with statements that were often wrong. For example, Albert initially believed that a shorter pendulum is easier to balance than a longer one. This misconception was quickly fixed when Anderson challenged him to balance a pencil on his hand (which is very difficult).

A less obvious question was to know where to locate the weight. Both students agreed that adding weight to the pendulum makes it more stable. But they were not sure whether it was better for the weight to be on the top or at the bottom of the pendulum. They made some guesses but agreed that the best thing to do was to try it out. After some test with weights placed at different positions on the pendulum, both students concluded that the weight is best placed at the bottom, which is in contrast with the generally accepted knowledge in the control field (top is better). At this point, their reasoning was based mainly on how it feels when balancing the pendulum with their hands; hence the inaccurate assessment. Despite the misleading conclusion, I was still satisfied with the experiment as it still led to some new realizations for the students and it got them engaged. But most importantly we consolidated the development of a methodology of verifying a hypothesis by creating a proof.

Later (in section 4.2.1.8) they would reexamine their hypothesis using more precise measurements via sensors and come up with a more accurate conclusion.

4.2.1.2. Albert and Anderson's First Program: Discrete Rules that Do Not Work

After they were familiar with the challenge, we started to work on the computer. I introduced them to the programming language and showed them how to sense and control the pendulum car. After some demonstrations, I engaged them to come up with a strategy to automatically balance the pendulum.

Now that they are confined to the information that was available (pendulum's angle, car's position) and the possible reactions (the car's speed and direction), the students came up with a set of discrete rules, which were carried over to the program they created. For example, the first system that Albert and Anderson came up with for the inverted pendulum consisted of the following two rules:

```
If beam_falls_to_the_left Then move_the_car_to_the_left
If beam_falls_to_the_right Then move_the_car_to_the_right
```

In fact, this idea is, more or less, what all of the students I have worked with came up as their initial strategy. In the Logo programming language, the idea could be implemented like this:

```
If angle < 245 [ setpower 8]
If angle > 245 [ setpower -8]
```

The number 245 in the program is a sensor value that was measured when the pendulum is pointing up right. The “setpower” command determines the speed and direction of the car. Eight is full speed one way and minus eight is full speed the other way.

When tested on the pendulum, this program makes the pendulum car move rather erratically as the slightest movement of the beam would cause the car to rush out in that direction.

4.2.1.3. Albert and Anderson’s Second Attempt: Uncovering the Limitations of Discrete Rules

When Albert and Anderson saw the results, their first impression was that the car is too “bouncy.” After some tests and reflections, they concluded without my intervention that the motor power should be more adaptive. That is, the farther the beam falls the higher the motor power. With this idea in mind, they divided the angle into a set of ranges and associated different motor power to each. So, the program looked like this:

```
If angle < 245 and angle > 225 [ setpower 6]
If angle < 225 and angle > 205 [ setpower 4]
If angle < 205 and angle > 185 [ setpower 2]
…
```

This time the car movement was smoother but it did not yield a better result. The car became less responsive and quickly failed. Albert and Anderson now think that the car was too slow. This led to many revisions of the angle ranges and the power level associated to them. During this tuning process, it became clear to me that the students were struggling with two things:

Problem I: Tuning the program was time consuming.

Changing the values in the rules (e.g. the sensor range and the associated motor power) was a tedious process. A lot of time was spent on just getting the right sensor values into the program. It was such a detour that the students sometimes lost sight of the big picture and the ideas they were trying to test. It also discouraged them from revising their program. After two to three iterations, the students started to need motivation from the researcher to continue.

Problem II: The system was not observable.

It was difficult for the students to identify which rule is being fired while observing the results. For example, they could not determine which rule was being executed the most or when the power level changed from one level to another. Without this information, it was difficult to identify problems and debug them.

4.2.1.4. "Discrete" vs. "Continuous" Approaches

At this point, I decided to intervene and offer a different approach that I thought would help alleviate the above shortcomings. This alternate approach was based on a framework I had developed prior to my work with Albert and Anderson. I will refer to it as the "continuous" approach. The name signifies its opposition to the students' rule-based "discrete" approach.

I created a quick demonstration of my continuous method that could perform the same task. As shown in the figure below, the angle sensor value is mapped onto a line object called "angle." (Note once again that the angle value is the raw sensor readings not the actual angle of the pendulum). A second line called "power" was created in parallel with the "angle" line. The value indicators (the green squares) of both lines are tied and moves together. Because each line can have different value ranges, the lines can be used as a scaling tool. For example, the lines can generate a motor power value that is inversely proportional to the pendulum's falling angle (See Figure 4-2 for a more detailed explanation). I refer to this approach of using on-screen objects to assist in calculation as Spatial Computer Paradigm (SCP). Please refer to the appendix for a complete description of SCP.

Figure 4-2: This example shows how two lines can be used to convert (scale) one value to another. The value positions (the squares) on both lines are tied together with a vertical line. When the value of the _angle line changes the value of the _power line changes as well. Thus, the power value will depend on the physical location on the _angle line. A conversion takes place because the lines have different value ranges. From the image above, an angle of 202 is converted into a power of -3.38.

I refer to this method as ***continuous*** because a slightest change on one line would cause the value on the other line to change as well. On the other hand, Albert and Anderson's previous approach is ***discrete*** because it relies on discrete rules for the conversion. The output does not change unless a different rule is triggered.

I demonstrated my approach to show how it is similar to their previous approach, but with added benefits that would alleviate the two difficulties they were having (although, as later observed, this was not necessarily always the case).

§ **Tuning.** If the power response to the pendulum angle needs to be changed, it can be done by resizing one of the line objects as demonstrated in the figure below.

§ **Ability to Observe**. Since this line approach allows a visual observation of the parameters involved in the system, the students can see the angle and the converted power values in real time.

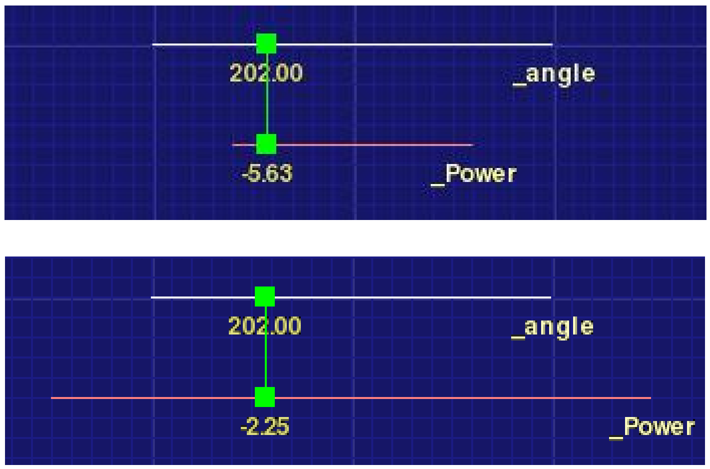


Figure 4-3: The conversion between two line objects can be influenced by changing the line's length. The upper figure shows how a shorter _power line increases the power value in response to the angle. An angle value of 202 is converted to a power of -5.63. The lower figure shows the opposite when the _power line is expanded. An angle value of 202 is now converted to a power of -2.25.

Although neither approach at the time could balance the pendulum, Albert and Anderson felt that the continuous approach may be a better choice due to the benefits described above. Albert verbally contrasted the new continuous approach to his older discrete one as being like dividing the angle into many, many, many tiny segments. So, no matter how detailed he makes his rules, it would not match the line object's resolution. Both students were comfortable adjusting the length of the line objects to influence the value conversions.

Although both Albert and Anderson saw and agreed that the continuous approach was better, they would later revert back to their rule-based approach. This will be described when the speed of the falling pendulum was taken into consideration in section 4.2.1.9 on page 57.

4.2.1.5. Moving Forward by Observing Body Movement as a Working Model

I knew from my control engineering experience that the system would not be able to accomplish the task if the angle rate of the pendulum was not considered. The question to me was how to get the students to think about this.

I decided to use a manually-driven working model as a starting point. We captured the pendulum car's motion while being manually balanced with the hand. A sensor-only pendulum car with the exact same configuration as the motorized car was used. Both students controlled the car by pushing it back and forth on a track. A camera was used to capture the movements. The sensor readings from the car and the track were also logged using the Logo programming environment.

Figure 4-4: Albert and Anderson making recordings of their own movements balancing the inverted pendulum.

As we reviewed the recordings, I challenged the students to see if there are any differences between what they are seeing and what occurred in their previous program. A particular challenge was to see if the car ever moves in the opposite direction to the pendulum's angle. This would violate their program which always tells the car to move in the same direction as the pendulum's angle.

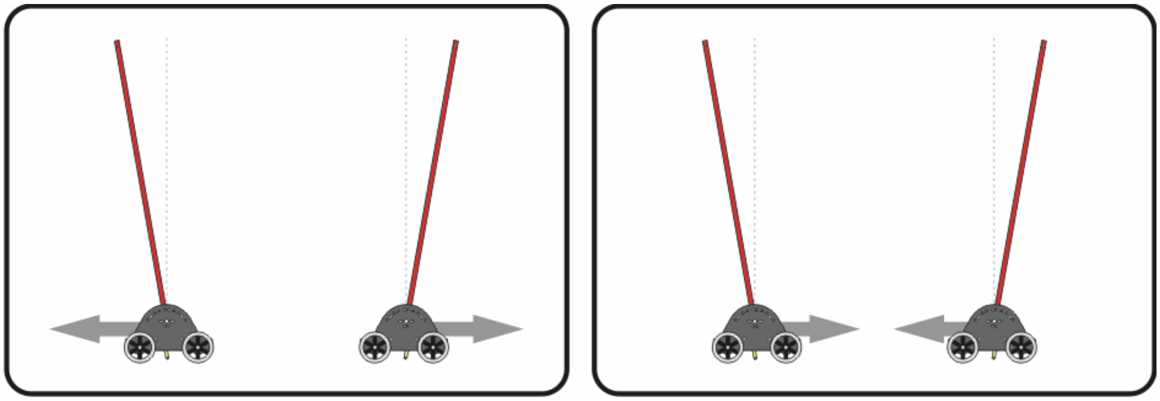


Figure 4-5: The left box shows the rules that are present in Albert and Anderson's program. That is, the car should always move in the direction of the pendulum's angle. The right box shows new states that were observed in the recording of hand balanced sessions where the car was seen to move in the opposite direction of the pendulum's angle.

It turns out that both students spotted many occasions where the car moved in the opposite direction to the pendulum's angle. These observations were done by looking at the physical pendulum while being balanced, replaying the recorded video in slow motion, and by analyzing the recorded sensor information using graphs as shown in Figure 4-6.

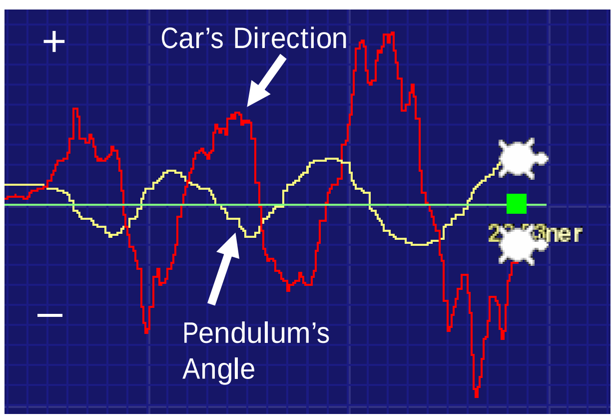


Figure 4-6: An example of a graph that was created to analyze the car's direction and the pendulum's angle. If the car were to always follow the pendulum's angle (as Albert and Anderson thought), both lines would always be on the same positive or negative halves (above or below the center horizontal line). The graph clearly shows that this is not the case. The car speed and pendulum's angle are often in opposition.

The students were not quick to be convinced of this new discovery. It was not until we produced and jointly deciphered the graph that the discovery became hard to deny. When balancing the car with their hands, both the students did not see that the car sometimes moves in the opposite direction from the pendulum's angle. When asked, they falsely confirmed that the car follows their initial description –that the car's direction always follows the beam's angle!

When I pointed out moments in the recorded video where their rules were violated, they still tried to defend their theory. Albert thinks the observation is a byproduct of how slow we are to react to the falling pendulum. He says "It is not on purpose. Logically it should not do that to work but somehow it does." Anderson also suggested that he thinks humans occasionally do it wrong. But we are capable enough to recover from those mistakes. The machine on the other hand is much faster and thus should not do the same mistakes.

However, when we made a graph of the car's direction and the pendulum's angle, it was rather obvious that the opposing motion is happening on a regular basis. This is when both students showed signs of accepting the graph's behavior as valid evidence for something new.

4.2.1.6. Introducing Speed

As mentioned before, my objective at this time was to get the students to think about speed as a new variable. My emphasis on speed or angle rate was derived directly from my previous experience with PID control systems. In a PID approach, one must consider three variables when aiming to accomplish a stable controlled system: Proportional, Integral, and Derivative. In the pendulum's case, "P" is the angle, "I" is the accumulated error of the angle, and "D" is the angle's rate or, in more general terms, speed.

The use of Derivatives allows the system to anticipate what to do instead of just reacting to what is currently happening. For example, when the pendulum is moving towards the center, the car can slow down or even move in the opposite direction before the pendulum reaches the center so that the pendulum would not overshoot to the other side.

PID control is a well known idea that is often treated as a formula. Given this mindset, I thought if Albert and Anderson were to balance the pendulum, I would have to get them thinking about the "D" (and potentially the "I") components. The following section shows how I tried to achieve this, but ended up with an unexpected outcome.

After the experiment, Albert and Anderson were looking for an explanation of the new discovery and a new direction for their program. I tried to influence them by explaining the role of the angle rate. Both students had difficulty understanding what I was trying to convey. I started by describing verbally the idea while showing them the recorded data and video. Then I drew a diagram on paper and acted out the cases where speed influenced the direction of the car. After many tries, they eventually started to see what I was getting to. Albert sees speed as a way to "save time on the next cycle" and in the end it will be a "giant time saving opportunity."

4.2.1.7. Tip Speed Control

In addition to my idea of using angle rate, Albert and Anderson had developed another theory based on their observations. They noticed from both the video and the data recordings that the top tip of the pendulum was the least moving portion of the system. Albert said "It is as if the car is swinging from the pendulum" while acting it out with his hand as shown in the figure below.

Figure 4-7: Albert acting out his observation that the top tip of the pendulum is moving the least and perhaps that is what they need to focus on.

The students also mentioned that when balancing the pendulum with their hands, it was a lot easier when they were looking at the top tip. These observations lead them to thinking that what they really needed was to regulate the movement of the top tip. And the best way to monitor the tip is by tracking its speed. Although the term speed is being used, it was of a completely different kind from the speed I was thinking of initially (angle rate vs. tip speed). The idea is a drastically different approach from the canonical PID method typically applied to this problem. Most important, in contrast to the previous angle rate idea, this top tip idea was straight forward to them. They were also more excited and enthused to try it out. In the end, they decided to go with their idea and abandoned mine.

4.2.1.8. A Slight Detour: Determining the Optimum Weight Position

After deciding on the strategy to pursue, Albert and Anderson went off a short detour to determine the optimum spot on the pendulum to place weights. It just happened that Albert and Anderson stared to argue whether it was better to put a weight at the bottom or top of the pendulum. They decided to perform a small experiment to properly analyze each case.

The experiment, as shown below, was conducted by running three tests each with weight placed at different positions on the pendulum: top, middle, and bottom. They then performed the task with their hands and recorded multiple tests for each setting. A graph was plotted comparing the motion of the tip and the car. Their measurement for success was a setup that was most controllable. In other words, the setup with the least tip and car movement is the best one.

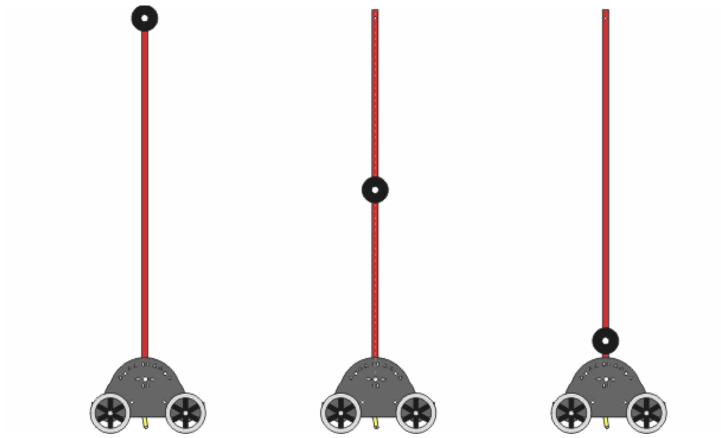


Figure 4-8: Albert and Anderson experimented to determine the optimum location for the stabilizing weight. The figure, from left to right, shows three locations that were tested: top, middle, and bottom. They then made recordings of the motions when being manually balance using their hand.

In order to conduct the test, they needed a way to track the pendulum's top tip. This information was not directly available to them as there was no sensor providing that information. However, they were able to obtain the tip's position by using a feature of the Logo environment that allows turtles to be stuck to each other. In brief, a new turtle was created and stuck onto the top tip of the beam object (also a turtle). When the beam moves, this new turtle moves as well. Thus, the new turtle's position is the desired top tip position. Please refer to the appendix in the section "Tracking the pendulum's tip" for a detailed description of how this operation was implemented.

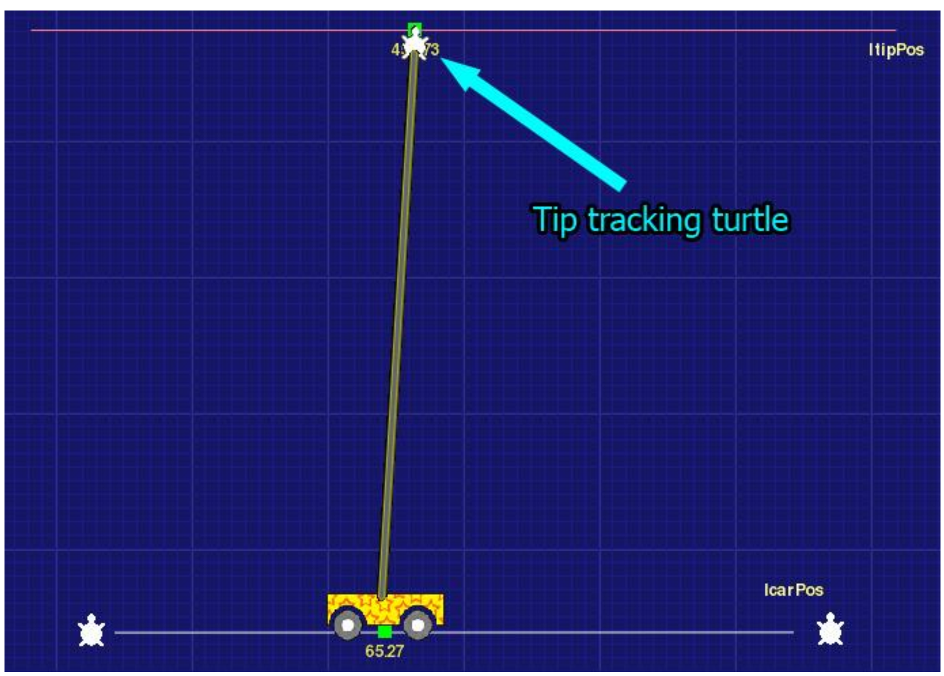


Figure 4-9: showing how the pendulum's top tip was determined by sticking a turtle to the on-screen pendulum object. The tracking turtle's X coordinate is the desired position.

Once all the needed information is recorded, the students plotted graphs comparing the tip and car motion of each setting. The following figures are examples of each setup.

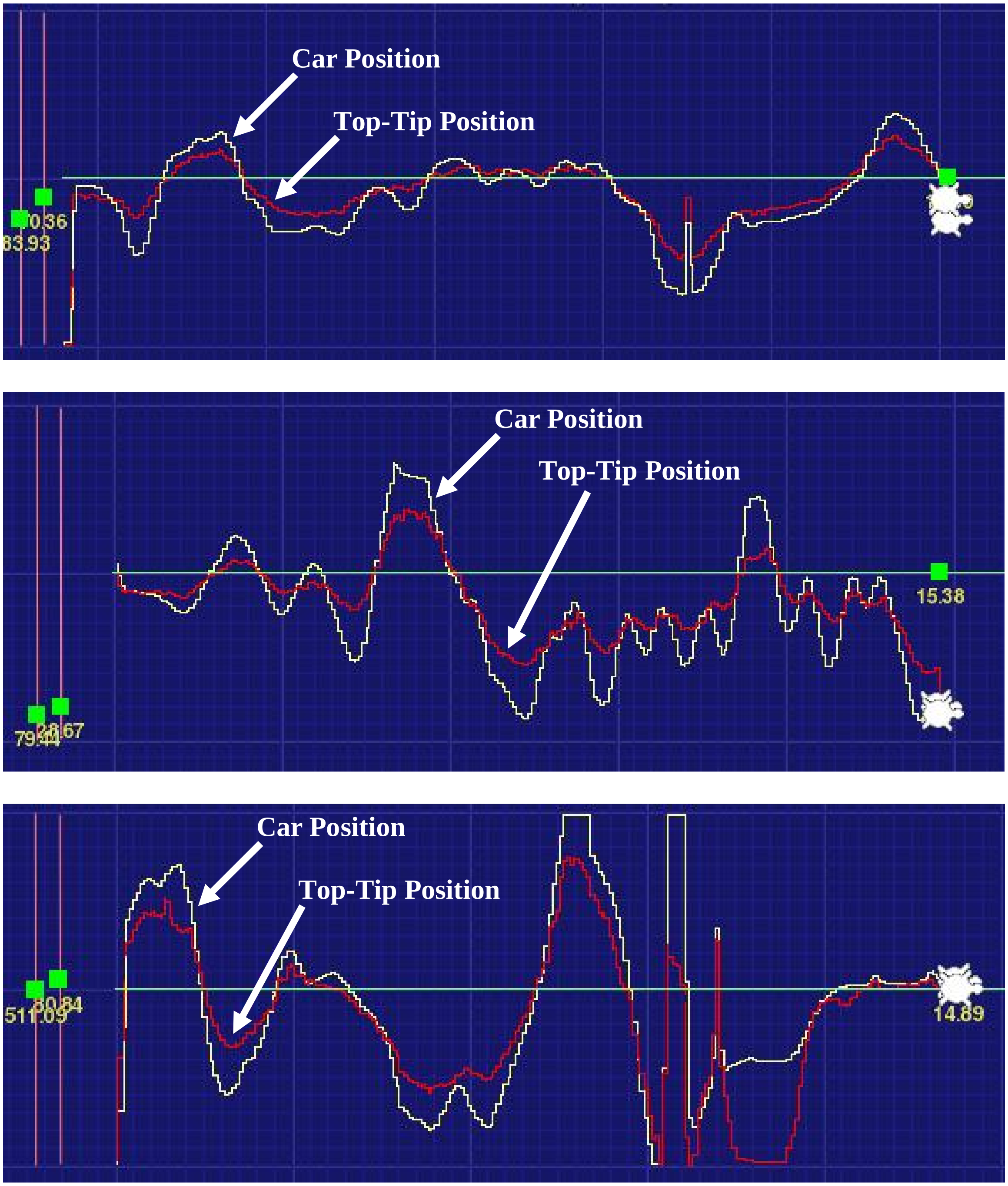


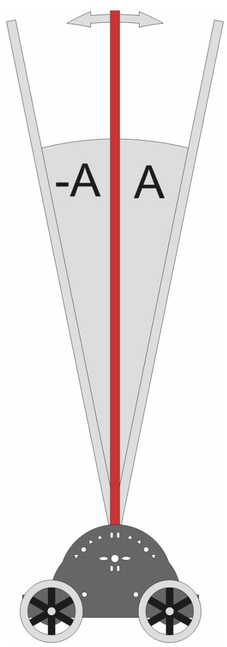
Figure 4-10: Graphs showing movements of the pendulum's top tip (red) and the car (yellow). The car was manually controlled by the hand. The top, middle, and bottom graphs show movements when weights are attached to the top, middle, and bottom respectively. Weight at the top tip has noticeably less movement.

