CDM

Computational Discrete Mathematics

Klaus Sutner

Carnegie Mellon University
www.cs.cmu.edu/~sutner

Outline

Bourbaki, Charles Denis Sauter

Bourbaki, Nicolas

Carnap's Notes of Frege

In a mathematical lecture two things are always mixed together:

1. *the pure inferences,*
2. *the commentary on them.*

This mixture has the potential to negatively influence mathematical rigor. In conceptual notation, assuming a complete understanding of it, words are superfluous.

The Bourbaki Volumes

  I Set theory
 II Algebra
III Topology
 IV Functions of one real variable
  V Topological vector spaces
 VI Integration
VII Commutative algebra
VIII Lie groups

Some Bourbaki Idiosyncrasies

- Presentation strictly linear, no external references.
- Problem solving is secondary to axiomatics.
- Algorithmic content is off-topic.
- Combinatorial structure is non-essential.
- Logic is treated minimally.
- Applications nowhere in sight.
- And (cela va sans dire) no pictures.

Introductory Calculus Course, 1970's

- The natural numbers form a monoid, described by the Peano axioms.
- $\mathbb{N}$ can be extended to a commutative group, $\mathbb{Z}$.
- Lo and behold, $\mathbb{Z}$ carries a ring structure.
- Rings are nice, but fields are better: localize to get $\mathbb{Q}$.
- Rationals are great, but there are lots of gaps: the Cauchy completion has none: voila $\mathbb{R}$.
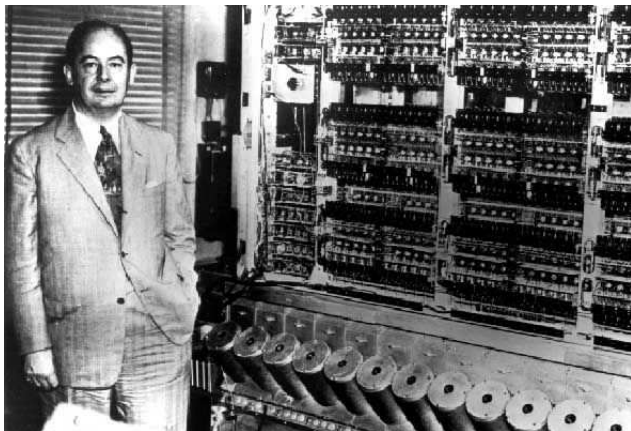- Now let's prove Stokes' theorem on $\mathcal{C}^1$ hypersurfaces.

Turing, Alan

What is Computation?
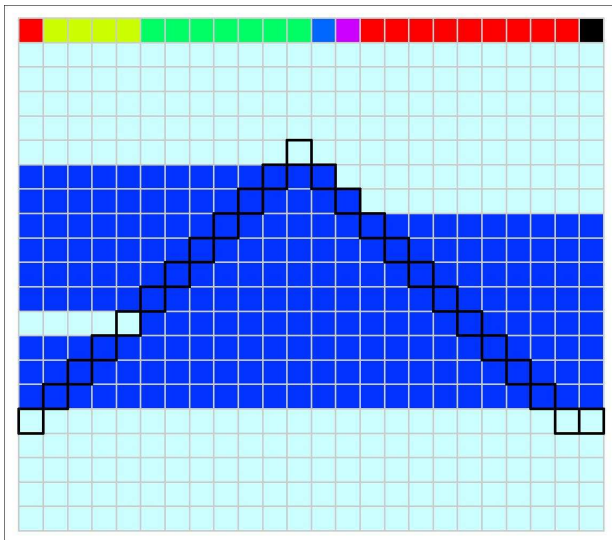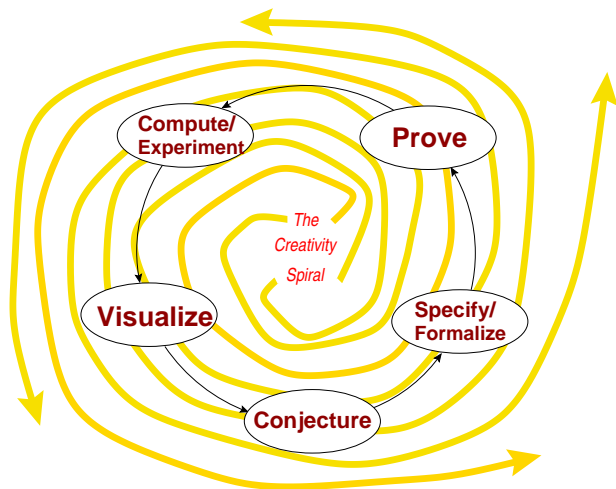
Computation

A Milestone

Plato

I believe that mathematical reality lies outside of us, that our function is to discover or observe it, and that the theorems which we prove . . . are simply our notes of our observations.

