Using Automata to Model Plant Behavior

Jasey Chanders

December 12, 2024

1 Background

I began this project by searching for an application of automata in a real world scenario as I felt like that would help ground the techniques we learned in class. I have always love nature, biology and plants so I began by looking for ways we could use automata to model them. This led me down a twisting path of different approaches which I will walk through in this report ultimately ending in using Discrete Finite Automata to predict disease outbreaks in crops.

2 Cellular Automata

The first type of automata that i found for modeling plants is called cellular automata and is very useful for modeling the natural world as it allows each cell of a model to be updated based on local rules (local conditions) instead of global ones, which is much more realistic of nature. A very basic example can be seen in "Introduction to Symbolic Dynamics and Cellular Automata" where they describe using celluar automata to model a pipe warming up. In the following figure the pipe is discretized into cells which indicate their tempature with color. Red shows a hot cell and blue shows a cold one.

Using the local rule that at every timestep any blue cell which is touching a red cell becomes red we can see how the temperature evolves over time.

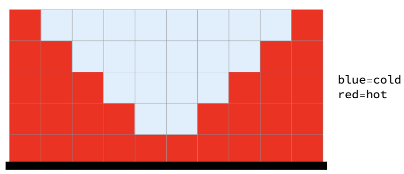
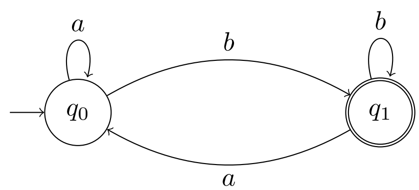


Figure 1.2. The rod is hot everywhere after four time steps, moving down one row at a time, under the rule described above.

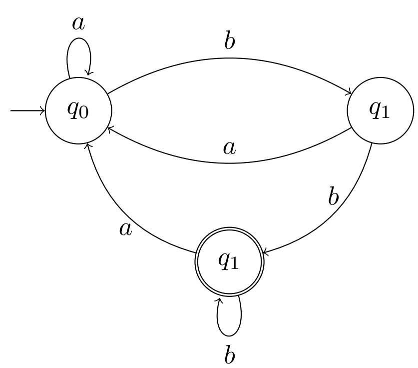
This method of modeling natural systems is quite useful and can be used on things such as crystal growth, Game of Life, and more. However, it wasn't really that applicable to what we were talking about in class so I moved on to other methods.

3 Mealy and Moore Machines

Next I found a paper titled ”Finding and defining the natural automata acting in living plants: Toward the synthetic biology for robotics and informatics in vivo” which while covering many different related topics had a section on using basic Discrete Finite Automata to model plant behavior. They discussed two basic automata called Mealy and Moore Machines. A Mealy Machine is a super basic automata that waits for an b, goes to the accepting state on that input, stays there as long as it continues getting input b and then returns to the initial state on input a.



The second automata they talked about is a Moore Machine which is similar to a Mealy Machine except it must see input b twice before it reaches the accepting state.



Venus Fly Traps are rather interesting example to model because their ”sensors” must be triggered twice in order to send the signal to closed their mouths. Thus they are wonderful examples of Moore Machines (at least at their most simplified, as most plant behavior is much more complicated).

4 Modeling Plant Disease Outbreaks

In a paper titled "Finite Automata Models in Agro-ecosystem and Plant Protection" the authors explored using Discrete Finite Automata to model outbreaks of disease in crops. They specifically focused on mildew in tomato and potato crops. This mildew is easy to model because they growth can be predicted by weather conditions and stopped with chemical treatments. In order to model this they defined an automata as follows:

Dangerous Intervals in which the meldew grows $D_1, D_2, ...D_m$ in which D_m signifies the mildew has grown too much for plant survival

Suppressive Intervals in which the mildew is killed $S_{m-1}, S_{m-2}, ...S_1$

Non-changing intervals in which the mildew neither grows nor dies Z

Levels of Disease Development $L_0, L_1...L_m$ where L_0 signifies healthy plants and L_m indicates plants beyond saving.

In order to create a transition function they define the inputs as: $X = \{S_{m-1}, S_{m-2}, ...S_1, Z, D_1..., D_{m-1}, D_m\}$ which can be simplified to $\{-m+1, -m+2, ...1, 0, 1, ..., m-1, m\}$

This allows them to define a DFA in the following manner

$$A = (Q, X, Y, q_0, \delta)$$

$$Q = \{q_0, q_2, q_3, q_4, q_5\}$$

$$X = \{a, b\}$$

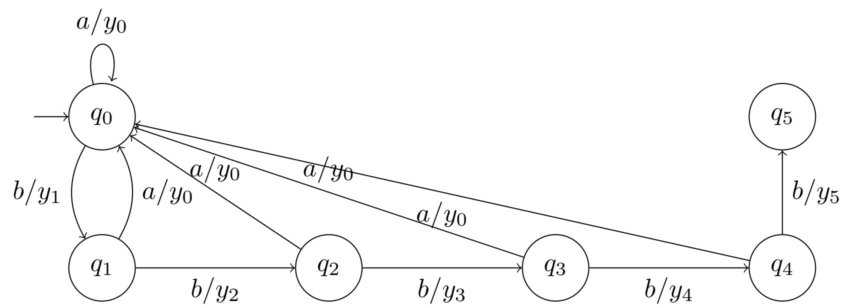
$$Y = \{0, 1, 2\}$$

$$\delta = \begin{cases} min(q+x, m), & 0 \leq x \leq m \\ max(q+m, 0), & x < 0 \\ -1, & x = m \end{cases}$$

- a: suppressive day
- b: dangerous day
- L_0: No danger to crops
- L_m: Crops are lost
- y0- "There is no danger for plants."
- y1- "The previous day was dangerous!"
- y2- "The two previous days were dangerous!"