As Albert and Anderson looked at the graph, they concluded that the weight at the top tip was the best option. It exhibited the least motion of the tip and the car, which they identified as being more stable.

To me, this mini-experiment showed me that both students are becoming familiar with not only the tools, but also the idea of proving a theory by setting up an experiment, analyzing results, and creating a criteria to which conclusions can be drawn.

4.2.1.9. The Return of the Struggle between Discrete and Continuous Approaches

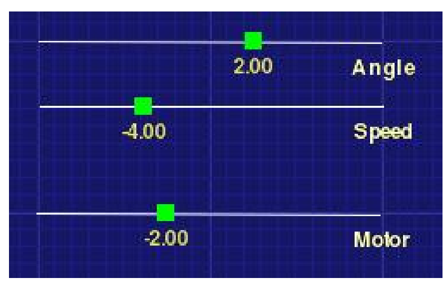
Once the weight position was out of the way, the students started to create their new program. The idea was that in addition to the pendulum's angle (their previous approach) the car's direction would also depend on the tip's motion (speed). In more precise terms, they identified special angle ranges called a "gray area." If the pendulum's angle was with in this gray area (-A to A in the figure below), the car's direction would be determined by the top tip's direction.




```
If angle > 0 [ moveRight ]
If angle < 0 [ moveLeft ]

If angle < A and angle > 0 and speed < 0 [ moveLeft ]
If angle > -A and angle < 0 and speed > 0 [ moveRight ]
```

Anderson and Albert's discrete method



A similar program using the continuous method. The motor output power is the sum of Angle and Speed parameters

Figure 4-11: (Left) Illustration of the gray area where speed was taken into consideration as well as position. (Right) Two approaches to implement the idea.

When Albert and Anderson started implementing their idea, they immediately reverted back to their discrete if-then approach. A simplified version of their program is shown in the upper left of
Figure 4-11. Basically, a new rule have been added telling the car to keep moving in the same direction until it catches up with the moving tip –even if the angle has flipped to the

other side. The gray area is the angle range where the car will be sensitive to the tip's speed as well as the angle.

The pendulum still quickly failed even with this improved program. In fact, what was observed was not so different from the previous program they had. All the problems with the previous program were still present and were actually worsened. Since there discrete approach had no gradient in the motor power, the problem with the car being too bouncy came back. The problems with debugging and tuning the system, which I had identified before, had also come back. But this time it was a lot worse!

With four concurrent rules in the program, it was impossible for Albert and Anderson to figure out which rule was causing what was observed. They were able to articulate issues that were directly observable. For example, "the car is still too slow" or "it is too bouncy," "it is not catching up." But they were not able to correlate the issues with the rules in their program. This presented a huge disconnect between the program and the observable outcome.

To me, this was another opportunity to demonstrate my continuous approach using the line objects. I hoped that this second exposure to the method would stick better and that they would use it instead of their discrete rules. However, it turned out that my continuous approach did not help them very much this second time around either.

I demonstrated to Albert and Anderson how a similar program could be created using line objects. A simplified version is illustrated in
Figure 4-11. There are two lines representing the angle and the tip's speed. The output motor power is the third line which value is the sum of the first two. Therefore, if angle and speed are both positive the sum or the motor power will be positive and the car will move in that direction. Since the sum is an analog value, there is a gradient in the output power and its value can be influenced by resizing the lines as described before.

This approach performed better on the pendulum car. However, Albert and Anderson had trouble understanding what was controlling the observed behavior. Although they could see the values moving on the line objects, they were not able to understand or explain why thing were happening the way they were. Anderson mentioned that "it seems to be doing better, but I'm not sure how." To my surprise, the situation was no different from where they were with their discrete approach.

The above situation revealed to me that there was a serious flaw in my approach. It may have delivered a better end result but from a learning perspective—the human comprehension aspect—it was not a success. Albert and Anderson continued playing with the line objects for a while. They re-sized them and observed the results. However, the process seemed artificial. Although they toyed around with the line objects and tried to figure out the best combination, they did not have a good explanation of what their actions were really doing to the system.

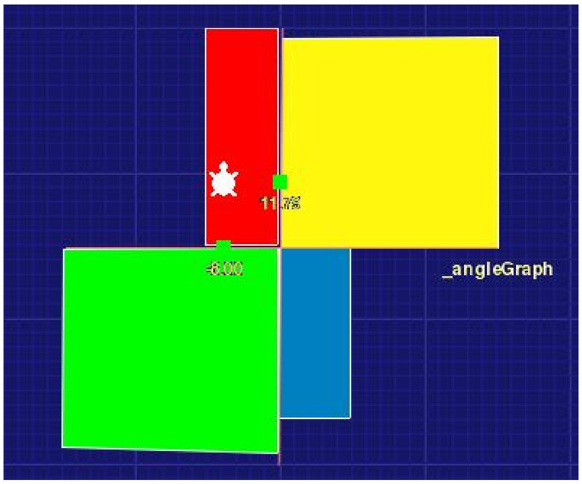
4.2.2. Phase II: Mix of Discrete and Continuous Paradigms

After some reflection of the initial experience with the students, I decided to look for a new approach that could alleviate the previous tension between the discrete and continuous approaches. Here are the lessons that I had learned:

- Discrete rules are important to students in their understanding of the program.
- The ability to see how rules affect the output was also key in the debugging process.
- The program also needs a gradient in the produced output, which was one of the main emphases of my previous continuous approach.
- If it is tedious to tune or adjust parameters, students become less motivated to debug and go through multiple testing iterations.

4.2.2.1. Spatially Defining States

With these observations in mind, I had experimented with a new way to help students advance by building on top of their discrete approach but at the same time trying to make connections to the needed analog method. The following shows the evolution of this effort, which in the end was much more successful.




```
if istouching "t1 "greenShape
    [goleft]
if istouching "t1 "blueShape
    [goleft]
if istouching "t1 "redShape
    [goright]
if istouching "t1 "yellowShape
    [goright]
```

Figure 4-12: (Left) Albert and Anderson's implementation of their state machine idea using color rectangles to define the region of each state. (Right) The needed Logo program.

Figure 4-12 shows Albert and Anderson's new program. The main difference of this new approach is the use of spatial properties of the screen to define discrete states. The inverted pendulum's angle and tip speed variables are plotted on a two dimensional graph (angle on the X axis and tip speed on the Y axis). Different regions representing a "state" are then defined by creating color rectangles. A turtle is then programmed to locate itself at an (X, Y) position according to the current measured angle/speed values. An "isTouching" command determines the rectangle on which the turtle is laying. This approach can be used to visually identify the state of the inverted pendulum. For example, if the turtle is touching the yellow shape (the upper-right box) it means the beam angle is leaning to the right and the tip is also moving to the right. In this case, the reaction, as seen in the program, is for the car to "go-right." The red and blue rectangles (the upper-left and lower-right boxes respectively) represent the gray area mentioned earlier.

Although this program is essentially the same as what both students had before, a big advantage is the ability to better see how the state changes while the program is running. That is, the students can see the turtle moving from one region to another.

In Logo, a turtle object can be used to draw lines. When a "pen down" command is executed the turtle object will leave a trail behind when it moves from one place to another. When this feature was applied to the program, the students were able to see a trail of how the turtle moved from one state to another. This was also how the graphs that were previously shown were created. This gave a chronological view of the system's behavior.

Figure 4-13 shows a comparison between the trails from a pendulum being controlled by the hand (left) and by Albert and Anderson's program (right). The latter graph is not entirely valid as the students had to help the pendulum when it fails. There was little tip movement (because they were grabbing it when it was falling) while a lot of angle movement was present. This is why the graph on the right is flatter than the one on the left. The manually controlled pendulum graph indicated that the pendulum's angle was kept to a minimum while the tip speed varies more.

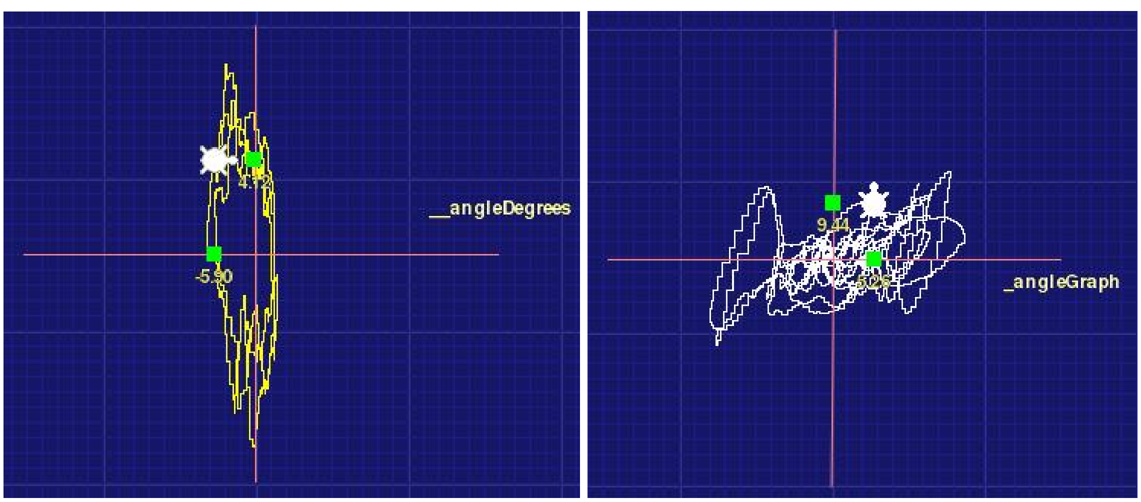


Figure 4-13: A comparison between an inverted pendulum controlled by the hand (left) verses one that was controlled by the student's program (right). The rectangles that defined the states were removed from the figures for the clarity of the lines.

4.2.2.2. Adding Gradient to the Calculations

Despite the improvements, Albert and Anderson's program was still limited by the fact that the programmed response was a simple go-left or go-right action. The pendulum still quickly fell. There was clearly a need for a gradient in their motor output. Thus, we experimented with a new way to extend the program to include mechanisms for calculating proportional quantities. Here is a description of how it worked:

Let's define four states for the position/speed relationship:

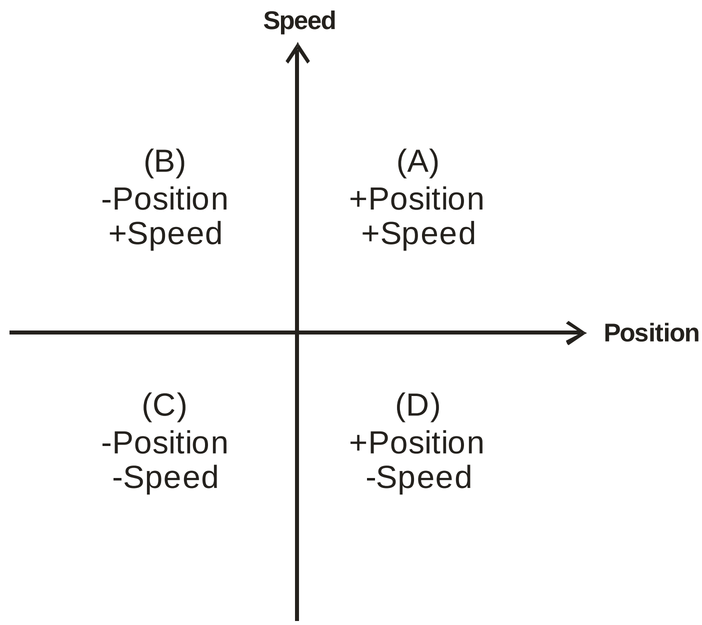


Figure 4-14: An illustration of the four states that represents the possible combinations of the position and speed variables.

State (A) and (C) are straightforward. The beam falls in the same direction as the speed. Utilizing the objects that we have on the screen, the motor power level that counters this motion can be proportional to the distance between the graphing turtle and the graph's origin. This produces a value that is proportional both to position and speed as illustrated in the figure below.

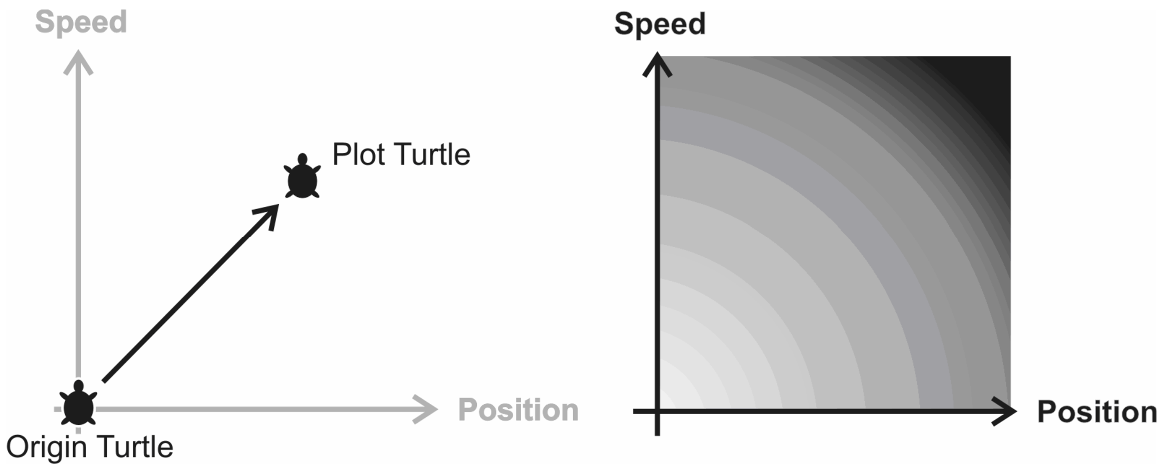


Figure 4-15: (Left) a turtle is placed at the graph's origin as a reference point. Plot turtle's location on the graph depends on the position and speed values. The distance between the two turtles is a vector with a length that determines the motor power. (Right) a color coded graph showing the motor power at each position/speed combination. The darker color means higher motor power.

States (B) and (D) are entered when the pendulum is still leaning one way but is starting to move the other way (speed and position have opposite signs). The vector from the graph's origin to the plot turtle doesn't work for this state because it will create a sudden jerk in the motor power when changing from state (C) to (B) or from (A) to (D) where the motor power needs to be reversed. A simple solution is to use only the speed component. This will result in an inverse power that always begins from zero (since speed = 0 at those problematic state borders). To keep the methodology consistent by using the turtle distance to determine motor power, the program will make sure the origin and plot turtles are always vertically aligned. The distance between the two will then depend only on the vertical positions as illustrated in the figure below.

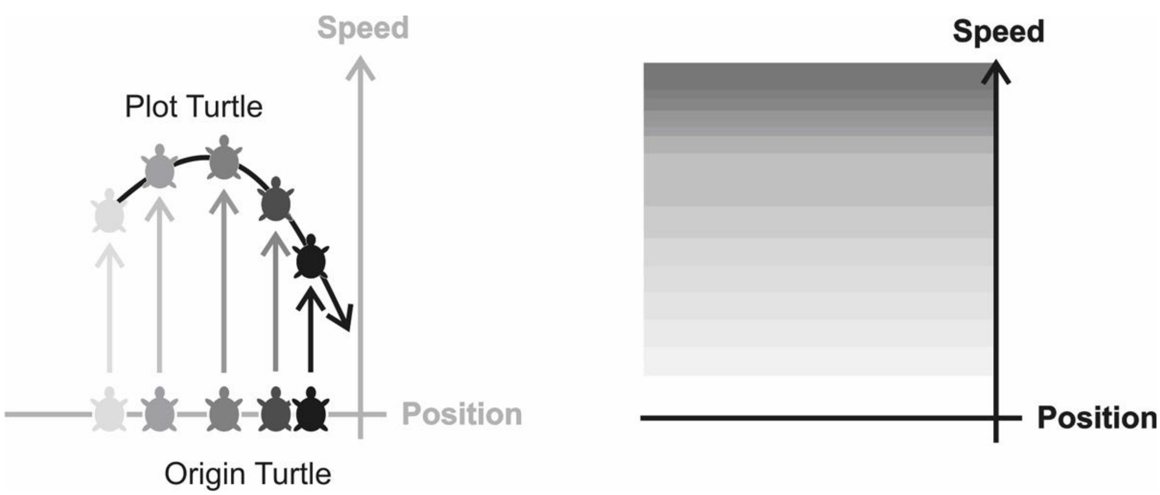


Figure 4-16: (left) to eliminate the motor jerks caused during state transitions, the position variable was removed from the distance vector in state B and D. This was accomplished by vertically aligning the origin and plot turtles. (Right) a color coded graph showing the motor power at each position/speed combination. The darker color means higher motor power.

This method allowed the pendulum to show promising signs of succeeding. We were able to get it working for 8-10 seconds. However, this approach has proven to be effective for the ball balance beam problem. In fact, it responds better to perturbation than the classic control theory (PID) approach that I have tried before.

4.2.3. Realization of the Phase-Plane Logic Control

I later discovered that this latest approach is significantly similar to a well-established engineering method called Phase-Plane Logic Control. This numerical method utilizes simple graphical means to establish stability [Gelb, Velde et al., 1968]. Phase-Planes have been most notably used in the space shuttle orbital control in the early 1980s. The space shuttle's on-orbit reaction control system (RCS) utilizes multiple jets distributed throughout the shuttle to maintain transitional and rotational control.

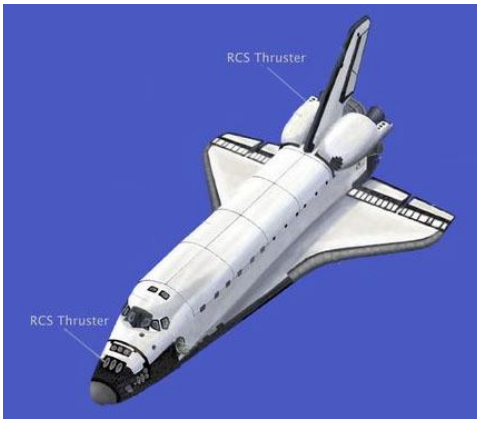


Figure 4-17: The space shuttle's on-orbit reaction control system (RCS) utilizes multiple thrusters distributed throughout the head and tail sections.

For each axis, a Phase-Plane of angular rate vs. attitude angle is constructed and divided into multiple regions as shown in the following figure. An appropriate rotation command (e.g. which jets to fire) is determined based on which region the shuttle's current state or phase point lies. Since the jets operate on an on-off basis, switch lines are used to determine when to "switch on" or "switch off" the jets. For example, in regions 4 and 8, the jets will fire until the phase point crosses the S13 switch curve towards the origin [Wie, 1998].

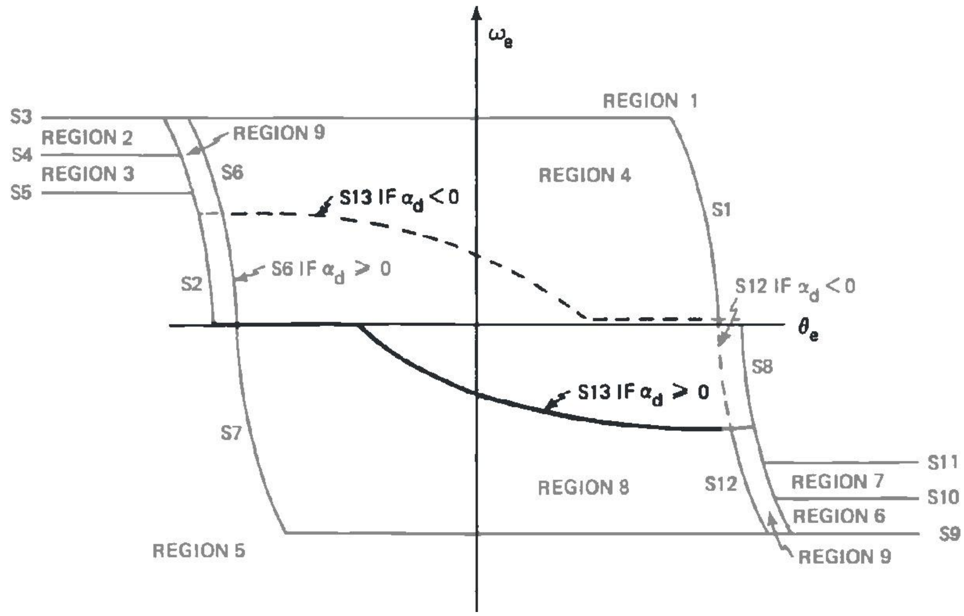


Figure 4-18: A Phase-Plane diagram used to implement the shuttle's RCS system. Different regions were defined to tell the craft what state it was in. Switch lines (S13) were used to determine when to fire the appropriate thrusters when the shuttle was in regions 4 and 8.

In comparison to the students' program, the resemblance is obvious. The figure below shows the four states that were used. Each state has been shaded to represent the motor power gradient. The darker color results in a higher motor power. If we create lines that represent the 50% motor power, they remarkably resemble the switch lines of the space shuttle control logic (compare the dotted and solid lines on the both figures).