G. H. Hardy

Buchberger's Spiral

Some CDM Idiosyncrasies

- Problem solving central motivation.
- Algorithmic content is crucial.
- Combinatorial structure is essential.
- Logic is the foundation.
- Applications are ubiquitous.
- Visualization is used extensively.
- Programming helps understanding.

The Golden Opportunity

The standard of correctness and completeness necessary to get a computer program to work at all is a couple of orders of magnitude higher than the mathematical community's standard of valid proofs.

Bill Thurston

CS Students

Many have great facility with programming – though not necessarily with the fundamental mathematical machinery.

One can exploit this facility to enhance their understanding of math.

And one can exploit understanding of math to improve one's ability to solve algorithmic problems.

Example: 00-free Sequences

Problem: Determine the number of binary sequences of length n that contain no two consecutive 0's.

Standard approach: compute by brute force, conjecture answer and prove it.

```
DeleteCases[Tuples[{0,1}, n], {___,0,0,___}];
```

Better: write a program that constructs only the 00-free sequences.

Computing with Relations

Really background material.

Computational aspects often not well-understood.

Example:
Given two equivalence relations, compute their meet.

Good solution: represent the relations as kernels.

Kernels

The kernel relation of a function.

$$x \; K_f \; y \iff f(x) = f(y)$$

For equivalence relations: let f be a choice function.

$$f(x) = \min(\, z \mid z \; \rho \; x \,)$$

Implementation: Just a plain array of integers.

Minimization

Typical application: (slow) algorithm to minimize DFAs.

```
//  h = meet( f, g )
for p = 1..n do
      i = f[p];
      j = g[p];
      if( (i,j) is new pair )            // membership query
            h[p] = val(i,j) = p;
      else
            h[p] = val(i,j);
```

Join and Closures

So meets are easy to compute, but how about joins?

Given a relation ρ on A and a relational property P, find the least relation σ that

- contains ρ: $\rho \subseteq \sigma$, and
- has property P.

σ is the P-closure of ρ.

Transitive Closure

Existence à la Bourbaki:

$$\rho^\star = \bigcap \{\, \sigma \mid \rho \subseteq \sigma \text{ transitive} \,\}$$

Slightly more constructive:

$$\rho^\star = \bigcup \rho^i = 1 + \rho + \rho^2 + \ldots + \rho^n + \ldots$$

Solves the join problem for equivalence relations:

$$\rho \sqcup \sigma = (\rho \cup \sigma)^\star$$

Algorithms

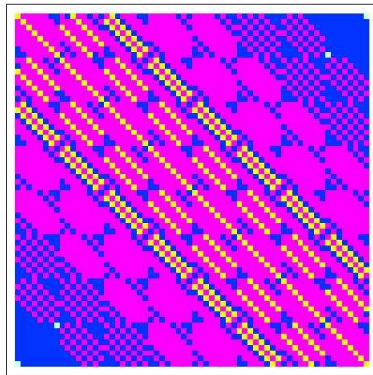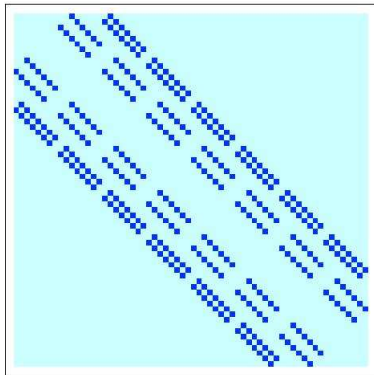Represent binary relations by Boolean matrices.

Get nice algebraic correspondence

$$\left\langle \operatorname{Rel}_{[n]}, \bullet \right\rangle \xrightarrow{\cong} \left\langle \mathbb{B}^{n \times n}, \cdot \right\rangle$$
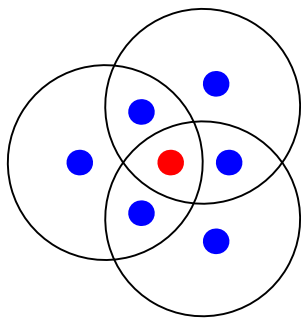
but not efficient for large n.

Represent binary relations by digraphs, use classical graph algorithms (DFS, BFS, Warshall).

Closure and Boolean Matrices

Flipping Pebbles

Operations:

- Flip all pebbles in one circle.
- Set all pebbles in one circle to red.

Problem: Start at all-blue and generate the shown configuration.

Invariants and Characterizations

Easy to handle by brute-force computation, but yields no insights.

