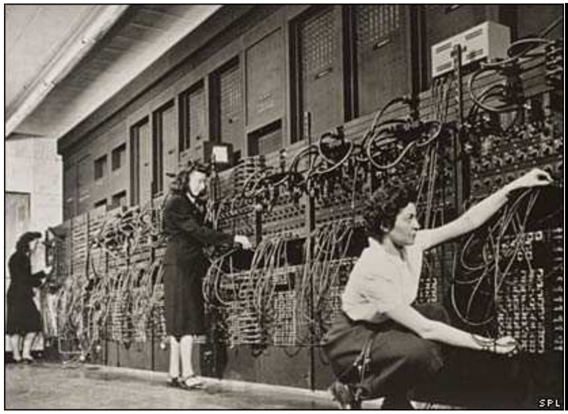
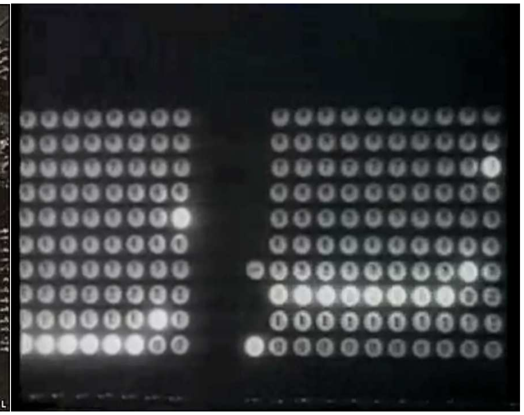
Reasoning with computer experiments in mathematics and computer science





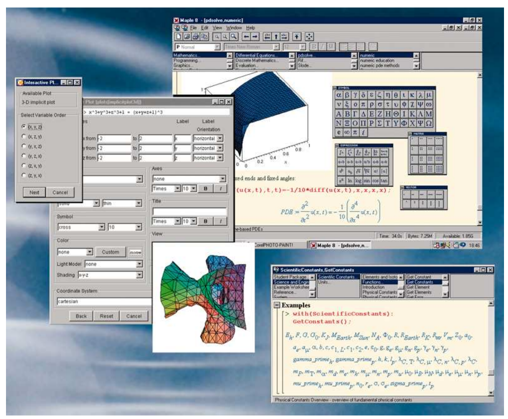
```
(define (initial-vec seq l n)
  (if (= l 0)
      seq
      (initial-vec (vector-append seq (vector (random n))) (- l 1) n)))
(initial-vec #(0) 300 2)


(define (add-period diff-periods n-period)
  (define (in-list? item seq)
    (cond ((null? seq) #f)
          ((equal? item (car seq)) #t)
          (else (in-list? item (cdr seq)))
          )
    )
  (cond ((= n-period (- 1)) diff-periods)
        ((in-list? n-period diff-periods) diff-periods)
        (else (append (list n-period) diff-periods))))

(add-period '(6) 1)

(define (determine-max r-max n-max?)
  (cond ((>= r-max n-max?) r-max)
        (else n-max?)))


(define (tag-it-vec count words v word ref-word ref-count)
  (define (deletion v word)
    (vector-tail word v))
  (define (produce words v word)
    (let ((appendent (vector-ref words (vector-first word))))
          (if (< (+ (- (vector-length word) v) (vector-length appendent)) 1) #()
              (deletion v (vector-append word appendent)))))
  (define (halt? word)
    (equal? word #()))
  (define (period? w1 w2)
    (equal? w1 w2))
  (define (period count ref-count)
    (- count ref-count))
  (let ((production (produce words v word)))
    (case (modulo count 1000000)
      ((0) (display count) (newline)(display (vector-length word)) (newline)))
    (cond ((period? word ref-word) (list count 0 1 (period count ref-count)))
          ((halt? production) (list count 1 0 (- 1)))
          ;((= count 100000) 1)
          ((= 0 (modulo count 1000)) (tag-it-vec (+ count 1) words v production word count))
          (else (tag-it-vec (+ count 1) words v production ref-word ref-count))
          )
    )
  )
```



Liesbeth De Mol
Centre for Logic and Philosophy of Science, Belgium
elizabeth.demol@ugent.be

Dedicated to the memory of Benoît Mandelbrot
A passionate and inspiring maverick
(November 1924-October 2010)

Intro.

Introduction (1)

⇒ **Motivation:** The increasing use of the computer in math seems to go hand-in-hand with growing significance idea "experimentation" in math – "computers [are] changing the way we do mathematics" (Borwein, 2008)

⇒ **Many Questions concerning "computer experiments" in math**

- To what extend is the computer *really* changing math?
- "What, if anything, is an experiment in mathematics?"
- What kind of knowledge can we expect from computer experiments?
- Difference "computer experiment" and "computer-assisted math"?
- etc

⇒ **Approach** Tracing "computer experiments" in math in the last 60 years to provide insight in some of the fundamental questions → Start from what mathematicians *themselves* consider as "experimental math"

Note: *Research in Progress!* Snapshot of the history of computer-assisted experiments in math

Introduction (2): Detailed approach

⇒ Guiding principle: "material" and "social" changes on the macro level (relative):

- Increase in speed and memory
- input-output devices
- Programming: languages, software and user interfaces
- miniaturization (vacuüm tube, transistor, chip)
- availability and knowledge of computer (computer time)

⇒ to understand changes on the level of:

- Mathematician(s)-computer(s) interactions: distribution of information and "responsibility"
  * Tools of storage of information: how is the (distributed) information during "experimentation" stored, where is it located and how is it distributed (in time and space)?
  * Processing of information: who processes and "creates" the information during "experimentation" and what are the main methods/approaches used during experimentation?

  ⇒ Major concepts: "internalization" and "time-squeezing"
- The views, goals and expectancies of the mathematicians

Introduction (2): Detailed approach

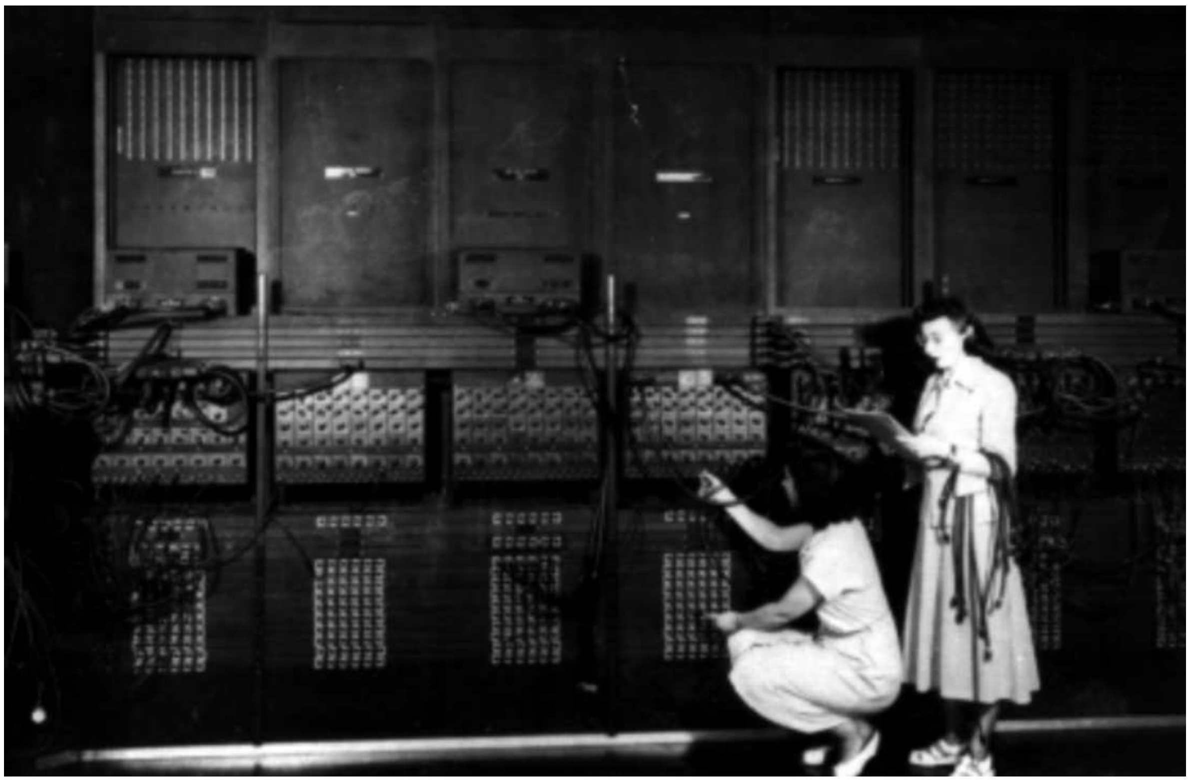
⇒ Study through concrete cases in the short history of computers and math

