The device physics we need to build thermodynamic computers

Suhas Kumar
The voids of computing

Present digital computers

- Floating point arithmetic
- Boolean logic
- Image processing
- Gene sequencing (+ other NP-class)
- Weather prediction

Data heavy

Arithmetic heavy

Nonlinear dynamics

Major limitations of digital computers:

- End of Moore’s law
- Von Neumann architecture
- Boltzmann tyranny
- Boolean logic
- Turing limit
A transistorless all-“memristor” Hopfield network

1. Synaptic memristors – nonvolatile storage
   • New device property: analog tunability
2. Neuronic memristors – volatile storage + nonlinearity
   • New device property: chaotic dynamics

Performance benchmarking – larger NP-hard problems

<table>
<thead>
<tr>
<th></th>
<th>memristor-HNN</th>
<th>GPU</th>
<th>FPGA</th>
<th>D-wave 2000Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock frequency</td>
<td>1 GHz</td>
<td>1.582 GHz</td>
<td>0.1 GHz</td>
<td>$10^4$ s</td>
</tr>
<tr>
<td>Time to solution (TTS)</td>
<td>0.3 $\mu$s</td>
<td>10 $\mu$s</td>
<td>&gt; 1 $\mu$s</td>
<td>$25,000$ W</td>
</tr>
<tr>
<td>Power</td>
<td>3.48 W</td>
<td>&lt; 250 W</td>
<td>70 W</td>
<td>250 MJ</td>
</tr>
<tr>
<td>Energy to solution</td>
<td>0.96 $\mu$J</td>
<td>&lt; 2.5 mJ</td>
<td>&lt; 2000</td>
<td>$4 \times 10^{-9}$</td>
</tr>
<tr>
<td>Solutions/s/Watts</td>
<td>$1.04 \times 10^6$</td>
<td>&lt; 2000</td>
<td>&lt; 2000</td>
<td>limited</td>
</tr>
<tr>
<td>Connectivity</td>
<td>all-to-all</td>
<td>all-to-all</td>
<td>all-to-all</td>
<td>limited</td>
</tr>
<tr>
<td>Cryogenic cooling</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Unpublished
A physics-driven computer program

- **System + Software**
  - Compact models + Architecture
  - New architectures, e.g.:
    - Hopfield networks,
    - Boltzmann machines

- **Devices + Interactions**
  - New device behaviors, e.g.:
    - Action potential, chaos

- **Materials + Physics**
  - Physical models
  - New physics, e.g.:
    - Thermal behavior during Mott transitions
Nonlinear electronics – why it’s important

All memristors are inherently nonlinear devices.

As devices are shrunk to the nanoscale, they interact with their environment → more state variables → nonlinearity is inevitable

$$\eta \propto T \left( \frac{k_B}{C_{th}} \right)^{1/2} \frac{4\pi}{R_{th} C_{th}}$$

Small devices can be driven by thermal noise, especially as they approach “kT”.

How do we make use of this?
Why local activity is important

Nonlinearity →
Local activity →
Chaos →
Edge of chaos →
Complexity and emergence

Chua, “Local activity is the origin of complexity”
Chua, “Neurons are poised near the edge of chaos”
Chua, “Local activity principle”
Extreme nonlinearity?

\[ i = Gv \]
\[ G = \psi T \uparrow \xi \]

Local activity \rightarrow the ability to amplify energy
Thermodynamics of electronic devices: e.g.: decompositions

All device/circuit models so far:

1. Behavior governed by \( i, v, P, Q \)

The two missing pieces of device models:

1. Behavior governed by thermodynamic quantities
2. Spontaneous symmetry breaking during instabilities.

\[ j \downarrow U = j \downarrow L (1-x) + j \downarrow H .(x) \]

Kumar et al., Nature Communications, (2018)
Kumar et al., Advanced Materials, 28, 2772 (2016)
How thermal noise interacts with local activity