The advantage that the students had over the traditional Phase-Plane approach was that they could implement the method by interacting directly with the regions as they see it on the screen. They could see the phase-point (the moving turtle) of the system in real-time. They could easily move and resize the regions to observe how the robot's behavior was affected. Compare this to the traditional implementation that requires the visual representation to be transformed into a less tangible form, such as a look-up table. The execution is harder to observe and it takes more time and effort to make any changes.

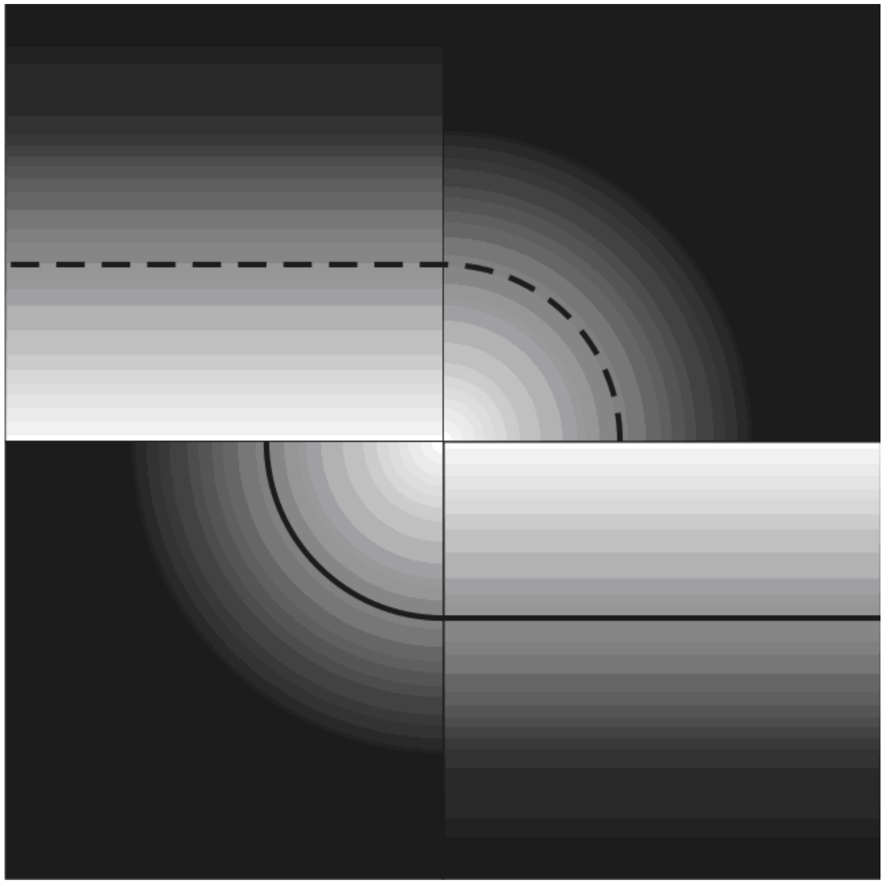


Figure 4-19: A color coded illustration of the Albert and Anderson's Phase-Plane implementation. The solid and dotted lines simulates where a switch line would have been if we had taken the 50% power level as the motor on/off position. The lines closely resemble the switch lines of the space shuttle controller.

4.3. Group 2: Greg and Joe

I will now discuss my work with Greg, 15, and Joe, 14. Greg and Joe worked together for the first seven sessions. Joe then moved to France, which left Greg working alone. Despite being alone, Greg continued working with me through out a twelve month period.

I divide my work with this group into four phases. The first describes the interactions before the students started to create computer programs. In contrast to Albert and Anderson, this group was immediately drawn to the robotic pendulum. During this time, many ideas and observations were made. These ideas sometimes come and go quickly and they did not seem to have any specific structure. This chaotic situation seemed meaningless at first, but the pockets of ideas occasionally played a role in their future work and helped them to understand what was going on and move forward.

Phase two describes the students' first attempt to create an autonomous system. It resembles Albert and Anderson's approach in which they focus on reacting to variables that change over time (e.g. beam angle). They also run into similar difficulties. This phase highlights the pathway Greg took to develop his understanding of a derivative (or speed) and its role in balance control. Despite the researcher's attempt to demonstrate the idea to him early in the learning process, Greg needed a much broader investigation of the subject which took time and spanned across phases three and four. His understanding was not neatly constructed step-by-step as described in a textbook. Rather, it was driven into many directions by the behavior of the pendulum car. Some directions were unrelated to others while some others lead to a dead end. But throughout the learning process, Greg had always maintained a coherent, even if not so accurate, picture of what he thought made the system balance. And as he gained more experience, his model of balance control (including a derivative) become more sophisticated as well.

Phase three describes a new direction that Greg had taken. It was based on his observation of a recording balancing the pendulum. He devised a prediction model that would hypothetically allow the car to know where to go ahead of time. This approach is drastically different from the typical "reactive" approach. This phase describes the developmental steps and the evolution of his ideas and also the struggles he faced along the way.

The final phase describes Greg's work on a new project. He wanted to move on to do something new and we ended up developing a computer simulation of a hover board. He applied what he had learned from his previous experience to the problem. The researcher also applied the Phase-Plane technique he developed with Albert and Anderson to the project. This phase shows Greg increased sophistication in thinking about a derivative. It also describes how the new Phase-Plane methodology allowed Greg to develop his understanding about stability through the analysis of shapes that were drawn in the Phase-Plane graph.

4.3.1. Phase I: Initial Reactions and Experiments: When Ideas Seemed Chaotic

During the first few sessions, a significant amount of observations and ideas were made and developed. The rate of which one idea changed to another was quite high and they seemed random and chaotic at times. Some ideas came and went in a few minutes while some others stuck around longer and sometimes induced experiments. Overall, there was not a clear order of how things were proceeding.

In the beginning this chaos was a bit of a concern to the researcher. I did not see where the students were heading. But a direction was eventually established once we got to the programming part. More importantly, the students would occasionally make a connection between what they were working on with these earlier discussions. This turned out to be helpful for the students to think about the situation at hand and move forward. Over time, many of these isolated pockets of knowledge became an integral part of the student's thinking. Thus, knowledge that seemed chaotic in the beginning eventually found its place in a coherent way.

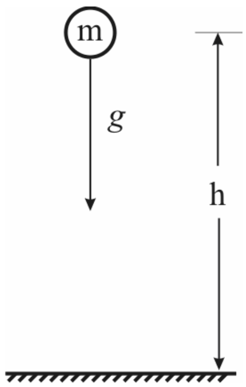
The following are some examples of the various discussions that the students had.

4.3.1.1. Potential Energy: Is it Easier to Balance near the Floor?

Greg referred to what he learned in school about potential energy. Because potential energy is a function of height, Greg thinks that the closer the pendulum is to the floor, the less potential energy it will have. And a pendulum with less energy will not fall as much. He tried out this idea by balancing the beam as close to the floor as he can (see the figure below).

Figure 4-20: Greg trying to balance the pendulum while keeping it as close to the floor as possible to reduce the potential energy. He believed this would make the pendulum fall less.

This idea is, however, a misconception. The definition of (gravitational) potential energy as it is taught in school is as follows.




```
Potential Energy = mgh

Where:

m = mass
g = gravity
h = Height of which the object is raised
```

Figure 4-21: A common definition of gravitational potential energy as taught in schools.

Greg did not realize that an object moving from altitude X to altitude Y gains the same amount of energy no matter how high the object is from the ground (assuming we are referring to a reasonable height). Thus, the effort needed to hold the pendulum upright is also the same regardless of the altitude.

Although Greg experiment did not yield results that corresponded with this theory, he came up with an explanation of his own. He thinks that his arms cannot move as freely when he is sitting down compared to when he is standing. So, this added difficulty nullifies the benefits that would have otherwise been observed. He preferred this explanation over my description above.

4.3.1.2. Potential Energy (Take II): Is it Easier with Weight at the Bottom?

After two weeks, Greg found another chance to revise his theory. It was during a discussion about whether it is better to place a weight on the top or the bottom of the pendulum. Both students were leaning towards having the weight at the bottom. I discuss this in more detail in the next few sections. However, one of the strong arguments Greg had was that putting weight at the top makes the beam more "fally." Because the weight is high it can fall more and gain more energy whereas having the weight at the bottom does not give it much space to fall and, therefore, does not give the weight a chance to gain as much energy.

Greg's revision of his theory actually holds up pretty well although he does not realize that putting weight on top also gives the tip more momentum of inertia, which is what makes balance easier to humans (and robots). Momentum of inertia slows down the initial movements and gives us more time to respond. However, the two students were confident about their theory and were hesitant about my momentum of inertia idea. They went ahead and worked with weights at the bottom for a while until this configuration started to show its weakness and gave them a hard time when programming the robot. It was then that they became more interested in moving the weight to the top.

The shift was not smooth. Although putting the weight at the top changed the behavior of the beam, it was not necessarily easier. They quickly noticed that although the beam would tend to stay still in the beginning, once it starts to fall, it is very hard to recover. Thus, my so called “correct” approach was, to them, just a “different” approach.

In the end, this whole process gave the students the benefit of seeing how weight can affect the pendulum differently depending on where it is placed. This is summarized in the following table.

	Weight at top	**Weight at bottom**
Benefits	Initially moves slowly which gives more time to respond	Easy to make the pendulum move
Drawbacks	Once it moves, it is harder to stop	Tends to move too fast and erratically

Greg and Joe’s arrival to the conclusion of the optimal weight position is significantly different from Albert and Anderson’s story. Albert and Anderson evaluated recordings and were convinced by analyzing graphs that shows less motion when weight is on top. Greg and Joe, on the other hand, made conclusions via a theoretical dialogue: a discussion of potential energy and momentum.

4.3.1.3. Obtaining a Holistic View: Is it Easier with the whole Car in Sight?

While Joe was trying to balance the pendulum car with his hand, he made an observation that might be easier if he can see the entire structure. Since the hand controlled pendulum does not allow Joe to stand away more than his arm reach, he cannot see the pendulum’s tip and the base car together at the same time.

This observation is, again, not entirely accurate. It is generally more important to see the top tip of the pendulum, as that is the part that people would normally try to regulate and maintain minimal motion.

Joe tried to step back as far as he can to get the best view of the pendulum. Since he cannot get too far back, he concluded that staring at the center of the beam yields the best compromise.

Greg contributed by trying to look at the pendulum at an angle. He climbed up a chair and looked down from a high angle (see figure below). Although he can see the entire structure, it was extremely difficult to reach down and move the pendulum car. Joe tried looking from below but encountered a similar difficulty.

Figure 4-22: Strategies used to gain a holistic view of the structure. Greg tried by looking down from a high angle (left) while Joe tried from a low angle (right). Both encountered difficulties moving the car.

Although their observations did not yield a useful result to them at the time, it established a discussion that later led them to think about "point of reference" as described next.

4.3.1.4. Point of Reference: Which Part of the Car Moves?

This is a debate that Greg and Joe briefly had relating to the point of reference idea. It started when Joe mentioned that he thinks the top tip of the pendulum stayed still while the base moved rapidly. Greg immediately countered this suggestion. He thought the base was the part that did not move because it was always connected to the same point of the car. So, it was the tip that was really moving.

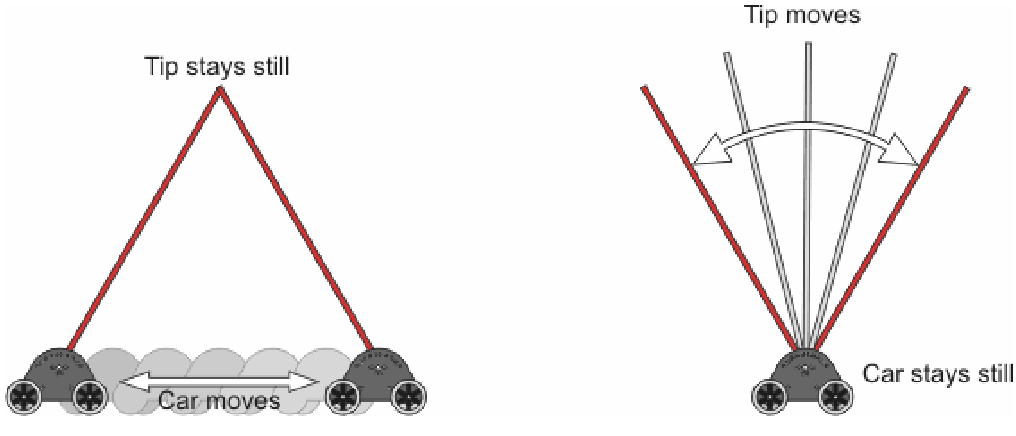


Figure 4-23: Two reference points. The tip can be seen as the part that tends to move the least in relation to the environment (left) or it can be seen as the part the moves the most in reference to the car (right)

Both students are correct. It just depends on what they are using as a reference. Joe is referring to the movement of the beam in reference to the environment. The tip is indeed the least moving section of the entire structure. Greg, on the other hand, is using the car as a reference. From the point of view from the car, the pendulum's top tip is the part that is moving the most.

Although nothing came out of this observation at the time and both students later moved on to other things, it would later be recalled while they were discussing different strategies that are based on these two frame of reference. This was particularly true when Greg was conceiving his prediction model in section 4.3.3

4.3.1.5. What Sensors to Use?

Throughout the course of the first few sessions, the two students would come up with suggestions of what kinds of sensors they think the car should have to accomplish the task. Although the suggested sensors were often redundant to what is already there, the exercise was useful in getting the students familiar with the sensors that were available and what information they can derive from them. The task forced them to think about the investigation systemically, in terms of the phenomena, the technology, and the methodology for coming to satisfactory solutions. Further more, some discussions gave them a reference point that later helped them in developing new balancing strategies.

This is particularly true for the discussion about how to sense the pendulum's tip position. Greg came to me at one point saying that he thinks it is important to know the position of the top tip. Since this sensor does not exist, we had a discussion about how we could use other existing sensors to calculate it. He did not use this idea right away. But it became an important part later in the implementation of his prediction method. See section 4.3.3.1 on page 78 for more information.

4.3.1.6. Dynamic Stability: Being Stable While Constantly Moving?

After spending some time trying to balance the stick on his hand, Joe made an observation that he feels it is easier to balance the pendulum by rapidly moving his hand back and forth to create a constant oscillation. He applied the idea to balancing the pendulum car and confirms how he feels. He explains that "there is more to see when the pendulum is moving so it is easier to do"

This idea stuck with him even though we did not immediately pursue it. The idea resurfaced a few sessions later after both students had started to create an automated computer program. He noticed that the programmed car was rapidly moving back and forth. Although the car behavior was due to the immaturity of their computer program, it intrigued Joe to become more serious about his constant motion idea.

Joe's constant oscillation approach is sharply different from the traditional method towards this problem. Instead of focusing on minimizing motion of the beam, we are actually promoting it. This initially seemed out of line to the researcher. But after observing how interested Joe was with the idea and how frequently it came up, I worked more with him to see how far we could go with it. This turned out to significantly influence the direction of their work. It can be seen in their later strategies that they specifically design the car to balance by moving back and forth.

But before they arrived at that stage, we needed a way to objectively verify Joe's idea. This required them to come up with a measure for what would count as "easy" versus "hard" to balance. Joe can feel it when it is easy to balance but what specifically does that mean? And how can we measure that? Joe and Greg came up with an idea to compare the movement of the pendulum's top-tip and the car. The "easier" strategy should yield less motion of the top tip. Although what I had in mind was to measure how long one can balance with each strategy, the students' approach also made sense. So, we went with their idea.

Later when I showed them how to make recordings of the physical car and play it back on-screen, Joe made a series of recordings of himself balancing the pendulum car. A graph was then created showing the movement of the pendulum's top tip versus the movement of the car (see the figure below).

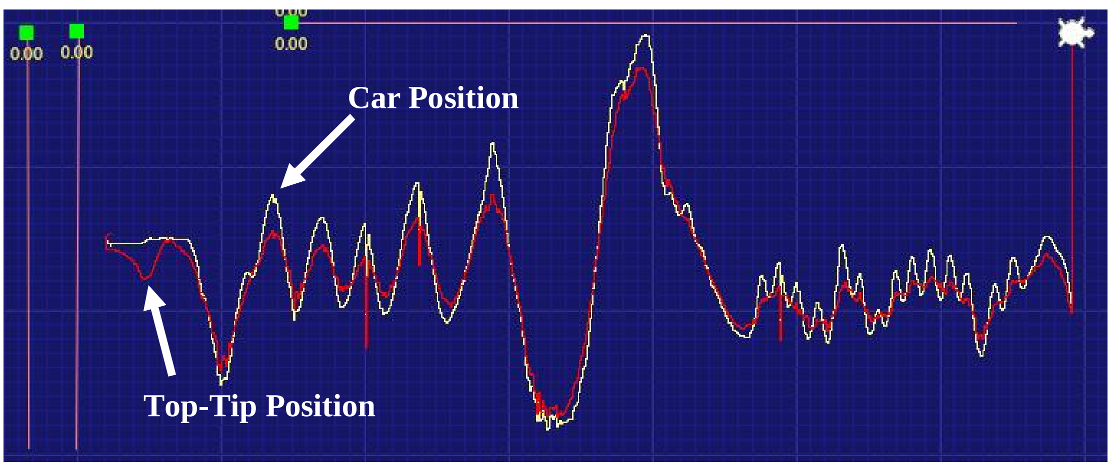


Figure 4-24: An example of Joe's recording. The red line represents the motion of the top tip and the yellow line shows the motion of the car. The recording was made while Joe was balancing the pendulum-car with his hand. Joe was trying to use graphs like this to prove that rapid motion of the base (right half of the graph) yields less motion of the top tip than trying to keep the beam still (left side of the graph).

Although the graph did allow us to confirm Joe's theory, the process of making this proof was the more interesting part.

In the beginning, I criticized Joe's conclusion by saying that the difference between the two strategies, as seen in the graphs, were not entirely decisive. Joe then tried to record many more samples to confirm his conclusions to me. Some of the graphs looked too convincing, so I accused him of manipulating the recordings by intentionally letting the pendulum fall in cases that goes against his theory. This led to a discussion about how one might manipulate data and "fake" a conclusion in ways that look very scientific. Many scientists are biased towards their desired conclusion and the evidence is sometimes distorted from the truth. This has nothing to do with ideas in balance, but it gave Joe a new perspective on scientific investigation and proof.

4.3.2. Phase II: Initial Programming and Encountering the Idea of a Derivative

This section describes the initial attempts both students made to control the pendulum car. In addition to describing the general development of their approach, a special emphasis has been put on how both students, Greg in particular, developed their understanding of role of a derivative (speed) in BCDS. As previously discussed in Albert and Anderson's cases, a derivative (speed or rate) is a crucial parameter in BCDS in addition to proportion (position) that is not straightforward for the students to comprehend. Also, the concept of a derivative is not usually introduced to students until they study calculus either late in high school or in university as it is deemed beyond the zone of achievement of students of this age. Unfortunately, it is also often introduced out of a context of use and solely as an abstraction.

Greg and Joe were first introduced to the concept though a discussion with the researcher early in the process. Unlike Albert and Anderson, Greg and Joe both appeared to understand the idea from the discussion. But once we moved on to the implementation stage, both students backed off and focused their efforts mainly on the pendulum's angle (the more straightforward parameter). However, when Greg started to realize that the car could not catch up with the falling beam using only the pendulum's angle, he expressed the need to make predictions that could hint the system of what to do ahead of time. In essence, that was the role of a derivative. The researcher believes that this was when Greg started to truly develop an understanding of what a derivative does, only that the formulation was different. In this sense, Greg "invented" the concept for himself with the need to solve a personally-felt problem.

4.3.2.1. Discussing the Idea of a Derivative: a Conversation that did not Stick

Greg and Joe started off similarly to Albert and Anderson. They created simple if-then rules that tell the car to move depending on the pendulum's falling direction. They were quick to adopt the line approach to make the car's speed proportional to the pendulum's angle. They had used the lines before to build manual controllers for the pendulum. Thus, they were already quite familiar with the idea.

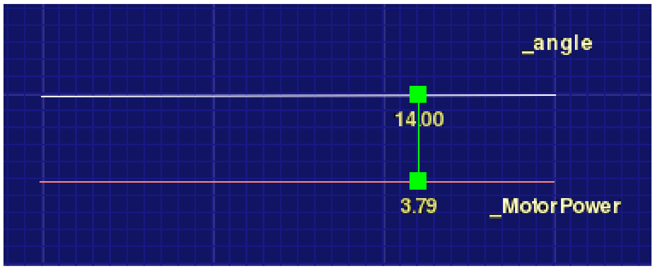


Figure 4-25: Greg and Joe's program using the line objects to convert the pendulum's angle to control the car's speed. The top line measured the pendulum's angle. Its value was mapped to the line under it, which was used to control the motor power on the car.

However, using only the pendulum's angle without the angular rate (speed) did not produce a working system. The pendulum car was never able to catch up with the falling beam. At first, the students believed that the problem could be solved by making the car move faster. But they soon learned that the problem was not solved while sometimes it had gotten even worse.

Because both students were rather quick to pick up the ideas so far, I decided to discuss the role of a derivative (speed) to see what they think of it. Similar to what I did with Albert and Anderson, Greg and Joe tried balancing the pendulum car by pushing the car back and forth with their hands. Unlike Albert and Anderson, though, both students appeared to accept and understand the role of speed rather quickly. When I made the point that the pendulum needs to slow down as it approaches the upright position. Both students were keen to accept the idea that speed was the variable to help make that happen. Because the conversation went so well, I naively assumed that the students had "gotten it."

Note that the students had decided to use the pendulum's top tip speed rather than the angular rate as the derivative variable. This was because they had talked to Albert and Anderson about it at school and felt that it was more straightforward[4].

The situation did not go as well once the students started the implementation. Their new program uses both the pendulum's angle and angle rate to control the car's speed. The approach was identical to the one I introduced to Albert and Anderson. Line objects were used to normalize the parameters and then the two values were summed to produce a power level for the motor that drove the car.

[4] See section 4.2.1.7 on page 54 for a discussion about tip speed versus angular rate.