- Case I: The ENIAC experience: Lehmer and von Neumann (1946-50)
- Case II: Mandelbrot and his set (1979-80)
- Case III: Brady and Busy Beavers (1966, 1983)
- (Short) Case IV: Wolfram, Mathematica and “A new kind of science”

⇒ Note 1: Gap between Case I and II-V

⇒ Note 2: Selection of the cases – “good” examples related to “new” developments on the macro level

Case I: The ENIAC experience: Lehmer and von Neumann

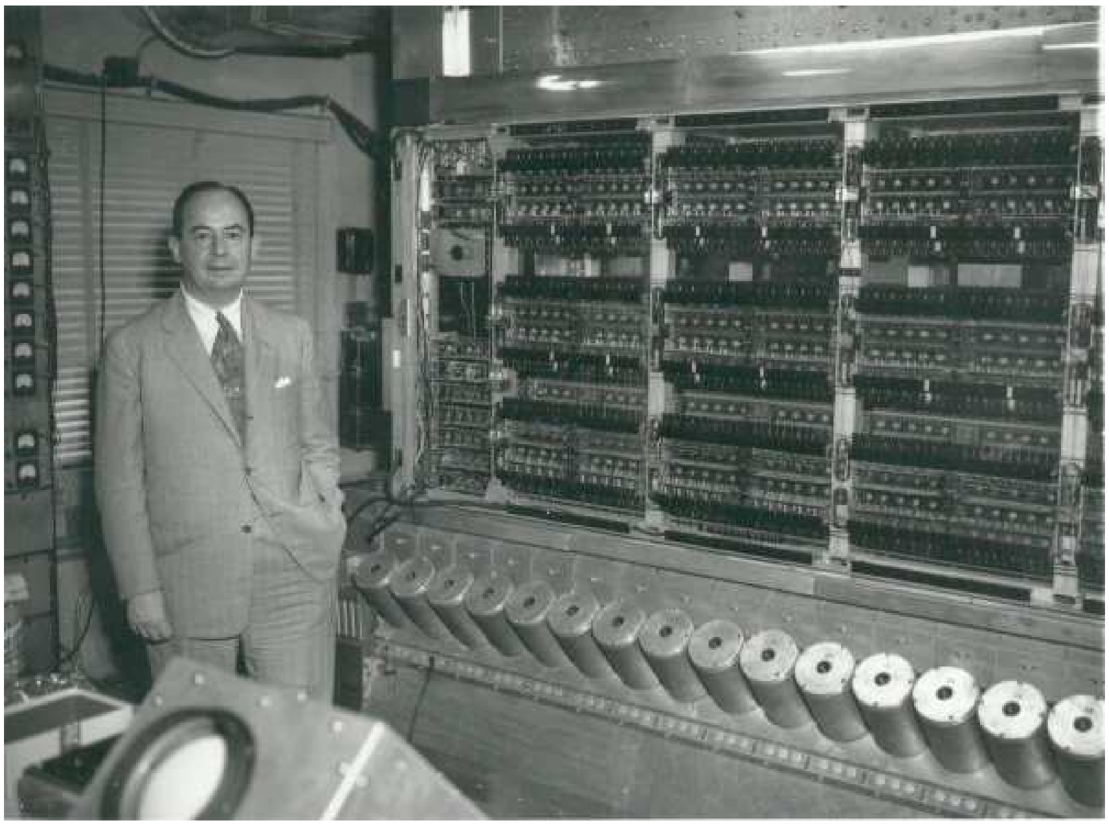


The behemoth ENIAC

- **Speed** addition time of 1/5000 second (revolutionary at that time!) / 0.5 seconds card reading, 0.6 seconds card printing + ENIAC1: parallelism
- **Memory** a constant transmitter (4 signs and 20 digits); 3 function tables 3×104 signed values; punched card input through CT
- **Arithmetic units** 20 accumulators (20 signed 10-digit numbers)
- **i/o** an IBM punched card reader and printer and a punch-card printer + IBM tabulator
- **Programming** "branching" + Local programming method: "The ENIAC was a son-of-a-bitch to program" (Jean Bartik) → externalized programs; more than just a "big" calculator
- **Miniaturization** 18.000 vacuum tubes; 1.500 relays and 40 panels to form 30 units
- **Availability and knowledge** Security clearance required to access computer → highly restricted access to some happy few

> *"The original "direct programming" recabling method can best be described as analogous to the design and development of a special-purpose computer out of ENIAC component parts for each new application [...] "* (Fritz, 1994)

“Johny”, ENIAC and the normality of π and e.



“Johny”, ENIAC and the normality of π and e. Context

Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin. von Neumann, 1949

- Stan Ulam – idea of using the “Monte Carlo” method for “computer simulations” (“numerical models”) in nuclear physics → picked up by von Neumann: “The statistical approach is very well suited to a digital treatment” → a need for random numbers → VN’s middle square method + external representation of random decisions on “neutron” punched cards
- “Early in June, 1949, Professor John von Neumann expressed an interest in the possibility that the ENIAC might sometime be employed to determine the value of π and e to many decimal places with a view toward obtaining a statistical measure of the randomness of distribution of the digits [...] Further interest in the project on π was expressed in July by Dr. Nicholas Metropolis [...]” (Reitwiesner, 1950)

How was ENIAC used to compute π and e?

- **Time issues** "Since the possibility of official time was too remote for consideration, permission was obtained to execute these projects during two summer holiday week ends when the ENIAC would otherwise stand idle"

$$e = \sum_{n=0}^{\infty} (n!)^{-1}$$

$$\pi/4 = 4 \text{ arctan } 1/5 - \text{ arctan } 1/239$$

$$\text{arctan } x = \sum_{n=0}^{\infty} (-1)^n (2n+1)^{-1} x^{2n+1}$$

- Computation cut into pieces of intervals of digits of π and e

How was ENIAC used to compute π and e?

