Ken’s Graduate Career

David L. Dill

Donald E. Knuth Professor in the School of Engineering, Emeritus

Stanford University

Context: CMU 1983 - 1992

- 1983: Prof. Edmund M. Clarke moved to Carnegie Mellon from Harvard shortly after co-inventing CTL model checking
- Ed started building a research group (I was 2nd PhD student at CMU)
- First model checking program written in LISP, struggled to examine a few thousand states.
- First applications were asynchronous circuits

Also: Ed co-founded CAV in 1989 – there was no dedicated research forum for model-checking before then.

Context: CMU 1983 - 1992

- Meanwhile, Prof. Randy Bryant invented “Binary Decision Diagrams” (BDDs) 1986
- Initially, for analyzing combinational circuits
- Main method for analyzing propositional logic until about 1996
  - Outperformed Davis-Putnam-Logemann-Loveland (DPLL) SAT methods until Chaff

1987 – Ken arrives at CMU

- I finished my PhD at CMU, joined Stanford in September 1987
- Ken started CMU PhD right after that.
  Ed Clarke: “I have this new PhD student who’s pretty good.”
- Ken developed symbolic model checking with BDDs within a few months (published significantly later).
- I was impressed since my first published paper was 4 years into my PhD program!

Ken and Symbolic Model Checking

Primarily responsible for the early success of symbolic model checking

- “10^20 states paper” (1990) using BDDs. First truly general treatment. Four co-authors, but Ken paper was founded on Ken’s work.
- SMV symbolic model checker – first practical BDD-based model checker. Widely used (no later than 1990)
- Demonstrated success on important practical examples (e.g., Gigamax cache coherence protocol, 1990).
- This combination caused people to take the approach seriously
  - Used in large hardware companies, commercial verification tools.

Symbolic Model Checking: 10^{20} States and Beyond*

J. R. BURCH, E. M. CLARKE, AND K. L. MCMILLAN

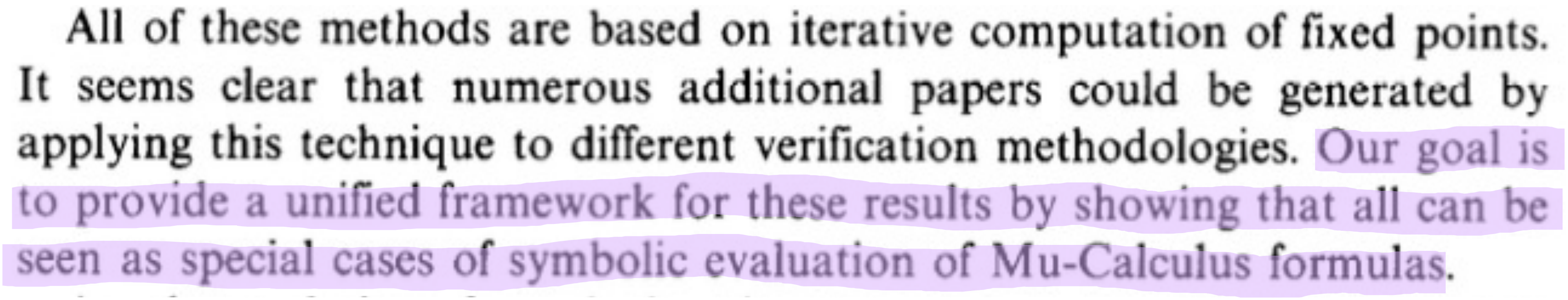
School of Computer Science, Carnegie Mellon University,
Pittsburgh, Pennsylvania 15213

AND

D. L. DILL AND L. J. HWANG

Stanford University, Stanford, California 94305

10^20 states paper goal



All of these methods are based on iterative computation of fixed points. It seems clear that numerous additional papers could be generated by applying this technique to different verification methodologies. Our goal is to provide a unified framework for these results by showing that all can be seen as special cases of symbolic evaluation of Mu-Calculus formulas.

More detail: 10^20 states paper

- Write computation as propositional mu-calculus formula (subsumes CTL)
  - Mu-calculus is propositional logic logic with nested least and greatest fixed points.

$$R = \nu P[V_0 \wedge \mu Q[\lambda y[\exists x[(P(x) \vee Q(x)) \wedge N(y, x)]]]]$$

- Implement with iterative fixed points with BDDs

Problems addressed in 10^20 states paper

- CTL model checking with fairness constraints
- Propositional Linear Temporal Logic (symbolic tableau procedure)
- Observational equivalence (strong & weak)
- Buchi automata language inclusion (when the larger language is represented by a deterministic Buchi automaton)

SMV model checking tool

- A major part of Ken’s early impact was showing that symbolic model checking could work in the real world.
- SMV BDD-based model checker
    - First generally usable tool for symbolic model checking
    - Flexible description language
    - Optimizations for efficiency
    - Well documented
    - Freely available for others to use

Gigamax cache protocol

- At that time, shared memory multiprocessors were still new
- Cache coherence in distributed systems was notoriously buggy
- Ken and J. Schwalbe of Encore Computer Corporation verified the Gigamax protocol using symbolic model checking
- Found several subtle bugs
- Published the paper in a computer architecture conference

Partial order reduction

Historical Context: (1989 – 1991)

Enumerating the states of a concurrent system seems wasteful, because it many states result from very similar interleavings of steps in concurrent processes.

