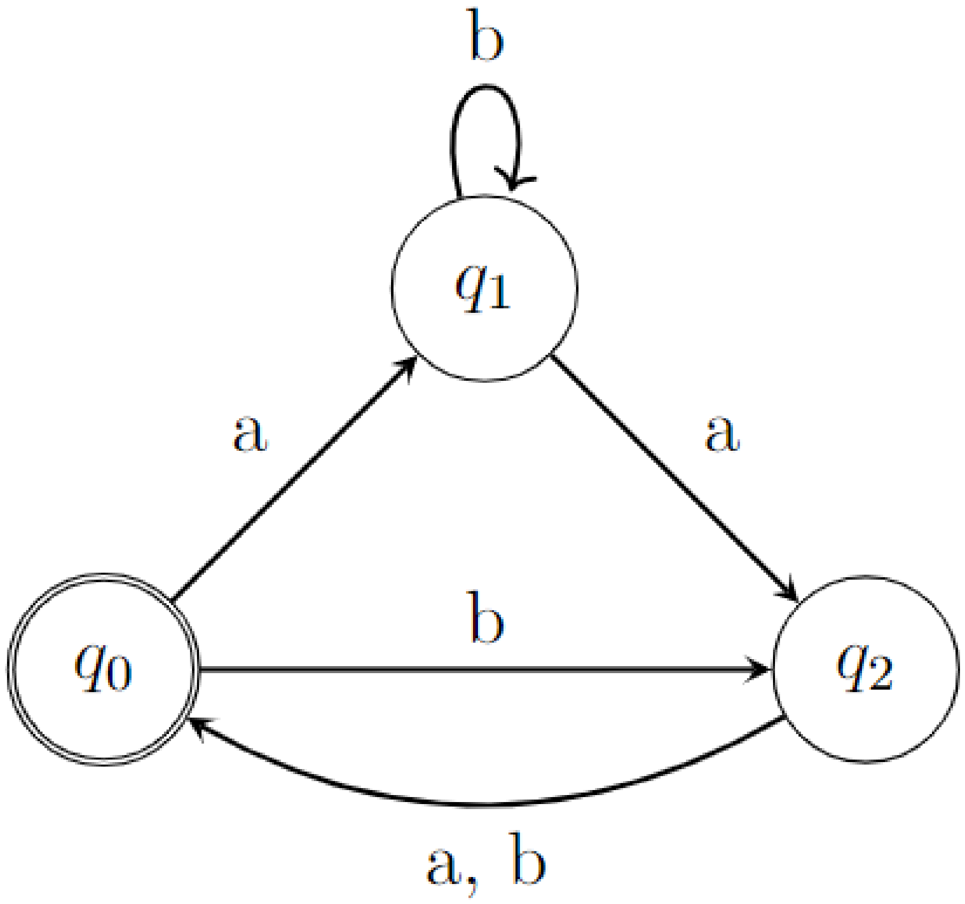
k-entry DFA

Šimon Huraj

k-entry DFA



k-entry DFA

k-entry deterministic finite-state automaton ($_kDFA$) is a quintuple $M = (Q, \Sigma, I, F, \delta)$ where

- Q is a finite set of states
- Σ is a finite set of input symbols
- **I is a set of initial states, $I \subseteq Q$**
- σ is a transition function, $\sigma : Q \times \Sigma \to Q$
- F is a set of final states, $F \subseteq Q$

