Note 0A: Overview

Welcome

As you've been told in 16A, the 16AB series is fundamentally about teaching you how to think like an engineer while making you stronger and more mathematically mature. The mathematical vehicle to do this is linear-algebra (and related concepts — especially in 16B the connections to calculus: differential equations most prominantly, but others as well) with the goal of you making these tools your own through your own hard work. This is done by showing you how you could have come with all the definitions yourself, as well as all the key proofs/derivations, and actually making you do many of these yourself — including deriving things not explicitly discussed in lecture/discussion, in ways that exercise the ways of thinking that are demonstrated for you in lecture and discussion. While mathematical curiosity is going to be fostered at points, the style is largely what is called "use-inspired[1]" where we show how each of the ideas emerge organically from engineering contexts, while being useful beyond the specific contexts that first motivated them.

The engineering contexts in 16AB have of course been carefully curated to allow both concepts *and* maturity to be built up step by step systematically. This is why 16A started with imaging, touched upon PageRank, engaged with touchscreens, and then went to the positioning module. This allowed you to develop the basic vocabulary of linear algebra, static linear circuit analysis, and inference by least squares and related ideas. Later in this note, we will go over the big picture application context for 16B: making cyborgs.

At times, 16AB use simple circuits to act as a bridge between the physical world and the much more generally applicable concepts that we want you to internalize. Circuits hit the "quadfecta" of being (1) simple (so simple that we can proceed without any physics prerequisites while still engaging basic physical intuition) vehicles for understanding the interplay between modeling, analysis, and design; (2) amazingly well-approximated by linear component models as well as easy-to-understand nonlinearities like switches, together with interconnection using simple graphs; (3) easy/safe to engage with in a hands-on fashion in lab; while also being (4) incredibly useful building blocks in the real world.

We also keep coming back to the pattern of learning from data, another widely useful building block for real world problems. **By the end of 16B, you will actually have built a foundational understanding of both basic circuits and basic machine learning.**

1 Making Cyborgs — the Application Thread

Throughout 16B, we will keep coming back to a big picture aspirational goal: making cyborgs. As engineers, we want to create a future in which machines can help human beings live better and fuller lives. One way to imagine doing so is cyborg technology. Imagine someone who has lost the use of a limb. We

[1] The term comes from a famous book by Donald Stokes: "Pasteur's Quadrant: Basic Science and Technological Innovation" that brought out how Engineering thinking is not encompassed by the naive dichotomy between "basic" and "applied" that exists in the popular view of the sciences. Instead, many key advances in the sciences (and mathematics) have actually been grounded in Engineering thinking where a broad set of goals creates the context for investigation and the systematization of knowledge. The popular example of "use-inspired" research is of Pasteur's foundational contributions to microbiology.

would like to be able to create an artificial limb that responds directly to their thoughts and helps them do what they want to do. In general, to have machines understand and respond to humans. This is futuristic technology (a civilization-scale endevour actually) that will take decades to fully realize, but that's exactly what we feel should motivate study in an introductory course like 16B. After all, those are the decades that you will be working!

How are we going to break this problem down into pieces that motivate aspects of the course? At a high level, there are three big tasks: extracting information from the brain and getting it into a computer, figuring out what the person wants, and then getting the physical machine to actually do what we want it to do.

1.1 The Blocks that Need to be Understood

Each of these big tasks itself breaks down into sub tasks. But first, there is a higher level question that we need to tackle. How should we divide the work between what information-processing should be done using analog circuits and what should be done in a digital computer? To answer this, the course starts out by getting a basic understanding of why we can't expect digital computation to be infinitely fast. Once we understand that, we can start into doing those tasks with the following building blocks that process the information in stages:

1. After our physical device records electrical signals in the brain, we need to "filter" or clean the signal in the analog domain to remove as much interference as we can.

2. The filtered signal needs to be sampled as slowly as we can get away with and we need to be able to implicitly interpolate (inside the computer) using a global approximation of the continuous-time signal as needed.

3. The resulting sampled information needs to be summarized (dimensionality reduced) using simple and fast computations in a way that is specific to the application. We're going to need to use learning from data to get this compression right at design time.

4. The summarized information needs to be classified/interpreted so that we can infer what the person wants their cybernetic implants to do for them. Once again, we're going to need to use learning from data to figure out the right mapping to do this.

5. The current state of the mechanical implants, the inferred goal, and knowledge of the dynamics of the mechanical implants needs to be combined to plan a trajectory that achieves the goal. Here, learning is going to have to be used to learn a model for these dynamics that we can use to plan.

6. The current state, the planned trajectory, and knowledge of the dynamics needs to be used to do real-time responsive control to ensure that our real-world machine actually follows the planned trajectory despite having model inaccuracies and disturbances coming in from the world.

We're not going to cover those blocks in the order above, although we will be starting at the beginning. Instead, we will cover them in the order that best allows us to discover the required tools and ways of thinking as we go.

It is fine if you don't understand what all this means right now. The point of the course is that you understand it by the end.

1.2 The Toy Version: Our Robot Car

To give you a hands-on way of engaging with the above aspirational vision of making brain-controlled robotic body parts, we will have a toy version that lets you experience these ideas. Instead of capturing signals from electrodes in brains, we will use voice capture using a microphone. Instead of a robot limb, we will have a robot car that you will build. But within this toy context, you will engage with the blocks above — including the learning parts.

