Monografías del Seminario Matemático García de Galdeano. **27:** 89–95, (2003).

Chaotic dynamical systems, deceptive computers, and New Instructional Technologies

Marc Artzrouni

Laboratoire de Mathématiques Appliquées
Université de Pau et des Pays de l'Adour
BP 1155, 64013 Pau Cedex, France.
e-mail: marc.artzrouni@univ-pau.fr

Abstract

Chaotic dynamical systems are defined and illustrated with the "doubling function". After less than 60 iterations the orbits become 0. This is due to the binary coding of real numbers in a computer. The simulation dramatically shows that computers may give deceptive (false) results when calculations are done a large number of times. The convergence to 0 does not occur with the "tripling function" but results are equally inaccurate if not visibly so. The presentation illustrates the use of Matlab's "Notebook" in the classroom. Those can be used to present computer simulations "embedded" within a Word 97 document that contains the body of the lecture.

Keywords: Dynamical systems, chaos, iteration, simulation.
AMS Classification: 37-01, 37-04, 39A11

1 The "doubling function"

The "doubling function" is defined on (0,1) by Q(x)=2x for 0≤x≤0.5 andQ(x)=2x-1 for 0.5≤x≤1. The Matlab program that graphs this function is given below. This program is "embedded" in, and runs from a Word 97 document. The Matlab function "Jacax2n" found in the program is the actual Matlab routine (not given explicitly here) that gives the doubling function. The graphical output follows below (the first diagonal is also plotted).

```
[x=0:0.001:1;
n=1; % order of iteration
Y=[0 1];
subplot(1,1,1);
hold on;
xlabel('x','FontSize',14,'FontWeight','b');
ylabel('Q(x)','FontSize',14,'FontWeight','b');
plot(Jacax2n(n),Y,'b');
plot(x,x,'--k')
hold off ]
```

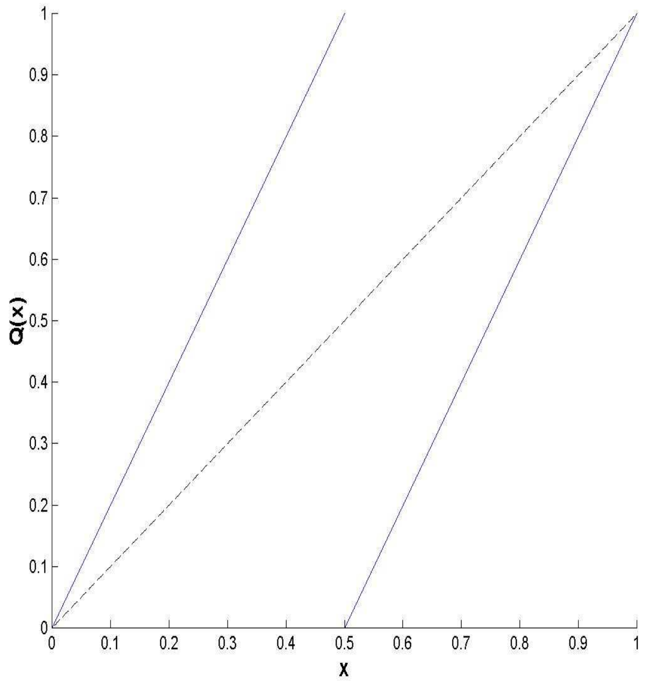

fig. 1 : Doubling function (and first diagonal)

2 Discrete dynamical systems

A discrete dynamical system is defined by a function Q(x), an initial point x(0), and the iterates generated by applying Q to X(0), then Q again to the result Q(X(0)), etc: x(1)=Q(x(0)), x(2)=Q(x(1)) and more generally x(n) = Q(x(n-1))=$Q^{(n)}$(x(0)) , n=1, 2, ...("$Q^{(n)}$" denotes the n-th iterate of Q). Again the program and the output are given below.

```
[x=0:0.001:1;n=1;
```

```
np=80;
npp=2*np-1;Td=zeros(1,np);Qd=zeros(1,np);
Xid=zeros(1,npp);Yid=zeros(1,npp);
% xid below: 0.95 periodic ; 0.1246 chaotic.....
xid=0.1246; yid=Jacadf(xid,1);Xid(1)=xid; Yid(1)=yid;Td(1)=1;Qd(1)=yid;
for k=1:np-1;
Td(k+1)=k+1;Xid(2*k)=yid;Yid(2*k)=yid;ypid=Jacadf(yid,1);Qd(k+1)=ypid;
Xid(2*k+1)=yid; Yid(2*k+1)=ypid; xid=yid;yid=ypid;
end
%
subplot(1,2,1);hold on;
xlabel('x','FontSize',14,'FontWeight','b');
ylabel('Q(x) + iterations','FontSize',14,'FontWeight','b');
plot(Jacax2n(1),Y,'b');plot(x,x,'--k',Xid,Yid,'r');
hold off;
subplot(1,2,2);hold on;
xlabel('n','FontSize',14,'FontWeight','b');
ylabel('Q^{(n)}(x)','FontSize',14,'FontWeight','b');
plot(Td,Qd)]
```

