λ parameter and sampling

Author(s): Koen Koning & Michael Lees

Objectives
This lab exercise has three objectives:

1. Develop a Python implementation of the Langton Parameter.
2. Recreate some of Langton's experiments for 1D CAs.
3. Explore different values of λ and its impact on dynamics.

Outcomes
By the end of the lab/assignment you should be able to:

1. Understand programmatically how to reconstruct a rule table Δ from λ.
2. Interpret behavior of λ with relationship to the Wolfram CA classification.

You should use the code you have developed in lab 2 as a starting point. To this you will add code that can generate CA rule tables from the Langton λ parameter. You should read Chris Langton's paper [1] before completing the assignment, accessible on the Google Classroom drive.

The goal

In order to implement the Langton experiments, a number of parameters should be added to the simulation. First, add a way to specify k, r and λ. Second, add two ways to specify the method for constructing the rule table, the *table-walk-through* method and the *random-table* method (see Langton sec 2.1).

The background

Read Langton's paper and understand why λ is interesting. Your objective is to implement a λ CA in Python and also perform analysis for varying λ. You can analyse the behavior of the CA qualitatively and/or quantitatively, using the same methods as earlier research (see [1, 2, 3]) or something different. The analysis part (below) will be left for you to decide what is interesting or relevant to show.

The implementation

You can use a great deal of the code from lab two as a starting point.

The main extensions are:

1. A function to build rule sets form input λ (using both methods)
2. Functions to measure CA behavior (e.g. information entropy)

That is all! It is probably best to focus on 1 first and then once you have a working CA read some of the material in the references at the end for ideas regarding measures.

The experiments

As you will most likely be initializing your CAs with random initial states you have to have some statistical confidence in your measurements. This may mean repeating a single parameter setting many times with different initial conditions. You can use the parameter sweep part of the framework, and set the number of repetitions to a number sufficiently high to minimize randomness.

Analysis

Your task is to decide on a suitable quantitative or qualitative measure of a 1D CA dynamics for varying values of λ. Take a look at [1], some suggestions might be:

- Shannon Information Entropy - Note that this can be done in many ways: per cell, per configuration, on the rules, etc.
- Transient Length - How long does the CA evolve before reaching a steady state or before it reaches a cycle. This can be approximated visually or calculated (can be hard to do - see [4])
- Build a Bayes classifier for the elementary CA rules using λ. Given a rule, you should be able to use this classifier to predict which class the rule is in. Also report on its accuracy. You will receive more points if you train the classifier using a test statistic (cycle length, transient length, entropy) instead of λ.

You should summarize the result of your analysis in a single figure including labeled axes and a figure caption. Be sure to show the repeated experiments (sampling) in the figure in some way, e.g., by using error bars or including the scatter points. Use the caption to explain how the measurements were done in a reproducible manner. The λ parameter should be part of your result figure (e.g., as x-axis).

Grading

You should submit only your program code Google Classroom. It should be functional and when run it should produce a single figure showing your analysis, including labeled axes and a figure caption with sufficient (reproducible) description (max. 250 words). Do not include a full written report. The figure may be shown on screen or it may be written to a file in the same directory.

Your grade will be based on, e.g., the correctness and interestingness of the result, the clarity of the figure, how you show the results of repeated experiments, whether you used a sufficient number of samples, and the quality of the figure caption in describing what we see and how it is computed.

Your code will be tested for plagiarism using special-purpose software.

This assignment will count 15% towards your grade for the practical assignments for the CA-part.

Deadline: Wednesday, December 6, 23:59.

References

[1] Chris G Langton. Computation at the edge of chaos: phase transitions and emergent computation. *Physica D: Nonlinear Phenomena*, 42(1):12–37, 1990.

[2] Kristian Lindgren and Mats G Nordahl. Complexity measures and cellular automata. *Complex Systems*, 2(4):409–440, 1988.

[3] Andrew Wuensche. Classifying cellular automata automatically. Santa Fe Institute Santa Fe, NM, 1998.

[4] Wentian Li and Mats G Nordahl. Transient behavior of cellular automaton rule 110. *Physics Letters A*, 166(5):335–339, 1992.