A good solution requires to think of the pebbles as a basis over a 7-dimensional vector space over $\mathbb{F}_2$.

$$X = \{x_1, x_2, x_3, x_{12}, x_{13}, x_{23}, x_{123}\}$$

Then one can use a parity check matrix.

$$P = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$
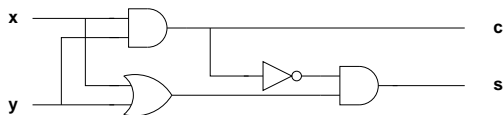
Logic

5.43

But in fact all propositions of logic say the same thing, to wit nothing.

L. Wittgenstein

Circuits

Half Adder

2-bit Adder
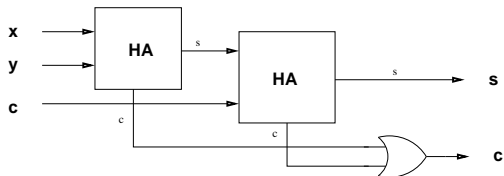
Boolean Functions

x	y	$H_{and}(x,y)$	$H_{or}(x,y)$	$H_{imp}(x,y)$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	1

Hard to manipulate combinatorially, equational reasoning is better.

Boolean Algebra

Lemma

$x + x = x$

Proof.

$$\begin{aligned}
x + x &= (x + x) \cdot 1 \\
&= (x + x) \cdot (x + \overline{x}) \\
&= x + x \cdot \overline{x} \\
&= x + 0 \\
&= x
\end{aligned}$$

$\square$

EQN

Famous conjecture from 1933, Robbins' Conjecture:

$$x = \overline{\overline{x+y} + \overline{x+\overline{y}}}$$
$$x + y = y + x$$
$$(x + y) + z = x + (y + z)$$

together imply the double negation property: $\overline{\overline{x}} = x$.

Proven in 1996 by the automatic prover EQP.

Algorithms for Propositional Logic

- Normal forms (negation, conjunctive, disjunctive)
- Matrix methods
- Davis-Putnam-Logeman-Lovelace

Apply to solve combinatorial problems.

Davis-Putnam-Logeman-Lovelace

$$(\neg p_{11} \vee p_{32} \wedge p_{23}) \wedge (\neg p_{12} \vee p_{31} \wedge p_{33} \wedge p_{24}) \wedge$$
$$(\neg p_{13} \vee p_{32} \wedge p_{34} \wedge p_{21} \wedge p_{25}) \wedge (\neg p_{14} \vee p_{33} \wedge p_{35} \wedge p_{22} \wedge p_{26}) \wedge$$
$$(\neg p_{15} \vee p_{34} \wedge p_{36} \wedge p_{23} \wedge p_{27}) \wedge (\neg p_{16} \vee p_{35} \wedge p_{37} \wedge p_{24} \wedge p_{28}) \wedge$$
$$(\neg p_{17} \vee p_{36} \wedge p_{38} \wedge p_{25}) \wedge$$
$$(\neg p_{18} \vee p_{37} \wedge p_{26}) \wedge (\neg p_{21} \vee p_{42} \wedge p_{13} \wedge p_{33}) \wedge$$
$$(\neg p_{22} \vee p_{41} \wedge p_{43} \wedge p_{14} \wedge p_{34}) \wedge$$
$$(\neg p_{23} \vee p_{42} \wedge p_{44} \wedge p_{11} \wedge p_{15} \wedge p_{31} \wedge p_{35}) \wedge$$
$$(\neg p_{24} \vee p_{43} \wedge p_{45} \wedge p_{12} \wedge p_{16} \wedge p_{32} \wedge p_{36}) \wedge$$
$$(\neg p_{25} \vee p_{44} \wedge p_{46} \wedge p_{13} \wedge p_{17} \wedge p_{33} \wedge p_{37}) \wedge$$
$$(\neg p_{26} \vee p_{45} \wedge p_{47} \wedge p_{14} \wedge p_{18} \wedge p_{34} \wedge p_{38}) \wedge$$
$$(\neg p_{27} \vee p_{46} \wedge p_{48} \wedge p_{15} \wedge p_{35}) \wedge$$
$$(\neg p_{28} \vee p_{47} \wedge p_{16} \wedge p_{36}) \wedge (\neg p_{31} \vee p_{12} \wedge p_{52} \wedge p_{23} \wedge p_{43}) \wedge$$
$$(\neg p_{32} \vee p_{11} \wedge p_{13} \wedge p_{51} \wedge p_{53} \wedge p_{24} \wedge p_{44}) \wedge$$
$$(\neg p_{33} \vee p_{12} \wedge p_{14} \wedge p_{52} \wedge p_{54} \wedge p_{21} \wedge p_{25} \wedge p_{41} \wedge p_{45}) \wedge$$
$$(\neg p_{34} \vee p_{13} \wedge p_{15} \wedge p_{53} \wedge p_{55} \wedge p_{22} \wedge p_{26} \wedge p_{42} \wedge p_{46}) \wedge$$
$$(\neg p_{35} \vee p_{14} \wedge p_{16} \wedge p_{54} \wedge p_{56} \wedge p_{23} \wedge p_{27} \wedge p_{43} \wedge p_{47}) \wedge$$
$$(\neg p_{36} \vee p_{15} \wedge p_{17} \wedge p_{55} \wedge p_{57} \wedge p_{24} \wedge p_{28} \wedge p_{44} \wedge p_{48}) \wedge$$
$$(\neg p_{37} \vee p_{16} \wedge p_{18} \wedge p_{56} \wedge p_{58} \wedge p_{25} \wedge p_{45}) \wedge$$
$$(\neg p_{38} \vee p_{17} \wedge p_{57} \wedge p_{26} \wedge p_{46}) \wedge (\neg p_{41} \vee p_{22} \wedge p_{62} \wedge p_{33} \wedge p_{53}) \wedge$$
$$(\neg p_{42} \vee p_{21} \wedge p_{23} \wedge p_{61} \wedge p_{63} \wedge p_{34} \wedge p_{54}) \wedge$$
$$(\neg p_{43} \vee p_{22} \wedge p_{24} \wedge p_{62} \wedge p_{64} \wedge p_{31} \wedge p_{35} \wedge p_{51} \wedge p_{55}) \wedge$$
$$(\neg p_{44} \vee p_{23} \wedge p_{25} \wedge p_{63} \wedge p_{65} \wedge p_{32} \wedge p_{36} \wedge p_{52} \wedge p_{56}) \wedge \cdots$$
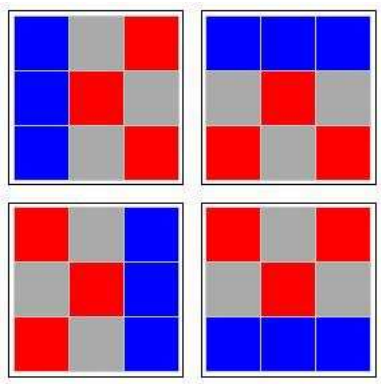
Knight's Tour

