DRAFT February 28, 2008

Background for the Quad Prize

Tommaso Toffoli (tt@bu.edu)
Electrical and Computer Engineering, Boston University, MA 02215

The a Quad Prize, to be awarded in coordination with the Automata-2008 workshop, wants to encourage filling a gap in our imagination of the simplest models of computation, and will be awarded to the first submission that will either exhibit a computation-universal 2-state cellular automaton on a 2-dimensional, 2×2-neighborhood—or on the contrary prove that none exist within that class. Short of that, a partial prize may be awarded for a significant partial result. Administrative details are given elsewhere; here we clarify the scientific/technical aspects of the competition.

1 A gap to be filled

The quest for models of computation based on the *simplest possible building blocks* is not (only) some sort of wizardly tournament for computer brats—the Harry Potters of the digital world. It is also a way to hedge our bets as to which approaches to computation are likely to be supported *most naturally and efficiently* by Nature. Our default presumption must be that Nature is *indifferent* to the "plans of mice and men;" that we are not likely to find, at the bottom of fundamental physics, interactions that look like a "four-input multiplexer" or a "JK-flip-flop" just because those logic functions are so cherished by the computer circuit designer. We must keep exploring ways to extract computational work from "fine-grained textures" of physics that, as far as we know, were not designed with computation in mind.

Since their very inception, *cellular automata* (CA) have been addressing the above question, namely, "What are the simplest-textured substrates we can manage to compute with?"

If we turn to CA that are alleged to support computation-universal capabilities, we have the following state-of-the-art for those with a state-alphabet of only two symbols and a small "radius of interaction" (D denotes the number of dimensions; N, the number of neighbors; and Y the approximate year of discovery.)

D	N	Y	Neighborhood
2	9	1970	Moore (3×3): Conway's game of Life[5]
2	5	1968	v. Neumann (cross), no 90° symm.: Codd's universal computer/constructor[3, p.27]
2	5	1971	v. Neumann (cross): Bank's universal CA[1]
2	4	—	Quad (2×2): *not known*
2	3	—	Trid (triangle): *not known*
1	3	2004	Elementary (3×1): Cook's universal CA[4]

Note that, number of states and number of neighbors being equal, there will be "more rejoicing in Heaven" for two- (or more-) dimensional CA than for one-dimensional ones. In fact, the latter are less challenging and less promising, both in terms of theoretical properties[1] and of prospects for useful implementations.

You may observe a gap at the bottom of the 2-D section, for the neighborhoods of size 4 ("Quad") and 3 ("Trid"), described in more detail in §2. It may well be that the gap is there simply because no CA of that kind are computation-universal. However—conceivably for the reasons given in §3—here we are facing virtually unexplored territory. Quad has some chances, since there is still no impossibility proof for it. Because of its extremely scarce resources, the chances that the Trid neighborhood may yield a computation-universal CA are quite remote. In any event, any successful Trid submittal will ipso facto qualify for the Quad prize—and trump a "merely Quad" entry.

[1] As soon as one goes above one dimension one gets mired, to use Lind and Marcus's words, in the "Swamp of Undecidability"[6]—which is, of course, a trademark of computation universality.

2 The Quad neighborhood

The 2×2 neighborhood—which we'll call "Quad"—is smaller than the 3×3 "Moore" neighborhood used, for instance, in Conway's game of Life, and is in fact, next to the "Trid" (see below), the smallest "natural" 2-D CA neighborhood.[2]

The canonical presentation of the Quad neighborhood is via the following quadruple of neighbor offsets

$$\{\langle -\tfrac{1}{2}, -\tfrac{1}{2}\rangle, \langle +\tfrac{1}{2}, -\tfrac{1}{2}\rangle, \langle -\tfrac{1}{2}, +\tfrac{1}{2}\rangle, \langle +\tfrac{1}{2}, +\tfrac{1}{2}\rangle\}.$$

In other words, the site located at $\langle x, y\rangle$ will at time $t+1$ get as inputs the states at time t of the four sites at $\langle x-\frac{1}{2}, y-\frac{1}{2}\rangle$, $\langle x+\frac{1}{2}, y-\frac{1}{2}\rangle$, $\langle x-\frac{1}{2}, y+\frac{1}{2}\rangle$, and $\langle x+\frac{1}{2}, y+\frac{1}{2}\rangle$.

Observe that the original site $\langle x, y\rangle$ is not among those four. While the spacetime mesh yielded by the Moore neighborhood may be visualized as a simple *cubic* crystalline lattice, the Quad neighborhood yields a *body-centered cubic* lattice, with the sites of the spacelike mesh layers alternating in time between integer coordinates $(\dots, -1, 0, +1, \dots)$ for both x and y and "half-integer" coordinates $(\dots, -1\frac{1}{2}, -\frac{1}{2}, +\frac{1}{2}, +1\frac{1}{2}, \dots)$.

[2] At least $n+1$ *non-collinear* neighbors are necessary for effectively n-dimensional interactions between sites. Quad and Trid are rotationally-invariant (with group generators of, respectively, 90° and 120°) *as neighborhoods*. Of course, the *rules* with which will they are implemented may break the symmetry.

Alternative presentations of the Quad neighborhood are available that, while perhaps less elegant, avoid half-integer coordinates. For a close analogy we may look at two ways of displaying the *same* table of binomial coefficients $\binom{t}{k}$; the first (A) is more compact but asymmetrical; the second (B), which sets $k = \frac{t+x}{2}$, restores symmetry but introduces "half-column tabbing" for odd rows.

```
t                t                          h=(t-x)/2        .
0  1       A     0          1     B             4  1       .   C
1  1 1           1        1   1                 3  1 4   .
2  1 2 1         2      1   2   1               2  1 3 6
3  1 3 3 1       3    1   3   3   1             1  1 2 3 4
4  1 4 6 4 1     4  1   4   6   4   1           0  1 1 1 1 1
   0 1 2 3 4  k    -4-3-2-1 0 1 2 3 4  x=2k-t      0 1 2 3 4  k=(t+x)/2
```

Of course these tradeoffs are a matter of which coordinate system one chooses. By using $(\frac{t+x}{2}, \frac{t-x}{2})$ coordinates instead of (t,k) we get the third form (C) of the table, which is both symmetric (with respect to the bisectrix) and packed on a tight orthogonal mesh. Similar considerations apply almost verbatim to the Quad neighborhood and its one-dimensional $(\{\langle -\frac{1}{2}, +\frac{1}{2}\rangle\})$ and higher-dimensional counterparts.

If you do not care about this kind of presentation issues, for the sake of the present competition feel free to use for the Quad neighborhood the presentation $\{\langle 0,0\rangle, \langle 1,0\rangle, \langle 0,1\rangle, \langle 1,1\rangle\}$, which yields a perfectly conventional CA format, with the "center cell" sitting in the lower-left corner of the 2×2 neighborhood. The evolution will be identical except for a constant-speed drift directed towards that corner.

Analogous considerations hold for the Trid neighborhood, with three neighbors forming an equilateral triangle around the destination site, and sites stacked in layers, time step after time step, like cannonballs (two variant lattices are possible).

3 A path less taken

Surprisingly, cellular automata on the Quad neighborhood have hardly ever appeared in the literature. But *why* should this CA class have remained practically unexplored? A likely reason is that the rank-and-file of CA users aren't even aware of its existence!

CA are traditionally introduced as a "solitaire" played on a sheet of quadrille paper. A *site* of the CA is identified with a square of this paper, or "cell," and its *state*—a symbol such as 0 or 1—is viewed as the "contents" of the cell itself. Start with an arbitrary initial configuration of 0s and 1s on your sheet. Take a second sheet of quadrille paper (a translucent one) and superpose it with good registration on the first. For each cell of the mesh use the CA "rule" to compute its new value from the current contents of its neighborhood as you see it on the bottom sheet; fill the cell with this value. When you're finished remove the bottom sheet and put a fresh one on top of the sheet that is left. And so on, *ad infinitum*.