“Partial order reduction” – reduce number of states searched by reducing this redundancy.

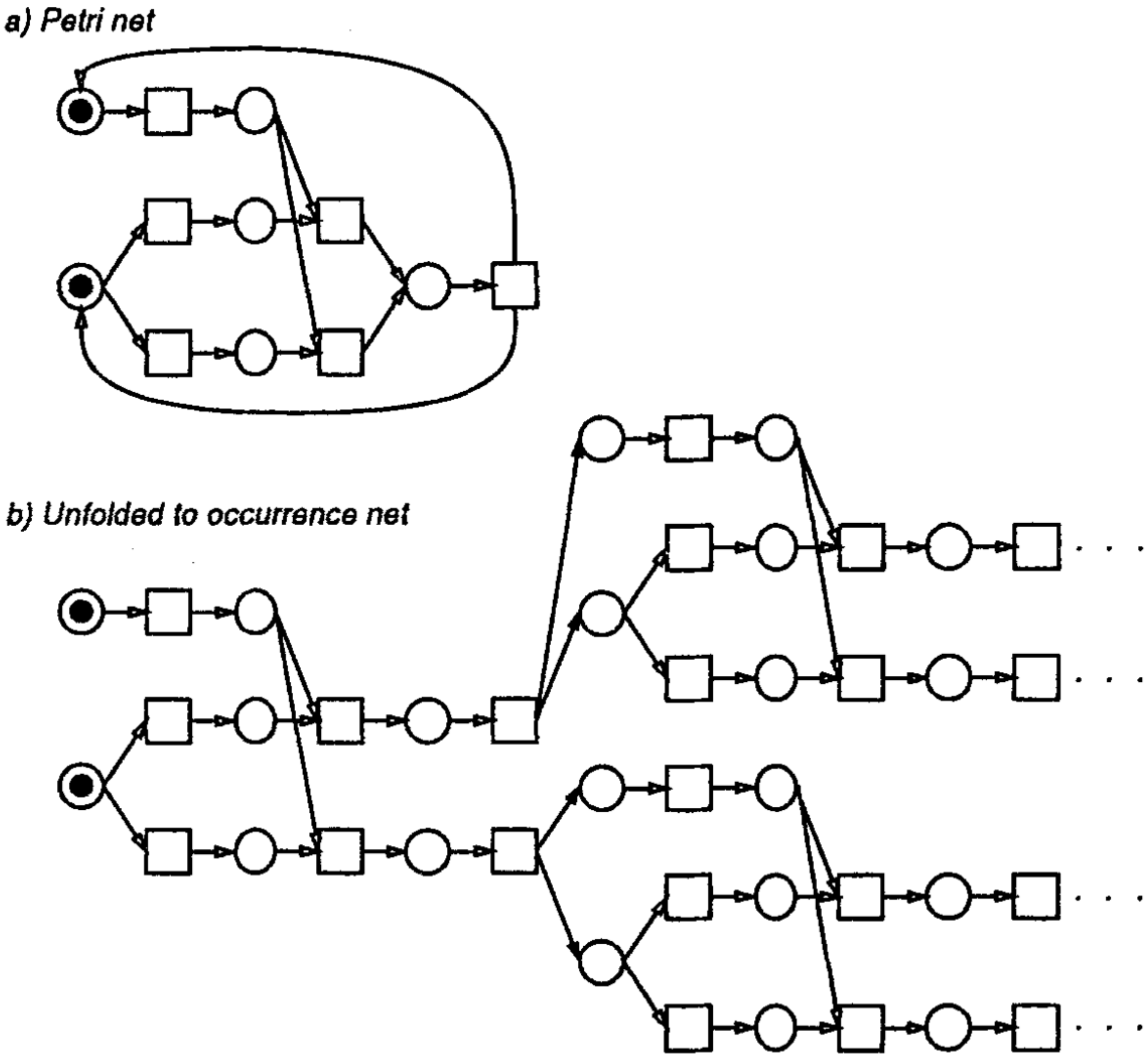
Ken: Petri net unfoldings

Petri nets are generalized finite automata that capture concurrency in their structure.

