Tech Know Learn (2011) 16:193–198
DOI 10.1007/s10758-011-9184-z

Computational Diversions: Slamming Door Comedy

Michael Eisenberg

Published online: 7 October 2011

Puzzles are an odd form of literature. For the most part, puzzle-writers—even the best of them—simply aren't concerned with narrative, plot, character development, or even simple plausibility. Take, for example, the "island of truth-tellers and liars" found in so many logic puzzles: what kind of a place would really and truly be populated with (exclusive) truth-tellers and liars, anyhow? Could such people exist? (Even my best friends aren't *exclusive* truth-tellers, and even my least favorite political figures aren't *exclusive* liars.) And squeezed together on an island, no less—wouldn't that lead to civil war? Or the farmer crossing the river—what's the man doing walking around with both a fox and a goose? Maybe he should just invest in a muzzle for the fox?

I was thinking about these matters the other day when browsing a fantastic puzzle book by Peter Winkler, entitled *Mathematical Puzzles: a Connoisseur's Collection* (Winkler 2004). The book is a gem, and (among the relatively simpler challenges) it includes the following:

> *Locker Doors.* Lockers numbered 1 to 100 stand in a row at the school gym. When the first student arrives, she opens all the lockers. The second student then goes through and recloses all the even-numbered lockers; the third student changes the state of every locker whose number is a multiple of 3. This continues until 100 students have passed through. Which lockers are now open? (Winkler 2004, p. 13)

Rather than provide the answer to this puzzle, I will instead direct the interested reader to Winkler's book; and if that constitutes a plug for the book, so much the better. But now, having justly praised Winkler's collection, I can perhaps be forgiven a literary critique. Who exactly are these students, running around opening and closing lockers? As a college professor, I've become accustomed to the inexplicability of teenage behavior, but this is just plain weird. No one runs around opening and closing doors, do they?

Well, perhaps they do, at that. In classic stage comedies (particularly the kind associated with the nineteenth-century French playwright Georges Feydeau), characters are always running about, opening and closing doors in complex mathematical patterns, just as in

M. Eisenberg (✉)
University of Colorado, Boulder, CO, USA
e-mail: ijcml-diversions@ccl.northwestern.edu

Winkler's puzzle. In a Feydeau farce, the doors aren't locker doors; rather, the characters keep entering and exiting the stage by moving between rooms while chasing each other, hiding from jealous spouses, and so forth. The form is so established that stage farces of this kind are in fact often called "slamming door comedies", since so much of the humor derives from the exquisitely precise timing of multiple doors opening and closing simultaneously.

With this as prelude, perhaps we can build a little program—a source of puzzles and challenges like that provided by Winkler—based on the idea of slamming door comedy. Here's the basic set-up: we imagine a row of n doors, each of which can be open or closed. Scattered among the doors are a group of p players, numbered 1 through p. At any given time state of the system, a player can be behind only one door; and multiple players can be behind a single door. Thus, to take an example: Fig. 1 shows a representation of a set of six doors and three players. At the particular time shown, players 1 and 2 are behind Door 3 (which is closed), and player 3 is behind Door 5 (which is open).

So far, our "slamming door comedy" is still rather static. We now introduce an element of movement into our program. Each player is given its own particular rule, the general form of which is as follows:

At a given time T, do the following:

a. Check to see if my current door is open. If not, we do not move (and in that case, we needn't bother with steps b–e below).
b. If the door *is* open, this player p will exit the door.
c. The player may choose to close the door upon exiting.
d. The player p now moves *delta* doors over. *Delta* may be a positive or negative number, and it is specific to each player (it might thus be best represented with a subscript, as $delta_p$).
e. Having moved to a new door, the player may choose to close the newly-entered door, or leave it open.

Thus, each player is essentially characterized by three parameters: the number of doors to move when its door is open, whether to close (or leave open) the starting door, and whether to close (or keep open) the newly-entered door. After all players have run their individual rules in succession, we have completed one time-step for the system, the time effectively increments to $T + 1$, and the players try their rules all over again at the next time step.

Suppose, for example, in our Fig. 1 system, Player 3 has the rule that says: leave your starting door open, move -2 spaces, and leave the new door open. Then, at the end of one time step, all three players would be behind Door 3 (which would now be open), as in Fig. 2.

