Bugs, Blobs, and Rotor-Routers

Jim Propp

(based on articles in progress by Ander Holroyd, Lionel Levine, and Jim Propp)

All these slides are on the web at `http : // jamespropp.org/bugs.pdf` so there's no need to take notes on anything you see here (only on the things that I say that you don't see!).

Puzzle #1: Each number between 1 and 5 is equipped with a light, which can be green or red. A bug is dropped on 3 and obeys the following rules at all times: if it sees a green light, it turns the light red and moves one step to the right; if it sees a red light, it turns the light green and moves one step to the left.

Show that the bug must eventually leave the system (either by exiting 1 heading to the left, or by exiting 5 heading to the right), and give a simple rule for predicting which of the two outcomes will happen.

Could the bug stay trapped in 1,2,3,4,5 forever, never escaping to 0 or 6?

If so, there must be some site that the bug visits infinitely often (3, say).

But then it must visit site 4 infinitely often (since half of the time when it leaves 3 it goes to 4).

But then it must visit site 5 infinitely often (since half of the time when it leaves 4 it goes to 5).

But on its first or second visit to site 5, the bug must go off to the right!

Contradiction!

So the bug must eventually escape.

But which way will the bug escape?

The trick to figuring this out is to notice that the the quantity

(NUMBER OF GREEN LIGHTS)

plus

(POSITION OF THE BUG)

is *invariant*:

If a green light turns red (and the bug goes right), the number of green lights goes down by 1, but the position of the bug goes up by 1.

If a red light turns green (and the bug goes left), the number of green lights goes up by 1, but the position of the bug goes down by 1.

In particular, if the bug goes from 3 to 0 (that is, it leaves the system heading left), then number of green lights must increase by 3; this can't happen if the number of green lights was 3, 4, or 5 to begin with.

On the other hand, if the bug goes from 3 to 6 (that is, it leaves the system heading right), then number of green lights must decrease by 3; this can't happen if the number of green lights was 0, 1, or 2 to begin with.

So, the bug must exit to the right if the green lights outnumber the red lights, and to the left if the red lights outnumber the green lights.

The order of the lights turns out not to matter; only how many of each kind there are.

Note that if you add a second bug to the system, it will exit the system on the opposite side. (If the first bug exited left, the second will exit right; if the first bug exited right, the second will exit left.)

If you add a third bug to the system, it will do the opposite of what the second bug did, that is, it will do the same as what the first bug did.

If you add lots of bugs to the system, one at a time, half of them will exit the system to the left and half will exit to the right.

Puzzle #2: Each positive integer on the number line is equipped with a green, yellow or red light. A bug is dropped on 1 and obeys the following rules at all times: if it sees a green light, it turns the light yellow and moves one step to the right; if it sees a yellow light, it turns the light red and moves one step to the right; if it sees a red light, it turns the light green and moves one step to the left.

Eventually the bug will fall off the line to the left, or run out to infinity on the right. A second bug is then dropped on 1, then a third.

Prove that if the second bug falls off to the left, the third will march off to infinity on the right.

Trick: Think of a green light as the digit 0, red as the digit 1, and yellow as the "digit" 1/2. The configuration of lights can then be thought of as a number between 0 and 1 written out in binary,

$$x = .x_1 x_2 x_3 \ldots$$

where, numerically,

$$x = x_1 \cdot (1/2)^1 + x_2 \cdot (1/2)^2 + \cdots \ .$$

This is the "value" of the lights.

Think of the bug itself as having value $(1/2)^i$ when it is in position i.

Then the value of the lights plus the value of the bug is invariant, that is, it does not change as the bug moves.

If you add more bugs, you find that half of them exit the system to the left and half of them exit the system to the right.

For more details, see Peter Winkler's recent book *Mathematical Puzzles: A Connoisseuer's Collection*. (See the Bugs on a Line problem on page 82, with solution on pages 91–93.)

Puzzle #3: Each positive integer on the number line is equipped with a blue or yellow light. All lights are initially blue. A bug is dropped on 1 and obeys the following rules at all times: if it sees a yellow light, it turns the light blue and moves one step to the right; if it sees a blue light, it turns the light yellow and moves TWO steps to the left.

