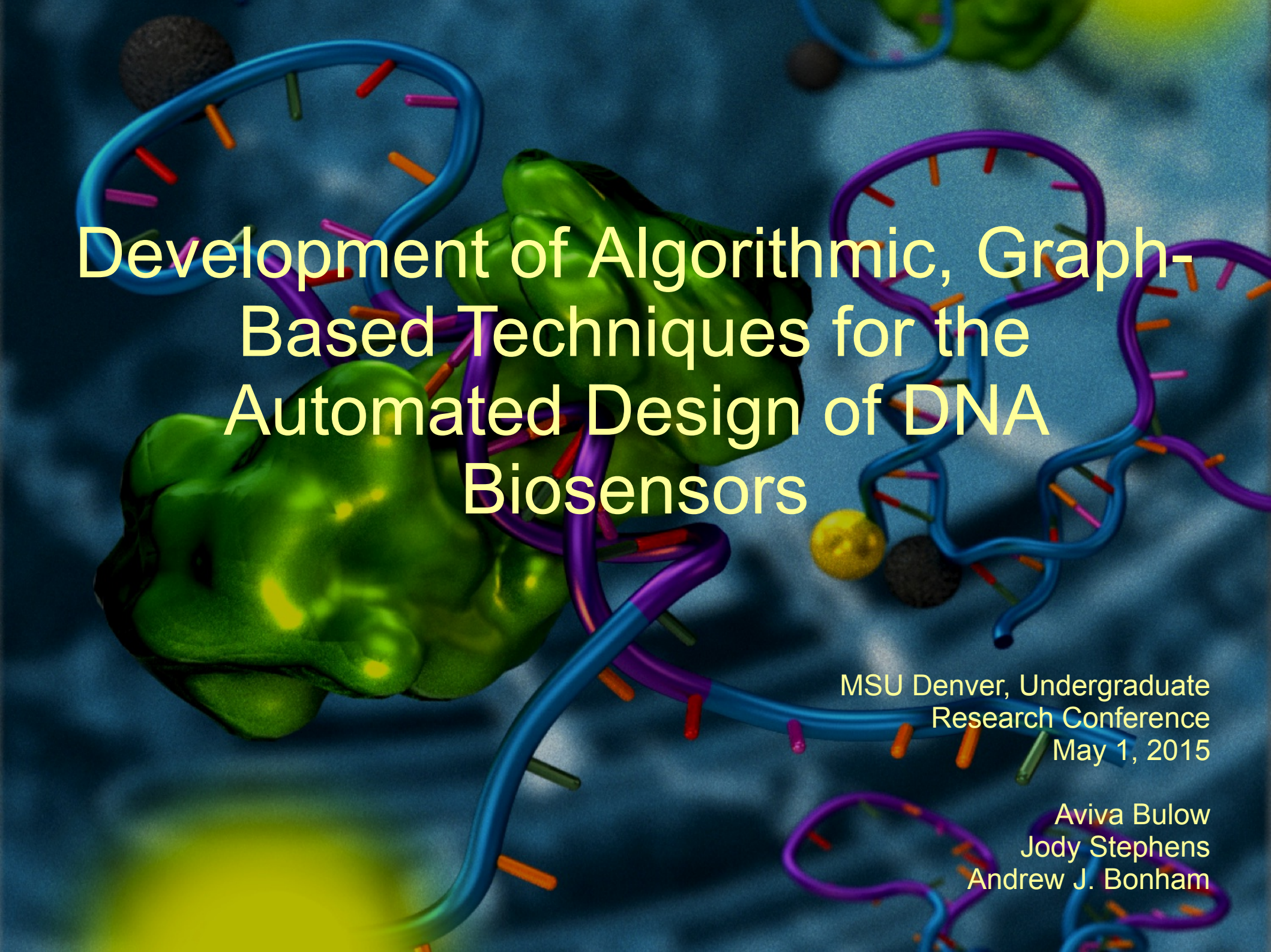


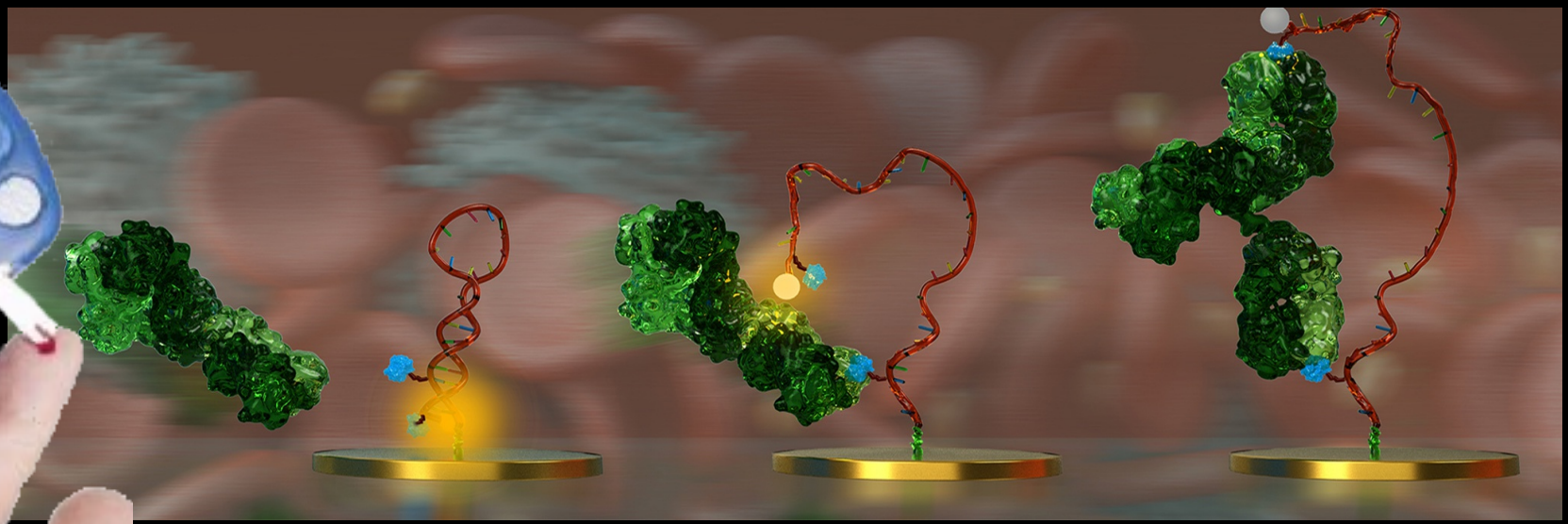
Development of Algorithmic, Graph-Based Techniques for the Automated Design of DNA Biosensors

MSU Denver, Undergraduate Research Conference
May 1, 2015

Aviva Bulow
Jody Stephens
Andrew J. Bonham

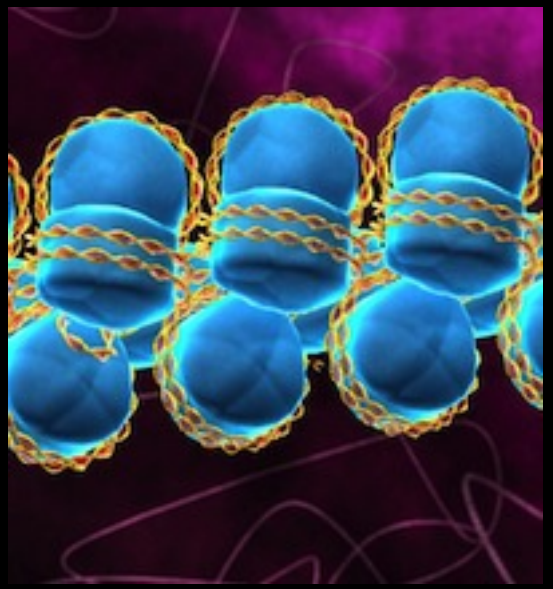
What's a Biosensor?