1. Smaller devices $\rightarrow$ more thermal fluctuations (< 50 nm)

2. More thermal fluctuations $\rightarrow$ higher likelihood of filament formation $\rightarrow$ failure likely

3. Dynamics become more interesting, and also noisier.

4. Most nonlinear transports allow for tunability of many of the above.
New physics in Mott insulators!

\[
\dot{i} \downarrow m = \left[ \sigma \downarrow 0 \ e^{-0.301/2k_B T} A\left\{ (k_B T/\omega)^{1/2} \ (1 + (\omega \sqrt{v \downarrow m /d}/k_B T)^2 + 1/2d ) \right\} \nu \downarrow m \right.
\]

\[
dT/dt = \dot{i} \downarrow m \nu \downarrow m /C \downarrow th - T - T \downarrow amb /C \downarrow th \ R \downarrow th
\]

\[
R \downarrow th (T) = \begin{cases} 1.4 \times 10^{16} \ (\text{for } T \leq T \downarrow MIT) @2 \times 10^{16} \ (\text{for } T > T \downarrow MIT) \end{cases}
\]

Strange behavior!

Confirmed by x-ray and thermal mapping

Kumar, Nature Comms. 8, 658 (2017)
Broad pointers

1. Practically, all future electronic devices will contain extreme nonlinearities

2. Any device model should account for
   1. Local activity
   2. Interaction with ambient state variables and their perturbations
   3. I, V, t, T – dynamics is important!
   4. Spontaneous symmetry breaking
   5. Most importantly, thermodynamic extremization

3. The search for device behaviors should be informed by simulating the performance of the architectures – this is where we do not use transistor emulators!

4. Continue to broaden the inventory of physical processes that lead to interesting device physics.
Mott transition and a “free energy well”
A chaos-driven computer
Classical analog annealing accelerators

Hopfield network

Storage

Weights

Energy

Solution

Nonlinear filter

Solution
Program rule:
\[ w_{ij} = -C_1 \delta_{il} \delta_{jk} (1-\delta_{lk}) - C_2 \delta_{lk} \delta_{ij} (1-\delta_{il}) - C_3 - C_4 \delta_{il} \delta_{lj} (1+\delta_{jk} - 1) \]

Energy function:
\[ E = - \frac{1}{2} \sum_{i,j} s_i s_j \sum_{k,l} w_{ij} \delta_{il} \delta_{jk} + \sum_{i,j} s_i s_j \]

US Patent App. 15/141,410
Statistics of many solutions with and without chaos

We only want *better* solutions quickly.

High precision $\rightarrow$ prohibitive slow downs

- Room temperature
- Scalable

The traveling salesman problem

Objective:
Find the shortest path

Constraints:
1. Visit every city once
2. Visit every city no more than once
3. Do not visit more than one city in a given stop
“Hard” problems

It is non-deterministic polynomial (NP) complete.

Other NP-complete/hard problems:

Gene sequencing/traveling salesman
Sudoku
Pokemon
Candy Crush
Vehicle routing
Open shop scheduling

A traveling salesman wants to visit every city in his territory.
Finding the shortest route is easy for a few cities. But the problem grows complex rapidly.
Challenges of analogue systems

Why did analogue computers die after the 1970’s?

Difficult to design
Difficult to reprogram
Did not scale
Digital emulators were error prone
Did not offer precision
Digital offered precision and scalability

In short, we used analogue systems for the wrong set of problems.

Example: graph coloring using analogue oscillators

Parihar et al., Scientific Reports, 7, 911 (2017)
Computationally hard problems and nonlinear dynamics

“Hard” problems

Nonlinear differential equations

Chaotic dynamics

Exponential reduction in time to solution

(exponential increase in energy expenditure)

Ravasz et al., Scientific Reports, 2, 725 (2012)
Memristors can emulate both synapses and neurons!

Pickett, Nature Materials, 2014
Kumar, Nature, 2017
Chua, “Neurons are poised near the edge of chaos”, 2012