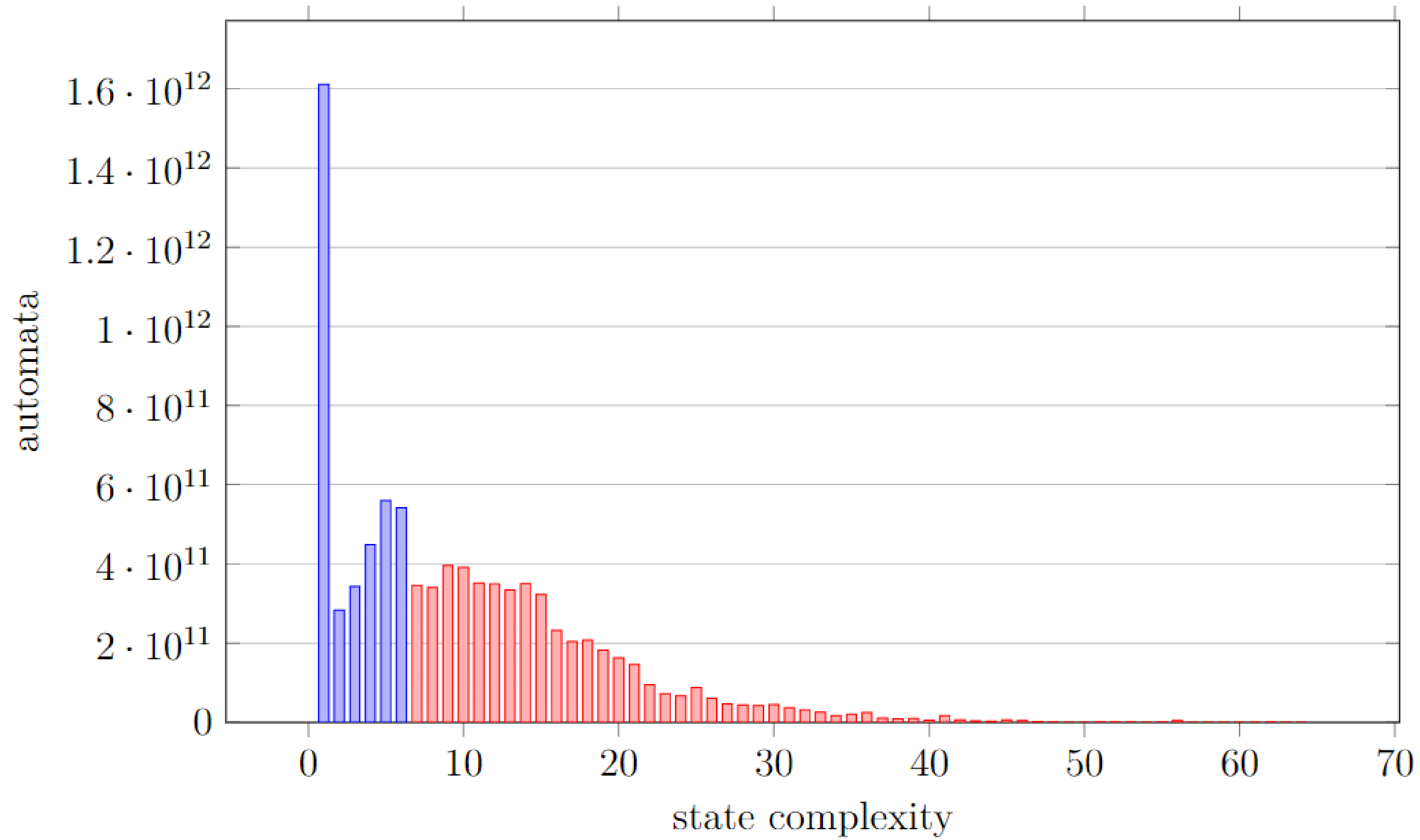
Goals

1. Develop a program that accepts an automaton as input and generates all automata with various choices of initial states. Expand this program to generate all n-state automata. Additionally, create a program capable of determinizing and minimizing the automaton to ascertain the state complexity of the language it represents. Furthermore, ensure that the program is designed to leverage parallel computing.
2. Investigate the deterministic state complexity of automata represented by nondeterministic automata, where the only nondeterminism is from a choice of initial states.
3. Examine the worst-case state complexity identified in 2.
4. Explore the range of all obtainable state complexities from 2.
5. Study the average state complexity from 2.