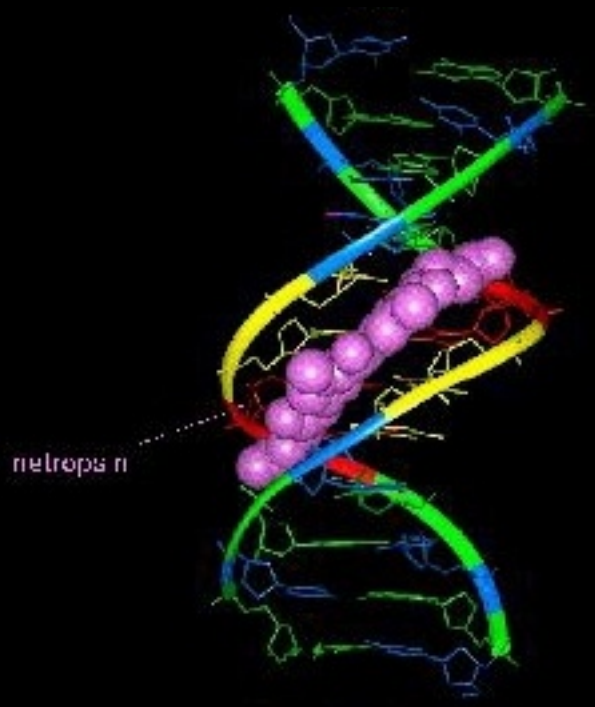
- Something, in this case DNA, used to detect biological or bioactive molecules.
- They are important for medicine and research.

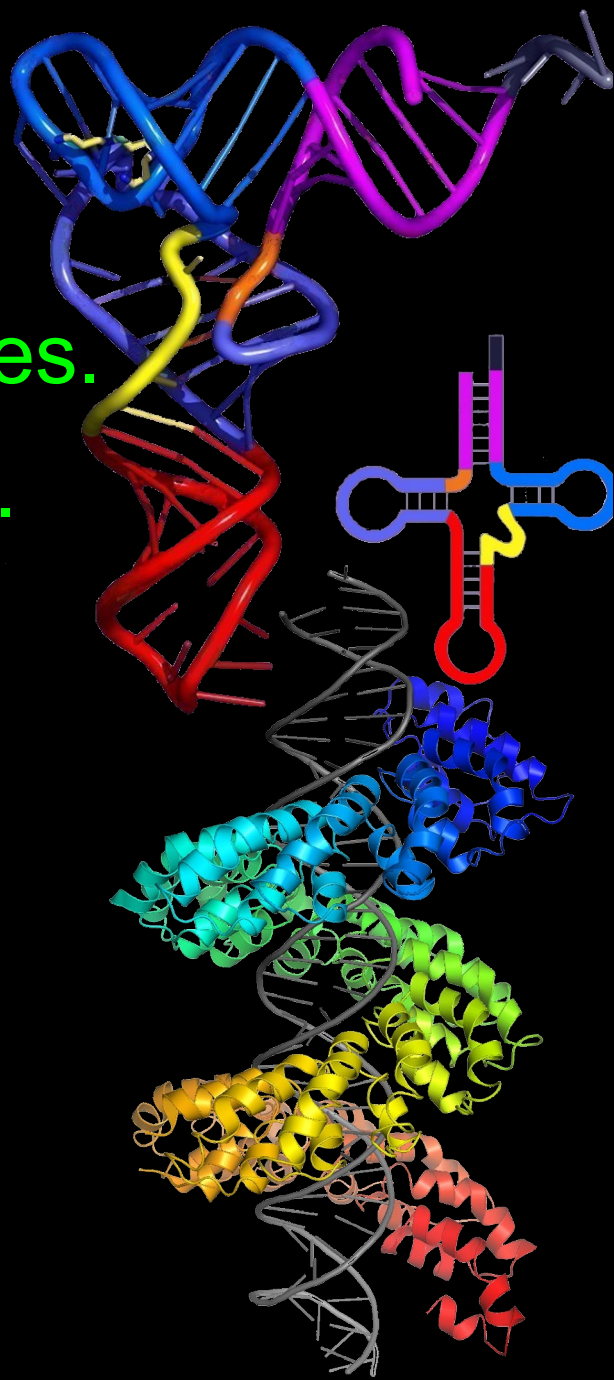


Why DNA?