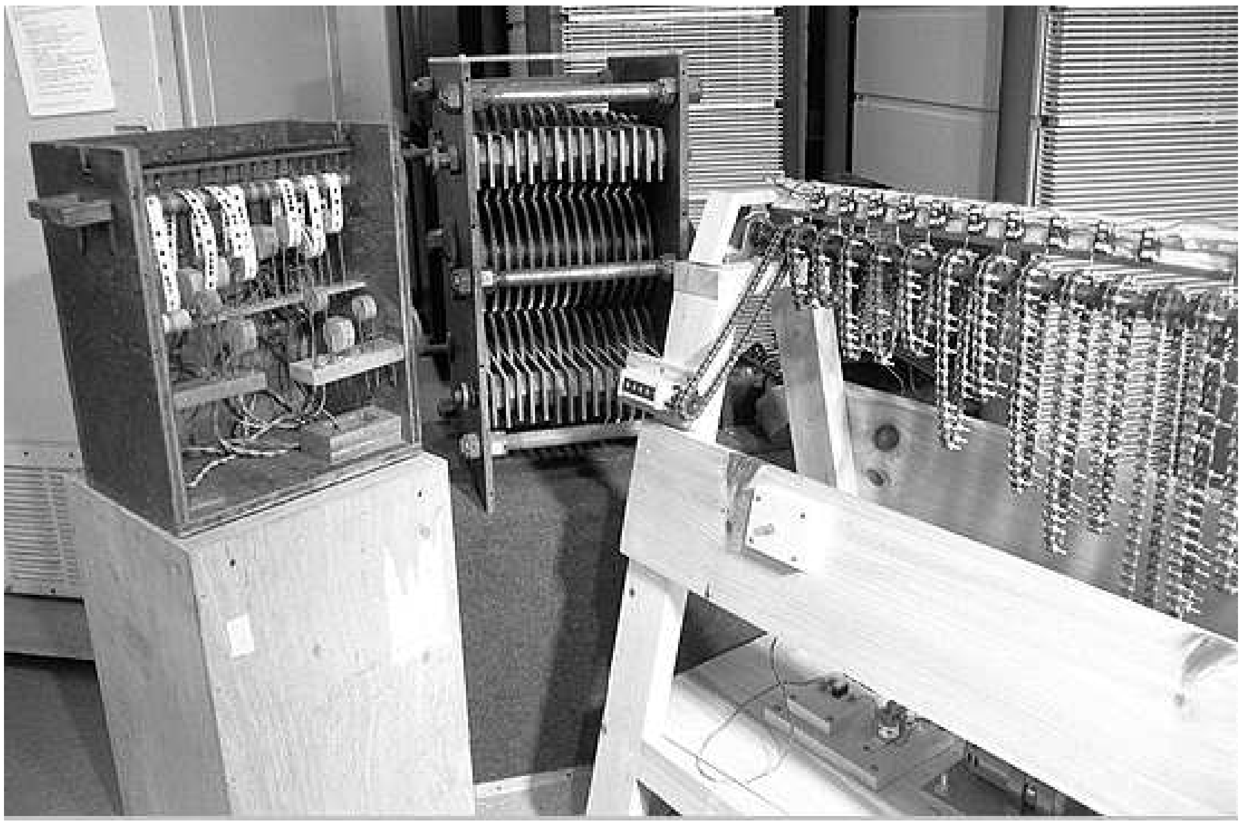
- **Possibility of error**: "In order to insure absolute digital accuracy, the programming was arranged so that one half applied to computation and the other half to checking. Before any deck of cards was employed to determine the next i digits, the cards were reversed and employed in the checking sequence to each division by a multiplication and each adition by a subtraction and vice versa [...]"
  The ENIAC determinations of both π and e confirm the [previously made] 808-place determination[s] of e and π

- **External/human exploration** The relevant fact about the distribution [of digits in e] appears upon direct **inspection**. [I]n order to see how this peculiar phenomenon develops as n increases to 2000, [the relevant statistical data] have been determined for smaller values of n. These numbers show that the abnormally low value of p_n which is so conspicuous at $n = 2000$ does not develop gradually, but makes its appearance quite suddenly around $n = 1900$. Up to that point, p_n oscillates considerably and has a decreasing trend, but at $n = 2000$ there is a sudden dip. [T]hus something number)theoretically significant may be occurring at about $n = 2000$"

π = 3.14159	26535	89793	23846	26433	83279	50288	41971	69399	37510
58209	74944	59230	78164	06286	20899	86280	34825	34211	70679
82148	08651	32823	06647	09384	46095	50582	23172	53594	08128
48111	74502	84102	70193	85211	05559	64462	29489	54930	38196
44288	10975	66593	34461	28475	64823	37867	83165	27120	19091
45648	56692	34603	48610	45432	66482	13393	60726	02491	41273
72458	70066	06315	58817	48815	20920	96282	92540	91715	36436
78925	90360	01133	05305	48820	46652	13841	46951	94151	16094
33057	27036	57595	91953	09218	61173	81932	61179	31051	18548
07446	23799	62749	56735	18857	52724	89122	79381	83011	94912
98336	73362	44065	66430	86021	39494	63952	24737	19070	21798
60943	70277	05392	17176	29317	67523	84674	81846	76694	05132
00056	81271	45263	56082	77857	71342	75778	96091	73637	17872
14684	40901	22495	34301	46549	58537	10507	92279	68925	89235
42019	95611	21290	21960	86403	44181	59813	62977	47713	09960
51870	72113	49999	99837	29780	49951	05973	17328	16096	31859
50244	59455	34690	83026	42522	30825	33446	85035	26193	11881
71010	00313	78387	52886	58753	32083	81420	61717	76691	47303
59825	34904	28755	46873	11595	62863	88235	37875	93751	95778
18577	80532	17122	68066	13001	92787	66111	95909	21642	01989
38095	25720	10654	85863	27886	59361	53381	82796	82303	01952
03530	18529	68995	77362	25994	13891	24972	17752	83479	13151
55748	57242	45415	06959	50829	53311	68617	27855	88907	50983
81754	63746	49393	19255	06040	09277	01671	13900	98488	24012
85836	16035	63707	66010	47101	81942	95559	61989	46767	83744
94482	55379	77472	68471	04047	53464	62080	46684	25906	94912
93313	67702	89891	52104	75216	20569	66024	05803	81501	93511
25338	24300	35587	64024	74964	73263	91419	92726	04269	92279
67823	54781	63600	93417	21641	21992	45863	15030	28618	29745
55706	74983	85054	94588	58692	69956	90927	21079	75093	02955
32116	53449	87202	75596	02364	80665	49911	98818	34797	75356
63698	07426	54252	78625	51818	41757	46728	90977	77279	38000
81647	06001	61452	49192	17321	72147	72350	14144	19735	68548
16136	11573	52552	13347	57418	49468	43852	33239	07394	14333
45477	62416	86251	89835	69485	56209	92192	22184	27255	02542
56887	67179	04946	01653	46680	49886	27232	79178	60857	84383
82796	79766	81454	10095	38837	86360	95068	00642	25125	20511
73929	84896	08412	84886	26945	60424	19652	85022	21066	11863
06744	27862	20391	94945	04712	37137	86960	95636	43719	17287
46776	46575	73962	41389	08658	32645	99581	33904	78027	59009
94657	64078	95126	94683	98352	59570	98258			

Figure 1: The first 2035 digits of π computed by the ENIAC, at the Ballistics Research Laboratory.

