M.J. Fischer, *et al.*,

The First Decade: mid-60's to 70's

Division of Engineering and Applied Physics, Harvard, 1965

I was TA & Mike was 1st year student in Applied Math (206?), taught by Oettinger

Wow, was Mike smart! – and experienced:

Already (at Michigan)

- equivalence (union/find) algorithms
- MAD compiler & parsing

Parsing

An *important theme* for Mike:

- paper on precedence parsing (1969 STOC)
- thesis on Macro-grammars (1968, Greibach)
- *LINGOL* with Pratt

Parsing

...This is an area in which I have not worked, but I can relay the comment of Harrison at Berkeley who told me shortly after reading Fischer's paper on the subject that he was convinced that he and Fischer were the only two people who *really* understood precedence grammars.

--letter from Meyer, Jan. 2 1975, to EECS Dept. Head supporting Mike's promotion to tenure.

Harvard DEAP '63 — '68

A romantic time --

pioneering a new discipline of theoretical Computer Science:

- Wang (& Cobham) on logic & computational complexity -- students Aanderaa, Cook
- Pat Fischer on Recursive Functions, Automata, "real-time" automata -- students Cole, Meyer, Ritchie, Rosenberg

Harvard DEAP '63 — '68

- Even & Greibach on automata & languages
  -- students Mike Fischer, Book

And nearby:

- Rabin, Blum: time-bounded complexity
- McNaughton/Papert, Krohn/Rhodes:
  Algebraic automata theory

Harvard DEAP '63 — '68: people

Stol Aanderaa	*G. Birkhoff*
Ron Book	Steve Cook
Shimon Even	*Pat Fischer*
Sheila Greibach	Albert R. Meyer
Tony Oettinger	Dennis Ritchie
Arny Rosenberg	*Hao Wang*
Alan Cobham	*Michael Rabin*
Ed Moore	Krohn-Rhodes
Manuel Blum	*McNaughton-Papert*

Harvard DEAP '63 — '68

...influenced Mike's future work:

- complexity of multiplication

-- improved "overlap" argument, following Cook/Aanderaa, gave O(*n* log *n*) lower bound for online multiplication (w/ Paterson/Meyer, SIAM-AMS Proc. 1974)

Harvard DEAP '63 — '68

-- string-matching
(w/ Wagner, JACM 1974;
w/Paterson, SIAM-AMS Proc., 1974)

-- converting off-line to online losing only a log *n* factor
(w/Stockmeyer, STOC 1973), leading to

-- fast parallel prefix circuits
(w/Ladner, JACM 1980)

Harvard DEAP '63 — '68

- translational argument, following P.Fischer/Ruby & Cook, for nondeterministic time hierarchy (w/Meyer, Seiferas, JACM 1978)
- bounds on formula size
  (w/ Meyer, Paterson, Vilfan)

Mike & me:

- Meyer to Carnegie Tech in '67
- Fischer joins at CMU in '68 (just missing Floyd)
- both to MIT Math/CS '69-75

Fischer-Meyer at MIT

Influential contributions:

- fast transitive closure
- economy of description
- λ-calculus schemata – CPS transform

(Only the last ever got refereed – 20 years later – but 100's of citations)

Fischer-Meyer at MIT

Less influential:

- priority arguments in complexity (w/Lynch, TransAMS, 1976) – "*sets that don't help*"
  -- but connected Nancy & Mike
- $\Omega(n \log n)$ bounds on formula size (w/Paterson, SICOMP, 1982)

Fischer-Meyer at MIT

Less influential:

- *failed* mutual exclusion (w/Rivest, Pratt)
  -- but insightful, fixed later

Fischer-Meyer at MIT

- Think-a-dot

February 6, 1974

MIT 6.045: Intro to Theory of Computation

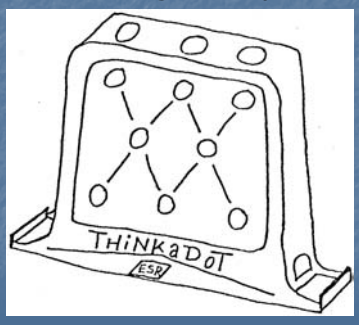
A. Meyer notes by M. Fischer

1. THINK-A-DOT

The mini-theory of the Think-a-Dot game which we develop in this section will serve as a paradigm of the kinds of questions we will investigate and the style of mathematics we will use in our studies of abstract machines.

Fischer-Meyer at MIT

- Think-a-dot – original implementation



Fischer-Meyer at MIT

Lasting important contributions:

- exponential complexity of decidable theories (w/ lots of students & Rabin)
- Dynamic Logic (w/ Pratt, later Ladner at UW)
- and of course …

Fischer-Meyer: PhD. students

Peter Bloniarz	Jean Ferrante
Mike Hammer	D.S. Johnson
Nancy Lynch	Ed McCreight
Robbie Moll	Nick Pippenger
Charles Rackoff	Joel Seiferas
Larry Stockmeyer	Bostjan Vilfan
Mitch Wand	Frances Yao

Open Problem

The complexity of multiplication was a significant focus of Mike's work.

Can integer multiplication be computed in linear-time?

Final Perspective

Theoretical Computer Science was a new field without most of today's distinct subdisciplines. With a small community of colleagues and now-prominent students, Mike made seminal contributions in this period to

* automata and formal language theory
* graph algorithms
* programming language theory
* computational complexity
* logic of programs

as well as working on • compilers, • graphics, • parsing, and • process synchronization

Final Perspective

We worked as a team -- with Paterson and our students -- for ***fun*** and ***romance***.

We thought our ideas would be valuable, but had no clue that a theoretician could be an entrepreneur.

We captivated students with our enthusiasm and engaging problems .

Some of what we did dead-ended – rightly so – but much remains the influential basis for current research.

Final Perspective

And, by the way, we did it while raising infants, maintaining large gardens, and (Mike & Alice) learning to figure skate.

Drumlin Farm, Concord MA, Fall 1970

Final Perspective

Is this perspective helpful in the context of today's richly developed field of computation theory?

I leave that for you to consider.