- DNA folds predictably.
- DNA can take many different shapes.
- DNA binds many important targets.

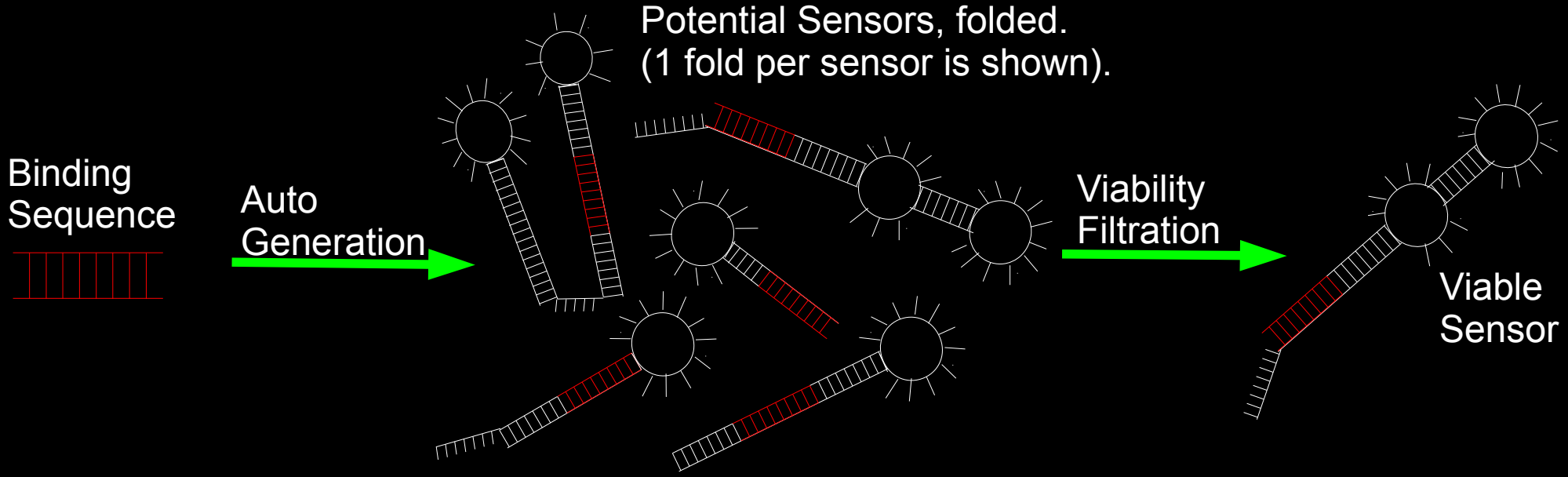






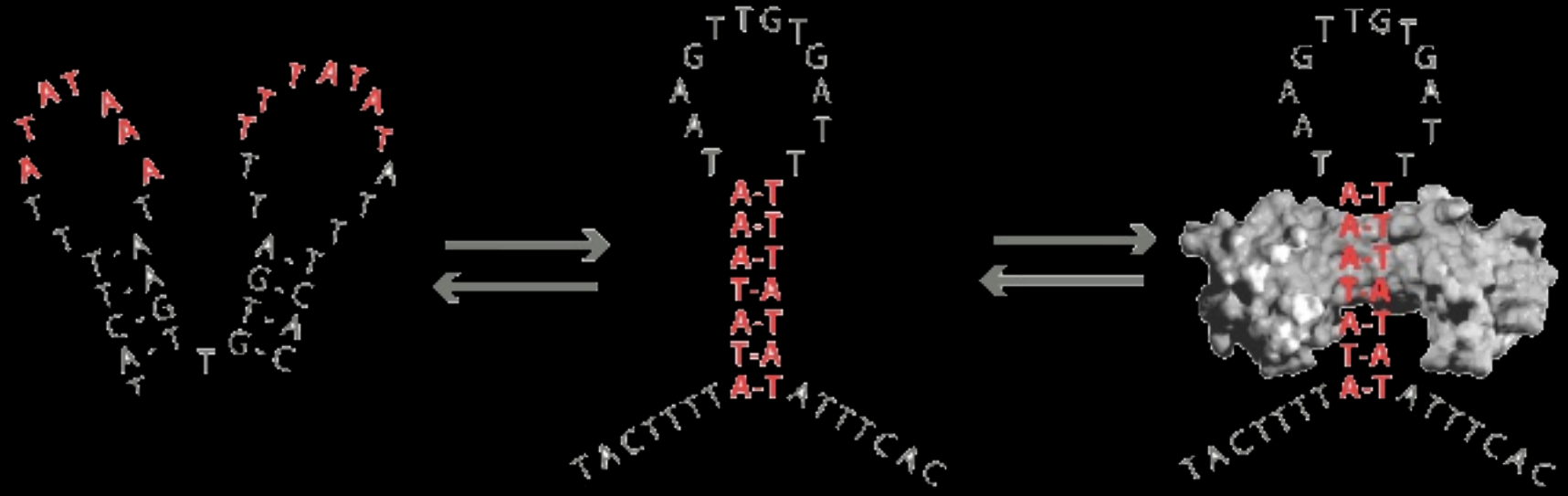
So What's the Big Idea?

