High Order Fibonacci numbers

Johan N. Herbschleb

October 20, 2010

Abstract

Using L-systems to generate Fibonacci-like numbers leads to fascinating phenomena. Even Benford's law pops up.

1 L-system

Introduction and example from Wikipedia (http://en.wikipedia.org/wiki/L-system)

> An L-system or Lindenmayer system is a parallel rewriting system ... most famously used to model the growth processes of plant development, but also able to model the morphology of a variety of organisms.... L-systems were introduced and developed in 1968 by the Hungarian theoretical biologist and botanist from the University of Utrecht, Aristid Lindenmayer (1925–1989).

⋮

```
Example 2: Fibonacci numbers

If we define the following simple grammar:

    variables : A B
    constants : none
    start  : A
    rules  : (A → B), (B → AB)

then this L-system produces the following sequence of strings:

    n = 0 : A
    n = 1 : B
    n = 2 : AB
    n = 3 : BAB
    n = 4 : ABBAB
    n = 5 : BABABBAB
    n = 6 : ABBABBABABBAB
    n = 7 : BABABBABABBABBABABBAB
```

> ... each string is the concatenation of the preceding two ....

. . . , if we count the length of each string, we obtain the famous Fibonacci sequence of numbers:

1 1 2 3 5 8 13 21 34 55 89 ...

For $n > 0$, if we count the kth position from the invariant end of the string (. . . right in Example 2), the value is determined by whether a multiple of the golden mean falls within the interval (k-1,k). The ratio of A to B likewise converges to the golden mean.

This example yields the same result (in terms of the length of each string, not the sequence of As and Bs) if the rule (B → AB) is replaced with (B → BA).

This sequence is a locally catenative sequence because G(n) = G(n-2)G(n-1) where G(n) is the n^{th} generation.

A generalization of this system is formed by production rules like:

```
delay = 0 : (A → AA)
delay = 1 : (A → B), (B → AB)
delay = 2 : (A → B), (B → C), (C → AC)
delay = 3 : (A → B), (B → C), (C → D), (D → AD)
```

⋮

The numbers 0, 1, 2, 3, . . . are called 'delay' in forming the next 'generation' (in biological terms). A 'delay' of 0 simulates growth of bacteria, of 1 leads to Fibonacci numbers and a lot of interesting phenomena in biology like number of flowerets in a sunflower, but not to flowers with 4 petals. A 'delay' of 2 leads to the series: 1, 1, 1, 2, 3, 4, 6, 9, 13, . . .

2 Generating formula

The previous section can be summarized as follows, where n, g and k are natural numbers, n stands for the number of elements in generation g and k stands for the delay in generations before an element can divide itself.

$$n_g = 1 \qquad \text{for } g \leq k$$

$$n_g = n_{g-1} + n_{g-k-1} \qquad \text{for } g > k$$

Two special cases:

$k = 0$ gives a geometrical series

$k = 1$ gives the Fibonacci numbers

The generations k,. . . ,$2k + 1$ give the natural numbers 1,. . . ,$k + 2$.

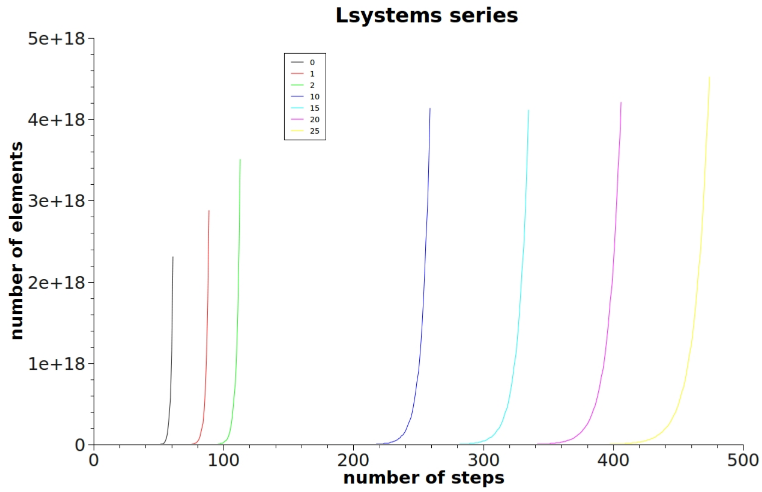

Figure 1:

3 Graphical representation

The number of elements was calculated for delays (k) 0,1,...,74 up to 500 generations (g) or the number 4611686018427387903 (half of the largest integer in my system), whichever came first.
Figure 1 shows the development of the series for some delays. After a large number (depending non-linearly on k) of steps all series seem to follow the same pattern.