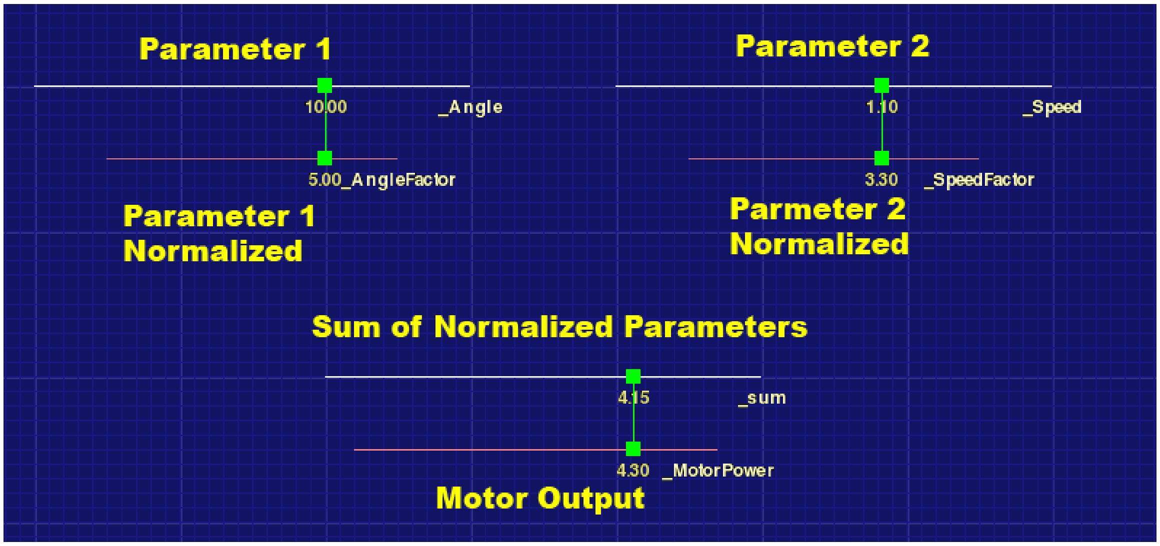


Figure 4-26: The initial program that used both the pendulum's angle and the tip speed to control the car. Both the input parameters (_Angle and _Speed) were normalized (_AngleFactor and _SpeedFactor) and summed together (_sum). The sum was then used to produce the motor power that drives the car.

Once we ran the program, it was then apparent to me that there was a large gap between discussing the idea and experiencing it in action. Whatever they had agreed about during the conversation was not revealed once they started to observe and debug their program. It was as if we had never talked about it. When the beam fell, both students went back to their previous thinking that the car was not moving fast enough. Although the pendulum worked slightly better than before, it did not make them reason about the problem differently.

There were two epistemological issues that I had observed. The first has to do with the approach chosen to implementation the idea. Both students had difficulty understanding what was going on looking at the line objects. This is the exact same issue Albert and Anderson had as described in section 4.2.1.9 "The Return of the Struggle between Discrete and Continuous Approaches" on page 57. Mixing the role of angle and speed by summing their quantitative values created an output that was difficult to comprehend. Although this approach was computationally efficient, it failed to act as mediator of the underlying ideas. Even when the students tried to manipulate the numbers, they were not able to make a solid connection between what they did and the output observed.

The second issue has to do with how much time the students have had to experience and explore the ideas for themselves. So far, they were mostly told about the importance of a derivative. It was done either through a conversation or a demonstration. Although the students were interacting with me all along and although they seemed to have built an understanding of the subject, it was not until they started to create their computer program that the phenomena came in full contact with them.

4.3.2.2. How the Idea of a Derivative Truly Emerged

The situation had now changed dramatically. The students' analysis of the problem had become fixated once again on the car's slowness. When I tried to bring back the conversation about speed, the students did not seem to "get it" anymore. But more importantly, they were so focused on their ideas about what was wrong that made my comments felt irrelevant.

They believed that the car was working well while the pendulum's angle was near the vertical position. But when the angle grew larger, the system quickly fails. The students wanted to figure out a way to detect when the angle from the vertical position grew beyond what they called a "safe zone." When that happens, the car should respond by rushing towards that direction at the maximum speed.

This was the time that I had developed the idea of using on-screen colored squares to identify discrete states. I had just finished adding the needed features to the Logo environment and this was a perfect opportunity to introduce the idea. The following figure illustrates how the students' "safe zone" idea was implemented.

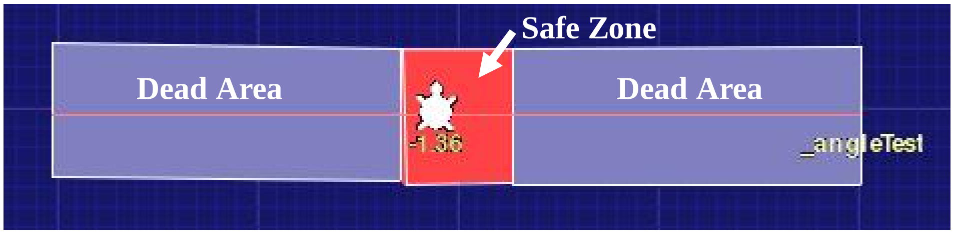


Figure 4-27: Using colored boxes to implement the different zones of the pendulum's angle. The box in the center was defined by the students as the "safe zone." The two boxes on the sides were the "dead area." The turtle moves sideways proportionally to the pendulum's angle from the vertical position.

The pendulum's angle controls the turtle's vertical position. Thus, a computer program can be written to detect when the turtle leaves the red box in the center (safe zone) and respond accordingly. The impact of this shift of approach was clearly observable. It worked well as a bridge between the analog nature of the sensor measurements and the concrete states the students wanted to define. This approach allowed the students to observe the current state (e.g. safe or not safe). And since the boxes can be easily resized, tuning (or adjusting the sensitivity) was straightforward.

The most important impact of this approach, however, was that it encouraged experimentation. This impact was clearly demonstrated in the case of Albert and Anderson. The same program could have been written using a set of "if-else" commands[5] but it would not have been straightforward to observe during execution and making modifications would have been a tedious process.

[5] See an example in section 4.2.1.2 on page 47 where Alex and Adam used if statements to test the pendulum's angle and control the car's speed accordingly

[Note that Joe had to move to a foreign country during this time. Greg was then working with me alone.]

Despite the improved implementation and Greg's tremendous attempts, the pendulum would not work, as speed was not considered. The more Greg tried to catch up with the falling beam, the worse it performed.

But after some time Greg developed a new idea. He stated that the car could never catch up with the falling beam because it was only reacting to what was currently happening. The system should be able to "predict" when the beam would leave the safe zone so that the car could respond before that happens.

This was an important development. The purpose of Greg's "prediction system" is essentially the function of a derivative. A derivative provides historical information about how a parameter has been changing, which is then used to make assumptions about what will happen next so that the system can respond accordingly.

When I asked Greg how he thinks a prediction can be made, he was not sure. He was interested to hear my idea of using speed to do the trick as written in the previous paragraph. But he was still reluctant and did not pursue it. I noticed that the idea of using the past to predict the future did not sit well with how he thought a prediction should be made. I became a bit frustrated at the time because I did not understand what was preventing him from getting what I was conveying to him.

After reflecting on what eventually happened afterwards, it became clear that Greg's exploration was more extensive than what was in the researcher's mind. Because of my previous engineering experiences, I always had the idea of using a derivative in the back of my head. Without this bias, Greg was more opened to investigate different strategies, poke at different methods, and build up his experience with the different aspects of the domain. This was necessary for him and perhaps it was why he was reluctant to take "speed" as the answer, even after acknowledging it in and earlier discussion. Instead, he chose to develop other strategies based on the behavior of his computer program. I believe this is good evidence that he was in the process of negotiating his conceptual understanding with the expression of the ideas through the computer and the reaction he was receiving from the world.

As the next sections would show, he would jump to a few other topics and learn about other ideas. The progression was logical but was also extremely organic. Towards the end, Greg's thinking would be more developed (see section 4.3.4.2 on page 88) and the programming approach had also evolved (see section 4.3.4.3 on page 89). And we eventually reached a stage where we were able to implement a program that uses a derivative in a way that resonated with Greg's thinking. This is a pathway towards understanding that would have differed greatly if I had imposed my own way of understanding on Greg.

4.3.3. Phase III: Making Predictions: Going Beyond Simple Reaction Systems

All of the strategies that have been employed by the students so far were based on a reaction model. That is, the response of the system (motor power) at any given time is calculated from the properties currently being measured (position, speed, etc.). Although a reaction system can work quite well (in fact most elementary control theory projects uses this strategy), it is based on a model that is rather limited when compared to human intelligence. Motions carried out by humans utilize far more sophisticated mechanisms based on models that, for example, allow us not only to react but also to predict what should take place ahead of time.

Greg and I have been talking about the importance of prediction for some time. But he was not able to translate what it means to make a prediction in concrete terms. It was only when we watched a video recording of a person balancing the pendulum that he could came up with an idea. This experience is significant for this research as it involves all the main aspects of the physio-syntonic learning framework. First, the idea emerged from the observation of body motion. Second, the need for prediction emerged from the learner as they observe the limitations of their current strategy. Lastly, the implementation of the idea relies heavily on spatial computation, which the programming environment is designed to support. The following section describes Greg's idea in detail.

4.3.3.1. The Triangle Idea

As Greg observed the video recording, he had noticed a triangle shape that is created when the pendulum car steadily oscillates from one side to another. The triangle consists of the top tip of the pendulum which is the least moving part, the car which moves rapidly from one side to the other (see the figure below). Greg thinks that this triangle is rather symmetrical. Thus, if we know the location and the angle of the pendulum car when it is at one corner, we can calculate the approximate location of the opposing corner, which is roughly where the car should move to. Thus, he can now predict where the car should go next.

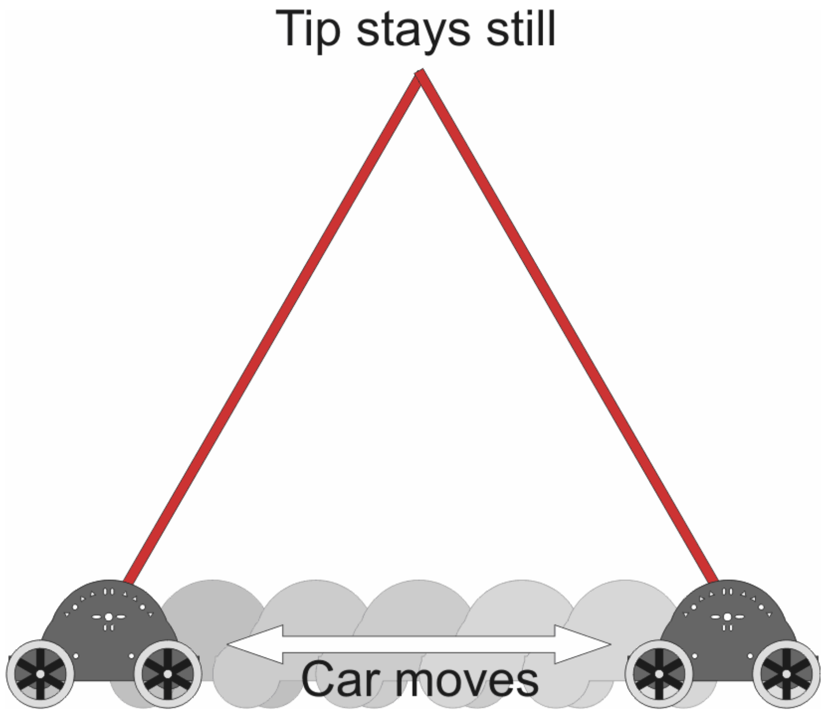


Figure4-28: The triangular shape noticed by Greg while observing a recorded session.

Although this strategy sounds promising, Greg realizes that the actual program will need to take account for the state of the pendulum (angle, speed) at each time interval as well. In other words, the end product is a mix of the prediction model, which guides the overall behavior, and the reaction model, which makes small corrections in response to the current situation. Greg's plan is to first create the prediction model to see how it works and then build the reaction system to add the required adjustments.

Since this triangle idea is highly visual and geometric, the spatial computing paradigm was highly useful in the implementation process. It is a three-step process as follows:

(1) A turtle is placed at the top tip of the on-screen pendulum. A "stickto" command is evoked to tell the turtle to follow the tip of the pendulum.
(2) The X coordinate of the tip-turtle is mapped onto a second turtle which is horizontally aligned with the pendulum-car turtle. This process places the second turtle at the bottom center of the triangle.
(3) Since the distance between the car and the second turtle is half the width of the desired triangle, this distance (D in the figure below) is used to set the X coordinate of a third turtle, which is the X coordinate of the second turtle plus the half-triangle length (minus if angle is less than zero). The position of this third turtle is the desired position for the car.

A final component is to tell the program when to make a prediction. In this case it is done only when the speed vector is inverted (that is when the car is at a corner of the triangle).

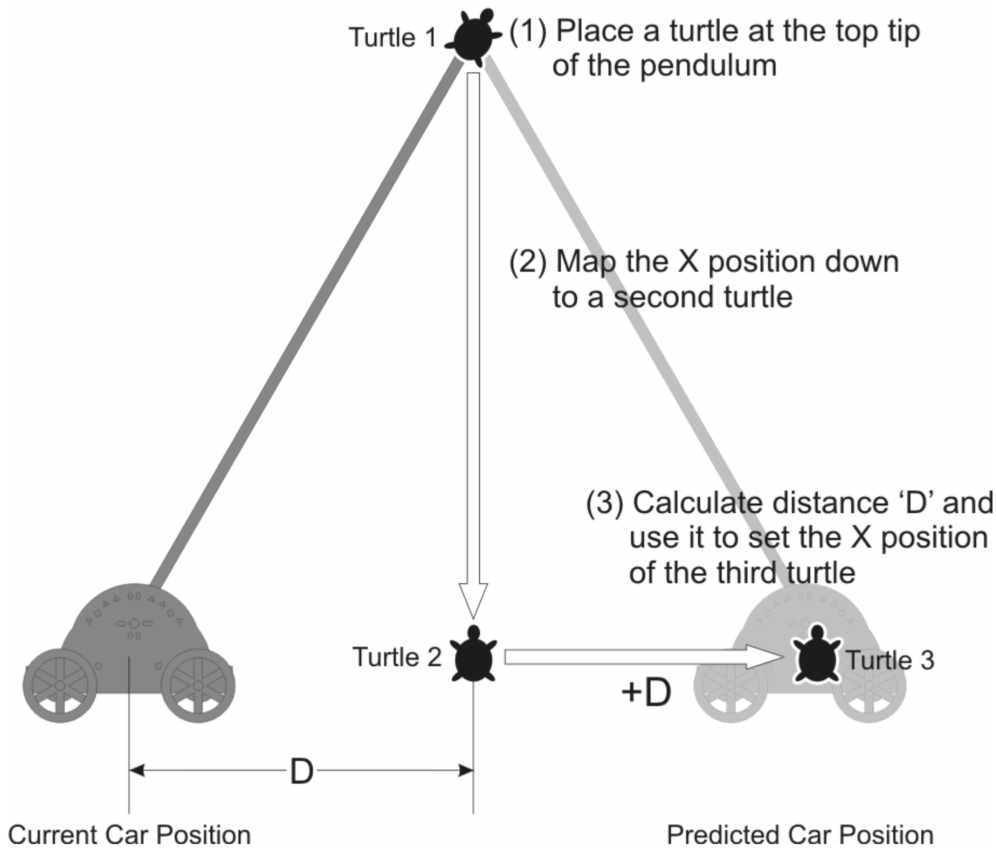


Figure 4-29: The three step process used to approximate the position to which the pendulum car should move.

4.3.3.2. Results of the Initial Prediction System

The prediction strategy required the pendulum car to be in a specific configuration to kick start the correct prediction. This requirement made it a bit harder to get the system going. But once it got into the correct back and forth rhythm, the pendulum car worked noticeably better compared to what we had before. This improvement was not so much reflected in the time the car could hold the beam, which was four seconds. But Greg commented that the car appeared to have more control of the beam. "It seems to know more about what it is doing than before."

4.3.3.3. Overshoot and Compensation

The results of the prediction system were logged on the computer and a graph was plotted as show in the figure below. When Greg looked at the graph, the first thing he noticed was that the pendulum car tends to "overshoot" the predicted position. Through some discussion with the researcher, we hypothesized that the momentum of the car is the cause. That is, in the physical world, the pendulum car cannot immediately reverse its motion. It has to gradually slow down, stop, and then reverse.

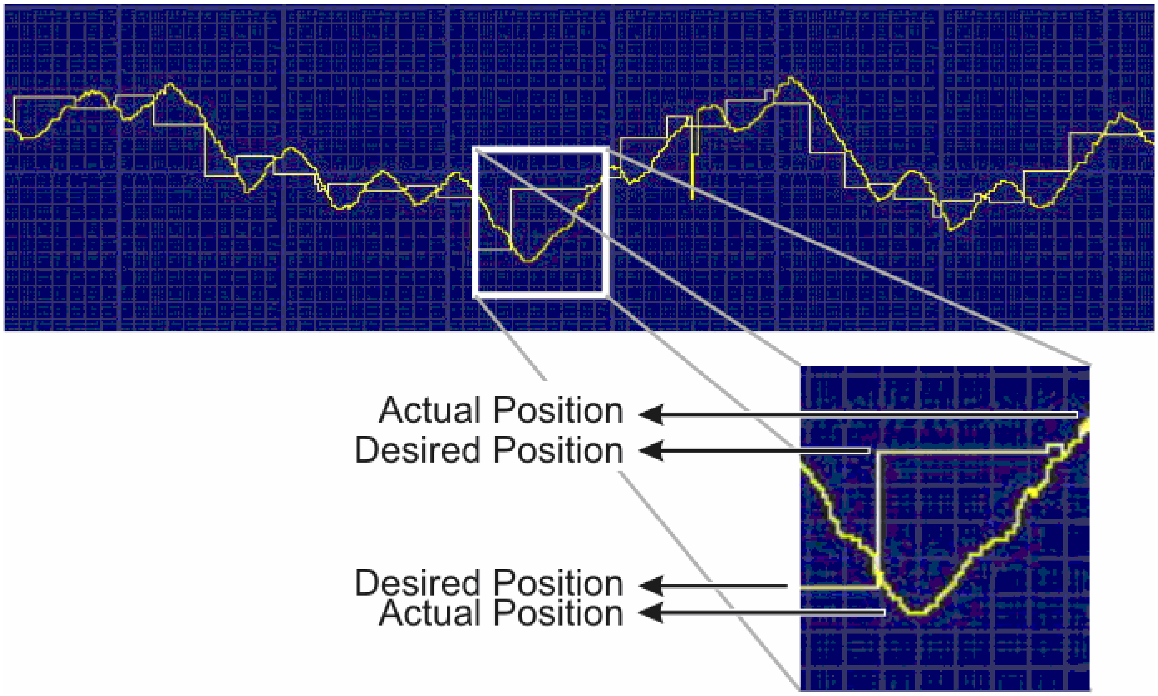


Figure 4-30: A graph showing the desired (predicted) position verses the actual car position. The blocky line shows the desired car position calculated by the prediction system while the curvy line shows the actual car trajectory. It can be seen that the car always exceeds the desired position causing what Greg has identified as an overshooting problem.

If this hypothesis is true, we should be able to "compensate" for this delay. We decided to decrease the threshold in the program so that it tells the car to reverse its direction slightly before it reaches the predicted position. After some tweaking, we were able to eliminate the overshooting in the graph and the pendulum car was able to work for two seconds longer (six seconds all together; an all time high). Thus, the hypothesis was concluded to be true by evidence.

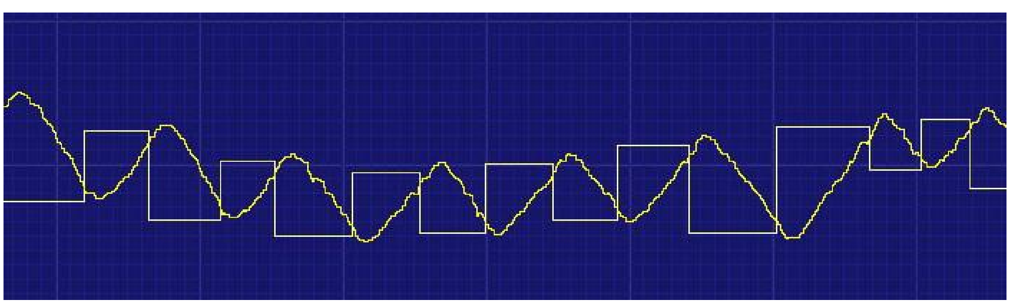


Figure 4-31: A graph after Greg compensated for the car's momentum in his program. It is clear that it has significantly reduced overshooting.

4.3.3.4. When Prediction Fails

Another observation made was that the prediction was not always correct. The predicted distance was often too far; meaning the pendulum's angle was too large and beyond recovery when the prediction was made. Greg and the researcher hypothesized that the prediction model works well only when the car and the pendulum moves in sync and forms the triangular shape as discussed earlier. But the pendulum is bound to veer off course. This error can be sudden or gradual but it eventually breaks the desired rhythm and starts falling. Since Greg's current model does not monitor the pendulum's angle between predictions, his program cannot deal with this situation.

Greg's reaction to this problem was to make more frequent predictions. If the previous predicted distance was not met in a given amount of time, the prediction will be updated. Thus, a prediction is now made either when the previous prediction was met or when it expires. The result of this approach was mixed and inconclusive. Observations showed that the frequent updates reduced the number of excessively large predicted distance, but it did not prevent the beam from falling. The behavior of the system appeared to be the same as before.

4.3.3.5. Mixing of Prediction and Reaction Systems

When the problem of the prediction strategy persisted, Greg brought back a portion of his previous reaction approach. That is, in addition to a prediction, he wanted the system to respond to the pendulum's angle as well. The more the beam falls one way the faster the car should move in that direction to catch up.

Greg averaged the outputs of his prediction and reaction methods and used it to control the motor. Averaging the values caused a significant change in the behavior of the system. But because the output of each method was easily manipulated by resizing the line objects, Greg was able to tune the system and observe different results. This tuning process mainly determines the significance of on parameter over the other. See the shaded box below for a more detailed explanation.

Mixing values using line objects

The pendulum car’s motor power ranges from minus eight to plus eight where the sign determines the spin direction. Greg’s final motor power output is an average of the two motor power decided by the prediction and the reaction systems.

- **The prediction system.** The motor power is proportional to the distance between the car and the desired position.

- **The reaction system.** The motor power is proportional to the pendulum's angle. The more the angle (the pendulum is falling) the more the motor power to move the car in the falling direction.

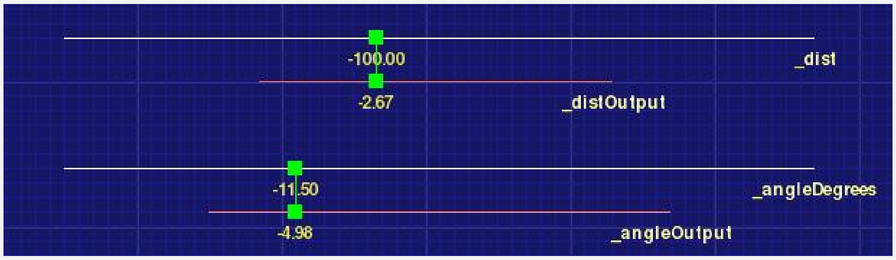


Figure A: Shows the line objects that are used to control the final motor output. The size (length) of each line affects the output.