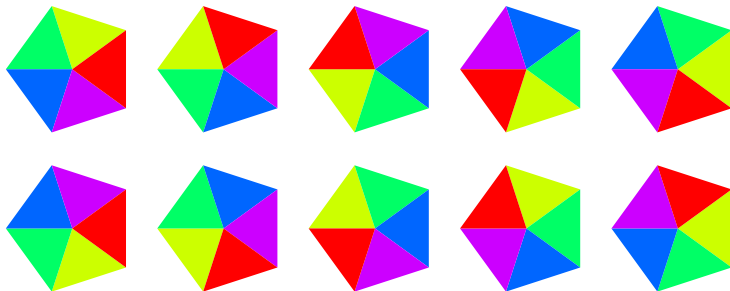Can even push a bit further: code the existence of a Knight's Tour as a propositional formula.

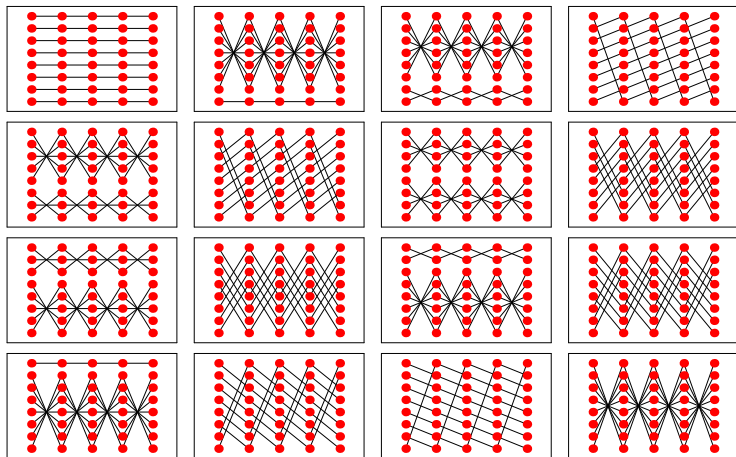Unfortunately, without external code 4×4 is the limit . . .
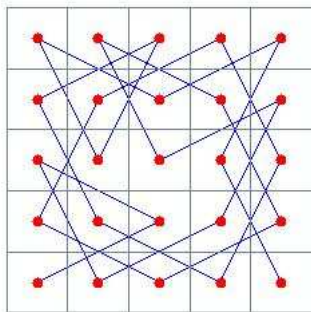
Tic-Tac-Toe

Dihedral Groups

As Permutations

Group Actions

$$\varphi : G \times X \to X$$
$$\varphi(a * b, x) = \varphi(a, \varphi(b, x))$$
$$\varphi(1, x) = x$$

Already familiar from transition systems: just more algebraic structure.