Lehmer and the first extensive number-theoretical computation on the ENIAC (Joint work with M. Bullynck)



Lehmer and the first extensive number-theoretical computation on the ENIAC

- "*[Lehmer] had programmed the problem and run it on ENIAC, with J. Mauchly serving as "computer operator", during the three-day weekend of July 4, 1946.* **The running time of the problem occupied almost the entire weekend, around the clock, without a single interruption or malfunction. It was the most stringent performance test applied up to that time, and would be an impressive one even today**." (Alt, 1972)

  "*I think what's particularly interesting about the number theory problem they ran was that this was a difficult enough problem that it attracted the attention of some mathematicians who could say,* **yes, an electronic computer could actually do an interesting problem in number theory**" (Alt, 2006)

- A special case of the converse of Fermat's little theorem

  Theorem 1 *If n divides $2^n - 2$ then n is a prime*

- **Goal I** Testing the machine

- **Goal II** Finding composite numbers to generate tables of primes

- **Goal III** Exploration of prime number tables in number theory

How was ENIAC used to compute composite numbers?

- The ENIAC was used to determine a list of exponents e of 2 mod p, i.e., the least value of n such that $2^n \equiv 1 \bmod p$ with p prime
- These exponents can be used to determine composite numbers of the form $2^{pq} - 2$ through the theorem:

  Theorem 2 *If p and q are odd distinct primes, then $2^{pq} - 2$ is divisible by pq if and only if p - 1 is divisible by the exponent to which 2 belongs modulo q and q - 1 is divisible by the exponent to which 2 belongs modulo p*
- Compute small numbers to compute big numbers
- A sieve was implemented on the ENIAC to determine primes relative to the first 15 primes, thus making use of the ENIAC's parallelism. The last prime p processed, after 111 hours of computing time, was $p = 4,538,791$
- Eratosthenes's Sieve:

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & \ldots \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & \ldots \end{pmatrix}$$

How was ENIAC used to compute composite numbers?

- **Computing from the machine's point of view** "The method used by the ENIAC to find the exponent of 2 modulo p differs greatly from the one used by human computer" (Lehmer, 1949)
  "In contrast, the ENIAC was instructed to take an "idiot" approach"

$$\Gamma_1 = 2, \Gamma_{n+1} = \begin{cases} \Gamma_n + \Gamma_n & \text{if } \Gamma_n + \Gamma_n < p \\ \Gamma_n + \Gamma_n - p & \text{otherwise} \end{cases}$$

  Only in the second case can Γ_{n+1} be equal to 1. Hence this delicate exponential question in finding $e(p)$ can be handled with only one addition, subtraction, and discrimination at a time cost, practically independent of p, of about 2 seconds per prime. **This is less time than it takes to copy down the value of p and in those days this was sensational**." (Lehmer, 1974)

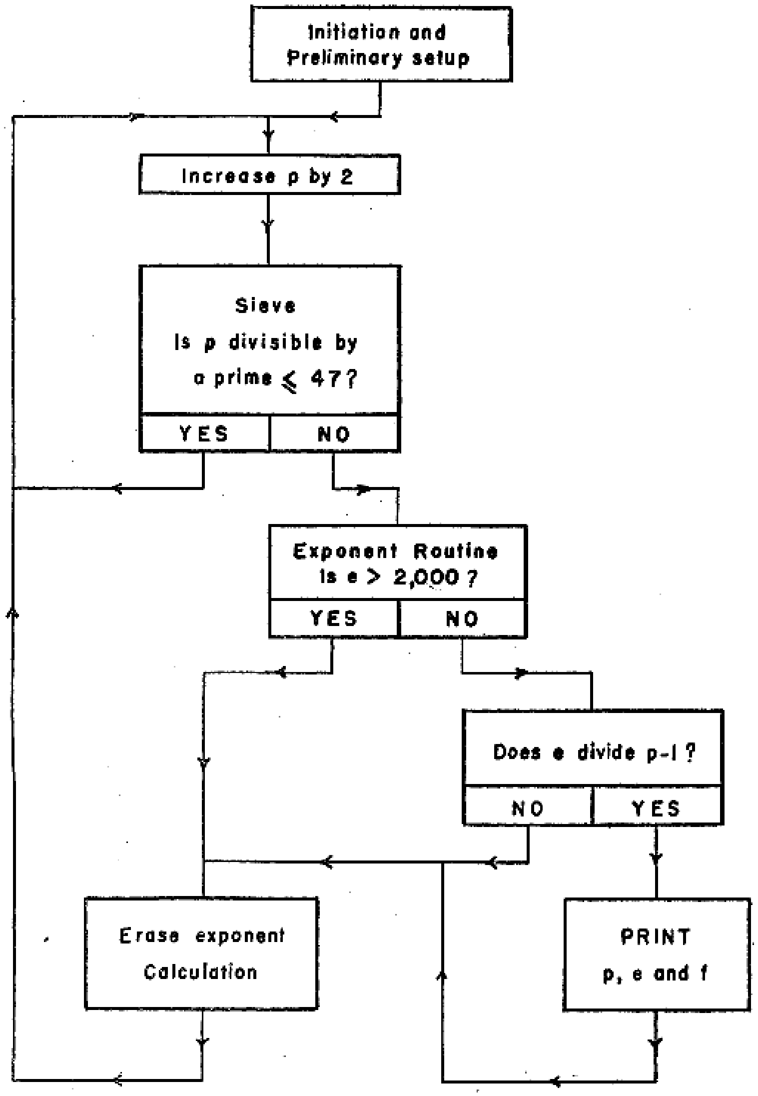
- **Internalization and heuristic program** "The "next value of p" [i.e. the next prime] presents an interesting problem to the ENIAC. [Circumstances] prevented the introduction [of] punched cards. [...] This means that the ENIAC should somehow compute its own values of p. To this effect a "sieve" was set up which screened out all numbers having a prime factor ≤ 47. [Else there is a need for] "much outside information [introduced] via punched cards [...] to be prepared by hand in advance" + 25 out of 11336 eliminated by hand

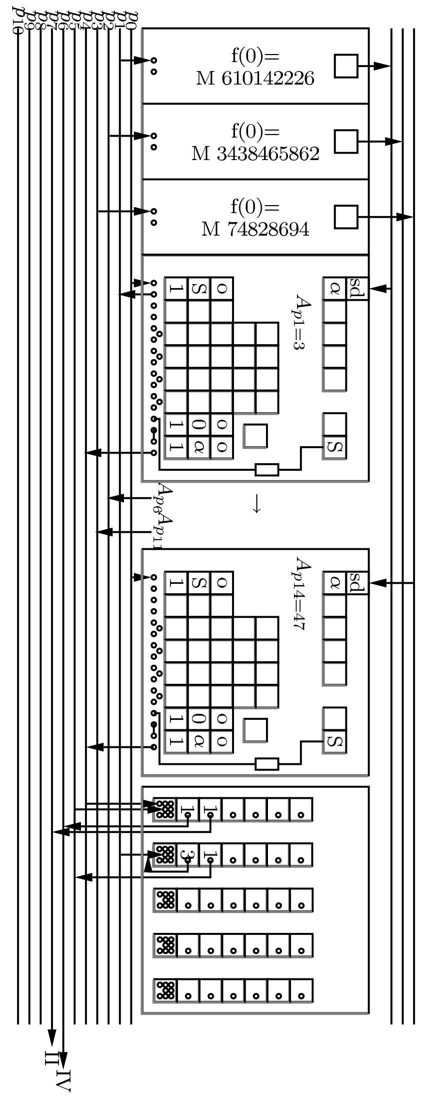
How was ENIAC used to compute composite numbers?

