Physically Scalable Computation
- an axiomatic approach

Steven Lindell, Haverford College
slindell@haverford.edu

Abstract: A consideration of the matter and energy requirements of physically implemented machines leads tentatively to a finer mathematical analysis of asymptotic space and time computability.

Given June 19, 2001 at rump session of Conference on Computational Complexity

CC.0

Computational Complexity

The conventional approach is to execute an algorithm on a computer system consisting of a *finite* **program** text (the software) on an *infinite* **machine** model (the hardware).

Assume problem size ***n*** is measured by how much 'room' the input (and/or output) occupy. Then let ***n*** go to infinity in order to do an asymptotic resource analysis of:

- **time =** # of program *steps* executed = *T(n)*
- **space =** # of machine *locations* utilized = *S(n)*

Considering a physical basis for measuring these was inspired by Feynman's study of "Mr. Turing's machines".

CC.1

Physical requirements

We propose a closer examination of the natural requirements of sequential computing devices.

- **density** = *matter* per unit space (bit *storage* per location)
- **power** = *energy* per unit time (modification *work* per step)

Our discussion of the fundamental units *step* and *location* builds on foundations provided by [explicated by]:

- *Alan Turing* [Wilfred Sieg] (analyzed human computor)
- *Andrei Kolmogorov* [Yuri Gurevich] (bounded-degree graphs)

CC.2

Postulates

A computation proceeds through *discrete steps* of time in a determined fashion.
- At each step, the machine resides in a *state*.
- A run is a(n) (in)finite sequence of states: *S*, …

Each state is the dynamic configuration of a statically connected network of *discrete locations* in space.
- Data can be symbolically **stored at each location**.
- Data can be **communicated between adjacent locations** along addressing channels.

At any given point in the run, data can only be altered or transferred at a single place.

CC.3

Physical Constraints

The state of the machine depends only on the processor state and a finite portion of the memory contents. We say it is *scalable* if it can be ***built*** and ***operated*** using a fixed implementing technology regardless of the ***size*** or ***length*** of the computation.

- uniform *density* = *O*(1) mass per unit space (**bounded storage** per location)

- uniform *power* = *O*(1) energy per unit time (**bounded work** per step)

RAM models, because of unbounded word size, violate both conditions.

CC.4

One Step

Divide each computational step into three phases:

- **Input**: Information is loaded from the focal part of memory.
- **Process**: An internal decision is made to change the 'state of mind'.
- **Output**: Information is stored back into memory, and the 'focus' is moved to an adjacent location.

Turing made precise the notion of single step of a computor, specifying *locality of movement* and a *bound on the amount of information* in:

- recognizable symbolic configurations
- internal states of mind
- potential executable actions

CC.5

Flow of Information

Also bound *the amount of work in the read/write phases* of I/O (including move operation).

- **Communication:** Information cannot travel over an unbounded distance (in one step) (*counterexample:* TM with two heads:

```
________________________________
________________________________
         ^       …       ^
```

- **Storage:** No (single) operation can change an unbounded amount of information (*counterexample:* TM with head reset):

```
|_______________________________
        <=       …       <=
```

CC.6

Summary

Analysis of computability is extended by physical considerations of energy and matter. The undefined terms *step* and *location* can be seen as arising naturally from assumptions regarding the quantization of time and space.

- Conventional approach is based on our *mathematical* understanding of single operations, iterated arbitrarily often.
- New approach tries to better understand the *physical* uniformity of arbitrary computations.

To go further, we should take into account global assumptions such as self-power and self-assembly.

CC.7