In the figure above, two pairs of lines are presented. The function of each pair is to convert value from one range to another. The white input lines (_dist and _angleDegrees) are the parameters governing the motor power. The value positions (the green dots) on the paired red output lines (_distOutput and _angleOutput) are linked to the white lines so that they move together. From the above figure:

- A pendulum angle of -11.50 degrees is converted to a motor power of -4.98
- The distance of -100 (distance to the desired position) is converted to a motor power of -2.67.

Thus, the final motor output would be `(-4.98) + (-2.67) / 2 = -3.825`

The size of the lines will affect the final motor output power. For example, Greg found from experience that the motor power output from the prediction system needed to be very large even with a small distance. So, his line objects looked like the following figure:

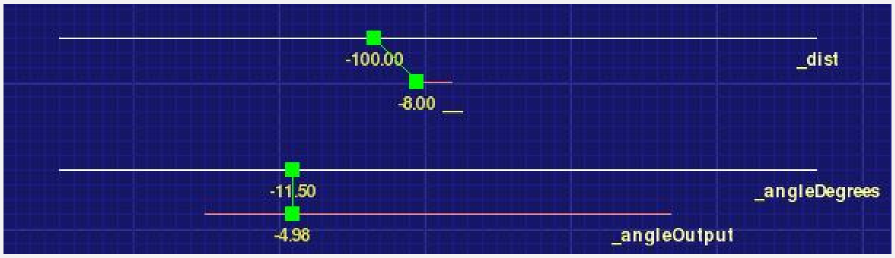


Figure B: Shows Greg's modification to the output of the prediction system making the output extremely sensitive. Note that the value on the output line never exceeds its boundaries. That is why the dots on the two lines are no longer aligned.

Now the _distOutput line is very short making it very sensitive. The distance of -100 now converts to a motor power of -8, which is the maximum power. The average output power is `(-8)+(-4.98) / 2 = -6.49`. This demonstrates how one can define how much influence one parameter can be on the final output.

4.3.3.6. Entering into the Realm of Multi-Variable Systems

Greg has started to use more than one variable to control the pendulum's behavior. This realm of multi-variable system can be complex and the output is often counter intuitive. In traditional control engineering, multi-variable control loops requires careful analysis and multiple iterations of tuning[6]. Understanding the tuning process is not straightforward. There have been extensive research and formalisms in this field to understand how to accomplish the desired system behavior (See [Skogestad and Postlethwaite, 2005] to get an idea). Although most of Greg's actions were based on trail-and-error, I was trying to help him gain some insight that could help him better understand the system's behavior as well. The following is an example.

As described earlier, Greg has used the distance to the predicted position and the pendulum's angle to determine the car's speed. He combined the two variables by calculating an average. When applying this method to the car, he observed that the car became less responsive and did not move very fast. To better see what is going on, we plotted a graph of the related values as shown in the following figure.

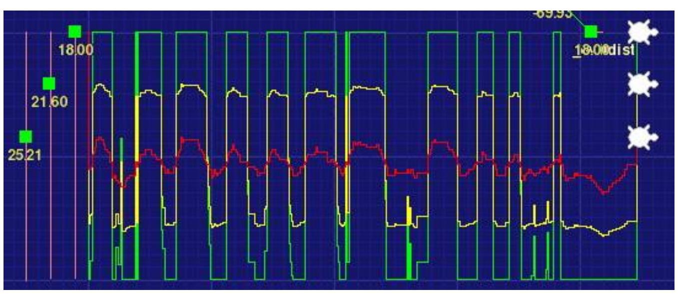


Figure 4-32: a graph showing how variables are mixed and the resulting output. Red (in the center) is the pendulum angle, green (the large blocky line) is the distance to the predicted position, yellow (in between the first two) is the average used to control the motor power.

[6] A PID control system is a good example. Three components measured from the target system; Proportional, Integral and Differential; are mixed to create a control signal. Tuning is an integral part of a successful PID control loop. Many times tuning is a trial and error process.

What Greg observed from the graph was a side effect of making an average. Since the value of the pendulum angle line (red in the center) is small, it keeps the output value (yellow in between the two other lines) relatively small. When compared to the distance-to-predicted-position line (the green large blocky line), which was previously the single variable used, it is clear that the average is always smaller and thus the car would move slower.

When Greg saw the behavior of the average graph and understood what was going on. He tried to increase the value of the output by increasing the influence of the angle line (red) while reducing the influence of the distance-to-predicted-position line (green). He did this by resizing the length of the line objects as described before. The following graph shows the new behavior. This time the value of the two input variables are very close to each other making the average almost identical to either input. This time the results were similar to the single variable approach. It did make the car move faster but the there were not a performance gain from using two variables. It was rather puzzling that the two input values could be as identical as observed in the graph. After some thought, we realized that both inputs were derived from the same variable –angle. After all, the distance-to-predicted-position is proportional to the pendulum's angle.

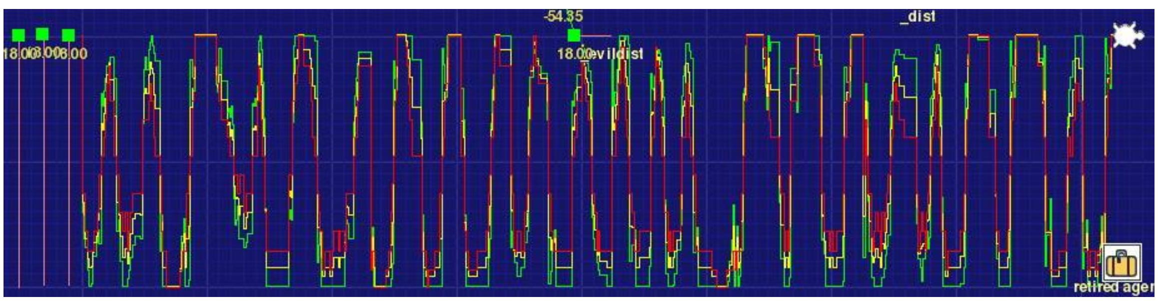


Figure 4-33: another graph showing the two input variables and the average output. Notice that the value of the two inputs are vary similar making the output values very similar to the inputs as well.

Although this discovery was a setback for Greg in terms of accomplishing his goal, the process was valuable as it allowed him to realize the problem. He may not have gained this insight had he relied only on trail-and-error.

4.3.3.7. Moving On and Taking a Break

As with the previous difficulties Greg has encountered, there has always been a new and promising alternative that emerges. This has become a normal process for how I work with Greg. This time, the particular alternative was to simplify the process by making a prediction of the pendulum's angle instead of the car's position. It made sense because we have learned from the previous approach that the car's position is a derivative of the pendulum's angle. And since we can directly measure the angle, there is no need to go through the long process of figuring out where the car should be.

Using the angle is also more accurate. For example, if the car moves too slowly the pendulum could still be falling forwards even when the car reaches its predicted position. Monitoring the angle is more accurate as it is not affected by the position of the car.

Greg has also learned from previous experiments that prediction alone is not sufficient. The car needs to adapt depending on how things are going as well. The prediction model assumes the pendulum will follow a certain pattern. But how to determine if things are following the desired pattern was still unclear. Greg previously used a simple time-out approach to refresh a prediction if it was not met within a short time period. The approach had little success because a prediction is not valid unless it is made at the end of the movement cycle.

During this time, I suggested that we could use the pendulum's angular rate as the supplement variable. The following are the reasons I gave to Greg:

A. Angular rate can tell whether the pendulum is moving in the correct direction. That is, we can detect when the pendulum reverses its fall prematurely and make a new prediction accordingly.
B. Angular rate can also tell us how fast the beam is approaching the desired angle. This can help us prevent overshooting.

Although the reasons to use the angular rate as given to Greg, he appeared to relate to it well. Perhaps this was because the benefits I described connected directly to his previous experience. Benefit (A) was rather straight forward as he had to think about it before. Benefit (B) relates to the overshooting experience that he had also experienced earlier (See section 4.3.3.3). Although it was formulated differently, he did not show any signs of doubt or confusion.

4.3.4. Phase IV: Shifting to a New Project: The Hover Board

Even though we were all clear about how to implement the new idea for the pendulum car, we did not get very far. Greg at the time had become interested in a different project, which had emerged from a conversation we had about other things he could do after finishing with the pendulum. At the time Greg had already worked on the pendulum project for more than eight months, so I decided to let him take a break from it and work on this new project. I also wanted to see how well he can adapt to a new problem after his experience with the inverted pendulum.

4.3.4.1. Conception of the Idea

Greg initially came up with a few ideas but the one that stuck to him the most was about how to make a "hover board." This is a device that looks like the flying skateboard depicted in the movie "Back to the Future Part II" as shown below.

Figure 4-34: The hover board as depicted in the 1989 movie "Back to the Future Part II."

Since it is not possible to build an actual hover board, we decided to create a simulation instead. I worked with Greg to create a simple character sitting on a hover board. We added gravity pulling the board towards the ground. We then determined what variables were needed and made them available as shown in the following figure.

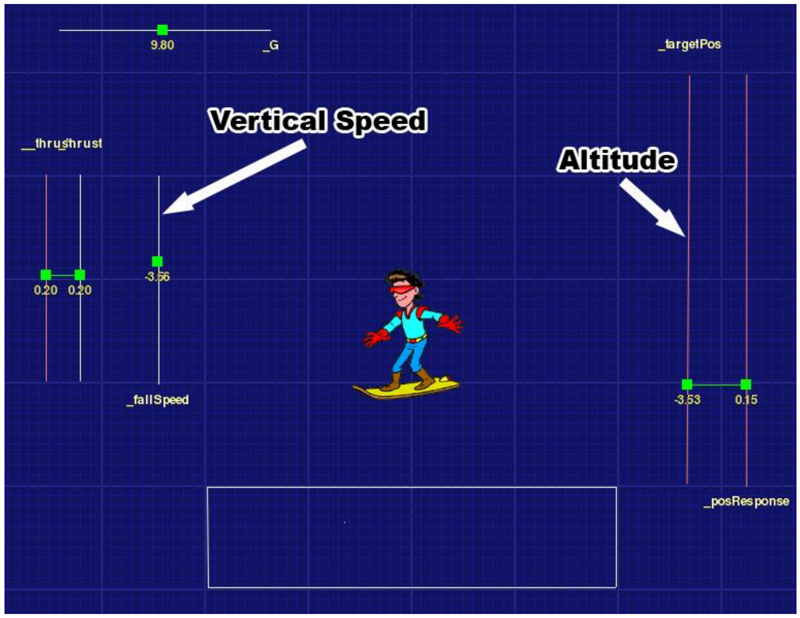


Figure 4-35: An overview of the hover board simulation. The main character on the board is pulled downwards by gravity. The board can thrust either towards or away from the ground. The board's altitude and vertical speed are provided through the vertical lines.

4.3.4.2. Greg's Fluency in Thinking about System Variables

It was rather clear to me that Greg had then developed an understanding of how a derivative (e.g. speed) in addition to a proportion (e.g. position) is a necessary variable in order to maintain balance. He asked for it even without my intervention. When I asked him why he thinks it is needed, he explained that he needs to know the direction to which the hover board is moving. For example, it is not necessarily a bad thing when the board is below or above the desired altitude as long as the board is moving towards it. We would not be able to detect these situations without knowing the direction.

Greg's emphasis on using speed to identify direction showed that he has picked out one particular aspect of the broader, and somewhat more complex, utility of a derivative. A control engineer may consider this as being naïve. But from this thesis's point of view, it does not mean his understanding of the subject was incomplete. Quite the contrary, he knew that there was more to speed than identifying the direction. For example, he uses the magnitude of speed in the program later when we continued. But these other aspects of speed were not yet as explainable. Speaking of speed in terms of direction, though not complete in any engineering standard, was coherent to him and it gave him a way to talk about a complex subject that may not have been easily graspable otherwise.

4.3.4.3. Introducing the Phase-Plane Idea

Although Greg knew the utility of both altitude and fall speed, it was rather clear that he was having difficulties figuring out how to deal with the two variables at the same time. He initially used a sum of both variables to calculate the thruster's power. But this approach led to the same limitations he had experienced before. Summing two variables does not yield an output that could be easily understood. He became frustrated when the hover board did not work and he was unable to see or articulate what was going on.

This was when I decided to introduce Greg to the "Phase-Plane" approach I discovered before with Albert and Anderson. I created squares that represent the possible combinations of altitude and fall speed. A turtle was then programmed to show the current phase of the system while the hover board moves around as shown in Figure 4-36 and Figure 4-37.

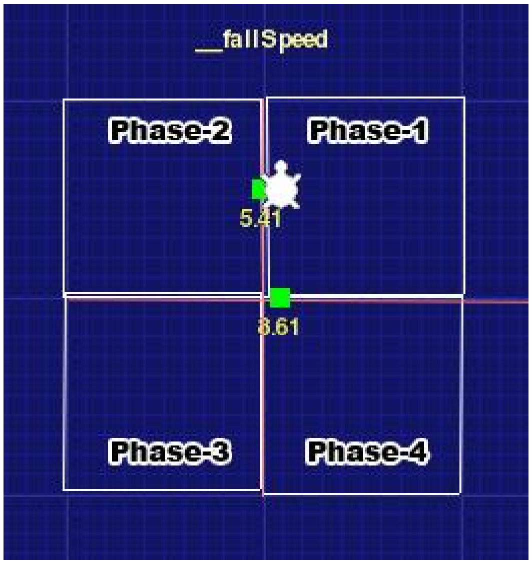


Figure 4-36: The Phase-Plane diagram that was introduced to Greg. There were four possible phases derived from the two variables under investigation. A turtle was used to show which phase the hover board was in at any given time.

The introduction of this Phase-Plane idea created a huge difference. First, the approach was not entirely new to him. He had utilized a similar approach of using shapes to detect an event before with the pendulum project, although it was limited to only one dimension. Secondly, the four phases gave Greg a straightforward way to create different behaviors he wanted (as shown in the figure below). He had the understanding from the beginning but before using the Phase-Plane approach he did not have a good way to express it in concrete terms.

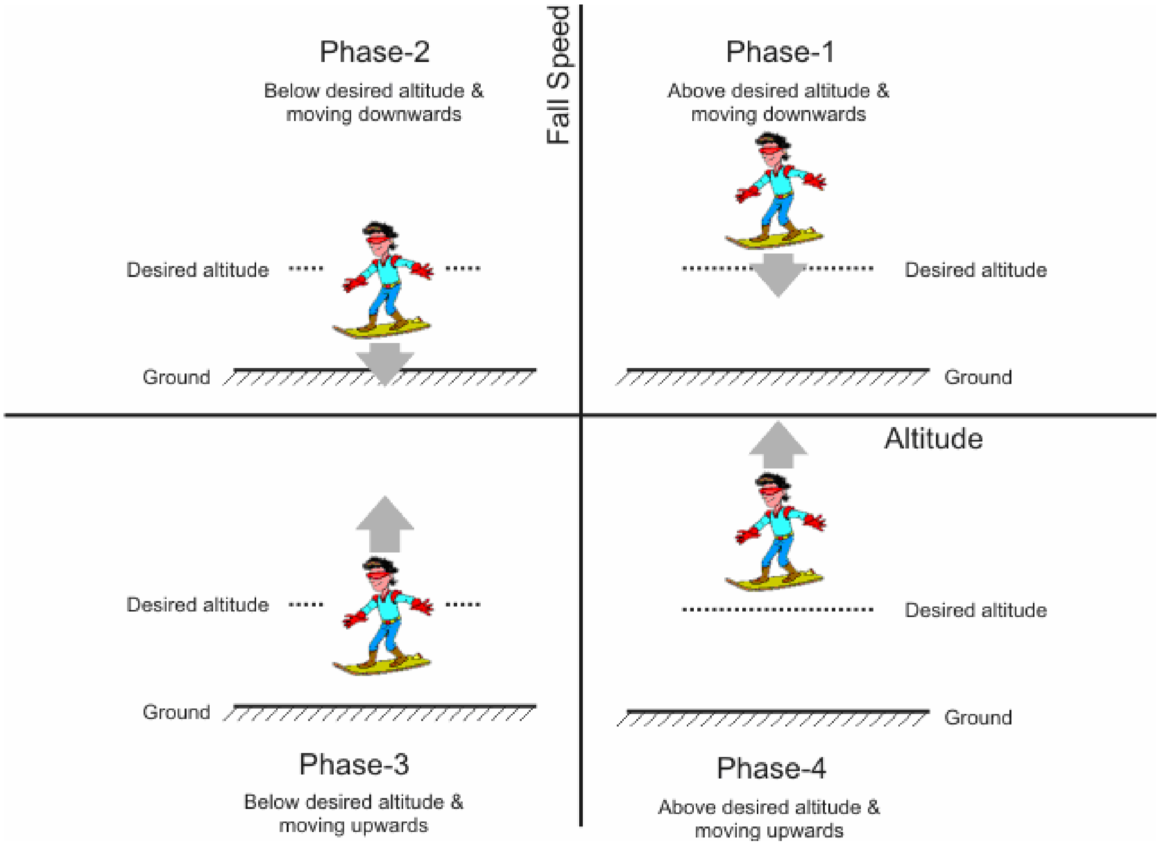


Figure 4-37: A graphical illustration of the four phases, which are the mix of two variables: Altitude and fall-speed.

With this new approach, Greg assigned behaviors to each phase as follows.

Phase-2: Fire the thrusters pushing the board upwards. The board is below the desired altitude. We must prevent it from crashing. The intensity of the thrusters is proportional to the falling speed.	**Phase-1:** Do nothing. The hover board is moving in the right direction. Let gravity do the job.
Phase-3: Do nothing. Same reason as phase-1	**Phase-4:** Do the opposite of Phase-2. The board is above the desired altitude. We should pull it back.

4.3.4.4. Using Shapes to Create a Grammar of Stability

Greg ran his program and watched how the turtle moved between the phases. Since the traveling turtle leaves a trail, Greg was able to observe its behavior. Greg's general strategy for what to do in each phase was quite accurate. But because of bugs in the implementation of his program, the behavior in Phases 1 and 3 was wrong. Doing nothing was translated into "keep the thrusters as is" instead of "turning the thrusters off." The hover board, thus, swung wildly and quickly went out of control. The following figure shows the resulting trail created by this program.

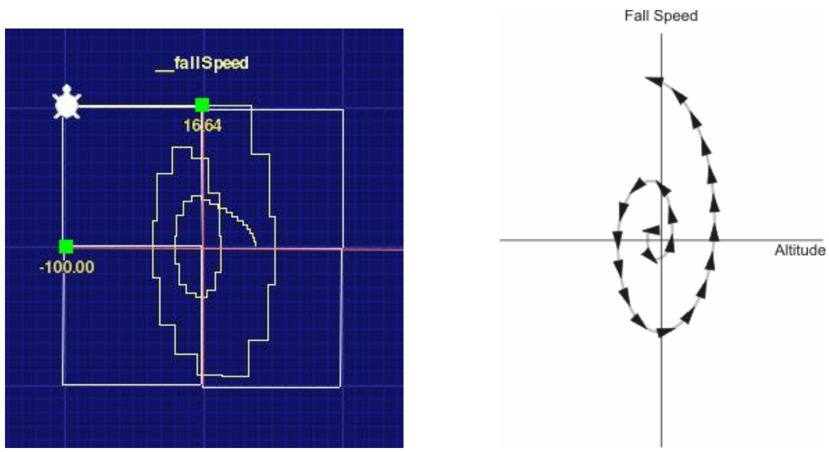


Figure 4-38: The Spiral. The outward spiral was caused by Greg's buggy program. The system failed to maintain stability.

After a few trials, Greg noticed that the trail in the Phase-Plane was an outward spiral. It is worth noting that Greg has now shifted all his attention to the Phase Plan graph and was no longer using the hover board character itself in the debugging process. His analysis of stability was based on the shapes drawn. For example he associated the term "big spiral" as a symbol of "getting out of control."

Because the graph was rather repeatable and consistent, Greg made an observation that the big spiral tended to fall off the edge in phase-1. The program in phase-1 was then given a careful analysis. This led him to discover and fix the bug. He actually enhanced it by not only turning off the thrusters but actually reversing them. With phase-3 still containing the bug, he observed a different behavior as shown in the next figure.

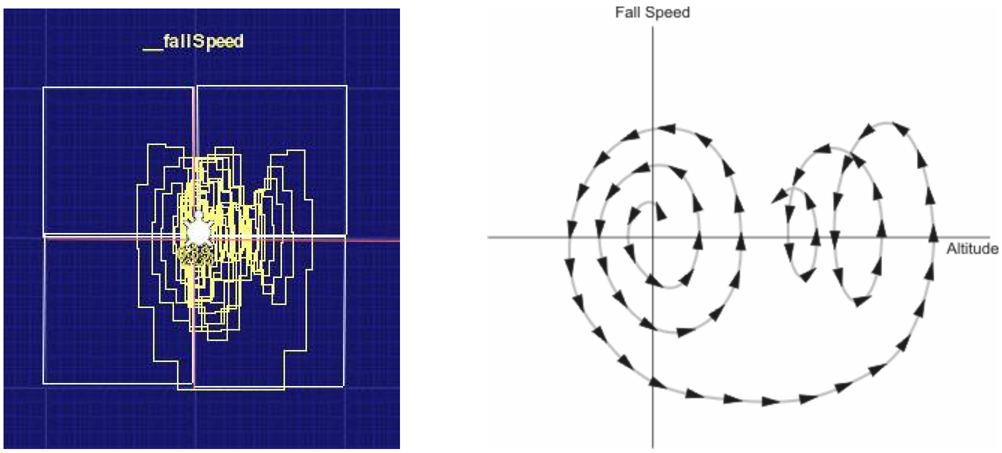


Figure 4-39: The Helix. The graph shows the trail after phase-1 (the upper right box) was fixed. The system behaved better but still produces occasional disruptions due to the remaining problem in phase-3, which still unnecessarily fires the thrusters. A helix was observed when the system was gradually recovering from the disruption.

Although the result was still a bit rough, the system did not fail. The behavior of this new system was rather intriguing to Greg as it occasionally produces a disruption that sends the hover board jerking off from the desired altitude. The system then gradually stabilizes itself before the disruption cycle takes place again. Greg names the motion while the system was trying to stabilize a “helix movement” as seen on the right most section of the graph in Figure 4-39

Later when the system was improved to eliminate the occasional disruption, a more stable behavior was observed and a “ring shape” was drawn in the Phase-Plane, as shown in the figure below. A large ring would indicate that, although the system does not get out of control, it still vibrates undesirably. Thus, a ring was associated with a stable system. But a large ring was unwanted.

