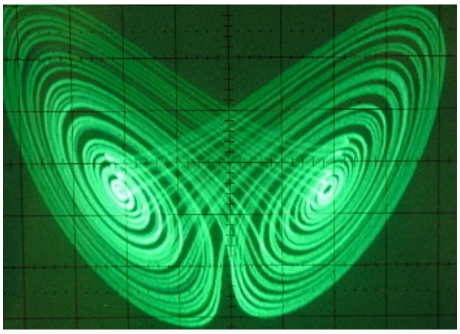
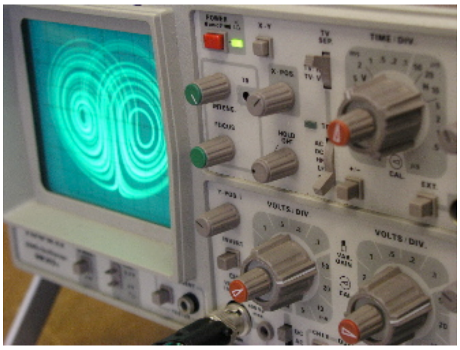
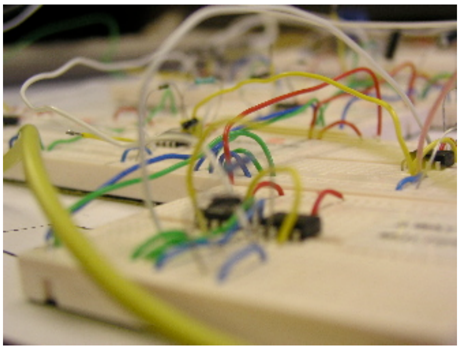
Analogue butterflies

Maarten Ambaum







To celebrate the Meteorology Department's 40th anniversary I designed an analogue computer to simulate the Lorenz 1963 model which produces the butterfly that became the cartoon for the science of chaos and the (un-)predictability of weather. Analogue computers are wonderful machines. They have some remarkable properties.

An analogue computer solves differential equations by representing values of variables by voltages in a circuit. Wires connect modules that perform specific arithmetic operations. For example, a subtraction module will have two input connectors and one output connector where the output voltage equals the difference between the input voltages (this particular module is in fact simply a differential amplifier with unit gain). The topology of an analogue computer is similar to that of our brain with the axons being represented by the wires, the cell body by the arithmetic modules, and the input ports by the dendrites.

Contrast this with a digital computer. In a digital computer variables are stored in memory spaces which are then occasionally operated upon by copying these memory spaces to the central processor which then changes the values of variables in other memory spaces.

Digital computers only change values of variables if the central processor says this should happen. In an analogue computer values are always consistent. There is no internal clock speed; calculations happen instantaneously. So, if for three variables a, b, and c we have $a+b=c$ then in an analogue computer this is always the case. In a digital computer this is only valid after the central processor has performed this addition and then only until either a or b are updated again.

Time integration is also a very natural process for an analogue computer. The input and the output voltages of a time integrating module are always consistently related: at all times the output is equal to the time-integral of the input. There are no time steps involved. Numerical instability of time-stepping routines is not an issue.

Analogue computers have no memory. This makes them essentially equivalent to the systems we try and simulate. A swinging pendulum does not have a memory of its previous states. One could connect a computer with analogue–to–digital converters if memory or exact measurements are required, or even a tape recorder if you want to stay away from anything digital.

Analogue arithmetic

Integration in time occurs by converting a voltage to a current through use of an operational amplifier and then using this current to charge a capacitor. The voltage over the capacitor is now the time-integrated value of the input voltage. Other arithmetic operations are also performed with the help of operational amplifiers. For example, subtracting two voltages happens through a differential amplifier of unit gain. Multiplication and other related operations are more difficult to implement. Dedicated analogue chip are used which contain log and antilog converters.

Dr Maarten Ambaum is a lecturer at the Department of Meteorology. He is interested in the fluid dynamics of atmosphere and oceans.
m.h.p.ambaum@reading.ac.uk

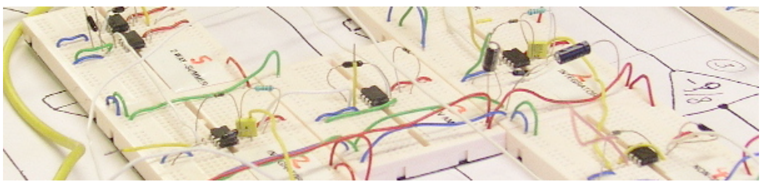
Analogue computers are hard to program: programming the computer is the same as building the computer. Clearly this is where digital computers are superior. An analogue computer is in effect an electronic copy of the system we try and simulate. So if we want to simulate a swinging pendulum we build an electronic system that oscillates exactly like the swinging pendulum. The computer has now become an electronic version of the swinging pendulum itself.

This is a very interesting property of analogue computers. Think of the Lorenz 1963 system. Apart from a very artificial set-up, there is no actual physical representation of the Lorenz system; it was designed as a mathematical system. Analogue computers are the only way we can get useful physical representations of such mathematical systems.

People who see an analogue computer for the first time often ask: how fast is it compared to a modern digital computer? In fact, their speeds are hard to compare. In a digital computer speed is limited by the clock speed of the processor and the speed at which variables can be loaded into and out of the processor. One such calculation may typically take a nanosecond or so (a billionth of a second). In an analogue computer the speed is limited by the speed at which the operational amplifiers can follow changes in input voltages (the *slew rate*). Operational amplifiers can change over time scales of a few nano-seconds. However, analogue computers do not perform calculations as such; they perform simulations. Asking how fast an analogue computer calculates is the same as asking how fast the swinging pendulum calculates how it moves.

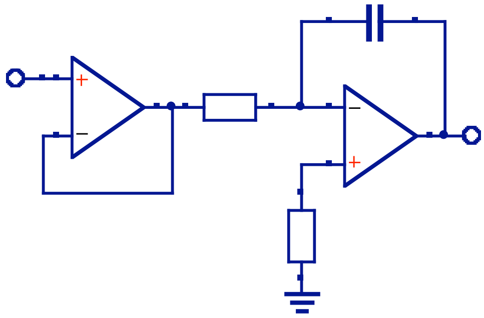
I have been mainly discussing an ideal analogue computer where electronic components are perfectly specified, have no noise, no drift, no temperature dependence, and no operating limits. The real world is not like that. Building an analogue computer requires knowledge of the component's limitations and the system's behaviour to make it work correctly. However, electronics have improved so much that modern analogue computers are much more stable than the early analogue computers.

I cheated in designing this computer: I used my Mac to simulate the likely behaviour of my analogue computer before it was built. I am in good company though: Seymour Cray used his Mac to simulate the next generation of Cray supercomputers.



$$
\begin{aligned}
\frac{dX}{dt} &= \sigma(Y - X)\\
\frac{dY}{dt} &= \rho X - Y - XZ\\
\frac{dZ}{dt} &= XY - \beta Z
\end{aligned}
$$

The Lorenz equations: E. N. Lorenz: "Deterministic nonperiodic flow" J. Atmos. Sci., *20 (1963), 130–141*.



Schematic of an inverting integrator. The output voltage equals minus the time integral of the input voltage

Dr Giles Harrison provided guidance in the design and the logistics. The prototype (pictured) was built in the Meteorology Department electronic workshop by Stephen Tames.