Program

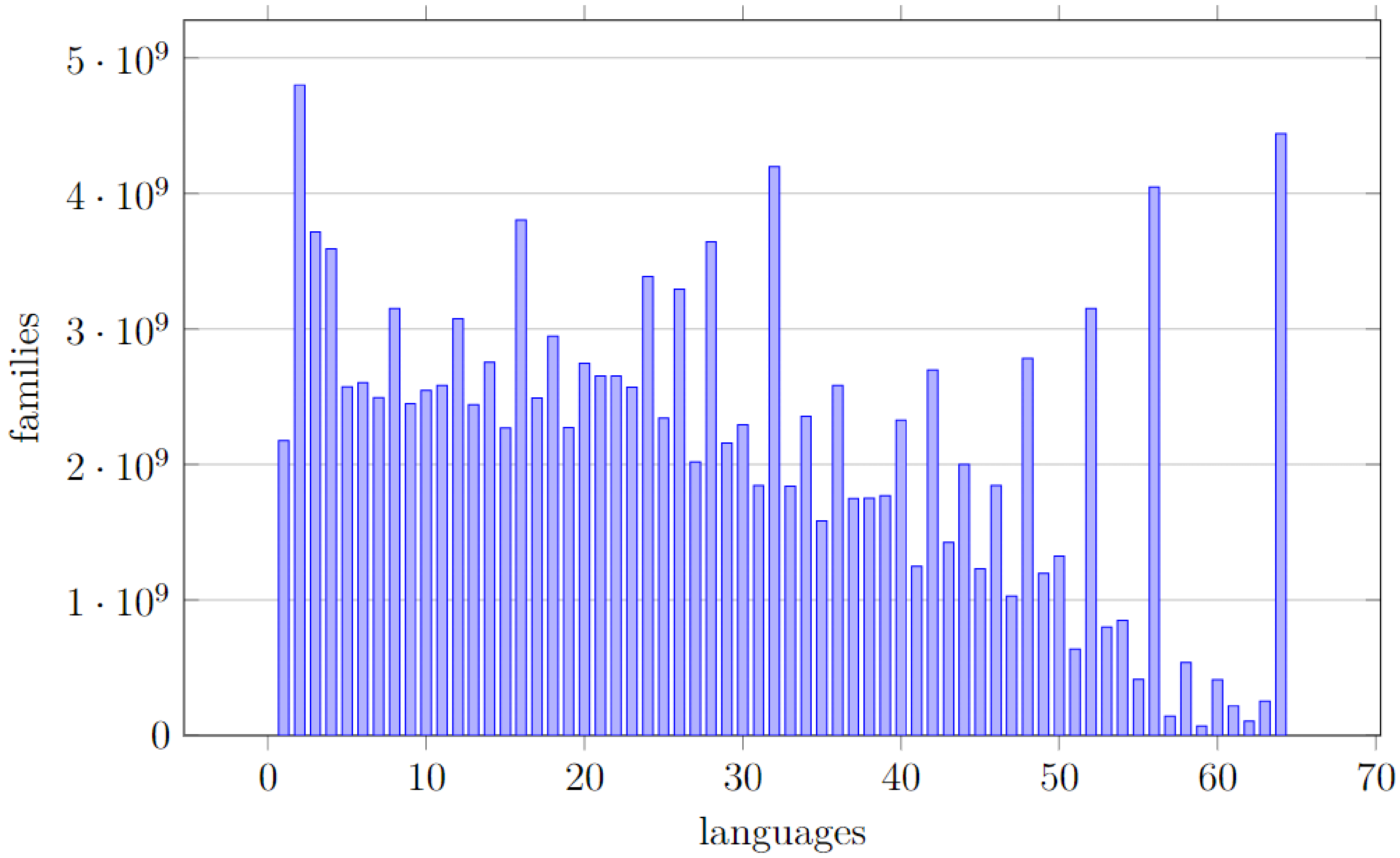
```
const uint8_t threadsCount = 8;
std::vector<std::thread> threads( n: threadsCount);
for (uint8_t i = 0; i < threadsCount - 1; i++) {
    threads[i] = thread( &: generateAutomata, i, chunkSize);
}
threads[threadsCount - 1] = thread( &: generateAutomata, threadsCount, rest);

for (uint8_t i = 0; i < threadsCount; i++) {
    threads[i].join();
}
```