In order to guess if $\lim_{g\to\infty} \frac{n_{g-1}}{n_g}$ exists, the relation $\frac{n_{g-1}}{n_g}$ for $k = 1, \ldots, 74$ was calculated for $g = 100$, $g = 300$ and $g = 500$ as long as $n_g < 4611686018427387903$. In case of a too large n_g the last calculated pair was used. See figure 2.

Clearly $\lim_{g\to\infty} \frac{n_{g-1}}{n_g}$ equals 0.5 for $k = 0$ and the golden ratio 0.618034 appears for $k = 1$. For the other delays the figure suggests strongly a tendency to a limit.

To get an idea of the density of the calculated numbers a scatter diagram was produced. For $1 \le n_g \le 10000$ these numbers were plotted on line 0, for $10001 \le n_g \le 20000$ on line 1, etc up to $n_g \le 10000000000000$.
See figure 3. Whether this distribution is denser than primes or not, I do not know.

4 Concluding remarks

The distribution of the first digit of calculated number of elements follows quite closely Benford's law:

```
freq (1) = 0.301      freq (2) = 0.177      freq (3) = 0.126
freq (4) = 0.096      freq (5) = 0.078      freq (6) = 0.066
freq (7) = 0.058      freq (8) = 0.052      freq (9) = 0.046
```

As I am not a mathematician, I do not have any proofs of my assertions.

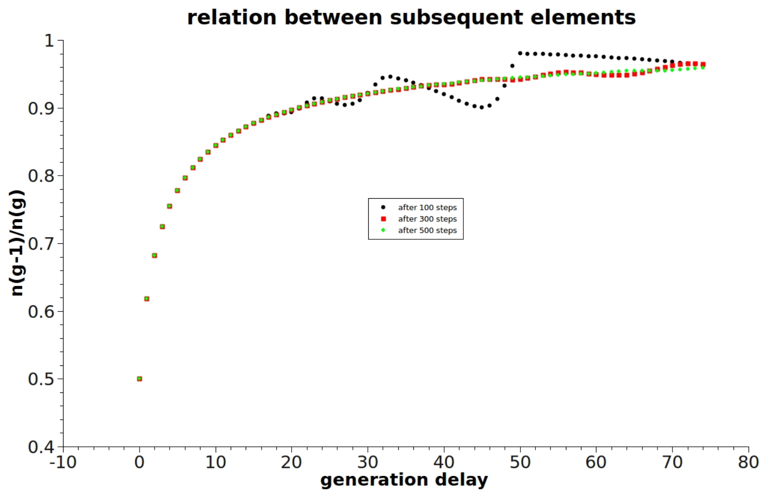

Figure 2:

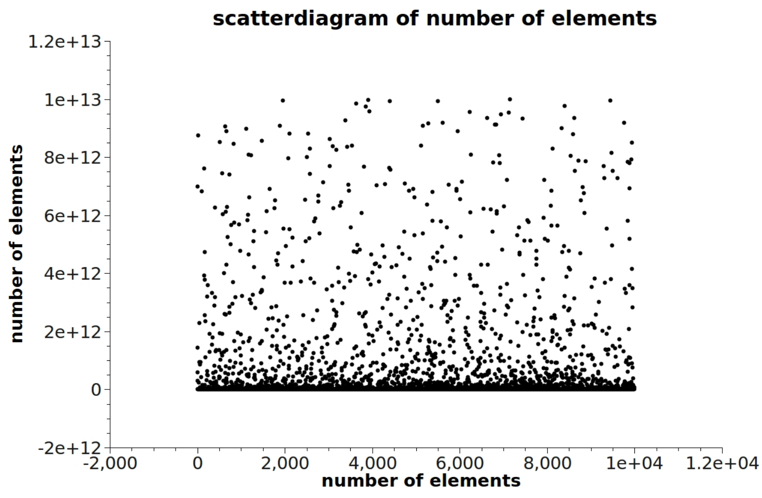

Figure 3: