Abstract

A key point of NKS is the computational universality of certain cellular automata. Sutner has constructed reversible one-dimensional cellular automata of intermediate (less than universal) Turing degree by simulation of intermediate-degree Turing machines.

Sutner's construction uses tracks that left-shift and right-shift their contents at every step, allowing for the propagation of signal bits that simplify the Reachability problem. He has shown that the behavior of the signal system is trivial on finite configurations.

We examine the signal system and show that its behavior is trivial even on infinite periodic configurations. It is impossible to construct domain walls or produce behavior more complex than parenthesis-balancing, so it cannot increase the degree of an automaton beyond the degree of the Turing machine being simulated.

The signal system is not strictly reversible: there are “Garden of Eden” states which can have no ancestor. However, this does not introduce ambiguity to the simulation of a legitimate Turing machine. We analyze
the consequences of semi-reversibility for the Confluence and Reachability problems.

Intermediate Turing Degree

A key point of NKS is the computational universality of certain simple machines. All universal machines are undecidable, but there is a broad spectrum of systems that are neither decidable nor universal.

Systems that are strictly decidable are said to have a Turing Degree of 0.
Universal systems are never decidable because they can simulate known undecidable problems such as the halting problem. These are said to have a Turing Degree of 1.
If a problem $A$ can be reduced to a problem $B$ by using an instance of $B$ as an oracle to solve an instance of $A$, then we say that the Turing Degree of $A$ is less than or equal to that of $B$: $A \leq_T B$
Problems of Intermediate Complexity

There is a brief discussion of intermediate degrees on pages 1130-1131 of *A New Kind of Science*, which gives an example of a problem with intermediate degree:

If we take a universal system and modify it so that if it ever halts its output is discarded and, say, replaced by its original input. The lack of meaningful output prevents such a system from being universal, but the question of whether the system halts is still undecidable.

Similar methods can be used to produce a problem whose degree is strictly between those of any given two problems.

CAs of Intermediate Complexity

We consider problems that can be modeled by examining the orbits of cellular automata – that is, their input configuration, and the steps of their evolution from that state.

Given a CA and two possible configurations X and Y, Reachability is the problem of determining whether Y can be reached from X by a finite number of iterations. Confluence is the problem of determining whether the orbits of X and Y overlap.

**Theorem 1 (Sutner).** For any two recursively enumerable degrees $d_1$ and $d_2$ there is a one-dimensional cellular automaton whose Reachability Problem is of degree $d_1$ and whose Confluence Problem is of degree $d_2$. 
Reversible Turing Machines

We can represent an instantaneous description of a Turing machine by dividing its transition function into separate read/write transitions, left-shift transitions, and right-shift transitions. If these transitions are all injective, then the Turing machine is reversible.

**Theorem 2 (Lecerf, Bennett).** For any r.e. set $W$ there is a reversible Turing machine that accepts $W$.

---

Reversible Cellular Automata

**Theorem 3 (Morita, Harao).** For any reversible Turing machine there is a reversible cellular automaton that simulates the Turing machine.

By using a partitioned cellular automaton with dedicated left-shift and right-shift tracks, we can represent instantaneous descriptions of Turing machines as configurations of the CA, and write rules to simulate the behavior of a given Turing machine.

\[
\begin{array}{c}
\cdots \cdots \cdots \cdots \\
\cdots \cdots \cdots \cdots \\
\cdot \cdot \cdot y \cdot \cdot \cdot \\
\cdot \cdot \cdot z \cdot \cdot \cdot \\
\end{array}
\Rightarrow
\begin{array}{c}
\cdots \cdots \cdots \cdots \\
\cdot \cdot \cdot x' \cdot \cdot \cdot \\
\cdots \cdots \cdots \cdots \\
\cdot \cdot \cdot y' \cdot \cdot \cdot \\
\cdot \cdot \cdot z' \cdot \cdot \cdot \\
\end{array}
\]

As long as the TM we are simulating is reversible, the CA will be reversible on valid configurations.
Complexity of Reversible CAs

Theorem 4 (Sutner). For any r.e. degree $d$ there is a one-dimensional reversible cellular automaton whose complete orbit has degree $d$.

Sutner uses the same three-track one-dimensional cellular automaton to simulate a reversible Turing machine accepting an r.e. set $W$ of Turing degree $d$.

Each cell in the left-shift and right-shift tracks also contains a signal bit. If two signal bits are in the same column, they cancel each other out.

When there are a finite number of signal bits in the improper state, the size of their light-cones provides an upper limit on the number of steps apart two configurations may be. Therefore the Reachability problem is decidable.

In the case that there are no improper signal bits at the end of the computation, we know that the behavior of the automaton is equivalent to the behavior of the reversible Turing machine being simulated, so the Reachability problem has degree $d$.

Therefore the complete orbit of the CA has degree $d$ as long as there are only finitely many improper signal bits.
Model of the Signal Tracks

We examine the behavior of Sutner's system when the signal tracks differ from the quiescent state in infinitely many places.

We model the behavior of the two binary signal tracks as a 4-color 1-dimensional automaton.

white = 0 on top, 0 on bottom (left-moving particle)
light = 1 on top, 0 on bottom (this is the quiescent configuration)
dark = 0 on top, 1 on bottom (collision)
black = 1 on top, 1 on bottom (right-moving particle)

In[93]:= Clear[RasterGraphics];
RasterGraphics[state_, MatrixQ, colors_Integer : 2, opts___] :=
Graphics[Raster[Reverse[1 - state],
colors - 1],
AspectRatio -> (AspectRatio /. {opts} /. AspectRatio -> Automatic), opts]

In[95]:= swapDualShiftRule = 4^^3332333233323332111011101110111011101110111011101110111011101110,
4, 1; configuration = 8883<, ... by Mathematica for Students
An Infinite Configuration that Produces Quiescence Immediately

In[99]:= Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, {configuration, {1, 3, 1, 0}}, 20, {All, All}], 4]];

Infinite Configurations that Produce Quiescent Windows

In[100]:= Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, {configuration, {1, 1, 3, 1, 1}}, 20, {All, All}], 4]];
In[101]:= Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, 
configuration, {1, 1, 1, 0, 1, 1}], 20, {All, All}], 4];

In[102]:= Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, 
configuration, {3}], 20, {All, All}], 4];
Some Random Configurations

In[103]:= Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, {configuration, {3, 0}}, 20, {All, All}], 4]];

In[110]:= Do[Show[RasterGraphics[CellularAutomaton[swapDualShiftRule, {Table[1, {100}], Table[Random[Integer, {0, 3}], {500}], 500, {All, All}], 4}], {5}];
The seeded quiescent region of size $n$ is typically overrun soon after $n/2$ steps. The interaction of random noise tends to eliminate all particles of a single direction fairly soon, at least in each half-universe consisting of every other cell in a checkered fashion.
The Ancestor Problem

The problem of determining whether a given configuration has an ancestor is nontrivial because collisions leave no information behind.

**Theorem 5:** A state has no ancestor iff it is of the form $\Sigma^* L \Sigma R \Sigma^*$, where $L$ and $R$ indicate left- and right-moving particles. In this case, the smallest odd number of cells between a left- and right-moving particle determines the maximum age of the system.

Domain Walls?

Borders of size $n$ will guarantee $2n$ steps of quiescence, and much more in the average case.
Theorem 6: It is impossible to construct effective domain walls with Sutner's signal bits.

Consider the half-universes mentioned above. If we have a wall of size $n$ within such a half-universe, it will be overrun as soon as the outside region produces $n$ particles moving toward that wall.
To determine the color of a cell \( n \) at time \( t \), we simply examine the initial cells between \( n \) and \( n + t \) (non-inclusive), treating black particles as left-parentheses and white particles as right-parentheses. If the parentheses are perfectly balanced, we will see the intersection of the initial cells at \( n \) and \( n + t \). Otherwise, we will see only the particle that was not overtaken, or quiescence if they were both overtaken.

**Theorem 7:** A periodic configuration of size \( s \) will become trivial after \( s \) steps of evolution.

If we have a periodic configuration of size \( s \), we can count the number of left- and right-moving particles and conclude that the one in excess will dominate the other, so that after \( s \) steps, we have only a single color remaining, in a periodic
The complete signal system is composed of two of these half-universe systems acting simultaneously without interaction, so the complexity is the same, although the period may not be.

Triviality of the Signal System

Since the behavior of the signal system is trivial even on infinite configurations, it cannot increase the degree of the problem beyond that of the Turing machine being simulated.
**References**