Honours Project: Does the Collatz-Algorithm show chaotic behaviour?

Co-Supervisor (CS): Prof. Stefan Gruner (email: sg@cs.up.ac.za),
Co-Supervisor (Math): Dr Ruaan Kellerman (email: ruaan.kellerman@up.ac.za).

(Inter-departmental Honours-Project: Mathematics & Computer Science)

The Collatz-Function in the Natural Numbers is based on the following very simple conditioned iteration: "If some input N>1 is even, let N be N/2 (and repeat), otherwise let N be 1+3N (and repeat), HALT when N=1".
The conjection by Lothar Collatz (1937), stating that this iteration will terminate down to N=1 for any natural input number, is formally-analytically un-proven to date, albeit it is empirically validated by every computer-experiment with every chosen input number made so far.
Thus the question may arise WHY was it not possible to formally-analytically prove the empirically validated Collatz-Conjecture from 1937 to date? What could be the reason for such stubborn un-provability since 80 years? Could the reason perhaps be CHAOS in the runtime-behaviour of the Collatz-Algorithm? Indeed we have literature which states that a structurally very similar function/algorithm, however defined for REAL input numbers (not natural numbers), is chaotic in its behaviour. Thus ---perhaps--- the original Collatz-Algorithm (defined for Natural Numbers), might be chaotic, too? --- which is also yet not known with certainty. Of course any answer (if any at all) to this question will also depend on how the very notion of 'chaos' is defined in the first place; thereby we want to work with one common notion of "chaos" which is well-known from the literature, and which is widely accepted, although a particularly 'keen' project-student could also come to criticize this very notion of 'chaos' (and perhaps provide a better one). A previous research project on the same topic ended inconclusive, though it made a 'valuable' start. In this project we want to continue where the previous project had ended without conclusion.
Students, who are interested in this project, should have basic knowledge in Number Theory, Numerical Computing, as well as in Computer Programming --- also including some familiarity with computer graphics software by means of which intuitive visual dot-plot graphics can be generated. Some initial literature will be provided, although the student is also expected to do some further literature-search and lots of theory-reading while at the same time doing some exciting computer-experiments. An outstanding honours-project on this topic, with an especially remarkable final result, could perhaps be extended into a follow-up Master-project.