This description pretty much covers the entire system, though if you try to write a program along these lines you'll find (as one always does) that there are various finer points to consider. For example, we'll stipulate that the doors have "wraparound": that is, a player who moves forward past the rightmost door reappears at the left of the row, and a

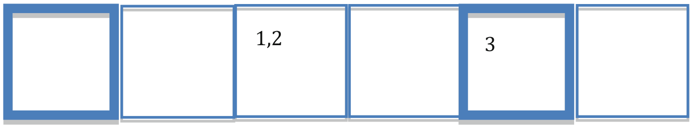

Fig. 1 A row of six doors. Open doors are represented with *thick* boundaries; closed doors with *thin* boundaries. Here, players, 1 and 2 are behind (closed) Door 3, and player 3 is behind (open) Door 5

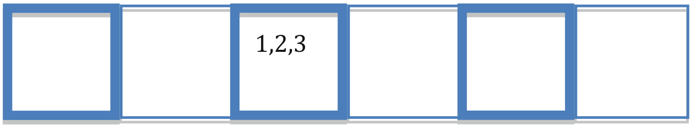

Fig. 2 Continuing from the situation in Fig. 1. Player 3 moves two spaces left, leaves the original door (Door 5) open, and leaves the newly-entered door (Door 3) open. Players 1 and 2 did not move, since (in Fig. 1) their door is closed

player who moves backward past the leftmost door reappears at the right end. (Another way of phrasing this is that we actually have a "ring" of doors rather than a bounded line of them.) We'll also stipulate that each player finishes its turn before the next one begins. Thus, in Fig. 2, suppose Player 1 has the following rule: close your starting door, move one space to the right, and close the new door. Then, at the next time step, Player 1 would move first and close Door 3 in the process, keeping Players 2 and 3 from moving. The result would look like Fig. 3, and from this point on, no player can move and the entire system is effectively over. (Or, as a playwright might put it, *finita la commedia.)*

The overall system bears some resemblance to a "standard" one-dimensional cellular automaton of the type discussed in (Wolfram 2002), but it exhibits some important differences as well. We can think of the state of any particular door as being characterized not only by the "open-closed" dimension, but also by the number and identity of players behind the door. For a given slamming-doors system with d doors and p players, we find that there are: $2^d \times d^p$ distinct states, assuming that there is no wraparound and all the doors are distinct: the first factor accounts for all the open/closed door states, and the second factor for all the possible player configurations. If we allow for wraparound, then in effect we reduce the number of states by a factor of d, since now we have d choices of which door to call "number 1"; the number of distinct states is now $2^d \times d^{(p-1)}$. Even for our little system of six doors and three players, then, there are 2,304 distinct states (taking wraparound into account), though many of them will be "inert" like the one in Fig. 3.

In the spirit of play, I wrote a small slamming-doors simulation (in Scheme) and have spent some time playing with it. It quickly became apparent that, even for a relatively small number of doors and players, there is a lot to experiment with. In my system, I chose to use twelve doors and four players. Figure 4 represents a thirty-step simulation (up to $T = 29$) in which all four players begin behind Door 1, which is open. The rules are as follows:

Player 1: Leave door open, move one space right, close door.
Player 2: Leave door open, move five spaces right, leave door open.
Player 3: Close door, move two spaces right, leave door open.
Player 4: Close door, move three spaces right, close door.

The figure shows the set of twelve doors at successive time steps: the top row is the initial time-step, the row below that represents the situation at $T = 1$, the row below that

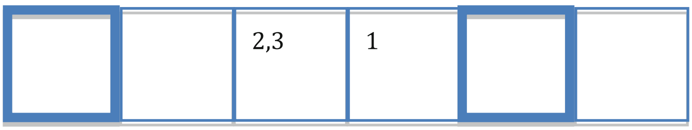

Fig. 3 Continuing from Fig. 2. Player 1 moves one space to the right, closing its starting door (Door 3) as it leaves. This prevents players 2 and 3 from running their programs, since their door is now closed. From this point on, the system will remain static