Generation



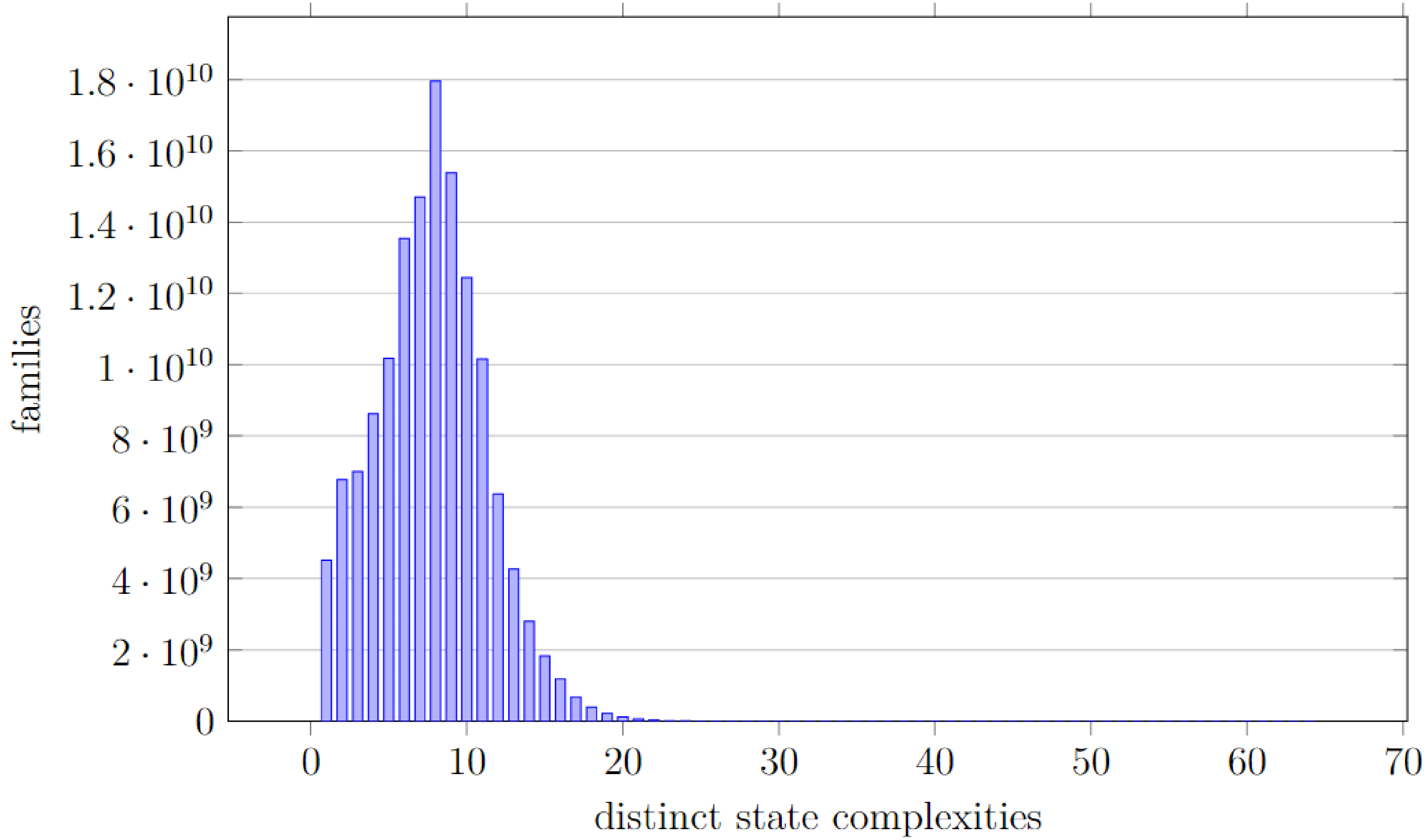
automata
$1.6 \cdot 10^{12}$
$1.4 \cdot 10^{12}$
$1.2 \cdot 10^{12}$
$1 \cdot 10^{12}$
$8 \cdot 10^{11}$
$6 \cdot 10^{11}$
$4 \cdot 10^{11}$
$2 \cdot 10^{11}$
0
0
10
20
30
40
50
60
70
state complexity

Generation



families
languages
0
$1 \cdot 10^9$
$2 \cdot 10^9$
$3 \cdot 10^9$
$4 \cdot 10^9$
$5 \cdot 10^9$
0
10
20
30
40
50
60
70

Generation



families
$1.8 \cdot 10^{10}$
$1.6 \cdot 10^{10}$
$1.4 \cdot 10^{10}$
$1.2 \cdot 10^{10}$
$1 \cdot 10^{10}$
$8 \cdot 10^{9}$
$6 \cdot 10^{9}$
$4 \cdot 10^{9}$
$2 \cdot 10^{9}$
0
0
10
20
30
40
50
60
70
distinct state complexities