- Lehmer's little problems, they were always too big for it. So consequently, you always had to be changing it or to think of something new and innovative in order to get a problem or ways that you could break the problem down into smaller portions. (Jean Bartik, 1973)
- **External/human processing** "The list of exponents furnished by ENIAC is sufficient for the extension of the table of composite numbers n dividing $2^n - 2$ to 10^9 and beyond. However the list presented herewith extends only from 10^8 to 2×10^8. **The labor of producing these composite numbers is still considerable.**"

TABLE OF COMPOSITE SOLUTIONS n OF FERMAT'S CONGRUENCE $2^n \equiv 2 \pmod{n}$ AND THEIR SMALLEST PRIME FACTOR p

n	p	n	p	n	p
100463443	7577	312773	3541	558011	6449
618933	4729	413333	6067	940853	503
860997	9649	495083	1987	120296677	229
907047	5023	717861	1013	517021	2341
943201	5801	111202297	5273	838609	433
101152133	5807	370141	883	121062001	1201
158093	3673	654401	6101	128361	6961
218921	8713	112032001	4001	374241	6361
270251	9001	402981	3061	121472359	4409
276579	6163	828801	6133	122166307	739
954077	1597	844131	3067	396737	2857
102004421	2381	113359321	761	941981	337 · 491
443749	4049	589601	331 · 571	123330371	691
678031	3583	605201	7537	481777	3881
690677	2069	730481	433	559837	4177
690901	5851	892589	919	671671	9631
103022551	6121	114305441	6173	886003	1187
301633	7873	329881	7561	987793	709
104078857	6679	469073	3089	124071977	2089
233141	2441	701341	1229	145473	397
524421	5903	842677	2459	793521	4561
105007549	1033	115085701	1801	818601	2281
305443	2833	174681	773	125284141	4231
919633	4603	804501	5381	686241	6473
941851	1051	873801	1051	848577	2897
106485121	7297	116090081	6221	126132553	5023
622353	433	151661	7621	886447	6793
743073	1699	321617	5393	127050067	5347
107360641	2161	617289	2357	710563	9787
543333	4889	696161	2161	128027831	11161
108596953	7369	998669	1459	079409	5437
870961	2609	117246949	1597	124151	2311
109052113	4993	445987	5419	468957	2927
231229	2699	959221	2053	536561	8017
316593	3697	987841	7681	665319	2383
437751	5231	118466401	1249	987429	4637
541461	6043	119118121	2729	129205781	6563
879837	2707	204809	2383	256273	739
110135821	3967	261113	4657	461617	10177
139499	6427	378351	911	524669	2939


Initiation and Preliminary setup
Increase p by 2
Sieve
Is p divisible by a prime ≤ 47?
YES
NO
Exponent Routine
Is e > 2,000 ?
YES
NO
Does e divide p-1 ?
NO
YES
Erase exponent Calculation
PRINT
p, e and f


f(0)=
M 610142226
f(0)=
M 3438465862
f(0)=
M 74828694
A_{p1}=3
A_{p14}=47

Distribution of (processing and tools of) information? Discrepancy between speed processing vs. retrieval and storage; restrictions on programmability (no intermediary language), processing power and memory (external storage and retrieval) and availability

- (Processing) one long computation put on ENIAC, without "responsive interactions" – extra tests by hand; major part of interpretation or further manipulation of information by hand.
- (Processing): Multi-purpose machine: centralization of different kinds of computations
- (Processing) Need for "computations" from the machine's point of view (speed and "machine thinking")
- (Processing and Storage) call-by-value-method: eg Lehmer's sieve; internalized random number generator: **partial internalization of processes**
- Storage *Data* during and after computation externally stored on *slow* punched cards → need for human intervention (reversal of the cards)

⇒ **Discontinuous process of "experimentation"** Separated phases of the experiment distributed over human and machine

⇒ Info from experiment are/must be **humanly practical**

⇒ BUT: The ENIAC experience: programmability and "time squeezing" – a gathering of different mathematician-machine meetings

The views of von Neumann on "computer experiments" in math

- **The computer instead of physical explorative experiments** "In pure mathematics the really powerful methods are only effective when one already has some intuitive connection with the subject, when one already has [some] **intuitive insight**. [T]here are large areas in pure mathematics where we are blocked by a peculiar interrelation of rigor and intuitive insight, each of which is needed for the other, and where the unmathematical process of experimentation with physical problems has pro- duced almost the only progress. **[C]omputing, which is not too mathematical either in the traditional sense but is still closer to the central area of mathematics than this sort of experimentation is, might be a more flexible and more adequate tool in these areas than experimentation**"

- **Human practicality of the information** "The reason for using fast computing machines is not that you want to produce a lot of information. **[W]herever there may be bottlenecks in the automatic arrangement which produces and processes this information, there is a worse bottleneck at the human intellect into which the information ultimately seeps.** The really difficult problems are of such a nature that the number of data which enter is quite small. All you may want to know is a few numbers, which give a rough curve, or one number. All you may [is] a yes or a no,”

The views of Lehmer on "computer experiments" in math

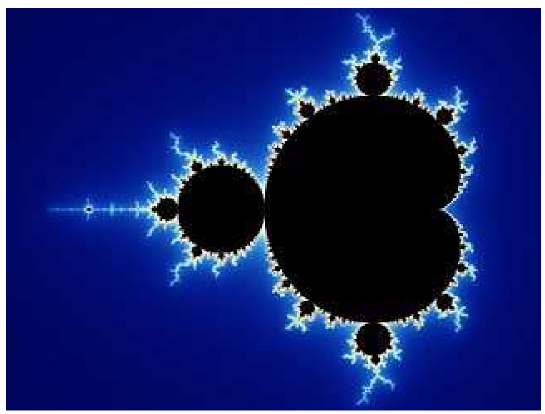
- **The computer as a means to explore number theory** "[T]he most important influence of the machines on mathematics should lie in the opportunities that exist for applying the **experimental method** to mathematics. [...] Many a young Ph.D. student in mathematics has written his dissertation about a class of objects without ever having seen one of the objects at close range. There exists a distinct possibility that the new machines will be used in some cases to **explore** the terrain that has been staked out so freely and that something worth proving will be discovered in the rapidly expanding universe of mathematics."
- Lehmer's classification of human-machine mathematics
  - Searching for counterexamples
  - Verification and exploration of cases of a proposition to find ideas for a proof (or formulate support for conjecture)
  - Construction and inspection of tables: "Not only is the publication of such tables impossible; even the inspection is well beyond human capability. It soon becomes apparent that it should be the machine's responsibility to make this inspection"
  - Verification of a large number of cases $\Rightarrow$ Lehmer's version of "true" theorem proving

The views of Lehmer on "computer experiments" in math

Humanly impractical problems "*In casting about for genuine theorems the proofs of which will tax the powers of a human being,* **we want to exploit the speed of the machine**. *This means that the proof must involve* **many thousands of steps** *all sufficiently different so that the* **outcome cannot be forecast**. *We must also exploit those features of the logical system of the machine that* **permit it to supervise and organize its own program. We should make it proceed in an unpredictable way by laying its own track ahead of it like a caterpillar tractor.** *At the same time it should keep a record of where it has been, so that it can return at a previous point and branch out along another path whenever it decides that this is necessary. Humans find this kind of work difficult even when it occurs in only moderate amounts.*"

Case II: Mandelbrot and his set





DEC-VAX-11/780 ("star")

- Follow up of the PDP-11. Release dat: October 25, 1977; first one installed at CERN
- **Speed** 3.4 MhZ (10^6 herz); 500,000,000 instructions/second
- **Memory** From 128kb up to 8 MB static memory! 1MB static and 4k ram
- **Miniaturization** Tektronix terminal, versatec printer
- **Programming** VMS operating system ("starlet"), with GUI and graphics support (!); support for multiple programming languages (FORTRAN, COBOL, BASIC? PASCAL, etc)
- **Miniaturization** Transistors
- **Availability and knowledge** $200,000 (sold until 1988); wider availability for universities and firms: "*In 1980, the programming skills and minimal computer power needed to produce some kind of picture were widely available to anyone who asked. In terms of exploratory research, everyone was in the same boat. [W]e could only work at night when we had only one competitor for the machine, chemist Martin Kaplus*" (Mandelbrot, 2004)"

Case II: Mandelbrot and his set

- Late 70s interested in the theory of rational maps of the complex plane (knowledge of work by Fatou and Julia) → "playing around" with quadratic Julia sets defined through iteration $z \rightarrow z^2 + c, c, z \in \mathbb{C}$
-

$$E_c = \{z_0 : |z_n| \rightarrow \infty\},\ z_n = z_{n-1}^2 + c$$

$$P_c = \{z_0 : z_0 \notin E_c \rightarrow \infty\}$$

  Julia set for c is the boundary of E_c.
  Fundamental dichotomy for Julia sets: connected and disconnected sets.