Computing with Polynomials

Cycle index polynomial for dihedral group D_{20}:

$$\frac{1}{40}(x_1^{20} + 10x_2^9 x_1^2 + 11x_2^{10} + 2x_4^5 + 4x_5^4 + 4x_{10}^2 + 8x_{20})$$

Substituting three colors:

$$\frac{1}{40}\big((c_1 + c_2 + c_3)^{20} + 10(c_1^2 + c_2^2 + c_3^2)^9 (c_1 + c_2 + c_3)^2 + \\ 11(c_1^2 + c_2^2 + c_3^2)^{10} + 2(c_1^4 + c_2^4 + c_3^4)^5 + 4(c_1^5 + c_2^5 + c_3^5)^4 + \\ 4(c_1^{10} + c_2^{10} + c_3^{10})^2 + 8(c_1^{20} + c_2^{20} + c_3^{20})\big)$$

Expand . . .

$$c_1^{20} + c_2 c_1^{19} + c_3 c_1^{19} + 10 c_2^2 c_1^{18} + 10 c_3^2 c_1^{18} + 10 c_2 c_3 c_1^{18} + 33 c_2^3 c_1^{17} + 33 c_3^3 c_1^{17} + 90 c_2 c_3^2 c_1^{17} +$$

$$90 c_2^2 c_3 c_1^{17} + 145 c_2^4 c_1^{16} + 145 c_3^4 c_1^{16} + 489 c_2 c_3^3 c_1^{16} + 774 c_2^2 c_3^2 c_1^{16} + 489 c_2^3 c_3 c_1^{16} + 406 c_2^5$$

$$c_1^{15} + 406 c_3^5 c_1^{15} + 1956 c_2 c_3^4 c_1^{15} + 3912 c_2^2 c_3^3 c_1^{15} + 3912 c_2^3 c_3^2 c_1^{15} + 1956 c_2^4 c_3 c_1^{15} + 1032 c_2^6 c_1^{14} + 1032 c_3^6 c_1^{14} +$$

$$5832 c_2 c_3^5 c_1^{14} + 14724 c_2^2 c_3^4 c_1^{14} + 19416 c_2^3 c_3^3 c_1^{14} + 14724 c_2^4 c_3^2 c_1^{14} + 5832 c_2^5 c_3 c_1^{14} + 1980 c_2^7$$

$$c_1^{13} + 1980 c_3^7 c_1^{13} + 13608 c_2 c_3^6 c_1^{13} + 40824 c_2^2 c_3^5 c_1^{13} + 67956 c_2^3 c_3^4 c_1^{13} + 67956 c_2^4 c_3^3 c_1^{13} + 40824 c_2^5$$

$$c_3^2 c_1^{13} + 13608 c_2^6 c_3 c_1^{13} + 3260 c_2^8 c_1^{12} + 3260 c_3^8 c_1^{12} + 25236 c_2 c_3^7 c_1^{12} + 88620 c_2^2 c_3^6 c_1^{12} + 176484 c_2^3 c_3^5 c_1^{12} +$$

$$221110 c_2^4 c_3^4 c_1^{12} + 176484 c_2^5 c_3^3 c_1^{12} + 88620 c_2^6 c_3^2 c_1^{12} + 25236 c_2^7 c_3 c_1^{12} + 4262 c_2^9 c_1^{11} +$$

$$4262 c_3^9 c_1^{11} + 37854 c_2 c_3^8 c_1^{11} + 151416 c_2^2 c_3^7 c_1^{11} + 352968 c_2^3 c_3^6 c_1^{11} + 529452 c_2^4 c_3^5 c_1^{11} + 529452 c_2^5 c_3^4 c_1^{11} +$$

$$352968 c_2^6 c_3^3 c_1^{11} + 151416 c_2^7 c_3^2 c_1^{11} + 37854 c_2^8 c_3 c_1^{11} + 4752 c_2^{10} c_1^{10} + 4752 c_3^{10} c_1^{10} + 46252 c_2 c_3^9 c_1^{10} + 208512 c_2^2 c_3^8$$

$$c_1^{10} + 554520 c_2^3 c_3^7 c_1^{10} + 971292 c_2^4 c_3^6 c_1^{10} + 1164342 c_2^5 c_3^5 c_1^{10} + 971292 c_2^6 c_3^4 c_1^{10} + 554520 c_2^7 c_3^3 c_1^{10} + 208512 c_2^8 c_3^2 c_1^{10} +$$