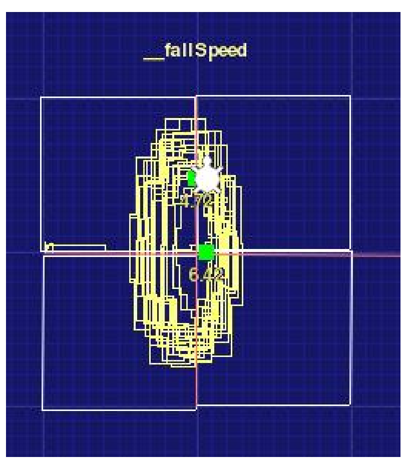


Figure 4-40: The ring. A ring shape was observed when the system stays in control. A large ring indicates large vibrations.

4.3.4.5. When Giving a Solution Can be Harmful

During the course of the project after Greg discovered the helix shape, I believe I made a mistake regarding his learning trajectory. Greg had come up with a few explanations and ideas to further investigate the phenomenon. For example, he thinks the program is more stable when it is doing the helix. Thus, he wants to find a way to keep the system in the helix stage. But because I knew that the solution was extremely simple, I did not give much credit to his ideas. Instead I decided to tell him the simple solution.

When Greg learned from me that the problem was easily solved by simply turning off the thrusters in phase-3, he considered the challenge complete and his enthusiasm had vanished. He no longer had the desire to continue working on the stabilization of the hover board.

Although there were other times when I believe giving a solution was a better choice (e.g. I was the one who told him to try the Phase-Plane approach), I believe doing so in this particular case did more harm then good. Had I not told him the solution, the circumstances would have allowed us to further discuss the underlying concepts. For example, I had planed to show him how the hover board could drift away from its desired altitude when the program does not take account for all the related parameters. But it was too late. I could not revive Greg's interest in the project. The opportunity had passed.

5. Reflections and Conclusions

This chapter will tie the observations I have made during the fieldwork to the theoretical discussions presented earlier. The chapter is divided into three sections. The first evaluates the idea of learning through observing body motions. It shows how the ideas have been expanded and how the tools have evolved to better support the learners' understanding of the situations involved.

The next section focuses on the domain knowledge that the students encountered along the way. Finally, I discuss the learning implications that became evident. I finish with some possible future directions.

5.1. Physio-Syntionicity: Learning by Moving the Body, Engaging the Mind, and Constructing with the Machine

5.1.1. Moving the Body: Learning by Reflecting on Recorded Motions

5.1.1.1. Recording Body Motions

The case studies have shown examples of how observing body motions while performing the desired actions can provide a rich resource for learners to develop ideas that they would then pursue. I have found that observing body motions contributes the most in guiding the general direction of the project. Strategies like the "gray area" (section 4.2.1.9 on page 57) or the "triangle" (section 4.3.3.1 on page 78) were clearly developed from body observations and had served as the overarching theme that governed the direction of their implementation.

After a strategy has been created from the observations, the learner could spend a long period of time in the implementation without referring back to recorded body motions. Thus, body motions were not tightly coupled with the actual implementation of the ideas. When students started to create programs, they were usually immersed in the mechanics of translating their thinking into programmatic terms. After some time and if the implementation becomes complicated they often loose sight of the larger scope of the project and they sometimes felt like they had reached a dead-end. This is when the observation of body motions, once again, could play a role in helping them to zoom out and advance forward in new directions.

For example, Greg had spent a long time working on the state-machine method. After some weeks, this approach become dominant and every new program he created uses this same kind of thinking. When he became stuck, he was not able to step back and think of other ways to go about solving the problem. That was when we decided to break the trend by revisiting the body motions. It allowed Greg to notice new aspects of the system and eventually devised the new prediction method, which was a completely different way of thinking from his previous state-machine approach. This kind of shift was greatly facilitated by Greg's engagement with the recordings of body motions.

Another aspect observed during the fieldwork was that the implementations of the students' ideas were not necessarily tied to the actual motion being observed. This is different from fields like Computerized Dynamic Posturography (CDP) where the model created must conform to the data gathered from the observation. The observations made by the students in this work were mostly inspirational. Once an idea was generated, the implementation was no longer constrained to the observation. Greg saw a "triangle" in the body recordings, but the strategies that he came up with to make a triangular motion was open-ended and was completely independent from the recorded data. That is, his triangle does not have to be the same as the triangle in the recorded data.

5.1.1.2. Recording Autonomous Robot Motions

While the recordings of body motion led to the development of high-level ideas, the recordings of motion under autonomous control have been found to play a bigger role in the implementation process. When programs failed, it was useful to analyze the recorded motions to figure out what went wrong. Its use ranged from simply playing back the recorded actions to creating graphs to compare two or more variables.

Thus, although the tools and methodology used to record and playback both body- and autonomous-motions were exactly the same, the two kinds of recordings had served very distinct roles.

5.1.2. Evolution of the Medium: Negotiating Meaning When the Body Meets the Mind

The evolution of how states were defined and how reactions were generated was the largest design change that took place in this work. It represents the result of the iterative design process employed. Here is a summary of the iterations:

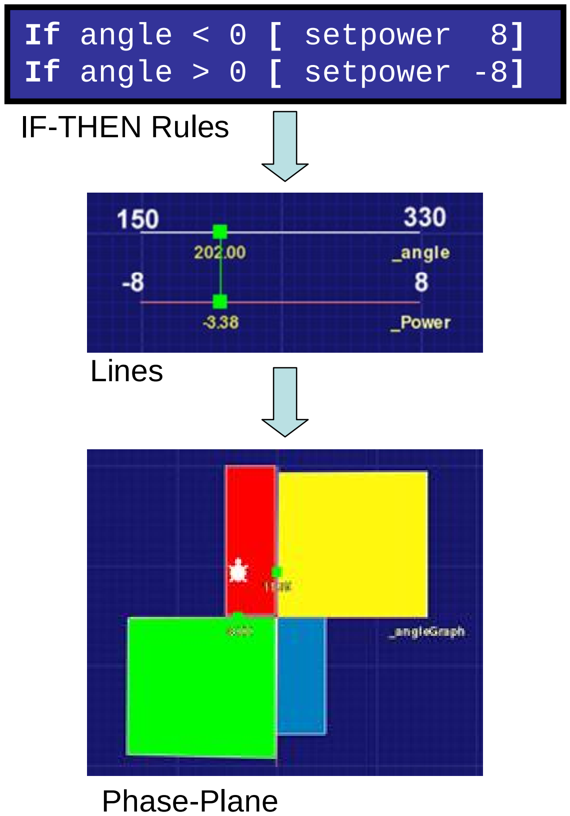


Figure 5-1: The representation of state evolved from simple IF-THEN rules (top) to the researcher's Lines approach (middle). The Phase-Plane controller (bottom) was a result of the researcher's observation of the students' thinking combined with the researcher's knowledge about what is needed.

Phase I: The IF-THEN Rules Approach

[See details and an example in section 4.2.1.2 on page 47.]

This was the approach that all the students used to implement their initial ideas. It allowed the students to quickly construct a program that matches their thinking. Despite its simplicity, this approach did not scale well. When rules become more complicated, tuning the numbers became tedious, and it was difficult to associate the rules with the robot's actions being observed.

Although there are ways to solve some of the above limitations of IF-THEN rules[7], I did not want to focus only on the textual representation. Since a significant part of this work has to do with visual representations of physical objects, I wanted to explore ways to benefit from properties of these 2D visual objects.

[7] For example see the Flogo programming environment [Hancock, 2003]

Phase II: The Lines Approach

[See details and an example in section 4.2.1.4 on page 49.]

I introduced the line approach as a solution to the problems of the student's IF-THEN rules. Although it worked better in controlling the robots, it did not resonate well with the learners. This was clear when the program became more complex. As a result, one group quickly went back to the more comprehensible IF-THEN rule approach (see section 4.2.1.9 on page 57) while the other became stuck, which was a setback to the development of their thinking (see section 4.3.2.1 on page 73).

At that point, it was clear to me that there was a conflict. From an engineering point of view, I had put an emphasis on making the control logic produce a smooth and proportional output. But to the students, it was more important to keep the methodology graspable by making the cause and effect (in the program) discrete.

Phase III: The Phase-Plane Approach

This approach was essentially the result of how I negotiated between my perspectives of what was needed and the observations I had made of the students' way of thinking. It started off as a simple way to visually define regions of a single variable. Using rectangle objects to define the regions allowed students to easily resize and thus modify the regions. Later, the idea was expanded to work with two variables. It allowed the four possible combinations of the two variables become observable and straightforward to understand. In essence, it was an approach that made the control states both practical and comprehensible.

I later discovered that this approach closely resembles a well known engineering method called the Phase-Plane Logic Controller [Wie, 1998]. This confirmed that the resulting direction of the design process was both valid and functional. It would have been difficult, if not impossible, for anyone to foresee that a Phase-Plane Controller would work well with children without spending time learning about how they think.

The implementation of the Phase-Plane Controller in this work is just the beginning. I have no doubt that much more can be done and many other topics can spin-off of it. But whatever the possibilities are, involving students in the process will remain the key aspect. Also, I do not have to be the only person creating these new additions. The Phase-Plane Controller was created using simple components (e.g. the turtle, shape, line objects) that are readily available in the programming environment. Thus, instead of only using a Phase-Plane Controller, the students were also involved in its construction as well. So, building tools was part of the project experience and modifying or improving them take place naturally. For example, Albert and Anderson added a technique to produce a smother motor output (see section 4.2.2.2 on page 61). Greg applied the phase-plan controller to think about stability (see section 4.3.4.4 on page 91).

5.1.3. Construction with the Spatial Computing Paradigm: A Computational Means for Human Comprehension

In addition to state definition, SCP was also meant to explore how many other operations could be performed using familiar properties of 2D objects. An extensive description of examples can be found in the appendix. Here I will focus on one example that best highlights the character of SCP.

In section 4.3.3.1 on page 78, I have described how Greg implemented a prediction model where the system would know where the car should move to when the starting condition is known. The figure below is a replica of Figure 4-29. It shows the approach taken by Greg, which accomplished the calculations by using the spatial properties of the on-screen objects, e.g. aligning and repositioning turtles.

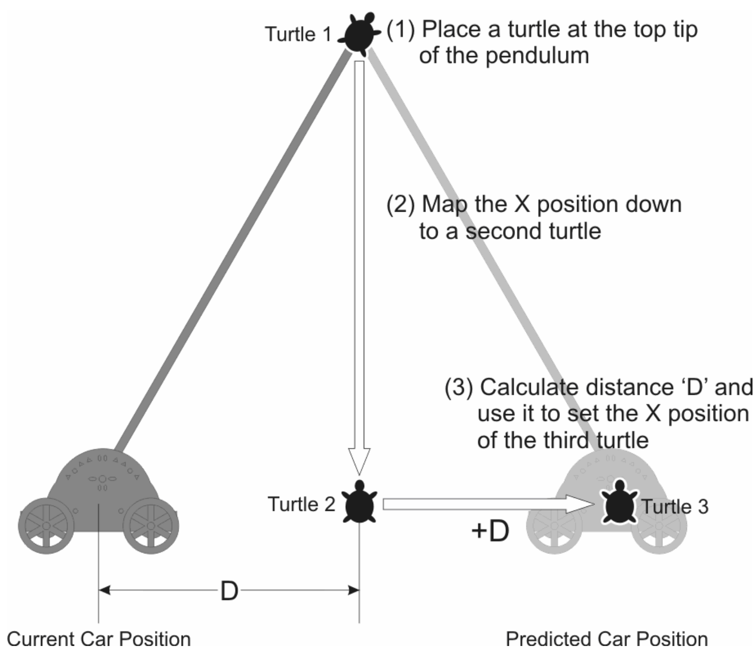


Figure 5-2: The operations performed by Greg to figure out where the car should move to given the starting position.

Greg's operations could have been implemented with a formula like the following:

```
Desired_X_Cor = Car.xcor + 2*(Pendulum_Len * cos (angle))
```

The operations would have been computationally more efficient with this formula than the spatial computing approach. However, Greg's SCP approach was connected to his experience and comprehensible to him. In addition, the speed of modern computers allowed the spatial computing approach to perform the task without significant delays. These are two different but viable ways to perform the same calculations. One is based on a formulaic representation while the other is more on spatial and computational forms.

This pluralism in thinking about problems in this area is not new but Greg's alternate SCP approach has never been as accessible to learners without the computational environment like the one demonstrated in this work. Before J.C. Maxwell's rigorous analysis of control systems in mathematical terms, engineers have invented mechanical means to handle such control systems [Lewis, 1992]. This goes all the way back to the float regulator for a water clock created 300 BC by the Greek Ktesibios. This mechanical approaches share significant resemblance to the actions involved in the Spatial Computing Paradigm. SCP allows learners to manipulate on-screen objects (e.g. turtles) in ways similar to physical objects. But since they are virtual and programmable, their behaviors are easily manipulated while still being precise. This allows SCP to be flexible enough to be used as a general purpose programming methodology.

Although the required calculations for this task can be accomplished both with a formulaic and computational means, each method has a different character and educational virtues. Formulaic analysis is the canonical medium used for this kind of operation while SCP approach is more progressive. The SCP approach often falls into a reasoning category associated with being less formal and relying on naïve or less precise reasoning processes, which cannot be taken too seriously. However, Greg's example demonstrates that computational representations such as SCP can, in fact, be quite formal.

This is not to say computational means are better or worse than the formulaic approach. What I am arguing against is a common mindset that views formulaic thinking as the only means deserving serious consideration. SCP is an example of a new form of thinking that this work has demonstrated its viability in the area of control engineering. Thus, alternate means to represent ideas should be treated at least on-par with the canonical formulaic ones.

The reason we should care about new representational forms like SCP is because of the quality they have that promotes learning in ways that cannot be done with paper and pencil. For example, the SCP approach was greatly successful with Greg because it connected well with his existing experience. Doing the actions by using turtles was straight forward and the results were directly observable. Greg could see the involved turtles move and it ensured him that his program is doing what he wants. Although it is true that Greg's fluency with the programming environment was a big advantage for him, the ideas that were used in the calculation process are based largely on our common understanding of objects in space. Sticking a turtle to the top tip of the pendulum is like gluing two objects together in the physical world. Aligning two turtles is something we could imagine doing by hand. The computer allows the action to be practical by doing it repeatedly and accurately. It is this quality that makes computational forms such as SCP worth paying serious attention. It opens up new learning pathways that highlight ***human comprehension***.

5.2. Domain Knowledge: Powerful Ideas in Balance Control that Emerged

The following is a summary of the important domain knowledge that came up during the field work.

5.2.1. The Role of Rate or Speed

Rate or Speed is one of the basic concepts in control engineering. But it was clear from the field work that students started off not being aware of how this parameter played a role in balancing the robots. Although the term was familiar and they may have learned about it in school, putting it into context was not straight forward.

Learning to appreciate the role of speed was the main storyline of the case studies. Each group had their own developmental path. Albert and Anderson became aware that using only the pendulum's angle was not sufficient after spending some time observing their body motions. Even with this awareness, my explanation about angular rate still did not stick. Instead, they developed their own definition of speed by using the speed of the moving tip of the pendulum. Although their approach may not be as efficient as mine, it made more sense to them and it still allowed them to move forward and build an awareness of "speed" as an essential factor in balance control.

Greg, on the other hand, took a longer route. He was not easily convinced that the pendulum's angle alone was not enough. Although at a conversational level it seemed like he understood the role of angular rate, it somehow became irrelevant when he put his ideas into practice. The theory in discussion was not connecting to the ideas in practice. However, he eventually developed an idea that we need to predict when the pendulum will fall so that the robot can compensate in advance. And, though many trails, he found that this prediction could be best done using speed. Although this process was long, it allowed the idea to really stick. This was clear when he moved on to work on the hover board project. Speed played a central role and it became part of his thinking right away.

5.2.2. Managing States

This topic closely resembles the ideas in Finite State Machines (FSM) which models a system's behavior by giving it a finite number of states [Arbib, 1969]. A set of variables or parameters determines how the system shifts from one state to the other. The way in which all the students translated their ideas into a computer program resonated with this approach. Simple IF-THEN rules were used to determine what state the system was in and what actions to take for each case.

The method evolved over time as shown in section 5.1.2 and we shifted to a more graphical representation with the Phase-Plane approach. This allowed the students to better think about managing multiple variable systems. The current state and its transition were observable through the graphical representation. The history or trajectory of the state transitions was also visible through the use of the turtle object, which can leave a trail showing its path.

After the Phase-Plane approach was introduced, the students become fluent in thinking about states and how a state changes when other parameters change. This was observed through their verbal explanation of the situation, the improved functionality of the robot, and their ability to apply this method to other similar projects.

At the moment, the number of states is still limited to four as the students have been focusing mostly on only two variables. But I wish to further investigate how this method could scale to support more complex state definitions when more variables must be considered. For example, multiple variable pairs could perhaps be defined and the result of each could be combined in a four by four grid.

5.2.3. Stability in Balance Control Systems

Stability is an important topic in balance control. It allows one to understand how different parameters contribute to a stable or unstable system. Multiple engineering formalisms exist that allows one to study and analyze the stability of a system. Nyquist Stability Criterion, Pole-zero Plot, and Frequency Analysis are notable examples. However, Greg encountered this topic of stability through a simple observation. He noticed that the turtle which was used in the Phase-Plane model was leaving trails that looked different depending on the situation. He identified different shapes that could reflect how stable the system was (e.g. an outward-spiral shape is unstable and a ring shape is stable). This approach was by no means accurate or complete when compared to any engineering standard, but it allowed Greg to start thinking about this complex subject and doing so in a way that was meaningful to him.

Making observations of nature's behavior is always a fundamental approach scientists use to construct new understandings. It has often led to major movements in the scientific community. The discovery of the butterfly effect by Edward Lorentz is a good example. It developed from a simple observation of how small changes could later yield significant changes in a system[8]. This idea has led to the development of chaos science as a new discipline [Jørgensen and Müller, 2000].

---

[8] In 1961, the mathematician and meteorologist Edward Lorentz was trying to predict the weather by running computations on various data. At one point, he wanted to re-run the calculations but he decided to take a shortcut by manually entering some numbers from his previous runs. He discovered that the results were entirely different from what he had before. He was puzzled as the results from a computer should be repeatable when the inputs are the same. He later found that some of the numbers that he manually entered were slightly rounded off and these slight changes had actually rendered a significant impact on his results. This led him to believe that it is not possible to predict the weather over a long time period. A slight change in weather conditions could yield a dramatic change over time. This observation seeded the development of chaos theory.

Thus, although Greg's observation is simple and crude at the moment, it can seed his interest and thinking of this topic in a similar way that has allowed Lorentz to later make such a huge impact on his understanding and to the scientific community.

5.2.4. Making Predictions

Greg had invested a considerable amount of time on his prediction model. Although his thinking was plain and simple, it did reflect the principle of an important topic in a traditional control engineering context. He first noticed that people tend to balance a pendulum in a way that resembles a triangular shape as shown in Figure4-28 (also shown below). Greg's idea was then to make use of this information by incorporating it into the control model.

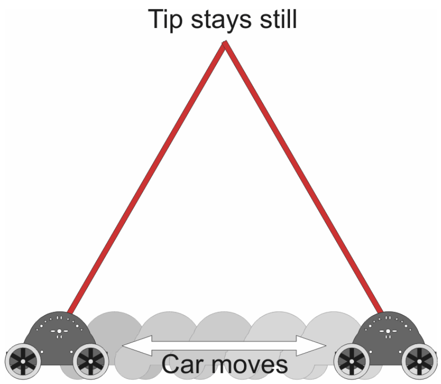


Figure 5-3: The triangular shape observed by Greg as shown in Figure4-28

In control engineering, the more we know about how the system should behave the better we can create a model to control it. This topic can relate to many control concepts such as Gain Scheduling and System Identification but the most relevant topic is perhaps the Model Reference Adaptive Control (MRAC) [Egardt, 1979]. MRAC is an adaptive control method commonly used when there is a good understanding of the plant and the performance requirements. This allows an engineer to define a reference model and create a control mechanism that can change the system properties and dynamics to be as close as possible to the model [Ioannou and Sun, 1996].

At a conceptual level, Greg's idea is essentially what the MRAC system is about. He had identified a pattern (or reference model) that the robot should follow (or adapt to). However, a concept like MRAC is typically expressed through means that rely heavily on mathematical forms that are significantly different from Greg's approach. While a classical MARC model would focus on defining algebraic and differential equations, Greg's computational approach focused more on creating computer programs that mimic the thinking that he had in mind. The focus here is not on comparing how better or easier one method is over the other. Rather, this case has shown how computational technology has connected an advanced engineering idea in ways that was both meaningful and personal to the learner. It shows how concepts typically considered too difficult have now become within reach of a much younger and broader audience.

5.3. The Learning Implications

5.3.1. Proofs and Refutations

The case studies have shown clear examples of the so called learners' misconceptions. In the beginning, none of the students were aware of how speed or rate played a role in the system. And as a consequence, they created naïve explanations of the phenomena (e.g. that angle of the pendulum alone was sufficient). But more importantly, the case studies have also shown that simply telling them about the flaws in their thinking does not immediately fix the problem. When I pointed out to Albert and Anderson from the video recordings of themselves balancing the pendulum that there were cases that could not be explained by there simple rules, both students refuted the counter examples. It was not until later when the counter examples accumulated and became so overwhelming that they started to become more open to a shift in their thinking. The illustration in Figure 5-4 shows this process from the case study.

This is not a matter of bad teaching or poor learning. Rather, it reflects the natural process people take in advancing their thinking. Imre Lakatos describes how refutations against a new idea (or proofs in a mathematical term) is normal in the history of the development of mathematics [Lakatos, 1976]. His portrayal goes against a classical perception that advances in mathematics are a steady accumulation of established facts and proven ideas. This description can reflect the developmental process of other scientific disciplines as well. Einstein's special theory of relativity was not immediately accepted by the scientific community. It was heavily criticized and corroborated prior to its gradual acceptance.

Thus, when focusing on contrasting understanding by connecting to learners' experiential knowledge, it is imperative that the teacher understands the importance of this logic of proofs and refutations. Teaching is not about rejecting misconceptions and imposing the proven ones on the learner. Rather, effective learning and teaching is about helping the learner make meaning of the events in the world that at times does not fit their current model and expectations.