The program given above is run interactively in the classroom and different behaviors can be illustrated by simply changing the initial point xid, e.g. with xid=0.95 the dynamics appear to rapidly go into a cycle; with xid=0.1246 the dynamics (seen in the graph below) appear to be "chaotic" - in a sense of course to be defined - until the 55th iteration roughly, whern the orbit goes suddenly to 0. We next define "chaos" and investigate the apparent "convergence to zero" of the orbit.

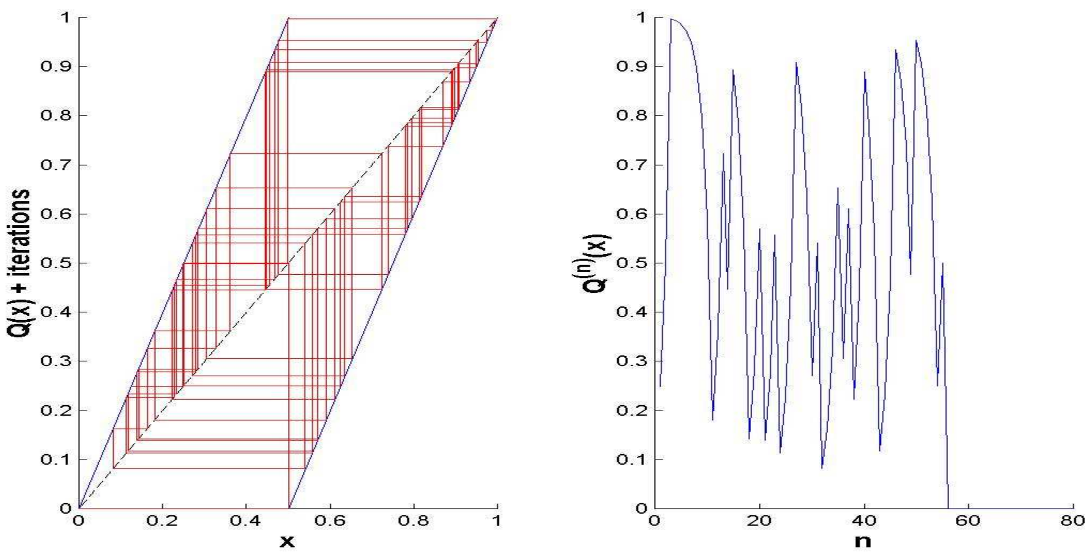

Fig. 2 : Phase space and time series representation

3 Chaotic dynamical systems

In the graph below we plot the third iterate $Q^{(3)}(x)$:

```
[clear;Y=[0 1];
x=0:0.001:1;
n1=3; % order of iteration
%df=Jacadf(x,n);
Y=[0 1];
subplot(1,1,1);
hold on;
xlabel('x','FontSize',14,'FontWeight','b');
ylabel('Q^{(n)}(x)','FontSize',14,'FontWeight','b');
plot(Jacax2n(n1),Y,'b');
plot(x,x,'--k')
hold off; ]
```