- y3- ”Situation becomes critical!
- y4- ”Situation is crucial! Treatment mandatory”
- y5- ”Plants are irreversibly destructed!”

This formulation gives the following automaton

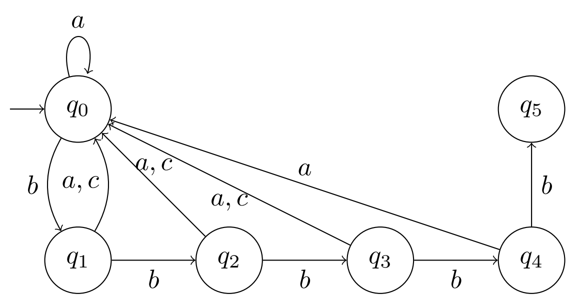


Results: “Installed in the field sensors ... measurements to the PC ... issued a daily prediction. ... The system successfully predicted 87% of the occurrences of mildew and two treatments which were not necessary”

5 Improvements

The above automata does model the system but only half of it as it leaves out an element of the model to model treating the crops. So I propose adding and input c, to indicate treatment of the crops. For the sake of a more interesting model I have chosen to show the treatment only being effective up until day q_3. I also propose simplifying the automata to remove the outputs as they are distracting and not necessary for the function of the model.

$$A = (Q, X, q_0, \delta)$$
$$Q = \{q_0, q_2, q_3, q_4, q_5\}$$
$$X = \{a, b, c\}$$
$$Y = \{0, 1, 2\}$$



Next I wanted to try to prove we could design a controller under which our system is safe and we never reach q_5. I did this in three different ways.

Defining input words: if we define w as the set of words on which our automaton functions we can define w such that:

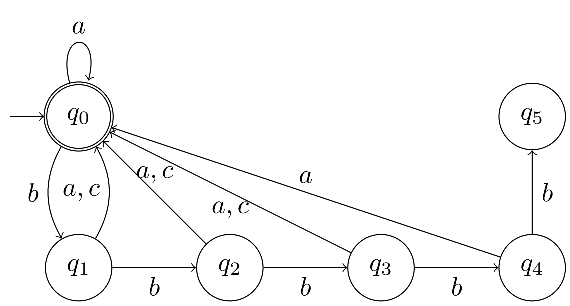
$$w = (b^{@}(a + c))^{*}\text{where @ indicates 0-3 occurrences of a given letter}$$

This results in words such as

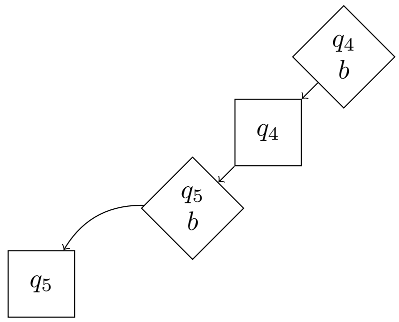
- bbba
- bc
- bbc

But excludes the word bbbbb which leads to our unsafe state of q_5

Infinite Automaton: Next I made the accepting state of the automaton q_0 and defined the automaton as infinite. This means that the automaton will only accept words in which we visit q_0 infinitely often and thus we must never visit q_5



Game Arena: Finally I used the game arena approach we talked about at the end of class to prove that we can design a controller that will never end up in q_5 because it can always choose option c when at q_3.



6 Conclusion

Though this exploration I have found several ways in which automata can be used to model plant behavior assuming the behavior can be sufficiently simplified. In that assumptions lies the biggest challenge to the utility of this approach as most natural behavior is anything but simple. Expanded models may be useful in prediction systems however automata are not likely to ever fully capture the vast complexity of plant behavior accurately.

7 Resources

- Introduction to symbolic dynamics and cellular automata. (2024). The Student Mathematical Library, 1–26. https://doi.org/10.1090/stml/108/01
- Kawano, T., Bouteau, F., Mancuso, S. (2012). Finding and defining the natural automata acting in living plants: Toward the synthetic biology for Robotics and informatics in vivo. Communicative amp; Integrative Biology, 5(6), 519–526. https://doi.org/10.4161/cib.21805
- Maneva, S., Manev, K. (2015). Finite automata models in agro-ecosystem and Plant Protection. International Journal of Computer Applications, 119(18), 1–6. https://doi.org/10.5120/21164-4223