

## Cracking the code: Understanding the mystery behind the computational tool

I have previously used MATLAB in my graduate work in computational electrophysiology and have found it to be an extremely useful platform for both simulation and for exploring complex science and engineering topics. Extending MATLAB as a computational teaching tool helps students to use a logical step-by-step approach to define and solve problems and to break technical theoretical concepts into concrete variables, constants, and conditions. When I am translating electrophysiology computational tools into the classroom, I follow several key steps to make sure students understand the underlying physical basis for the model, the applications and limitations of numerical models, the power of computation, and the utility of simulation for exploring how different variables impact the computational model. In the first step, the students are taught the physical basis for the equations and the foundation of the model. Many times when students learn computation in a stand-alone class they are often disconnected from the actual physical reality of the model. In the context of electrophysiology, students first learn about the biology of the excitable cell, then they learn the engineering model and its associated equations. This first step is especially important when using computation as a teaching tool in science and engineering because it helps to put the computation in context beyond simply programming equations. In the second step, the students learn how to perform numerical methods by hand and they learn the limitations of numerical methods. It is important for students to understand the underlying math principles that power the computational tool so that they can make informed decisions about the accuracy and reliability of the information that the tool provides. In the context of electrophysiology, students begin with a simple numerical method such as Forward Euler which uses data from a previous time step to calculate the next time step. Using small-scale examples, they explore how changing small variables such as the size of the time step can affect the answer, and solving the equations by hand also gives them an appreciation for how valuable computational tools become as the problems get more complex. Once the students understand the physical and mathematical basis for the problem, the third step is to translate the equations into MATLAB code to create the computational tool. In entry level courses, I will provide a skeleton code and have the students fill in the appropriate code to complete the required calculations. Students know how to complete the problems by hand, so they are also able to verify that the output of the program makes physical sense. The process of applying the scientific or engineering knowledge in a logical format helps to reinforce the underlying principles and it also allows students to solve more complex problems than can be done by hand. The final step is to use simulation to explore the impact of different variables on the model. In electrophysiology, students are able to visualize the response of the action potential to threshold voltage and current stimulus and verify that the model response is correct based on their understanding of the biological and physical principles they learn in the first step. Being able to visualize the response of the model to many different parameters in real-time without actually doing physical experiments helps to reinforce understanding of the biological and physical concepts and makes computation a powerful and cost-effective tool for teaching science and engineering.

Because I have experienced the power of computation for teaching complex ideas at the undergraduate level, I would also like to incorporate MATLAB into my engineering courses at the advanced high school where I teach. This workshop will allow me to get exposure to a wide range of experienced educators who can give advice on the best way to teach basic computational skills and can give a broad range of specific examples of how to use MATLAB to reinforce science and engineering principles.