- Designing a DNA biosensor by hand is considered cruel and unusual punishment in many states. Fealden solves this, here's how:



What was Fealden 1.0?

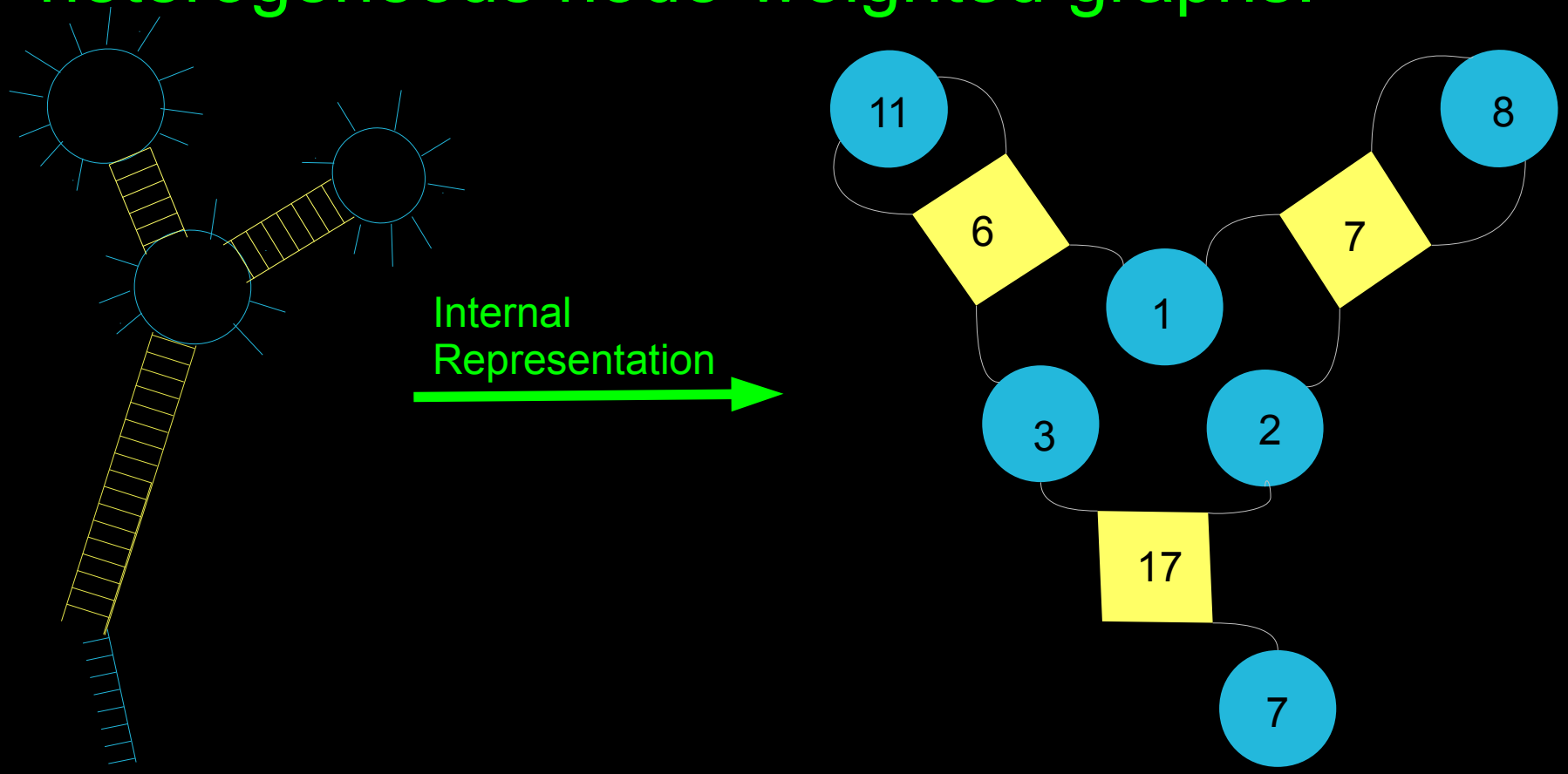
- Fealden 1.0 used the following assumptions to determine if biosensors were viable.