- For certain c, some points z always converge to a finite stable cycle of size n → attempt to classify Julia sets according to n (only connected sets!)
- **Fact:** The prisoner set P_c for $z \rightarrow z^2 + c$ is connected iff the orbit $0 \rightarrow c \rightarrow c^2 \rightarrow c^2 + c \rightarrow \ldots$ remains bounded.
- 1980: Exploration of the map $c \rightarrow c^2 + c$
- The Mandelbrot set:

$$M = \{c \in \mathbb{C} | c \rightarrow c^2 \rightarrow c^2 + c \rightarrow \ldots \text{remains bounded}\}$$

Case II: Mandelbrot and his set

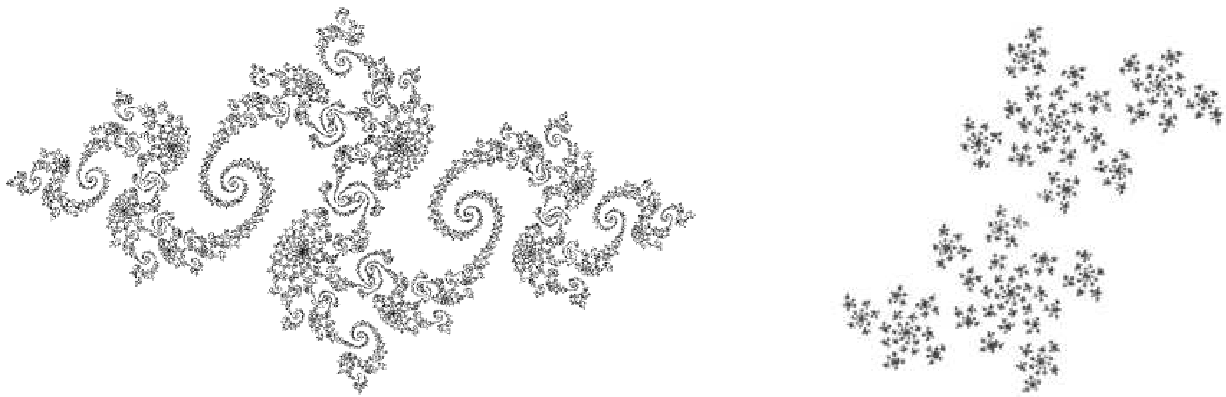


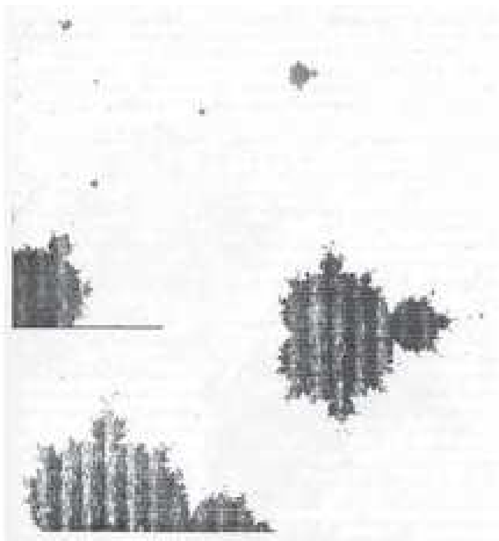
Figure 2: A connected and a disconnected Julia set

From seeing and knowing to discovering and results (1)

→ Exploration that resulted in several conjectures and (ultimately) proofs

- M as a road map for Julia sets: Classification of Julia sets with stable finite cycles n as smaller and smaller "sprouts" of M
- Connection between left-side of the cardioid and period-doubling bifurcations
- Apparent "specks of dirt" on print-out are "real" → zoom-in reveal "island whose shape is like that of M, except for a non-linear deformation. Each island is, in turn, accompanied by sub-islands, doubtless ad infinitum" (Mandelbrot, 1980)
- Julia sets with c's in specks of dirt + connectedness of these J-sets + theory of bifurcation: Conjecture M as a connected set (proven in 1982 – Douady & Hubbard)
- Observation: the hieroglyphic characer of M: "it includes within itself a whole deformed collection of miniature versions of all the Julia sets
- Observation self-similarity → Conjecture fractal dimension $M = 2$ (proven in 1991, Shishikura)

From seeing and knowing to discovering and results (2)



Possibility of machine error, intuition and knowledge “early originals looked awful: filled with apparent specks of dust that the Versatic rpinter produced [T]his complicated matters. But for the skilled, meticulous, and tireless observer that I was, mess was not a reason to complain but a reason to be particularly attentive”
“By a theorem Julia and Fatou, those Julia sets are connected. Therefore the broken-up appearances is necessarily due to the discrete variables used in computation. These graphs were important to my thinking because they sufficed to show hat the broken-up earlier early M set pictures were compatible with connectedness”
Machine-aided, human-directed exploration “When seeking new insights, I look, look, look and play with many pictures (One picture is *never enough*)”
“Incidentally, a picture is like a reading of a scientific instrument. One reading is never enough. Neither is one picture. More precisely, my discoveries of new mathematical conjectures relied greatly on the quality of visual analysis and little on the quality of the pictures”

Distribution of (processing and tools of) information? Increased speed and memory; stored programming + programming language; (clumsy) printing devices

- (Storage) *Internal* storage of thousands of "data"
- (Processing) Internal machine "processing/translation" of "low-level" data to humanly practical format, i.e., graphical picture of M
- (Processing) Development of efficient machine-adapted visualization techniques – encirclement of M $\rightarrow$ iteration on internally stored information
- ("Processing") Terminology and concepts directly inspired by pictures $\rightarrow$ result of interfacing between human and machine "processing" – **pictures as an interface**

$\Rightarrow$ "low-level" data are no longer humanly practical – internal machine computations "represented" in a humanly digestible way.