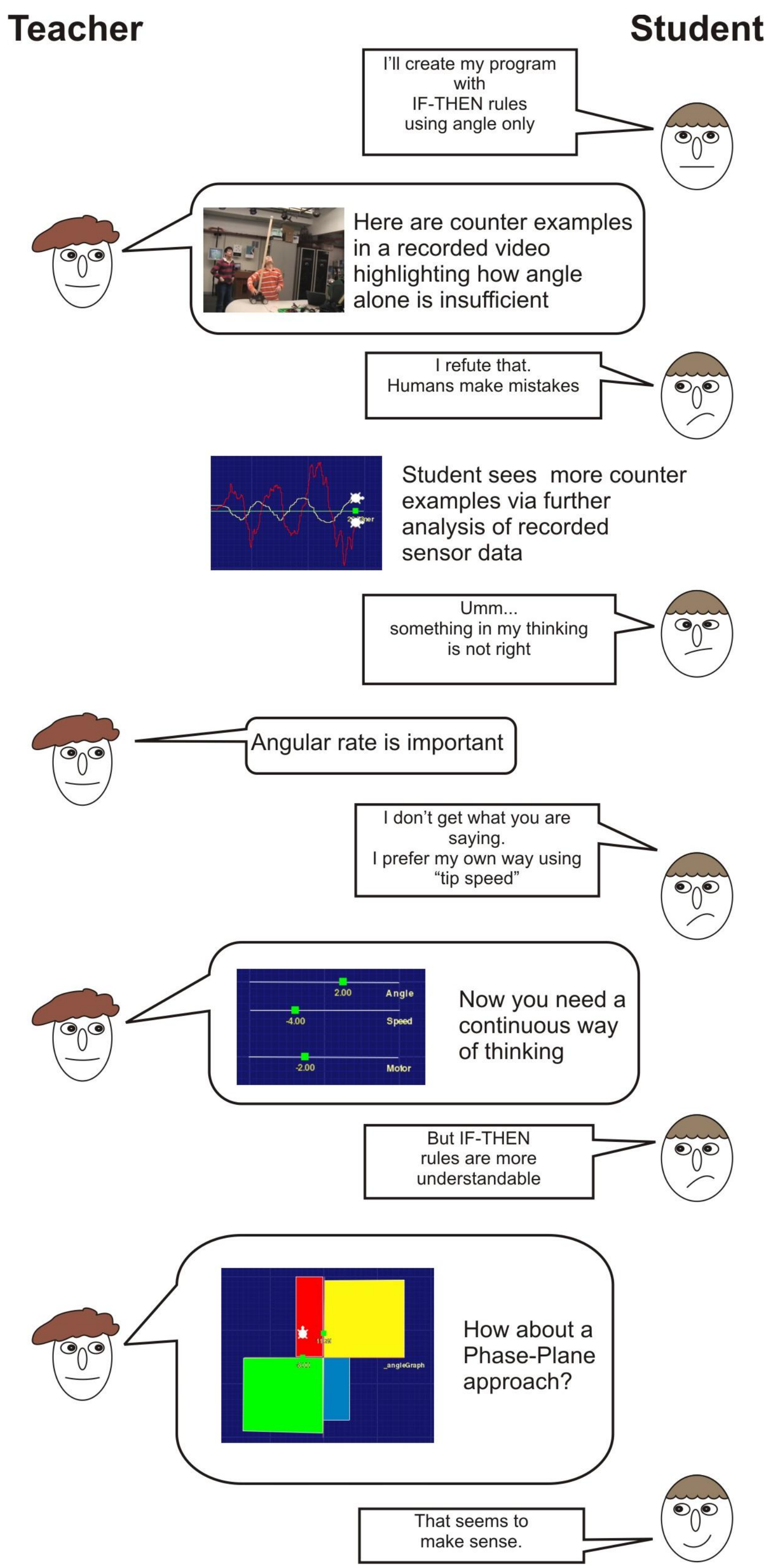


Figure 5-4: An illustration of the conversations that took place during the fieldwork with a student. It highlights the learning process where evidence of something different or better is not always immediately accepted by the learner.

5.3.2. Progressions in the Development of Understanding

The case studies have shown that when Albert and Anderson were first asked to think about their actions while balancing the pendulum (live and not through recordings), they were not able to see how their actions were actually different from their descriptions. They insisted that their thinking corresponded to their actions even though it was obvious to the researcher that they were not so!

The situation resembles many of Piaget's experiments with children. For example, when a fixed number of beads are shown to a child at an age of four, the child can be induced to say there are more beads when the beads are spread out compared to when the same amount are piled together. For an adult, the number of beads is an objective fact that does not depend on the spatial configuration of the elements to be counted. On the contrary, for a child who has not developed the conservation of number, spatiality prevails over numerosity. That is, the idea of numerosity, or quantity, does not hold in front of the conviction that "more spread out equals more" [Papert, 1980]. Most importantly, conservation cannot be learned though explanation, demonstration, or proof. Children eventually develop the correct understanding, but only when they are naturally ready, as they have built full structures around the individual concept. Also, once an understanding (of conservation or other concepts) has been developed, people usually forget the process they went through to arrive at were they now are [Watzlawick, 1984].

From this perspective, Piaget's experiment supports the Lakatos discussion. It exemplifies how expert advice may be completely alien or irrelevant to the world view of a novice. A student puzzled by the strange behavior of a gyroscope already knows that the gyroscope could stand upright despite his or her expectations. To receive a formal proof of the phenomenon is unlikely to make the student feel any better. What the learner needs in this case is not a better understanding of the subject matter but of him/herself [Papert, 1980]. This is accomplished by having time to experiment and build up the knowledge structures that would eventually help leverage his or her way to the next level.

I do not advocate that there should not be any form of expert advice. The case studies have shown how I have played a key role in providing guidance to the students. But it has always been a negotiation process that allowed the students to decide for themselves what to make out of the guidance being given to them. For example, I deliberately told Albert and Anderson the importance of angular rate in balancing the pendulum. They did not take my "expert advice" as is. Instead, they created the tip-speed approach instead, which they could better understand and relate to.

5.4. Future Work

Based on the findings and observations that have been made during the course of this work, there are several directions that are worth exploring further. The following is a comprehensive list.

5.4.1. Going Deeper

As a design research, one of the important goals is to investigate new ways that technology could help learners construct their understanding based on their own ways of reasoning. The most vivid result of this design process is the Phase-Plane approach where the learners' "discrete" ways of thinking of a complex situation can be accommodated, while still being flexible enough to yield a functioning system. The result was powerful. I believe there can be much more to improve this Phase-Plan approach. For example, the space shuttle version of the controller (section 4.2.3 on page 63) uses a concept called switch lines which would be interesting to experiment with children.

In addition, I believe the Phase-Plane approach is only one example of many more approaches that could be developed to help learners think about balance control. Working with more students and on a more diverse set of problems would make these other possibilities evident.

5.4.2. Going Broader

Many of the ideas that the students were able to encounter are actually fundamental to other areas outside of balance control. This is the basis of Papert's emphasis on "idea power." The power of the idea is felt when one sees how it can be useful in many situations. This discussion is actually quite prominent even in university-level control engineering education. There have been extensive discussion about how control should not be viewed as supplementary but rather a central theme for many other engineering topics (see [Bissell, 1999] for an example).

It would be interesting to see how students can realize and apply their experience from a control situation in areas such as social, biological, and economic systems. This can give the learner a much broader perspectives of systems thinking, modeling, and design.

5.4.3. A Deeper Investigation of Using Patterns and Shapes as Means for Computation

The possibilities of the Spatial Computing Paradigm in helping learners perform various operations have only been skimmed in this work. I believe there is still great potential for performing computation by tapping into our understanding of physical objects. Work in tangible media interfaces have revealed some examples in this direction. For example, in urban planning, when the design space becomes tangible and sharable, it enriches people's ability to express, reflect, and discuss their ideas [Ben-Joseph, 2001].

A second aspect of this direction is to investigate how far the approach can scale. While the processing power of today's personal computers has allowed graphical operations to be performed at an acceptable rate, there are other aspects to be considered as well. One example is to investigate how to mange the limited screen real-estate. Graphical representations take up valuable screen space. How can we make sure we can perform more complex operations while maintaining a coherent picture of the entire process? Such coherence would play an important role in maintaining the understandability of the operations. Would learners be able to comprehend their operations when the number graphical objects involved starts to grow?

5.4.4. An Epistemological Investigate of Discrete Versus Continuous Ways of Thinking

I have discussed at length in the case studies how the students tend to think of balancing actions in discrete terms (via if-then rules.) It took them some time to realize that the actual behavior of a working balancing system best contains properties that are also continuous. Their realization took place with the aid of the computation medium. Thus, it would be a viable investigation to better understand and explain this behavior. Is it the case that discrete thinking is just the way people are? Is it an approach to allow one to comprehend complex systems that are not graspable otherwise? Or is the discrete thinking just a matter of the materials we have to observe the phenomenon? If so, then how can the computational medium impacts the ways in which people perceive their balancing actions? What about other complex systems in general?

Finally, as a leap of faith, perhaps the outcome of the above investigation can be used to explain the way we think about learning as well. Is there a parallel between the way people think of balancing actions as discrete sets of isolated rules and the way people also often perceive education as a discrete process that can be segregated and compartmentalized and move learners from one state to the next in a linear fashion? Further more, can the shift in the medium allow for a different view that resembles a more continuous perspective where the various aspects of learning are considered more as a whole?

6. Appendix: The Spatial Computing Paradigm (SCP)

SCP is a methodology developed to perform computational operations using on-screen objects. It is developed based on the idea of simulating a physical object on the computer screen and giving them properties that enables them to be used as a computational tool. The goal is to provide access to computational ideas in ways that can connect to peoples' spatial intuition about objects in the real world. The following are case studies of how this computing model has been used.

6.1. Value Transformation

Throughout the course of the project it is common for students to manipulate numbers. For example, when reading the pendulum's angle sensor, the numbers obtained are raw values that do not correspond to the actual angle of the pendulum. This is due to the nature of the angular sensor which is usually a potentiometer that changes its resistance when turned. So, students often need to transform this resistance value into the actual pendulum angle before using them.

6.1.1. Basic Transformations

Let's pretend that the sensor values obtained for the angle -60 and + 60 degrees is 10 and 100 respectively. The Logo code to convert this number range would be like this:

```
SensorMin = 10
SensorMax = 100
AngleMin = -60
AngleMax = 60

angleDegrees = (sensorValue-sensorMin / (SensorMax - SensorMin)) *
               (AngleMax – (AngleMin)) + (AngleMin)
```

As an alternative, I have developed line objects that behaves similar to a value slider but utilizes on-screen properties to perform the same task.

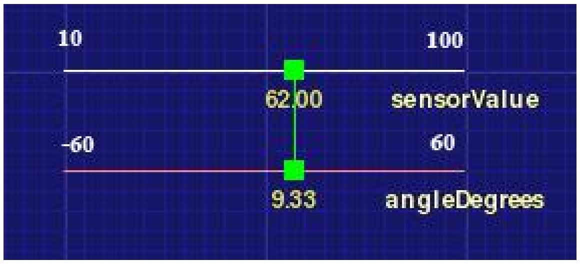


Figure 6-1: An example of how line objects can be used to scale and convert values

(A) Linear Approach

In Figure 6-1, the top white line (sensorVal) holds the raw sensor value. The line's min and max range are set to 10 and 100 respectively. When the value on the line is set, it is shown as a green square at a location proportional to that value (Figure 6-1 shows where it is for the sensor value 62). The bottom red line (angleDegrees) is a special "cloned" line. Its value is positioned to be vertically aligned with the parent line. And since the min/max range is different (-60 to 60), the value reading is different from the parent. Thus, from the figure, the value 62 on the parent line is transformed into 9.33 on the cloned line.

(B) Angular Approach

Since line objects need not be a simple straight line, a circular line and a turtle can also be used to perform a conversion. The basic arrangement is to place a turtle in the center of a circle. The value on the circular line is the converted to the turtle's heading using the "towards" statement.

```
t1, towards "sensorVal
```

The above statement will set the t1's heading to the current value location on the sensorVal line (the green square in Figure 6-2 left). The sensor value to turtle heading conversion can be accomplished by arranging the circle so that the value 10 and 100 is at turtle heading -60 and 60 degrees respectively (Figure 6-2 right).

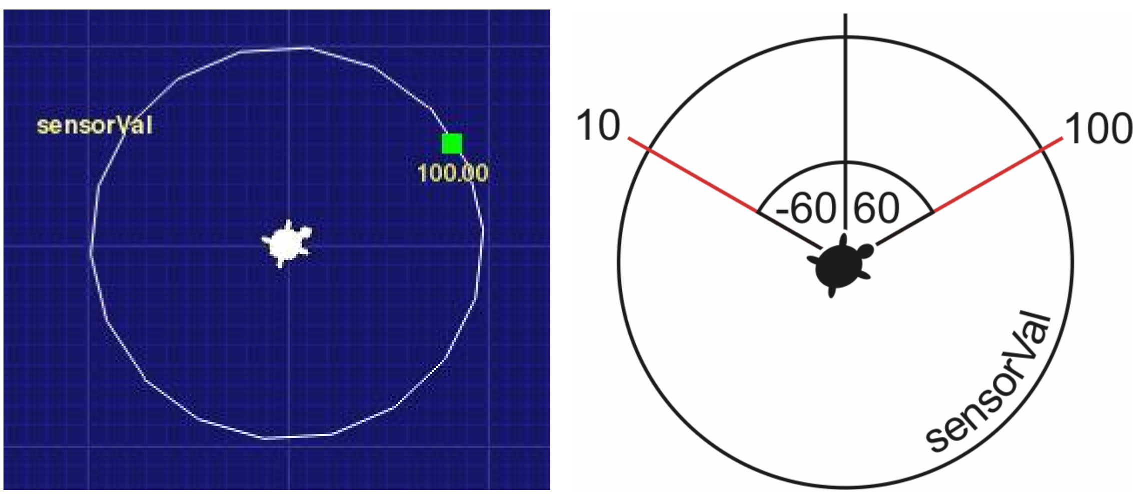


Figure 6-2: Another method to convert sensor data to an angle value by using a circle line and a turtle. (Left) a screen shot of the process in action. (Right) a diagram showing the needed alignments to make the conversion work.

6.1.2. None-Linear Transformation

In some cases the input or the desired output does not have a linear relationship with each other. For example, many sensors behave non-linearly and cannot be easily calculated mathematically. In this case, interpolation techniques are often a viable solution. Another common situation for such non-linear transformation is when an output signal (e.g. motor power) needs to be more sensitive to one input range than another. Consider a simplified case where the pendulum's angle from the center is used to determine the car's speed. The simple rule is that the greater the angle the faster the car should go. But students often want the motor to be more sensitive to the initial fall (angle has a small value) than later when it is usually too late to recover. This could be accomplished using conversion lines that are shaped as shown in the following figure.

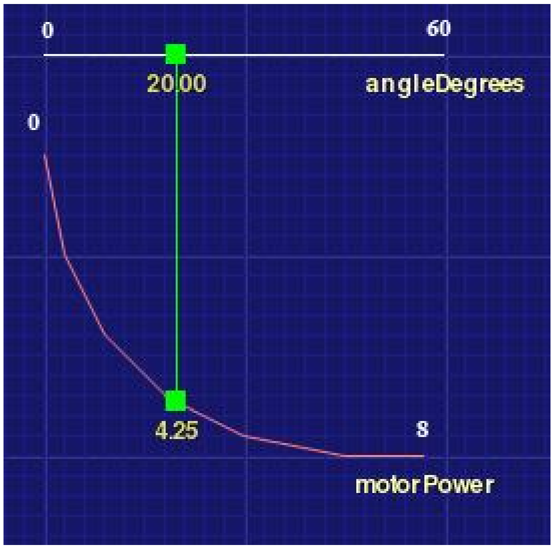


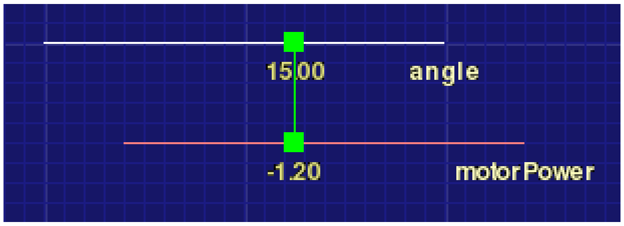
Figure 6-3: An example of a non-linear relationship between the input and output

The top angleDegrees line is the angle (0-60) while the bottom motorPower line is the motor power (0-8). The value positions of both lines are vertically aligned as described before. But the motor power line is steep in the area where angle value is small (towards the left). Thus, a small change in the angle line will cause a large change in the motor power line. The opposite is true for the area where the angle value is large.

6.1.3. Other Value Manipulation Operations

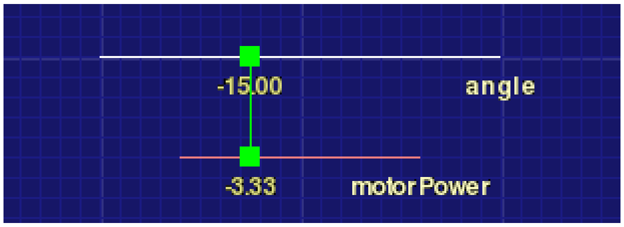
Some other common operations that can be done with line conversions include:

Offsetting



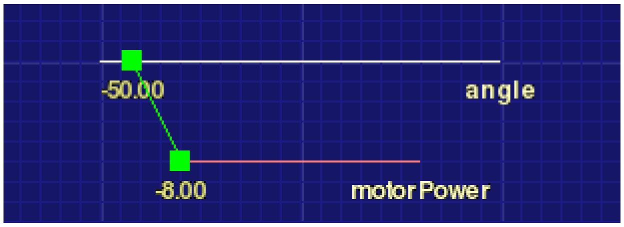
Lines can be shifted (left or right in this figure) to create an offset

Dynamic Scale Factor



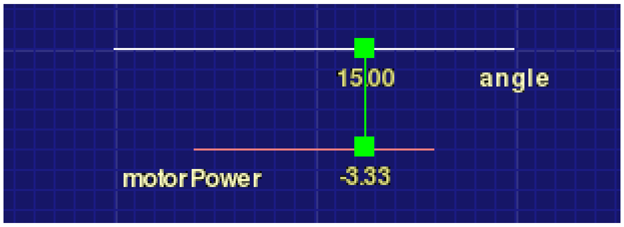
Lines can be stretched or shrunken to alter the scaling

Limits



Values on the child line (red) will never exceed its limits

Direct/Inverse Proportion



Flipping the lines will determine whether the transformation is direct or reverse

6.2. Creating Physical Relationships between On-screen Objects

When objects in the physical world are connected, the motion of one object would affect the other. We expect our wrist watch, hat, shoes, eye glasses, etc. to remain on specific parts of our body no matter how the body moves. This is common-sense and can be taken for granted. A more elaborate use of object relationships can lead to impressive mechanical structures. Auto engines utilize object relationships to convert the linear motion of the piston to the desired rotational motion for the wheels. Charles Babbage attempted to create a mechanical difference engine in the 19th century utilizing a tremendous amount of gearing. Arthur Ganson is renowned for his work with kinetic sculpture, which produces artistic motion from simple object relations[9].

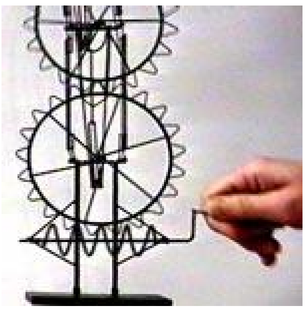


Figure 6-4: An example of Arthur Ganson's kinetic sculpture

Although the utility of object relationships are abundant in the physical world, these relationships do not inherently exist on the computer screen. It has been observed in this work that some of these relationships could be useful to the learners. Thus, a "stickto" primitive has been implemented. When one object is stuck to another, its position and heading would change depending on the reference object. Here are two examples:

6.2.1. Placing a Moving Car on a Rotating Beam

Consider the balance beam example. The beam is constantly rotating. The car that is moving on the beam must move relative to this motion. This affects both the screen position and heading of the car object as shown in the following figure.

[9] See http://www.arthurganson.com for more images and video of his work

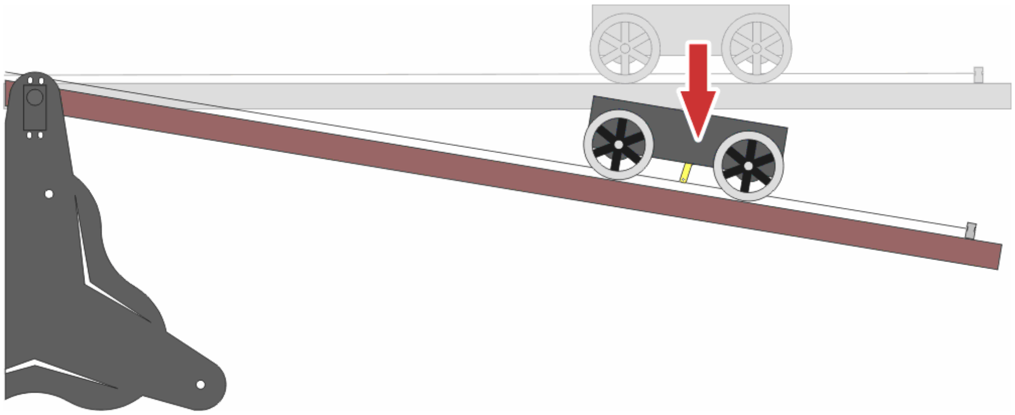


Figure 6-5: The car position and heading needs to change in relation to the beam.

This relationship can be created by issuing the following command:

```
Car, stickto “beam
```

With this relationship established, the car can simply and accurately move around on the rotating beam as shown in the figure below.