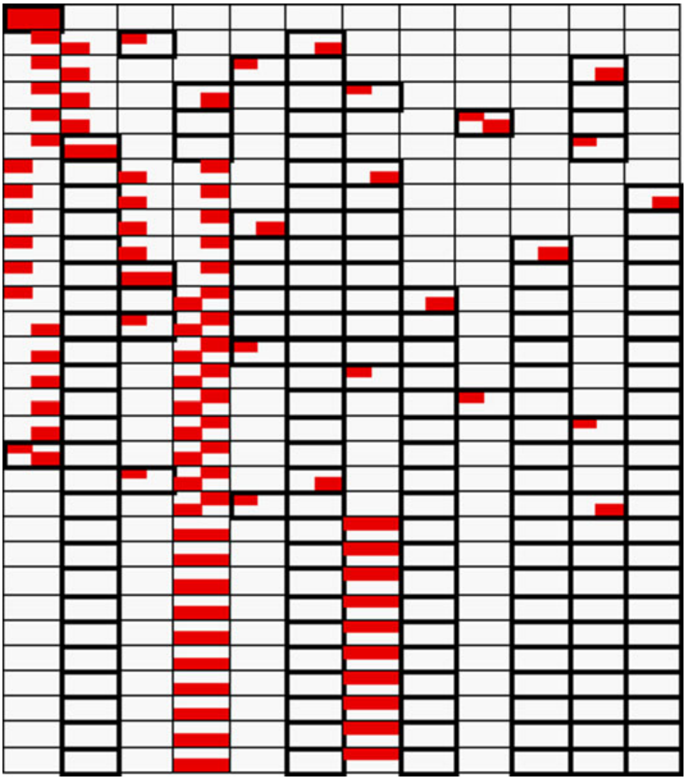

Fig. 4 A slamming-doors simulation in which pairs of players end up trapped after 20 time-steps. Closed doors are shown in *thin lines*; open doors in *thick lines*. Player 1 is a *filled bottom left-hand quarter rectangle*; player 2 a *filled bottom right-hand rectangle*; player 3 the *top left-hand rectangle*; and player 4 the *top right-hand rectangle* (see the more thorough description of this run in the text)

the situation at $T = 2$, and so forth. As in the earlier figures, a thick boundary represents an open door, and a thin boundary a closed door. Within each door, a filled quarter-rectangle represents one of the players: the bottom left-hand quarter is player 1, the bottom right-hand quarter is player 2, the top left-hand quarter player 3 and the top right-hand quarter player 4. Thus, in Fig. 4, you can see that players 1, 2, and 3 move from door 1 at the first time-step; since player 3 closes the door upon leaving, player 4 is blocked from moving at this step. Player 4 finally does get to move at $T = 6$, when player 3 arrives back at the starting door and leaves it open just in time for player 4 to move out. At $T = 20$ the entire system has arrived at a static configuration, in which players 1 and 2 are trapped behind closed Door 4, and players 3 and 4 are trapped behind closed Door 7.

If the players have a general tendency to close doors (either the door they just left or just entered), slamming-doors runs will generally end in static configurations like the one in Fig. 4. Often, pushing the system in a slightly more open-door direction keeps the players moving. For example, in Fig. 5, we show yet another slamming-door run: in this one, player 4 now closes the door it is leaving, moves six spaces right, and leaves the door of its arrival open (the other three players are identical). This less-stuffy player 4 now causes the system to keep churning for at least thirty steps. The new run begins (for the first five time-steps) with the same configurations as the previous run, since player 4 doesn't actually get a chance to move until just at the end of $T = 6$.

Conceivably, slamming doors simulations could lead to informative mathematical investigations. In playing with the system, the idea of relative primality is quickly foregrounded: after all, if you have twelve doors, then only players who move in jumps that are relatively prime to twelve (namely, players 1 and 2 in our original simulations) can possibly visit every door. They may *not*, as things happen, actually visit each door if they happen to get trapped at an early time-step. In Fig. 4, for instance, player 1 only manages to inch three doors over to the right before being trapped.

A more sophisticated experimenter might choose to explore notions such as the entropy of a simulation. The concept of entropy for one-dimensional cellular automata is thoroughly discussed in Wolfram's aforementioned book [especially the notes toward the end of the book; see also (Wolfram 1984)], and there are a variety of alternative entropy measures for these systems; the intent of all these entropy measures is to get a sense of the unpredictability or "disorder" of a given simulation. Intuitively, for our slamming door simulations, we would like to say that the Fig. 4 system has a low entropy value (since it's ultimately highly predictable), and that in Fig. 5 has a higher entropy value.