Average State Complexity

Theorem 3.2 *Let $n \in \mathbb{N}$, $n \geq 1$, then average state complexity of a language represented by an n-state family is at most $5/8 \times 2^n$.*

Average State Complexity

$$\frac{\sum_{i=1}^{n}\sum_{j=1}^{i}\binom{n}{i}\binom{n}{j}}{2^n - 1} \leq \frac{5}{8}2^n$$

$$S = \sum_{i=1}^{n}\sum_{j=1}^{i}\binom{n}{i}\binom{n}{j} = \sum_{1\leq j\leq i\leq n}\binom{n}{i}\binom{n}{j} = \sum_{1\leq i\leq j\leq n}\binom{n}{i}\binom{n}{j}$$

Average State Complexity

$$2S = \sum_{1\le i\le j\le n} \binom{n}{i}\binom{n}{j} + \sum_{1\le j\le i\le n} \binom{n}{i}\binom{n}{j} =$$

$$= \sum_{1\le i\le j\le n} \binom{n}{i}\binom{n}{j} + \sum_{1\le j<i\le n} \binom{n}{i}\binom{n}{j} + \sum_{1\le i=j\le n} \binom{n}{i}\binom{n}{j} =$$

$$= \sum_{1\le i,j\le n} \binom{n}{i}\binom{n}{j} + \sum_{i=1}^{n} \binom{n}{i}^2 =$$

$$= (2^n - 1)^2 + \binom{2n}{n} - 1.$$

$$S = \frac{(2^n - 1)^2 + \binom{2n}{n} - 1}{2}.$$

Average State Complexity

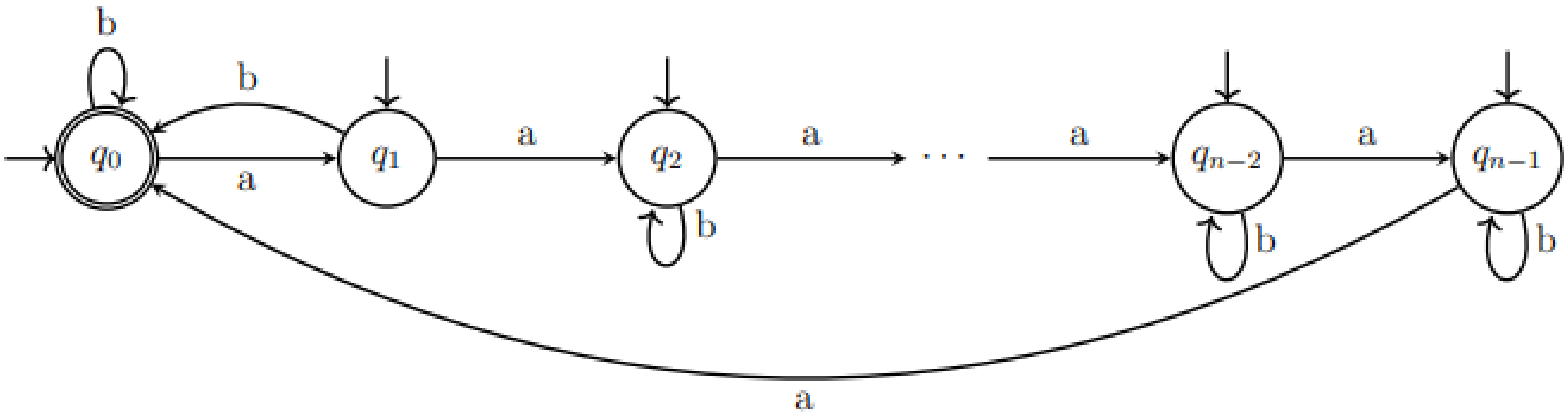
$$\frac{(2^n - 1)^2 + \binom{2n}{n} - 1}{2(2^n - 1)} \leq \frac{5}{8} 2^n$$