$\Rightarrow$ **"Time squeezing"**: Many smaller "experiments" and the flow of informations during this process of "experimentation" squeezed in a "reasonable" time-frame $\rightarrow$ increased involvement with the machine ($\approx$ "real-time" manipulations of the M-set)

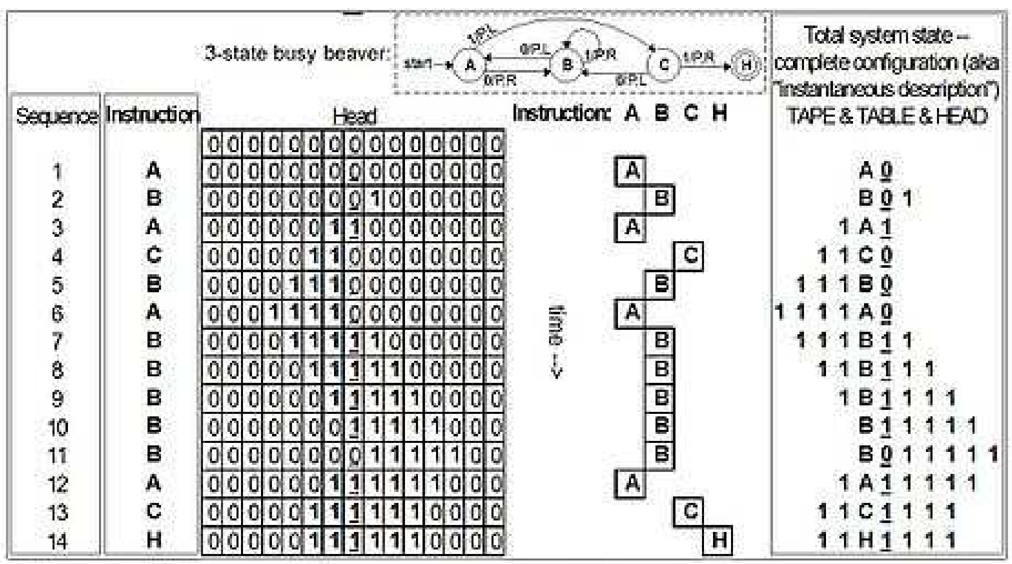
$\Rightarrow$ **More "continuous" process of human-machine "experimentation"**

$\Rightarrow$ **Increased interaction** during the "experiment": mixing of computation, exploration and interpretation in a process

Mandelbrot and his set (1980)
Mandelbrot's views

- My goal [with my 1980 paper on M] was to revive experimental mathematics by reporting observations triggering new mathematics"
- "A discovery is made when the tools [...] are available to an individual with the motivation, acuity, and inspiration to use them, and I was motivated to sniff out the ramifications of those speck of dirt"
- "The 'fate' that drove me to revive the theory of iteration, first chose me to **reinvent the role of the eye** in a field, mathematics, where it and explicit computation had become anathema, about as unwelcome as they could possibly be"
- "The interest that many mathematicians took in my observations/conjectures concerning the Mandelbrot set represented **an historical change** and a return to a far less ideological view of their craft."
- $\approx$ Lehmer: "Yesterday, "generality at all cost" was in the saddle. Today, "special" problems are more readily recognized as compelling."

Case III: Brady and Busy Beavers



Progress of the computation (state-trajectory) of a 3-state busy beaver

Case III: Brady and Busy Beavers.

Machine used? "A Turing machine simulator written in **a machine independent form of BASIC** is available from the author upon request"

The Busy Beaver problem Determine for any class of Turing machines TM$(m, 2)$ with m states and 2 symbols the maximum number of 1s $\Sigma(m)$ respectively the maximum number of computation steps $S(m)$ left on the tape by some $T \in$ TM$(m, 2)$ that halts when started from a blank tape. First formulated and proven recursively unsolvable by Rádo, 1962.
⇒ Early on, computer-assisted studies and proofs (!) of the Busy Beaver problem for particular m:
Some results

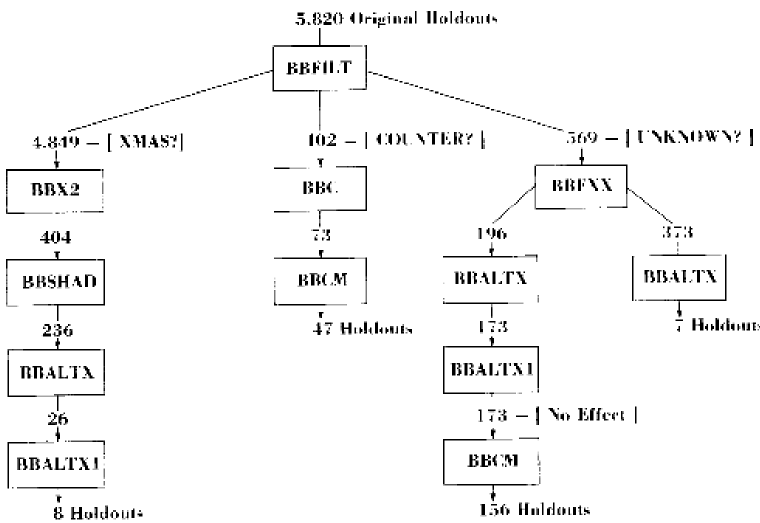
- $S(2) = 4, \Sigma(2) = 2$, Rádo (1962)
- $S(3) = 21, \Sigma(3) = 6$, Rádo and Lin (1965)
- $S(4) = 107, \Sigma(4) = 13$, **Brady (1983)** and Kopp (cited by Machlin and Kopp/Stout (1990))
- $S(5) =?47,176,870, \Sigma(5) =?4098$, Marxen and Buntrock (1990)
- $S(6) = 2.5 \text{ x } 10^{2879} \Sigma(6) > 4.6 \text{ x } 10^{1439}$, Terry and Shawn Ligocki (2007)

Computing Busy Beavers (1) (Brady, 1966)

- Notation instruction: (state, number read, number printed, move left/right, next state)
- The number n of Turing machines $T \in T(m, 2)$: $n = (4m + 1)^{2m}, m = 4, n = 6,975,757,441$
- Approach: Determine the set of halting machines by reducing the number of "hold-outs" to 0.
- Brady's 1966 reduction to 5820 hold-out: Tree normalization and backtracking
  1. Eliminate machines for which (1, 0, 1/0, L/R, halt); idem for (1, 0, 1/0, R/L, 1)
  2. Exclude the symmetrical left-right machines and retain the right-left machines (or vice versa).
  3. Generalization idea 1 (backtracking): prove that machine is in infinite loop by showing with backtracking that halting state cannot be reached → Generation of instructions as they are needed (e.g.: (1, 0, 1, R,2), only 8 out of the 16 possible next instructions need to be generated .

Computing Busy Beavers (2) (Brady, 1983)

⇒ Heuristic and experimental process of **Identification** (through exploration of hundreds of printouts) and **automated detection** of several types of infinite loops.



⇒ *"[I]t must be remembered that the filtering [BBFILT] was a heuristic technique based upon experimental observation."* → tentative classification based on the rate at which new squares are visited

⇒ Unpredictability and the need of making decisions in finite time ⇒ 218 remaining holdouts: "examined by means of voluminous printouts of their histories along with some program extracted features": discovery of a new loop: "Tail-eating dragons"

Computing Busy Beavers (3)

- Many different programs internalized: "More than 18 other programs were written, for various housekeeping purposes, simulating and displaying machine behavior, exploring other reduction and filtering possibilities etc. In all, at least 53 files were created and maintained for the project. Keeping track of what resembled a large scientific experiment became a major task in itself."

- **The problem of error** "While not all of the exploratory activities are reproducible, the runs [can] be reproduced, so that by utilizing the techniques described in this paper the proof can be corroborated. [...] Proofs of "correctness" of the programs used are not practical. Independent verification is the only means we currently have at our disposal."

Distribution of (processing and tools of) information?