$$c_2^9 c_3 c_1^{10} + 4262 c_2^{11} c_1^9 + 4262 c_3^{11} c_1^9 + 46252 c_2 c_3^{10} c_1^9 + 231260 c_2^2 c_3^9 c_1^9 + 693150 c_2^3 c_3^8 c_1^9 + 1386300 c_2^4 c_3^7 c_1^9 +$$

$$1940568 c_2^5 c_3^6 c_1^9 + 1940568 c_2^6 c_3^5 c_1^9 + 1386300 c_2^7 c_3^4 c_1^9 + 693150 c_2^8 c_3^3 c_1^9 + 231260 c_2^9 c_3^2 c_1^9 +$$

$$46252 c_2^{10} c_3 c_1^9 + 3260 c_2^{12} c_1^8 + 3260 c_3^{12} c_1^8 + 37854 c_2 c_3^{11} c_1^8 + 208512 c_2^2 c_3^{10} c_1^8 + 693150 c_2^3 c_3^9 c_1^8 +$$

$$1560534 c_2^4 c_3^8 c_1^8 + 2494836 c_2^5 c_3^7 c_1^8 + 2912112 c_2^6 c_3^6 c_1^8 + 2494836 c_2^7 c_3^5 c_1^8 + 1560534 c_2^8 c_3^4 c_1^8 + 693150 c_2^9 c_3^3 c_1^8 + 208512$$

$$37854 c_2^{11} c_3 c_1^8 + 1980 c_2^{13} c_1^7 + 1980 c_3^{13} c_1^7 + 25236 c_2 c_3^{12} c_1^7 + 151416 c_2^2 c_3^{11} c_1^7 + 554520 c_2^3 c_3^{10} c_1^7 + \ldots$$
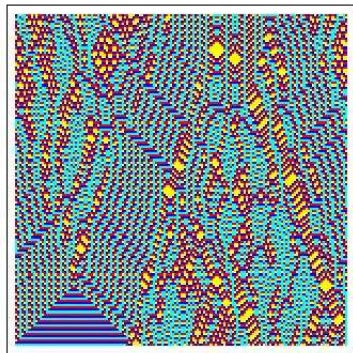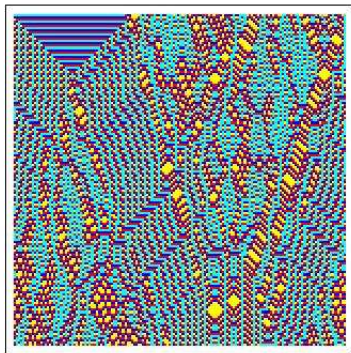
Elementary Cellular Automata
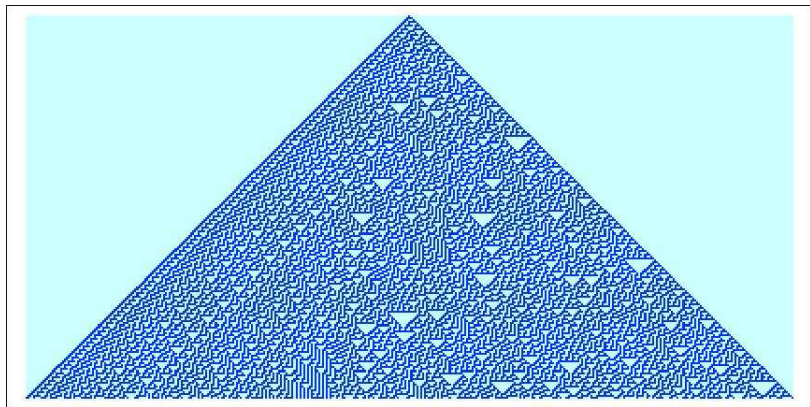
Firing Squad
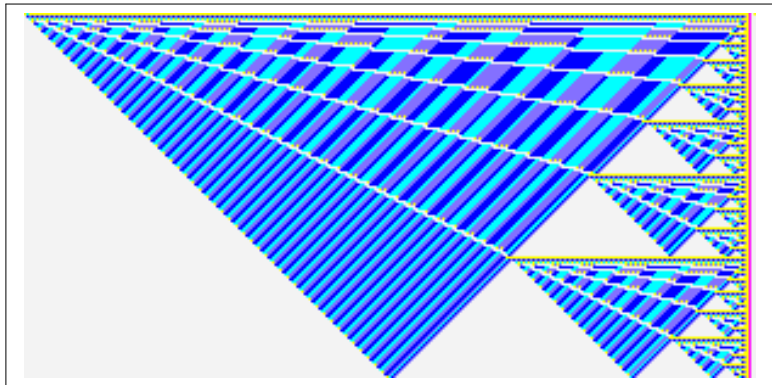
Fredkin Automata

Automata Theory

Excluded blocks in cellular automata naturally lead to counting problems for regular languages:

$$\gamma_L(n) = |L \cap \Sigma^n|$$

Calls for use of generating functions.

$$g(x) = \sum_{n \geq 0} \gamma_L(n)\, x^n$$

Polynomial Equations

$L \subseteq \{a, b\}^*$: all words of length at least 5.