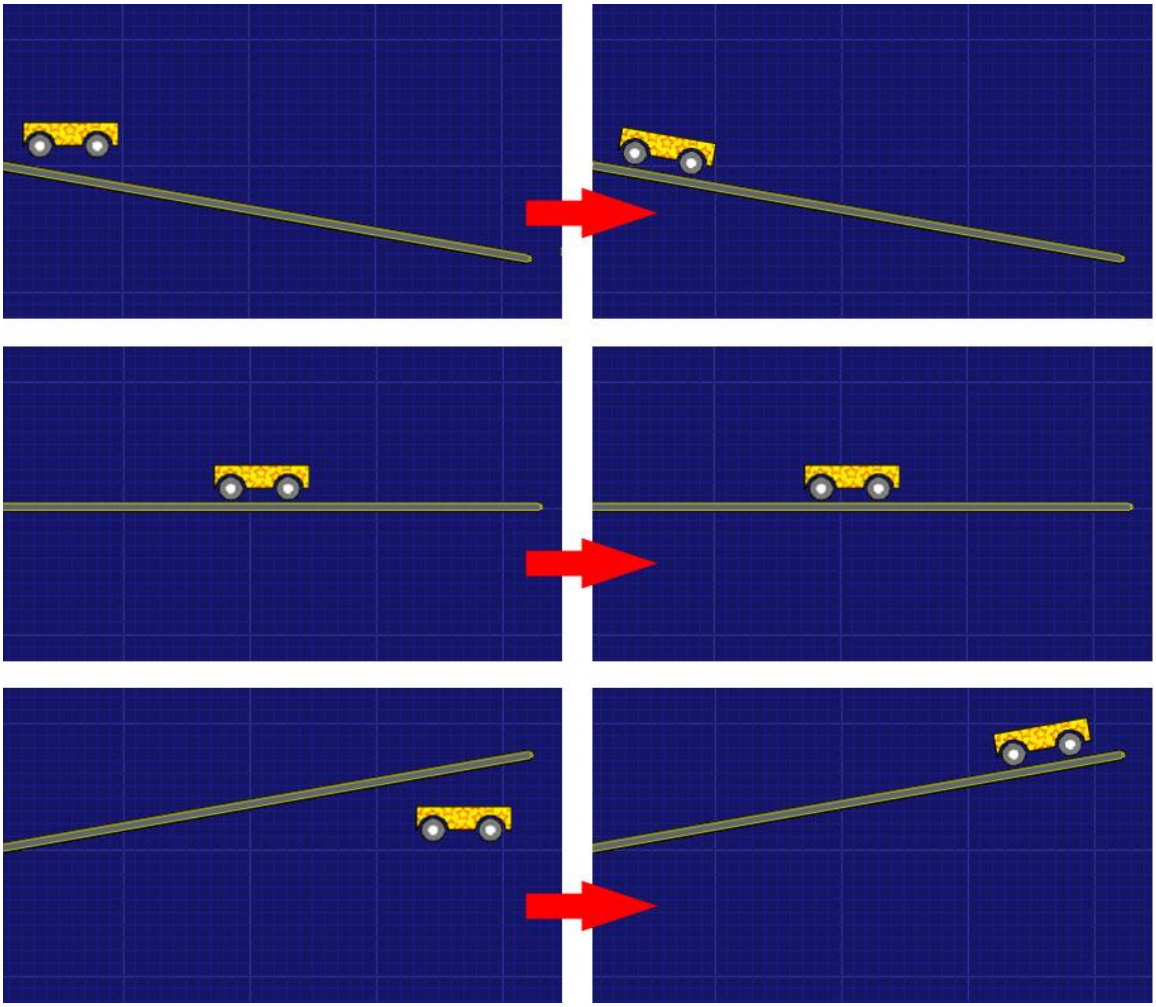


Figure 6-6: The left column shows the car's linear movement with no relationship with the beam. The right column shows the same car motion after establishing a "stickto" relationship with the beam.

6.2.2. Tracking the Pendulum's Tip

During the course of the inverted pendulum project, there were at least two occasions when the learners wanted to figure out the position of the pendulum's top tip. Since the information being sensed form the physical model does not include the tip position, it can not be figured out directly. However, because the on-screen model has the same aspect ratio as the physical object, the tip position can be tracked by "sticking" a turtle to it. The turtle's position will then reflect the desired tip position.

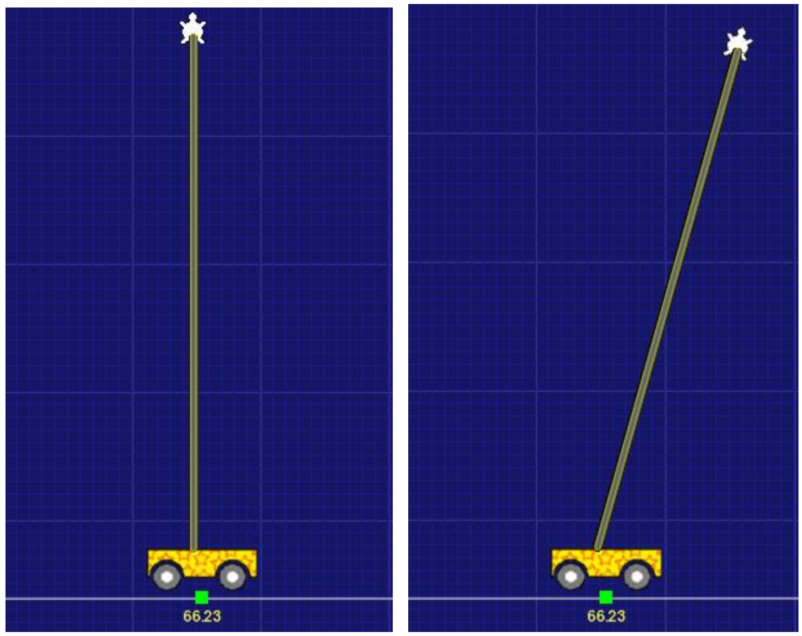


Figure 6-7: (Left) a turtle is placed on the top tip of the pendulum. A "stickto" command is then issued to make the turtle stick to the beam every where it goes. (Right) shows how the position and heading of the turtle follows the top tip position of the beam.

6.3. Graphing

Graphing is a common way used by learners to visualize information. There is no special module built in to the system for graphing. Graphing is done through the use of line and turtle objects. Once the graphing mechanism is constructed, learners to perform many adjustments to produce the plot they want.

A typical graph is created using two line objects and one turtle. The first line represents the value under investigation (Y axis). The second is the time line (X axis). The turtle is used to draw the graph itself.

Figure 6-8: An example of a simple graph. The `_carPlot` line is the car position value while the `_timer` line is an auto increment counter line. The turtle position is the (X,Y) combination of the value positions on the screen.

A simple program to create the graph in Figure 6-8 is as follows:

```
To updateGraph
      carPlot,
      setx _timer.xval
      sety _carPlot.yval
end
```

Once the program is created, the _carPlot, and _timer lines can be moved or stretched to align or zoom in/out of the values.

Multivariable graphs can be created simply by adding more turtles and lines as shown in the figure below.

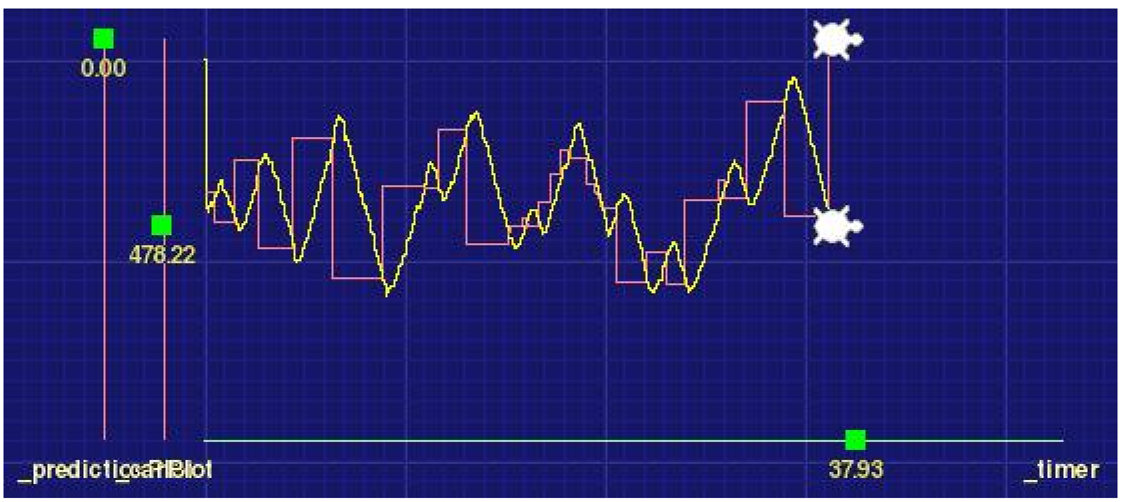


Figure 6-9: An example of a multi-variable graph. Two turtles and two value lines are used.

6.4. Finite State Machine

As students started to realize that both position and speed are important factors to balance the inverted pendulum or the car on beam, they started to define “states” and associate different actions for each of them. Consider the following definition of states and actions for the inverted pendulum:

A) **If** the beam is leaning to the left **and** the tip is also moving to the left **then** move left
B) **If** the beam is leaning to the right **and** the tip is also moving to the right **then** move right
C) **If** the beam is leaning a small amount to the left **and** the tip is moving to the right **then** slow down
D) **If** the beam is leaning a small amount to the right **and** the tip is moving to the left **then** slow down

These definitions can be transformed into rectangular areas on a 2D graph as shown in the following figure.

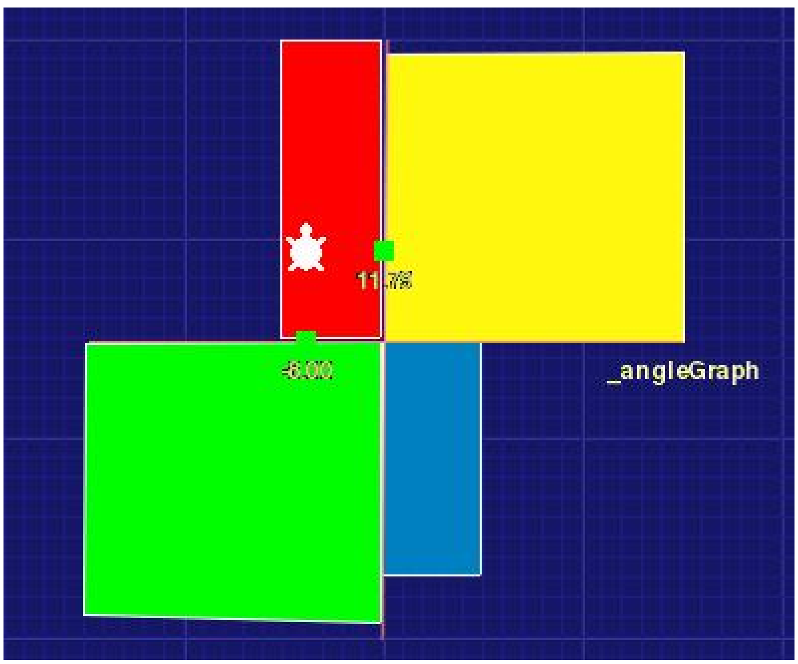


Figure 6-10: A graph with angle on the X axis and tip speed on the Y axis. Different colored rectangles are created to define different states of the system.

A turtle (t1) is moved to the according X and Y locations in real-time as the pendulum car is in action. Thus, the location of the turtle determines the current state. State detection and reactions are implemented through the following Logo code.

```
if istouching "t1 "greenShape
    [goleft]

if istouching "t1 "blueShape
    [goleft]

if istouching "t1 "redShape
    [slowDown]

if istouching "t1 "yellowShape
    [slowDown]
```

Essentially, this approach utilizes a spatial model to define states. It was easy to see how the turtle travels in and out of these states. The language (commands) used in the Logo program reflects the events that is straight forward to the students. It is worth mentioning the contrast to a more traditional approach as the difference is considerably more significant than the previous example.

```
If angle > 0 and speed > 0 [ moveRight ]
If angle < 0 and speed < 0 [ moveLeft]

If angle < 27 and angle > 0 and speed < 0 [ slowDown ]
If angle > -27 and angle < 0 and speed > 0 [ slowDown]
```

When I worked with students using this approach, only those who had programming experience were able to maintain a good understanding of what it is doing. Even so, the program was difficult to debug in all cases. This is mainly because it was difficult to see what is happening during program execution.

I am not trying to make a case that one approach is better than the other. But if learning is our goal, being able to connect to the modes of interaction that learners can better relate to is more desirable.

6.5. Conditions

Consider a simple example. During one of the experiments working with the inverted pendulum car, the car moved too fast sending itself off the table. The car dropped and broke into pieces. From that accident, there was an idea to implement a guard check to stop the car when it reaches the edges. Since we already have the car's motion simulated on the screen, this guard check was easily done by placing two turtles on each side of the model.

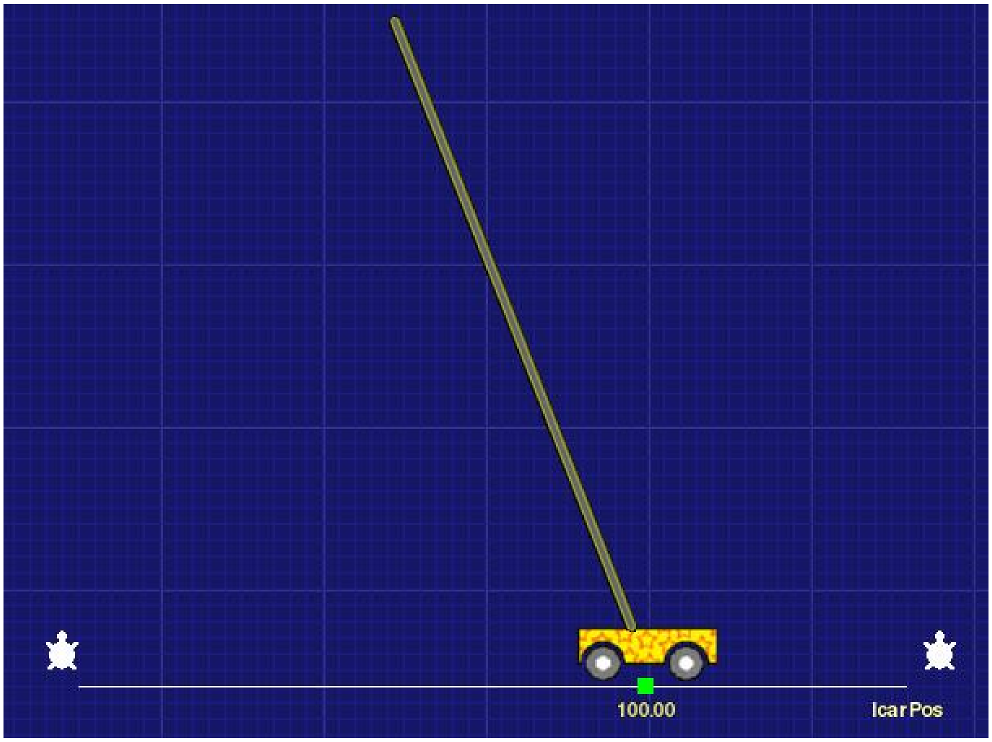


Figure 6-11: Screen shot of how two turtles are placed at each side of the screen to stop the car from falling off the table.

A Logo procedure was then added to detect and stop the car when it touches either one of these two “guard” turtles. Here’s how the code looks like:

```
to guard_check

       t1,
       if isTouching “leftGuard or isTouching “rightGuard [ stopall]

end
```

leftGuard and rightGuard are the names of the left and right turtle respectively. Now let’s compare it with a more traditional approach.

```
to guard_check

       if sensor1 < 88 or sensor1 > 245 [ stopall]

end
```

The former approach utilizes the properties of on-screen turtles while the latter digs into the underlying sensor values. Both methods work just fine. To be fair, the second approach is more efficient as it does not require a simulation model of the car position. But in terms of human comprehension, the first approach connects to our experiential knowledge that objects can move and collide. As described by Papert and Wilenski, this is the quality that makes the idea being expressed more “concrete” which is beneficial especially for those who are new to the concepts being articulated.

References

Ackermann, E. K. (2004). Constructing Knowledge and Transforming the World. A Learning Zone of One's Own: Sharing Representations and Flow in Collaborative Learning Environments. M. T. a. L.Steels. Amsterdam, Berlin, Oxford, Tokyo, Washington, DC., IOS Press. **1:** 15-37.

Arbib, M. A. (1969). Theories of Abstract Automata. Englewood Cliffs, N.J., Prentice-Hall, Inc.

Astrom, K. J., Klein, R.E , Lennartsson, A. (2005). Bicycle dynamics and control: adapted bicycles for education and research. Control Systems Magazine, IEEE. **25:** 26-47.

Ben-Joseph, E., Ishii, H., Underkoffler, J., Piper, B., Yeung L. (2001). "Urban Simulation and the Luminous Planning Table: Bridging the Gap between the Digital and the Tangible." Journal of Planning Education and Research **21**: 195-202.

Bissell, C. C. (1999). Control Education: Time for Radical Change? IEEE Control Systems. **19:** 44-49.

Blikstein, P., Cavallo, D. (2003). God hides in the details: design and implementation of technology-enabled learning environments in public education. Eurologo 2003, Porto, Portugal.

Bonato, P. (2003). "Wearable sensors/systems and their impact on biomedical engineering." IEEE Eng Med Biol Mag **22**(3): 18-20.

Cavallo, D. (2000). "Emergent Design and learning environments: Building on indigenous knowledge." IBM Systems Journal **39**(3 & 4): 768-781.

Cavallo, D., Blikstein, P., Sipitakiat, A., Basu, A., Camargo, A., Lopes, R., Cavallo, A. (2004). The City that We Want: Generative Themes, Constructionist Technologies and School/Social Change. International Workshop on Technology for Education in Developing Countries, Joensuu, Finland.

Dewey, J. (1963). Experience and education. New York,, Collier Books.

Egardt, B. (1979). "Stability of Adaptive Controllers." Lecture Notes in control and Information Sciences **20**.

Freire, P. (1970). Pedagogy of the oppressed. [New York], Herder and Herder.

Gelb, A., Velde, V., et al. (1968). Multiple-input describing functions and nonlinear system design. New York,, McGraw-Hill.

Hancock, C. M. (2003). Real-time programming and the big ideas of computational literacy**:** 121 p.

Harel, I. and Papert, S. (1991). Constructionism : research reports and essays, 1985-1990. Norwood, N.J., Ablex Pub. Corp.

Herr, H. and Wilkenfeld, A. (2003). "User-Adaptive Control of a Magnetorheological Prosthetic Knee." Industrial Robot **30**: 42-55.

Herr, H., Wilkenfeld, A., et al. (2002). Patient-Adaptive Prosthetic and Orthotic Leg Systems. The 12th Nordic Baltic Conference on Biomedical Engineering and Medical Physics, Reykjavik, Iceland.

Hirai, K., Hirose, M., et al. (1998). The Development of Honda Humanoid Robot. IEEE International Conference on Robotics and Automation, Leuven, Belgium.

Hoyles, C. (1993). Microworlds/schoolworlds: The transformation of an innovation. Learning from computers : mathematics education and technology. C. Keitel and K. Ruthven. Berlin ; New York, Springer-Verlag**:** 10-17.

Ioannou, P. A. and Sun, J. (1996). Robust adaptive control. Upper Saddle River, NJ, PTR Prentice-Hall.

Jørgensen, S. E. and Müller, F. (2000). Handbook of ecosystem theories and management. Boca Raton, Fla., Lewis Publishers.

Klein, R. E. (1989). Using Bicycles to Teach System Dynamics. Control Systems Magazine, IEEE **9:** 4-9.

Kolberg, E., Reich, Y., et al. (2003). "Project-based High School Mechatronics Course." Internation Journal of Engineering Education **19**(4): 557-562.

Lakatos, I. (1976). Proofs and refutations : the logic of mathematical discovery. Cambridge ; New York, Cambridge University Press.

Lewis, F. L. (1992). Applied optimal control & estimation : digital design & implementation. Englewood Cliffs, N.J., Prentice Hall.

Lieberman, J. (2004). "A Robotic Ball Balancing Beam." from http://web.media.mit.edu/~xercyn/sight/rbbb/rbbb.pdf.

Martin, F., Mihkak, B., and Silverman, B. (2000). "MetaCricket: A designer's kit for making computational devices." IBM Systems Journal **39**(3-4).

Martin, F. G. (1994). Circuits to control--learning engineering by designing LEGO robots**:** 255 leaves.

Miles, M. and Huberman, A. (1994). Qualitative Data Analysis : An Expanded Sourcebook. Thousand Oaks, CA, Sage.

Miller, D. P., Stein, C. (2001). " Creating autonomous roboticists." Intelligent Systems **16**(2): 20-23.

Nashner, L. M., Black, F. O., et al. (1982). "Adaptation to altered support and visual conditions during stance: patients with vestibular deficits." J. Neurosci. **2**(5): 536-544.

Papert, S. (1971). Teaching children to be mathematicians vs. teaching about mathematics. [Cambridge, Mass.], Massachusetts Institute of Technology, A. I. Laboratory.

Papert, S. (1980). Mindstorms : children, computers, and powerful ideas. New York, Basic Books.

Papert, S. (1999). Papert on Piaget. Time Magazine**:** 105.

Papert, S. (2000). "What's the big idea? Toward a pedagogy of idea power." IBM Systems Journal **39**(3 & 4): 720-729.

Papert, S. (2002). "The Turtle's Long Slow Trip: Macro-educological Perspectives on Microworlds." Journal of educational computing research **27**(1&2): 7-27.

Piaget, J. (1985). The equilibration of cognitive structures : the central problem of intellectual development. Chicago, University of Chicago Press.

Popovic, M., Englehart, A., et al. (2004). Angular Momentum Primitives for Human Walking: Biomechanics and Control. IEEE International Conference on Intelligent Robots and Systems, Sendai, Japan.

Resnick, M. (1994). Turtles, termites, and traffic jams : explorations in massively parallel microworlds. Cambridge, Mass., MIT Press.

Resnick, M., Ocko, S., Papert, S. (1988). "LEGO, Logo, and Design." Children's Environments Quarterly **5**(4).

Schuler, D., Namioka, A., Ed. (1993). Participatory Design: Principles and Practices, Lawrence Erlbaum Associates.

Sipitakiat, A., Blikstein, P., Cavallo, D. (2004). GoGo Board: Augmenting Programmable Bricks for Economically Challenged Audiences. International Conference of the Learning Sciences, California, USA.

Skogestad, S. and Postlethwaite, I. (2005). Multivariable feedback control : analysis and design. Chichester, England ; Hoboken, NJ, John Wiley.

Teng, F. C. (2000). Real-time control using Matlab Simulink. Systems, Man, and Cybernetics, 2000 IEEE International Conference on, Nashville, TN, USA.

Wall, C. r., Weinberg, M., et al. (2001). "Balance prosthesis based on micromechanical sensors using vibrotactile feedback of tilt." IEEE Transactions on Biomedical Engineering **48**(10): 1153-1161.

Watzlawick, P. (1984). The Invented reality : how do we know what we believe we know? : contributions to constructivism. New York, Norton.

Wescott, T. (2000). "PID Without a PhD." from http://www.embedded.com/2000/0010/0010feat3.htm.

Wie, B. (1998). Space vehicle dynamics and control. Reston, VA, American Institute of Aeronautics and Astronautics.

Wilensky, U. (1999). "NetLogo." from http://ccl.northwestern.edu/netlogo.