Eventually the bug will fall off the line to the left, landing at either -1 *or 0. A second bug is then dropped on 1, then a third, and so on. Each successive bug that is added falls off the line to the left, landing at either* -1 *or 0. (Prove this!)*

Show that the number of bugs that land at -1*, divided by the number of bugs that land at* 0*, converges to* $\Phi = (1+\sqrt{5})/2 = 1.618...$*, the "golden ratio".*

Trick: Once again, you construct an invariant; this is not built on base 1 (like Puzzle #1) or base 2 (like Puzzle #2), but on "base Φ" (or, alternatively, "base Fibonacci").

For more details, see Michael Kleber's article "Goldbug Variations" in the Winter 2005 issue of *The Mathematical Intelligencer* (available January 2005).

Q. Why are these puzzles interesting?

A. They illustrate the way in which quasirandom walk mimics properties of random walk.

The building-blocks for these gadgets are called *rotor-routers*.

Machines built out of rotor-routers are *deterministic*: their behavior does not involve any element of chance.

That is: you can predict in advance what they will do.

Equivalently: if you start two copies of the system in the same initial state, they will evolve in exactly the same way.

So these systems are not random.

But what if you randomize them?

E.g., for Puzzle #1, the bug just chooses randomly at each time-step whether to go right or left.

Fact #1: If a bug starting from 3 does random walk on $\{1, 2, 3, 4, 5\}$, where its chance of jumping one step to the left and its chance of jumping one step to the right are both equal to 1/2 ("random walk"), then the bug has a 100% chance of eventually escaping, and its chance of escaping to the left and its chance of escaping to the right are both 1/2.

Fact #2: If a bug starting from 1 does random walk on $\{1, 2, 3, \ldots\}$, where its chance of jumping one step to the left is $1/3$ and its chance of jumping one step to the right is $2/3$ ("biased random walk"), then the bug has a 50% chance of eventually escaping to the left and a 50% chance of wandering off to the right.

Fact #3: If a bug starting from 1 does random walk on $\{1, 2, 3, \ldots\}$, where its chance of jumping two steps to the left and its chance of jumping one step to the right are both equal to $1/2$, then the bug has a 100% chance of eventually landing on -1 or 0; the probability of landing on -1 is $1/\Phi$ and the probability of landing on 0 is $1/\Phi^2$. (Note: $1/\Phi + 1/\Phi^2 = 1$.)

Q. Why the golden ratio?

Assume that we already know that with probability 1, the bug eventually lands on either -1 or 0.

Let p be the probability that the bug eventually lands on -1, so that $1-p$ is the probability that it eventually lands on 0.

If we start at 1 then with probability 1/2 we go immediately to -1, and with probability 1/2 we go immediately to 2. So

$$p = (1/2)(1) + (1/2)(q),$$

where q is the probability that the bug will eventually land on -1 if it starts from 2.

We have assumed that we already know that with probability 1, a bug that starts at 1 eventually lands to the left of 1.

It follows that a bug that starts at 2 will eventually land to the left of 2.

Furthermore, when the bug first lands to the left of 2, with probability p it lands on 0, and with probability $1 - p$ it lands on 1.

So

$$q = (p)(0) + (1 - p)(p).$$

We now have two equations and two unknowns.

Substituting and solving, we get

$$p + p^2 = 1.$$

Q. In what way is quasirandomness better than randomness?

A. The Law of Large Numbers "kicks in sooner".

For Fact #1, if we put N bugs through the random system, the number of bugs that escape to the left will be about $N/2$, with a typical error on the order of $\sqrt{N}$.

But if we put N bugs through the non-random system of Puzzle #1, the number of bugs that escape to the left will be about $N/2$, with an error no greater than $1/2$.

Ditto for Fact #2 and Puzzle #2 (replacing "no greater than $1/2$" by "no greater than 1").

To see how Puzzle #3 goes, use the applet

$$\texttt{http}://\texttt{www.math.wisc.edu}/\sim\texttt{propp}/\texttt{rotor}-\texttt{router}-\texttt{1.0}/$$

with the Graph/Mode selector set to "1-D Walk".

Set the Graph/Mode selector to "2-D Walk" to see a quasirandom gadget for approximating $\pi/8$.

Set the Graph/Mode selector to "1-D Aggregation" to see a quasirandom gadget for approximating $\sqrt{2}$.

Finally, set the Graph/Mode selector to "2-D Aggregation" to see a quasirandom gadget for growing circular blobs. See

`http://www.math.wisc.edu/~propp/million.gif`

Lionel Levine and Yuval Peres proved in 2005 that these blobs really do become true circles in the limit. But the internal structures are still completely mysterious.

(This version of the rotor-router applet was created by UW undergrads Hal Canary and Francis Wong.)