The minimal DFA for L:

	1	2	3	4	5	6
a	2	3	4	5	6	6
b	2	3	4	5	6	6

Leads to a linear system of polynomial equations:

$$
I - T^{\mathrm{tr}} x = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
-2\,x & 1 & 0 & 0 & 0 & 0 \\
0 & -2\,x & 1 & 0 & 0 & 0 \\
0 & 0 & -2\,x & 1 & 0 & 0 \\
0 & 0 & 0 & -2\,x & 1 & 0 \\
0 & 0 & 0 & 0 & -2\,x & 1 - 2\,x
\end{pmatrix}
$$

Solving

Which system we can solve very easily (in the right environment).

The solution vector is

$$1, 2\,x, 4\,x^2, 8\,x^3, 16\,x^4, \frac{32\,x^5}{1 - 2\,x}$$

and the generating function is

$$g(x) = \frac{32x^5}{1 - 2x}$$

Nice, but what exactly are the coefficients in $g(x) = \sum \gamma_L(n) x^n$?

More Solving

$$32x^5 = c_0 + \sum_{n \geq 1} (c_n - 2c_{n-1}) \, x^n$$

From this one can see that $c_n = 2^n$ provided that $n \geq 5$, and 0 otherwise.

The last step can be handled by a program (at least for sufficiently simple automata).

Open Environment

CS majors tend to be extremely opinionated about the relative merits and demerits of various systems, programs and languages.

No point forcing a particular environment on them when they are already more comfortable and experienced in another.

So, anything goes: Java, C, C++, ML, perl, python, Maple, Mathematica, . . .

Computational Support for Experimental Mathematics

- Document-centric user interface:
  - compute,
  - visualize,
  - store,
  - document.
- Interactive, expressive prototyping language.
- General compute engine to manipulate data.
- Reasonably large algorithm base.
- Fast code to generate sizeable structures.
- Seamless connection between the modules.
- Strong support for graphics.

My Workhorse: Mathematica

- Mathematica kernel for general manipulation.
- `Automata` algorithm library.
- Context sensitive online help and examples.
- Notebook frontend as user interface.
- MathLink communication protocol.
- External $C++$ code for fast computation.

Preserving Content

> *"Universities generate content every day through their courses and seminars. Then they throw it away. There is a certain charm with this approach, but it is not cost effective. Universities operate like renaissance quartets based on live performances. . . . Content storage and reuse are also important to test and ameliorate performance and to generate an institutional memory."*

> *Dennis Tsichritzis, "Reengineering the University,"*
> *CACM June 1999*

Preserving Content, II

"Jesus saves, and so should you."

Dana Scott

Blackhole$^{\text{TM}}$ Course Destruction System

Popular course management systems are not a solution.

In fact, they exacerbate the problem.

Delivery is not the issue; the trouble starts after the final.

CCaps: Persistent, Personalized Storage

The objective of the proposed research is to design, implement and operate a system that will store, organize, index and make searchable course content, and preserve that content reliably over long periods of time. Course Capsules will not only archive the university generated learning objects, but also the creative work of the students. Content markup will allow the Capsules to respond intelligently to requests by the user, and to provide a knowledge base both for the period of active learning while on campus, and for professional development after graduation. For the faculty, Capsules will form a trusted repository of course material and will encourage sharing and reuse.

The impact of the proposed research is to provide better efficiency in terms of both effort and cost in the creation, use and reuse of course content for the faculty, and sustained, intelligent access to academic knowledge for the students.

Proof of Concept: Mathematics

- XML is expressive and flexible enough to make standards reasonable.

- It is easily parsed, generated and transformed.

- It can be transmitted easily over the web.

- It can be rendered in standard browsers.

- It is handled by third party software.

- It solves the persistence problem.

From Presentation to Semantics

A simple expression

$$ax^2 + bx + c$$

By convention probably a polynomial of degree 2.

One element in $\mathbb{R}[x]$ or maybe $\mathbb{Z}[x]$, or $\mathbb{Q}[x]$, or $\mathbb{C}[x]$, or ...