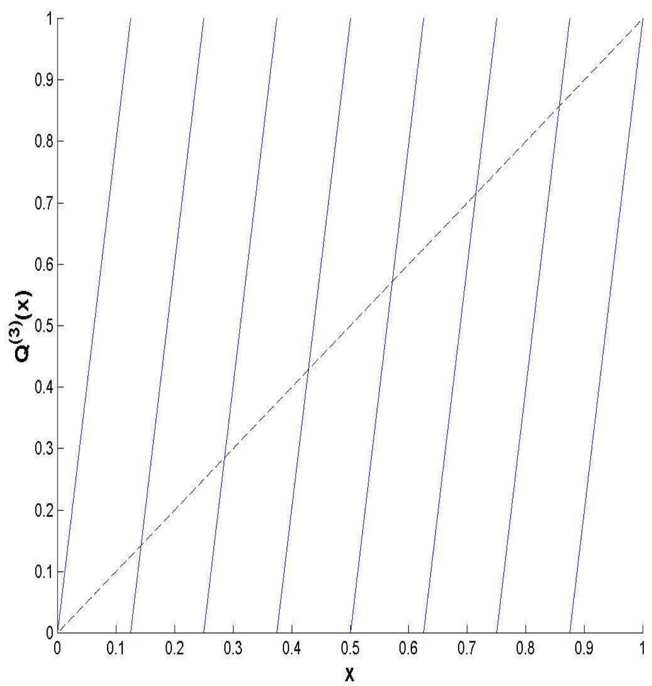

Fig. 3 : Third iterate $Q^{(3)}(x)$ of doubling function (and first diagonal)

Periodic points x of period three are the eight points for which $Q^{(3)}(x)=x$, i.e. where the graph of $Q^{(3)}(x)$ intersects the first diagonal. The reader is easily convinced that for

n going to infinity the function $Q^{(n)}$(x) consists of an increasing number of increasingly steep parallel segments. Periodic points are evenly distributed on the (0,1) interval and their number grows without bound when $n \to \infty$. The doubling function has the two defining properties of a chaotic dynamical systerm:

1. Periodic points are dense on the domain of the function.
2. For any two intervals I_1, I_2, there is $x \in I_1$ and n such that $Q^{(n)}(x) \in I_2$.

4 Apparent convergence to 0 of the orbit

Regardless of x(0) the computed orbit $Q^{(n)}$(x(0)) becomes 0 for n reaching approximately 55. This is due to the "finite binary coding" of a number x between 0 and 1. Indeed, an initial value x(0) between 0 and 1 is approximated as

$$x(0)_c = \frac{a_1}{2} + \frac{a_2}{2^2} + ... + \frac{a_k}{2^k} \tag{1}$$

where $a_p = 0$ or 1 for all p, k is say 50, and here and elsewhere a subscript "c" means "the "coded" value of". The "code" for x(0) is thus the finite binary sequence (a_1, a_2, ...,a_k). If x(0)≤0.5 then $a_1 = 0$ and

$$\begin{aligned} Q(x(0)_c) &= 2x(0)_c = 2\left(\frac{a_1}{2} + \frac{a_2}{2^2} + ... + \frac{a_k}{2^k}\right) = \\ &\frac{a_2}{2} + \frac{a_3}{2^2} + ... + \frac{a_k}{2^{k-1}} + \frac{0}{2^k} \end{aligned} \tag{2}$$

so that the "code" for the first iterate Q(x(0)$_c$) is $(a_2, a_3, ..., a_k, 0)$. The reader can easily verifiy that the same holds true for x¿0.5. If "$\stackrel{c}{=}$ means "has the coded representation" we then have

$$x(0)_c \stackrel{c}{=} (a_1, a_2, ..., a_k) \tag{3}$$

$$Q(x(0)_c) \stackrel{c}{=} (a_2, a_3, ..., a_k, 0) \tag{4}$$

$$Q^{(2)}(x(0)_c) \stackrel{c}{=} (a_3, a_4, ..., a_k, 0, 0) \tag{5}$$

and finally

$$Q^{(k-1)}(x(0)_c) \stackrel{c}{=} (a_k, 0, 0, ..., 0); \qquad Q^{(k)}(x(0)_c) \stackrel{c}{=} (0, 0, 0, ..., 0); \tag{6}$$

This shows that as n increases, the "coding" by the computer of the iterates becomes increasingly inaccurate. The "coded" value of the k-th iterate is 0 and remains 0 for iterates of higher order.

An obvious question is whether there is an analytical expression for $Q^{(n)}(x)$, which would make it possible to calculate $Q^{(n)}(x)$ exactly and compare the result with the computed value. The doubling function is simple and an expression for $Q^{(n)}(x)$ can be written down - however that expression does not help much. Indeed, when trying to evaluate that function, numerical problems arise, similar to the ones encountered in the iteration. It thus seems very difficult to compute a 60th iterate of the doubling function - except of course with a more powerful computer - or one that does exact "rational arithmetic". However, any computer has a finite precision and in the case of the doubling function it may be the 500th or the 5,000th iterate that becomes 0, but eventually the orbit does reach 0.

