Faulty Universality

Bruno Durand

LIRMM – CNRS – Université de Montpellier II

November 26th 2011

1. introduction

One can compute in a faulty medium

An informal statement derived from real theorems

THM

Peter Gàcs

There exists a Universal cellular fault tolerant cellular automaton in

- 3D + time – *easy*
- 2D + time – *less easy*
- 1D + time – *difficult*

A set of theorems with *deep* consequences (e.g. ergodicity)

The author excepted:

- only one person (Lawrence Gray) says he understands the proofs
- a few people say they think the proofs are correct
- nobody is willing to explain the proofs

We (B.D., A. Romashchenko, A. Shen) can reconstruct a proof for 2D by using several powerful techniques combined.

Models are needed

A model for faults

We alternate

1. an iteration of the Cellular automaton
2. a perturbation with a small probability

Models are needed

A model for faults

We alternate

1. an iteration of the Cellular automaton
2. a perturbation with a small probability

Several models are possible

"For almost all fault sequence…"

Models are needed

A model for faults

We alternate

1. an iteration of the Cellular automaton
2. a perturbation with a small probability

Several models are possible

"For almost all fault sequence..."

A model for universality

Several possibilities :

- Turing universality
- intrinsic universality among CA

Models are needed

A model for faults

We alternate

1. an iteration of the Cellular automaton
2. a perturbation with a small probability

Several models are possible

"For almost all fault sequence..."

A model for universality

Several possibilities :

- Turing universality
- intrinsic universality among CA

Problem: if the computation model is too complex, then one can cheat.

What we would like to explain in this talk

Fault tolerance **implies** a complex computation model (necessary condition independent of proofs)

- an encoding fonction
- a halting condition
- a decoding fonction

If the computation model allows too complex encoding, halting, or decoding, then one can cheat.

Examples :

What we would like to explain in this talk

Fault tolerance **implies** a complex computation model (necessary condition independent of proofs)

- an encoding fonction
- a halting condition
- a decoding fonction

If the computation model allows too complex encoding, halting, or decoding, then one can cheat.

Examples :

THM

- the encoding computes (instead of our CA)
- the halting condition computes
- the decoding computes

The situation without faults is much more

- an encoding function maps a finite object into a finite zone
- a finitary halting condition : appearance of a state or bounded pattern
- a decoding fonction that reads a finite word in the medium

Example :

A. Gajardo, E. Goles, A. Moreira THM
The *Langton ant* in the plane is Turing-universal

Many others :

J. Conway THM
The *Game of Life* is Turing-universal

But more and more complex models are needed (Damien Woods' talk).
See N. Ollinger *Universalities in Cellular Automata*.

2. remembering one bit forever

Toom’s rule

DEF

A cellular automaton

- binary alphabet
- in the plane
- majority of center, top, right

Finite patterns disappear :

Toom's rule is fault tolerant

Our technique (with A. Romashchenko): a hierarchy of islands of errors in the space-time diagram

Easy to be convinced

Not trivial to prove

Does not work in 1D

Reading the conserved bit

Toom's game :

- Martin chooses $x = 0$ or $x = 1$
- Marcos fills the plane with x
- Ivan alternates Toom/faults/Toom/faults/… as many times as he wants

Eric would like to find x
with probability 1

Reading the conserved bit

Toom's game :

- Martin chooses $x = 0$ or $x = 1$
- Marcos fills the plane with x
- Ivan alternates Toom/faults/Toom/faults/… as many times as he wants

Eric would like to find x
with probability 1

Alexandro's solution :

"The measure …ergodicity …convergence…"

True but not constructive enough

Reading the conserved bit

Toom's game :

- Martin chooses $x = 0$ or $x = 1$
- Marcos fills the plane with x
- Ivan alternates Toom/faults/Toom/faults/… as many times as he wants

Eric would like to find x
with probability 1

Alexandro's solution :

"The measure …ergodicity …convergence…"

True but not constructive enough

Anahi's solution :

"Let's put there an ant and see the limit frequency of what it observes"

Much better !

Reading the conserved bit

Toom's game :

- Martin chooses $x = 0$ or $x = 1$
- Marcos fills the plane with x
- Ivan alternates Toom/faults/Toom/faults/… as many times as he wants

Eric would like to find x
with probability 1

Alexandro's solution :

"The measure …ergodicity …convergence…"

True but not constructive enough

Anahi's solution :

"Let's put there an ant and see the limit frequency of what it observes"

Much better !

THM **B.D. A.R.**
Any constructive asymptotic solution is OK

Reading the conserved bit

Alexandro's solution :

"The measure …ergodicity …convergence…"

True but not constructive enough

Toom's game :

- Martin chooses $x = 0$ or $x = 1$
- Marcos fills the plane with x
- many times as he wants

A.R.

EXT This hierarchical construction can be used to prove theorems in percolation theory

B.D. A.R.

Any constructive asymptotic solution is OK

… limit … frequency of what it observes

Eric would like to find x

with probability 1

Much better !

3. fault tolerance and models of computation

Computation models with faults

Encoding

The code of the simulated machine and its imput *must be duplicated in an infinite number of locations*

Neither constant nor (fully) periodic at infinity

If you want to specify that the input and the machine are independent in the encoding you need 2 dimensions for encoding

Halting and Decoding

Halting : *the appearance of a finite pattern is not enough* since any patterns appear infinitely often because of faults. We need at least a limit frequency (see Toom's game).

Decoding : if dimension less than 2, *the reading zone cannot be bounded* (and even cannot be uniform). The reason is that if the output may (must!) be partially destroyed…

Faulty universality

In dimension 2:

We have a construction with a very complex encoding/decoding.

It *seems* that the computation is performed by the cellular automaton. We would like to *prove* it

For this, the standard solution is to give properties of the encoding and decoding function that ensure this

OPB **Open problem:** find such properties

In dimension 1:

Faulty universality

In dimension 2:

We have a construction with a very complex encoding/decoding.

It *seems* that the computation is performed by the cellular automaton. We would like to *prove* it

For this, the standard solution is to give properties of the encoding and decoding function that ensure this

OPB **Open problem:** find such properties

In dimension 1:

Maybe for Eric's 65th birthday !

Faulty universality

In dimension 2:

We have a construction with a very complex encoding/decoding.

It *seems* that the computation is performed by the cellular automaton. We would like to *prove* it

For this, the standard solution is to give properties of the encoding and decoding function that ensure this

OPB **Open problem:** find such properties

In dimension 1:

Maybe for Eric's 65th birthday !

The end.