- This throws away many, sometimes superior biosensors, and is quite inflexible

Can We Fix It?

- Yes We Can!! – with graph theory!
- Let's represent the DNA sequences as heterogeneous node-weighted graphs.



What's So Great About Graphs?

- We have no pre-conceived notions about what shape makes a good sensor.
- We have a flexible program.

What Have You Done?

- I have the graph!
- And a distance metric.

```
def construct_graph_SSNode(self, currentNode, currentIndex):
        currentNode.set_start(currentIndex +1)
        isLastNode = True
        length = 0
        for i, v in enumerate(self.foldData[currentIndex::]):
            if v[1] == 0:
                self.ptrList[i+currentIndex] = currentNode
                length = i+1
            else:
                isLastNode = False
                currentNode.set_length(i)
                nextNode = None
                if self.ptrList[v[1]-1] == None:
                    nextNode = node.DSNode(currentNode)
                    self.construct_graph_DSNode_strand1(nextNode, currentIndex + i)
                else:
                    nextNode = self.ptrList[v[1]-1]
                    self.construct_graph_DSNode_strand2(nextNode, currentIndex + i, currentNode)

                currentNode.set_downstreamDSNode(nextNode)
                break
        if isLastNode:
            currentNode.set_length(length)
```

What's Left to Do?

- Short Term.
  - Finish distance metric
  - Validity Metrics.
  - Triage poor sensors.
- Longer Term
  - Graph “seed” for generation.
  - Add extra functionality discussed.
  - User friendly front end.

Wait, What?

- Biosensors detect important targets.
- DNA makes good biosensors.
- It's torture to design DNA biosensors by hand.
- Fealden automates DNA biosensor design.
- Fealden 2.0 uses graphs to add flexibility and functionality to the program.
- I have finished some important steps in revamping Fealden.
- I have even more left to do.

Who Can I Blame for This Talk?

- I would like to acknowledge Jody Stephens for his pioneering work on this project, and for recruiting me to work on something which is so much fun.
- I would like to thank Dr. Bonham for all the time, effort, encoragement, and ideas he has put into myself and this project.
- This work is in progress at MSU Denver, thanks Metro!