In 1992, Ken published a startlingly original approach to partial order reduction

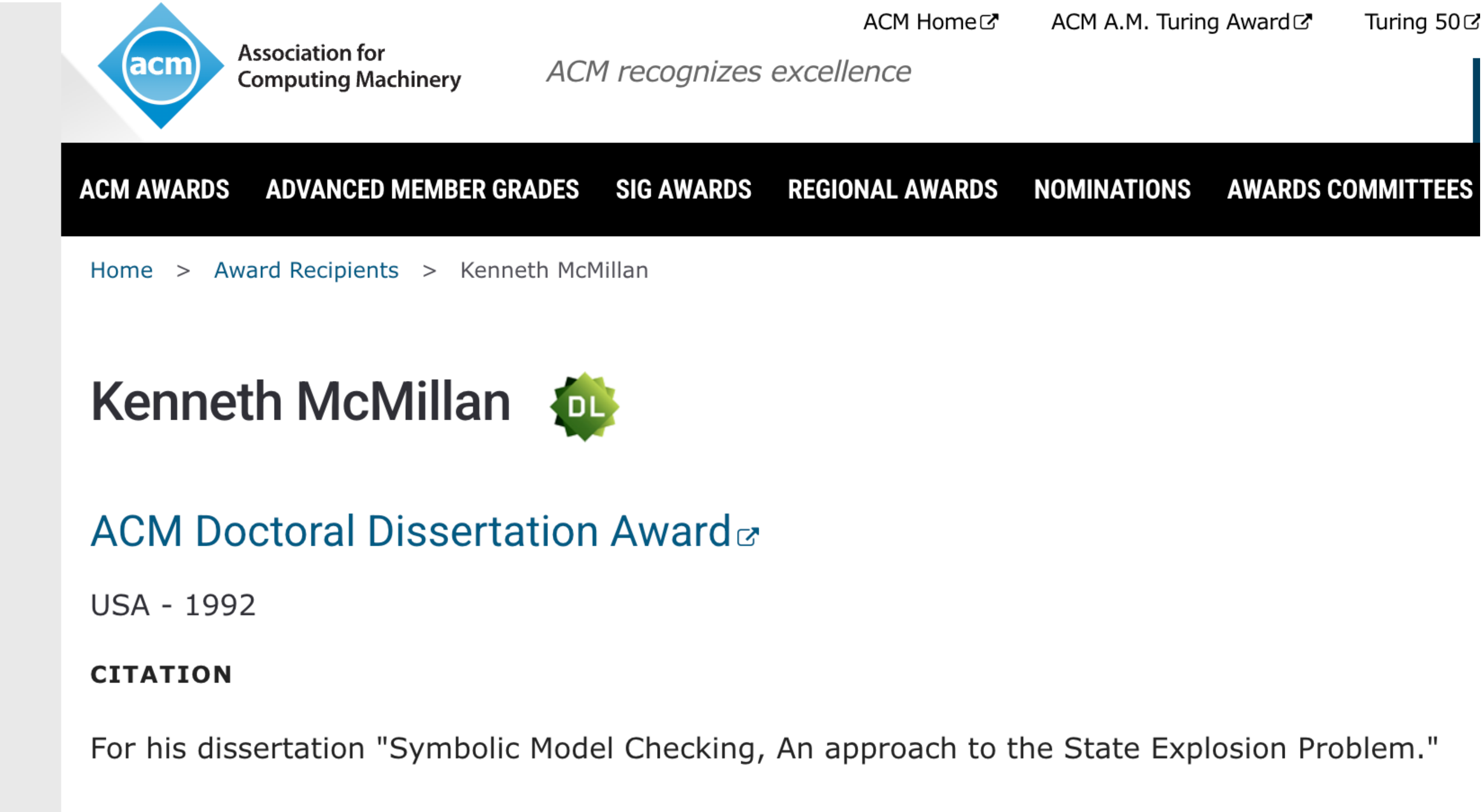
- Unroll Petri net in a certain way
- Stop when all markings (states) are represented
- Solves reachability (“coverability”) and deadlock

This started a whole new branch of research



a) Petri net

b) Unfolded to occurrence net



Kenneth McMillan

ACM Doctoral Dissertation Award

USA - 1992

CITATION

For his dissertation "Symbolic Model Checking, An approach to the State Explosion Problem."

Summary

- Ken emerged as one of the leading researchers in formal verification while still a junior PhD student
- Research during that time created the field of symbolic model checking
  - Theory + practice made it credible
  - We use different representations now (e.g., SAT-based), but the basic idea of symbolic model checking thrives
- There were other major contributions (unfoldings, others)
- Later work continued the trend