5 The tripling function

The tripling function R(x) is defined as R(x)=3x, or 3x-1, or 3x-2 depending on whether $0 \leq x \leq 1/3$, $1/3 < x \leq 2/3$; $2/3 < x \leq 1$.

```
[x=0; x=0:0.001:1; n=1;
np=70;npp=2*np-1; T=zeros(1,np);
Qit=zeros(1,np); Xi=zeros(1,npp); Yi=zeros(1,npp);
xi=0.64; yi=Jacatf(xi,1); Xi(1)=xi; Yi(1)=yi;
T(1)=1;Qiter(1)=yi;%
for k=1:np-1;
T(k+1)=k+1;
Xi(2*k)=yi;Yi(2*k)=yi;
ypi=Jacatf(yi,1);
Qiter(k+1)=ypi;
Xi(2*k+1)=yi; Yi(2*k+1)=ypi;
xi=yi;yi=ypi;
end
%
subplot(1,2,1); plot(Jacax3n(1),Y,'b');
hold on;plot(x,x,'--k',Xi,Yi,'r');hold off;
subplot(1,2,2); plot(T,Qiter); ]
```

With the tripling function, there is no convergence of the orbit to 0, but results are equally inaccurate after 55 iterations - but no visibly so as with the doubling function since there is no convergence to 0 of the "coded iterates".

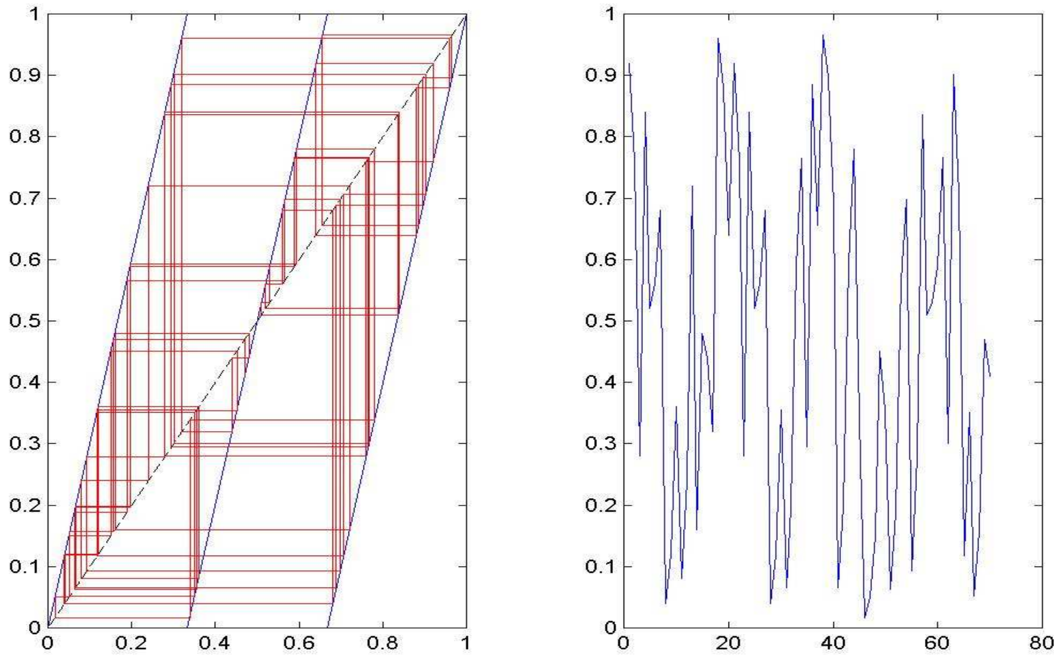

Fig 4. : Phase space and time series representation (tripling function)

6 Conclusion

In this paper we have briefly outlined an example from chaotic dynamical systems of the use of Matlab "Notebooks" in the classroom. This system has the advantage that a computer program can be embedded directly into a lecture prepared with Word 97. Simulations are done live, in real time, and concepts graphically illustrated. Ideally, the file would be made available to students who could then experiment further themselves with the programs contained in the lecture.

References

[1] K. Alligood, T. Sauer, J. Yorke (1996) *Chaos : An Introduction to Dynamical Systems* (Textbooks in Mathematical Sciences). Springer-Verlag, Berlin-New York.

[2] M. Barnsley (1993) *Fractals Everywhere*. Morgan Kaufmann Publishers, New York.

[3] R.L. Devaney. *A furst course in chaotic dynamical systems, Theory and experiments*. Addison-Wesley, Reading, Mass. (1992).

[4] HO Peitgen, H. Jurgens, and D. Saupe. *Chaos and fractals.* Springer Verlag. New York, Berlin. 1992.