Computer Algebra

The internal representation in Mathematica (presentation markup).

```
Cell[BoxData[RowBox[{
    RowBox[{"a", " ",
      SuperscriptBox["x", "2"]}], "+", " ",
    RowBox[{"b", " ", "x"}], " ", "+", " ", "c"}]],
      "Input", CellLabel->"In[62]:="
]
```

Parsing

Retains just enough semantics to process the expression.

```
Plus[c, Times[b, x], Times[a, Power[x, 2]]]
```

Towards MathML . . .

```
XMLElement["math", {"xmlns" ->"http://www.w3.org/1998/Math/MathML"},
  {XMLElement[ "apply", {}, {XMLElement["plus", {}, {}],
   XMLElement[ "apply", {}, {XMLElement["times", {}, {}],
     XMLElement["ci", {}, {"a"}],
     XMLElement[ "apply", {}, {XMLElement["power", {}, {}],
       XMLElement["ci", {}, {"x"}],
       XMLElement["cn", {"type"->"integer"}, {"2"}]}]}],
    XMLElement[ "apply", {}, {XMLElement["times", {}, {}],
      XMLElement["ci", {}, {"b"}],
      XMLElement["ci", {}, {"x"}]}],
      XMLElement["ci", {}, {"c"}]}]}]
```

Slightly more semantics: function application, some typing.

. . . and OpenMath

```
XMLElement["OMOBJ", {}, {XMLElement["OMA", {},
{XMLElement["OMS", {"cd"->"arith1", "name"->"plus"}, {}],
 XMLElement["OMA", {}, {XMLElement["OMS", {"cd"->"arith1","name"->"times"},{}]
 XMLElement["OMV", {"name"->"a"}, {}],
 XMLElement["OMA", {}, {XMLElement["OMS", {"cd"->"arith1","name"->"power"},{}],
        XMLElement["OMV", {"name"->"x"}, {}],
        XMLElement["OMI", {}, {"2"}]}]}],
 XMLElement["OMA", {}, {XMLElement[ "OMS", {"cd"->"arith1","name"->"times"},{}]
        XMLElement["OMV", {"name"->"b"}, {}],
        XMLElement["OMV", {"name"->"x"}, {}]}],
        XMLElement["OMV", {"name"->"c"}, {}]}]}]
```

Content dictionaries, all symbols bound to appropriate theories.

Comments and Formal Properties

```
<Definition id="c6s1p4.d1" for="monoid">

<CMP>
A <with role="definiens">monoid</with> is a structure (M, *, e)
where (M, *) is a semi-group and e is a unit for *.
</CMP>

<FMP>
∀M, *, e.s_grp(M, *) ∧ unit(e)  →  monoid(M, *, e)
</FMP>

</Definition>
```
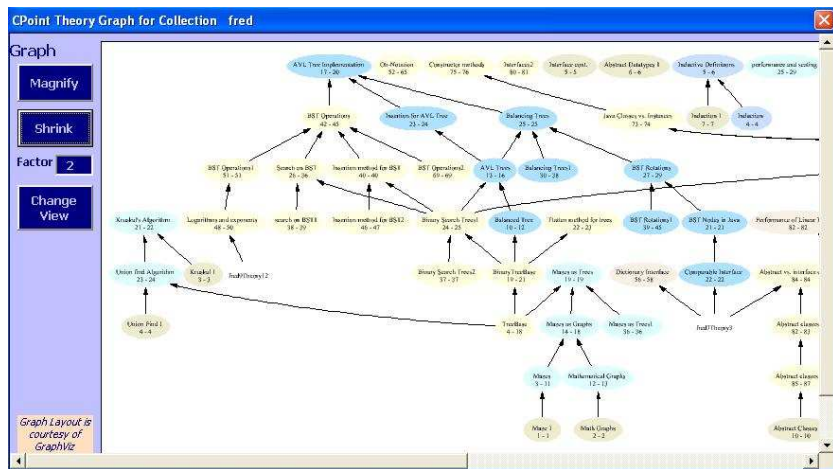
The Good/Bad News

The initial conversion can be done relative painlessly within Mathematica.

But the final result is (at best) very raw OpenMath.

Requires significant additional editing, currently in rather primitive environments.

At present, almost prohibitively labor-intensive.

Legacy Material

Availability

All course materials (except for exams) are posted on the web, including the computational machinery.

Generates some interesting email queries.

FCEs

For whatever it's worth:

- Course rating 4.52, 4.89 and 4.27

- Instructor rating 4.61, 5.00 and 4.73

Student Comments

- Theoretical aspects and connections are a great thing that I learned in this course.

- This class covers a lot of ground, but it isn't impossible to understand, and it is extremely interesting.

- [. . .] could make studying the mathematics of a rock interesting.