For our purposes, we can employ the very simplest definition of entropy, based on ideas from information theory. Let's illustrate the idea using the run of Fig. 5. Take a look at

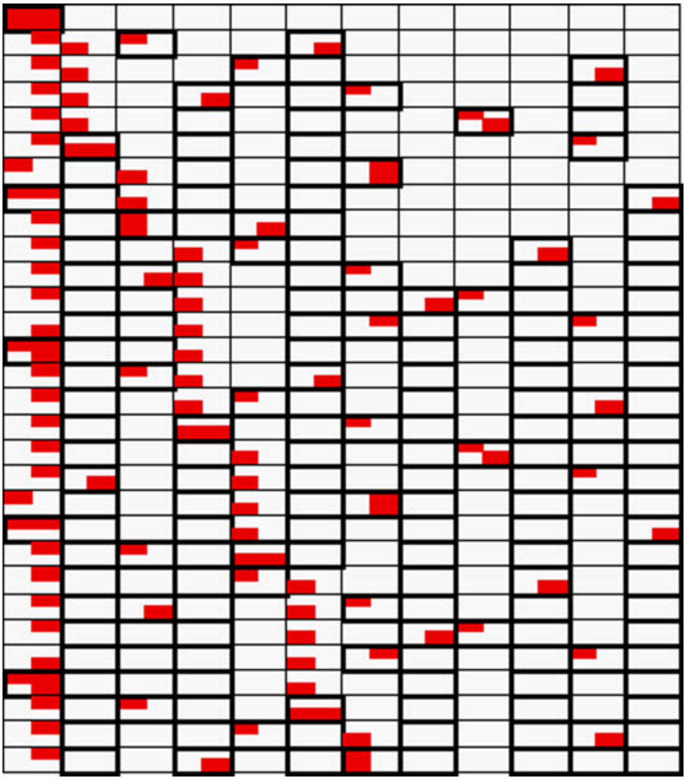

Fig. 5 The same run as shown in Fig. 4, except with a new player 4 who moves six spaces and leaves the door open upon arrival. Note that up through $T = 5$, until player 4 actually gets a chance to move, the simulation looks the same as in the previous figure

Door 7—the seventh column in the figure—and observe how that door opens and closes over the course of the simulation. In 30 recorded time steps (including the initial $T = 0$ situation), Door 7 is open exactly 9 times. Thus, we can say that for this brief simulation, the probability of Door 7 being open is 9/30, or 0.3, and the probability of its being closed is 0.7. A first estimate of the Door 7 entropy over time (H) is found by the classic information-theory formula:

$$H = -P(\text{open}) \log_2 P(\text{open}) - P(\text{closed}) \log_2 P(\text{closed})$$

Since P(open) is 0.3 and P(closed) is 0.7, the entropy estimate for Door 7 is 0.881 bits. We could get a better estimate by letting the simulation run longer, and also arrive at entropy estimates for the other doors, as a rough measure of how much unpredictable activity is involved in a particular slamming-doors simulation.

We might note that the entropy measure for any door in Fig. 4 will be 0, if we begin our measurement after time 20; in this case, every door remains open or closed at each time step. Thus, the values of P(open) and P(closed) will be 1 and 0 (or 0 and 1) respectively for each door, and our entropy formula returns 0. This fits with our intuition that the Fig. 4 simulation is more ordered and predictable (and less interesting) than that of Fig. 5. More sophisticated (and perhaps informative) entropy measures—like those discussed in the references by Wolfram already noted—could now be tried, but I'll leave those explorations to the adventurous reader.

For those more interested in hacking, there are many other sorts of experiments and variants on the basic slamming-doors idea that one could construct. One might allow players to toggle doors or move about according to a probabilistic rule (e.g., a player might leave the starting door open two thirds of the time when it moves; or it might flip a coin to decide whether to move right or left). We might choose to allow players to move in a random order at each particular time step, rather than always (say) giving player 1 the first chance to move as in our current simulation. We might choose not to use "wraparound" for the doors, but rather to let players simply stop when they reach the end of the row (without reappearing on the other end). We might have some players only attempt to make a move at (say) every other time-step, rather than at every time-step.

Beyond these numerous possibilities—and returning to the theme with which this column began—it might be fun to implement a slamming door simulation more like a true narrative, and in a much more pictorially appealing way than my own simple program. You might, for example, want to depict little characters actually opening and closing doors and running back and forth horizontally on the screen. A particularly nice touch would be to include sound effects for doors that open and close. The effect could be something like a mathematical stage farce, all enacted according to complex rules. Actually getting a laugh out of a computer simulation is an idea that might have made the master himself, Georges Feydeau, cackle.

Readers who come up with their own versions or variants of the slamming doors system are, as always, delightedly welcome to contact this column at: ijcml-diversions@ccl.northwestern.edu.

References

Winkler, P. (2004). *Mathematical puzzles: A connoisseur's collection*. Natick, MA: A. K. Peters.

Wolfram, S. (1984). Universality and complexity in cellular automata. *Physica, 10D*, 1–35.

Wolfram, S. (2002) *A new kind of science*. Champaign, IL: Wolfram Media, Inc.