In that scenario, while the *contents* of a cell may change from step to step, it makes sense to think of the cell itself, viewed as a *container*, as an enduring landscape feature—one having a permanent identity. But with neighborhoods like Quad the new translucent sheet is placed on top of the old one with an offset of *half a square* in both the x and y directions, so that the center of a cell of the new sheet falls right on the point where four cells of the old sheet meet. Thus a cell at time $t+1$ cannot be viewed as inheriting the identity of any specific cell from time t.

It thus appears that Quad-neighborhood CA have mostly been passed over by CA researchers because "they didn't even show up in the catalog they gave us!" and by LG researchers (see below) because a Quad neighborhood with a CA-like rule is typically less interesting for their purposes than one with an LG rule.

This very arrangement that is uncommon in CA is on the contrary quite common in *lattice-gas automata* (LG). While, in a CA, data passively reside in a cell waiting for the local map to "look at them" and produce new data depending on what it sees, in LG data spend their time as *signals* continually traveling *between* sites; the latter represent *events*. An event is the locus of interaction of signals whose paths intersect; signals get modified by going through an event. A staggering of events in spacetime, geometrically much as that of cells in the Quad neighborhood, is thus in lattice gases a very common way to arrange things (cf. the "Margolus" neighborhood[7]) rather than the exception. Even when the geometry is thus similar, the main difference remains that in a lattice gas n signals converge onto an event and n *distinct* signals diverge out of it; in a CA, on the other hand, n signals converge into the local map but only *one* signal comes out of it. This signal then goes through a $\times n$ fanout, from which it emerges again as n signals; but, unlike in a LG, these are *all identical*.

4 Evaluating the evidence

Proving that a CA having few states and few neighbors is computation-universal is not expected to be an easy task. To arrive, from raw elementary effects on the scale of one site, at functional modules capable of recognizable logic, arithmetic, and control functions, a whole hierarchy of useful intermediate structures and effects may have to be discovered first. The "logic" of these intermediate structures may be elusive and hard to explain, because indeed there may be no real rationale at that level—only epigenetic behavior that just "happens" to work right.

The Automata-2008 organizing committee will choose a panel of three judges and give them extraordinary latitude to collect and evaluate evidence for a claim, to get the help of other experts, to ask for clarifications and additional evidence, and to decide if and when they will be in possession of a claim worthy of an award. Should no fully satisfactory entries present themselves, the panel may decide to award a partial prize for a significant partial result. Likewise, the panel may decide to split the prize between several entrants.

It is quite possible that the exhibited computation-universal behavior will rely on a suitable preparation of the initial configuration, even to the point of postulating a regular, indefinitely-extended initial background pattern, as was done in the Codd and Cook constructions mentioned above. Again, it will be up to the panel to decide what extent of prepping of the initial state is to be deemed "fair."

Finally, the panel may accept, in lieu of proof, a demonstration that the proposed construction gives correct results in a number of representative tasks.

We encourage the contestants to be clear; to provide literate documentation and, if appropriate, usable simulation tools; and to suggest themselves some verification criteria or techniques appropriate for their prize entry.

List of references

[1] Banks, Roger, "Information processing and transmission in cellular automata," *PhD thesis*, MIT Dept. Mechanical Eng. 1971.

[2] Berlekamp, Elwyn, John Conway, and Richard Guy, *Winning Ways for your Mathematical Plays, Volume 2: Games in particular*, Academic Press 1982.

[3] Codd, Edgar, *Cellular Automata*, Academic 1968.

[4] Cook, Matthew, "Universality in elementary cellular automata," *Complex Systems* **15** (2004), 1–40.

[5] Gardner, Martin, "The fantastic combinations of John Conway's new solitaire game 'Life'," *Sc. Am.* **223**:4 (April 1970), 120–123.

[6] Lind, Douglas, "Multi-dimensional symbolic dynamics," *Symbolic Dynamics and Its Applications*, Am. Math. Soc. 2004, 61–79. (Same as *Proc. Symp. App. Math.* **60** (2004))

[7] Toffoli, Tommaso, and Norman Margolus, *Cellular Automata Machines—A New Environment for Modeling*, MIT Press 1987.