In fact, 16B is *the course* where you get to engage in a hands-on way with the entire modern machine-learning pipeline from building the data collection aparatus to collecting the data to learning from the data to actually deploying your learned model in the real world. And our main educational goal is that you actually understand every step of what is going on here, not as being able to call some magical libraries with the right incantations, but in a way that you understand fully what is going on and why it is working or not.

2 Behind the Application Thread: the Key Ideas

As you can see from the above, at a high level the distinction between 16A and 16B is that in 16A everything was largely static. In 16B, we will be engaging with dynamics a lot more. Mathematically, this means that we are really going to be leaning on not just the 16A prerequisite, but also calculus. In addition, there are ideas from our prerequisite 61A that we will be building upon, especially philosophically.

The ideas and concepts in the course are cumulative, and the story will keep building and deepening. At the same time, we will expect that your maturity will also keep developing alongside. While there are certainly going to be concepts and ideas that seem to be specific to each building block, we are actually going to keep circling back to a few core patterns of thought over and over again.

The main patterns are (in descending order of prevalence):

- Changing variables/coordinates — finding a coordinate system or way of looking at the problem that makes it as simple and transparent as possible.
- Learning from data.
- Iteration/Recursion/Induction — solving problems by taking one step at a time.
- Approximation — finding a way to divide the problem into the most important part, and the part that you ignore (possibly just for now).

Each of these is something that you have already been exposed to in 16A, 61A, and your calculus courses — but we're going to be greatly strengthening your understanding of their potential and versatility, especially as they combine together.

Again, it is fine if you don't understand how these work together just yet. The educational goal is that you understand by the end of the course since these are incredibly powerful ways of thinking that will serve you well across many different areas.

Along the way, we will also build a lot of mathematical and modeling tools. **Our core promise to you is that you will always know why we are doing whatever we are doing**, and even more importantly, we aspire to helping you understand how in principle, you could have discovered all these mathematical and modeling tools on your own. After all, we are laying the foundation for a long career in which you will constantly have to be learning and making discoveries for yourself.

3 How to Succeed in 16B

The short answer for how to succeed is don't fall behind. Attend lecture, take your own notes, and participate in discussion. Read the official course notes as needed[2] and strive for actually understanding. Accept the fact that you might be confused when you first encounter any given HW problem — but try. Don't be afraid to try something that doesn't succeed at first. Overcoming what you don't understand and your own fears is a part of the learning process. Work with others and ask for help from course staff. This is a course in which truly understanding all the labs, homeworks, and discussions is the key to success on exams and beyond.

Note 0B has a checklist of things you can do to ensure your success in the class.

Acknowledgements

The reality is that these notes are the fruit of the combined effort of many people. In particular, credit for the overall vision of the Linear-Algebraic/Machine-Learning side of 16AB and how to present these ideas belongs largely to Prof. Gireeja Ranade. The overall aesthetic[3] of 16AB owes a great deal to both Gireeja Ranade and Elad Alon, the main instructors for the pilot Sp15 offering of 16A and main designers of the entire 16AB course sequence.

Lots of faculty have also contributed to and shaped the development of the technical material in 16B. These include Anant Sahai, Claire Tomlin, Murat Arcak, Miki Lustig, Babak Ayazifar, Michel Maharbiz, Kris Pister, and Jaijeet Roychowdhury. The first set of comprehensive notes for students was a reader put together by Murat Arcak that covered the material after the first module. In Fall '18, efforts led by Elad Alon initiated initial notes for the first module in 16 style. In Spring '19, the combined efforts of Anant Sahai, Kris Pister, and Jaijeet Roychowdhury led to some updated notes for the first module, and then in Fall '19, Anant Sahai led the effort to both prune down the course scope and get all the material standardized into a unified set of notes. For these unified 16B notes, the efforts of 16AB TAs Nikhil Shinde, Aditya Arun, and Rahul Arya have been invaluable in capturing the spirit of the course while getting the details right. It is also important to recognize all the feedback from students taking the course.

The notes exist in an ecosystem of resources together with the discussion worksheets and most importantly, the homework problems and their solutions. The labs and course project complement this written and computational material with hands-on experience with many of the course ideas. Anyway, a major new and radically different freshman course sequence doesn't just happen — lots of people in the EECS

[2]Some people benefit from skimming course notes ahead of lecture and then reading them carefully after lecture. Others benefit from attending lecture without reading ahead, and then reading the appropriate sections of the official course notes carefully only after lecture in conjunction with their own notes. You should do what works for you, however 16B is definitely a course in which there is a big benefit to actively attending in-person lecture.

[3]Proper credit here also acknowledges the strong cultural influence of the faculty team that made EECS 70 and shaped it into a required course: Alistair Sinclair, Satish Rao, David Tse, Umesh Vazirani, Christos Papadimitriou, David Wagner, and Anant Sahai, along with pioneering TA's like Thomas Vidick, all helped make and polish the notes for 70. The 16AB + 70 sequence all follow the aesthetic that mathematical ideas should have practical punchlines. 16AB further emphasizes the rooting of mathematical intuition in physical examples as well as virtual ones.

As far as linear algebra and differential equations go, the treatments in 16B in particular also owe a significant intellectual debt to William Kahan's notes and approach to dealing with this foundational material. For circuits, there is also a definite philosophical influence of Chua, Desoer, and Kuh's perspective. The unified view of computation and physical systems also owes a philosophical debt to Lee and Varaiya. All of the Berkeley faculty mentioned in this paragraph are now retired (and some are no longer among us), but their insights continue to reverberate here in this fresh treatment.

department worked together for years to make all this a reality.

Contributors:

- Anant Sahai.