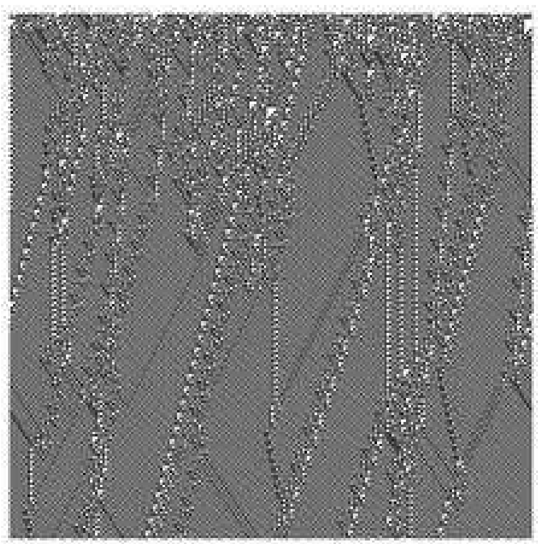
Increased speed and memory; portable programming language; (clumsy) printing devices

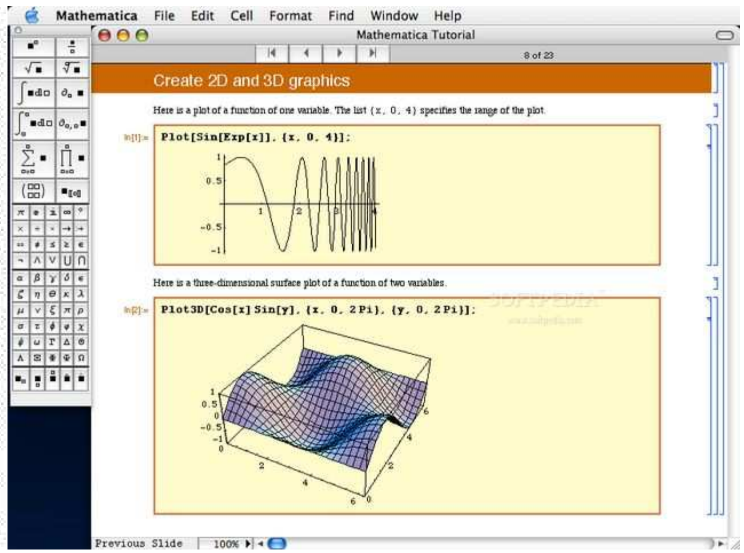
- (Processing and external Storage) "processing/translation" of simulated Turing machine behavior on display and print-out
- (Processing and Storage) Information that flows through Brady's flowchart of flowcharts
- (Processing) The machine does not do one thing (eg visualizing aspects of M) but many different things: Lehmer's flowchart vs. Brady's flowchart of flowcharts
- (Processing) Significant Internalized inspection and exploration of TM-behavior (human impracticality) → part of the proof is necessarily "unknown" (≈ un-surveyability) ⇔ "seeing" and "hearing" the computations of ENIAC
- ("Processing") Terminology and concepts directly inspired by human-made inspection of the print-outs

⇒ **Continuous and integrated process of human-machine "experimentation"**: Exploratory activities distributed between human and machine.

⇒ **Increased interaction and time-squeezing** Intertwinement of human and machine contribution. Proof within the interaction → Towards a human-computer collaboration

(Short) Case IV: Wolfram, Mathematica and a "new kind of science"





(Short) Case IV: Wolfram, Mathematica and a "new kind of science" (1984-1988)

- **Machine used?** the C language computer program;"CA: an interactive cellular automaton simulator for the Sun Workstation and VAX"; Connection Machine computer;...
- **Some technical (observational) results:** four classes of behavior; conjecture universality rule 110 ("*This paper covers a broad area, and includes many conjectures and tentative results. It is not intended as a rigorous mathematical treatment.*"); random number generator based on rule 30
- **Complex behavior simple programs:** "It is remarkable that such a simple system [rule 30] can give rise to such complexity. But it is in keeping with the observation that mathematical systems with few axioms, or computers with few intrinsic instructions, can lead to essentially arbitrary complexity. And it seems likely that the mathematical mechanisms at work are also responsible for much of the randomness and chaos seen in nature."
- **Complexity in physics** Undecidability and intractability in physics: "It is the thesis of this paper that [problems of computational irreducibility] are in fact common

⇒ **Before Mathematica:** Most of the basic results on CA already found

⇒ **Start development of a general theory inspired by computer science**

(Short) Case IV: Wolfram, Mathematica and a "new kind of science" (1988–now)

- **Mathematica (1988)**: "I first conceived of Mathematica because I needed it myself"
- "[T]he visionary concept of Mathematica was to create once and for all **a single system that could handle all the various aspects of technical computing**–and beyond–in a coherent and unified way."
- 2002: the long-awaited publication of "A new kind of science", based on theory of cellular automata as models for physical systems. Main method: "computer-based models and experiments"
- Connection Mathematica and "A new kind of science"?
- ≈ Maple and "Mathematics by Experiment" (Borwein and Bailey, 2004)

(Short) Case IV: Wolfram, Mathematica and a "new kind of science" (1988–now)

What is the significance of software like Mathematica and Maple for the development of "experimental mathematics"? (See e.g. Sorensen, 2010: fact-gathering vs. interactive exploration)

- "interactive exploration" is not the sole domain of Maple or Mathematica (See Cases)
- Significance from the perspective of information distribution, "internalization" and "time squeezing"

⇒ (Processing) Pre-programmed internalization and centralization of different aspects of "experimentation": statistical tools, graphics tools, special algorithms (user-friendliness)

⇒ **Time-squeezing** No wasting time on programming the tools; "real-time" manipulations and computations

⇒ **More continuous and integrated human-machine experiments**

⇒ **Increased (faster) interaction** Possibility of more "direct" interaction with the emulated/simulated objects studied.

⇒ Wider accessibility and integration of knowledge: development of "general" and "integrated" theories

Discussion

Discussion

- From the "behemoth" ENIAC to Mathematica/Maple: process of changing mathematician-machine interactions in terms of changing distribution of informations induced by macro-developments: from a discontinuous process of computer-assisted experimentation to a more continuous and integrated one
- (From a micro perspective) Change on the level of the method of experimentation: not one "smaller" experiment but many "phases" and "aspects" of experimentation integrated into one ("time squeezing" and "internalization")
- Changing views "experimental mathematicians"?

⇒ "**time-squeezing**" and "**internalization**": the "revolution" that started with ENIAC

Discussion: Many questions, thoughts...

- $\Rightarrow$ Distributed computing? The internet? Social aspects of math and computing (mailing, blogs, publishing, etc)
- $\Rightarrow$ **A(n) (computer) experiment in math?**? Mathematical (computer) experiment $\neq$ computation: Explicit integration of "pure" computations ("nature") with exploration, concept-formation, conjecturing, etc *and* heuristic and probabilistic programming $\Rightarrow$ **Not reasoning *with* but *in* computer experiment**
- $\Rightarrow$ **To what extend is the computer *really* changing math?** What is the difference between e.g. Brady's "explorations" and Gauss' "explorations"?
- $\Rightarrow$ "Progress" and the necessity of hiding the "source"? $\approx$ Heidegger
- $\Rightarrow$ **A "paradox" of mathematician-computer interactions?** Growing distances between mathematician and physical computer and time-squeezing results in more direct and intertwined interactions that reflect upon our thinking on "experimental math"
- $\Rightarrow$ "In any event, whenever [the] stage [of high baroque] is reached [in mathematics], the only remedy seems to me to be the rejuvenating return to the source: **the reinjection of more or less directly empirical ideas**." (Von Neumann, 1947) $\Rightarrow$ reinjection of time into mathematics as a fundamental question for computer-assisted math?