$$2^{2n} - 2 \cdot 2^n + 1 + \binom{2n}{n} - 1 \leq \frac{5}{4} 2^n (2^n - 1)$$

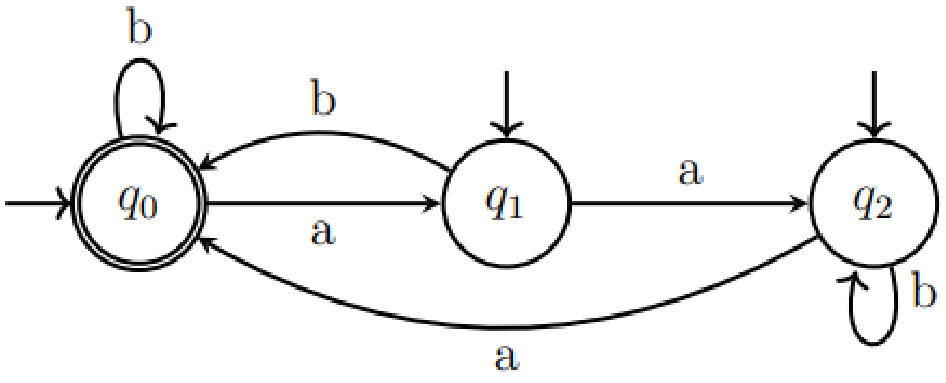
$$\binom{2n}{n} \leq \frac{4^n}{4} + \frac{3 \cdot 2^n}{4}.$$

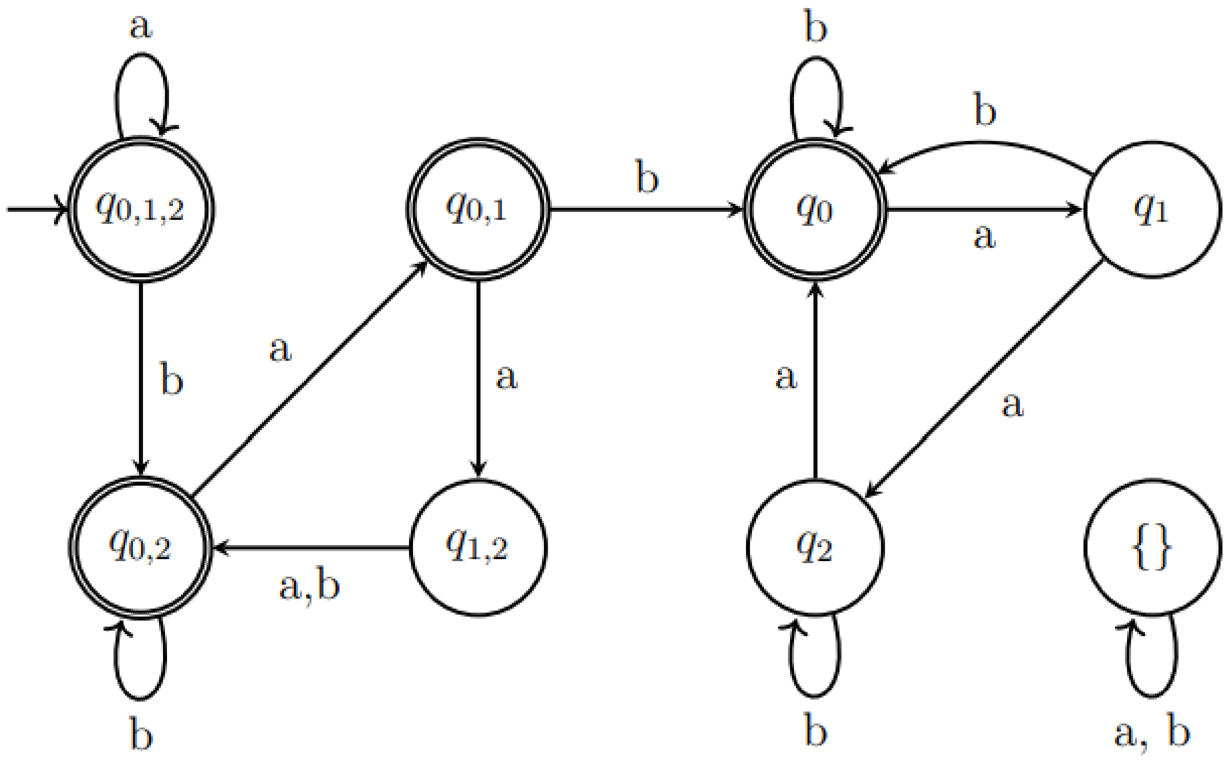
Worst-case state complexity

Lemma 2.1 *For every $n \in \mathbf{N}$ there exists a n-state ${}_k$DFA such that equivalent minimal DFA has exactly $2^n - 1$ states.*

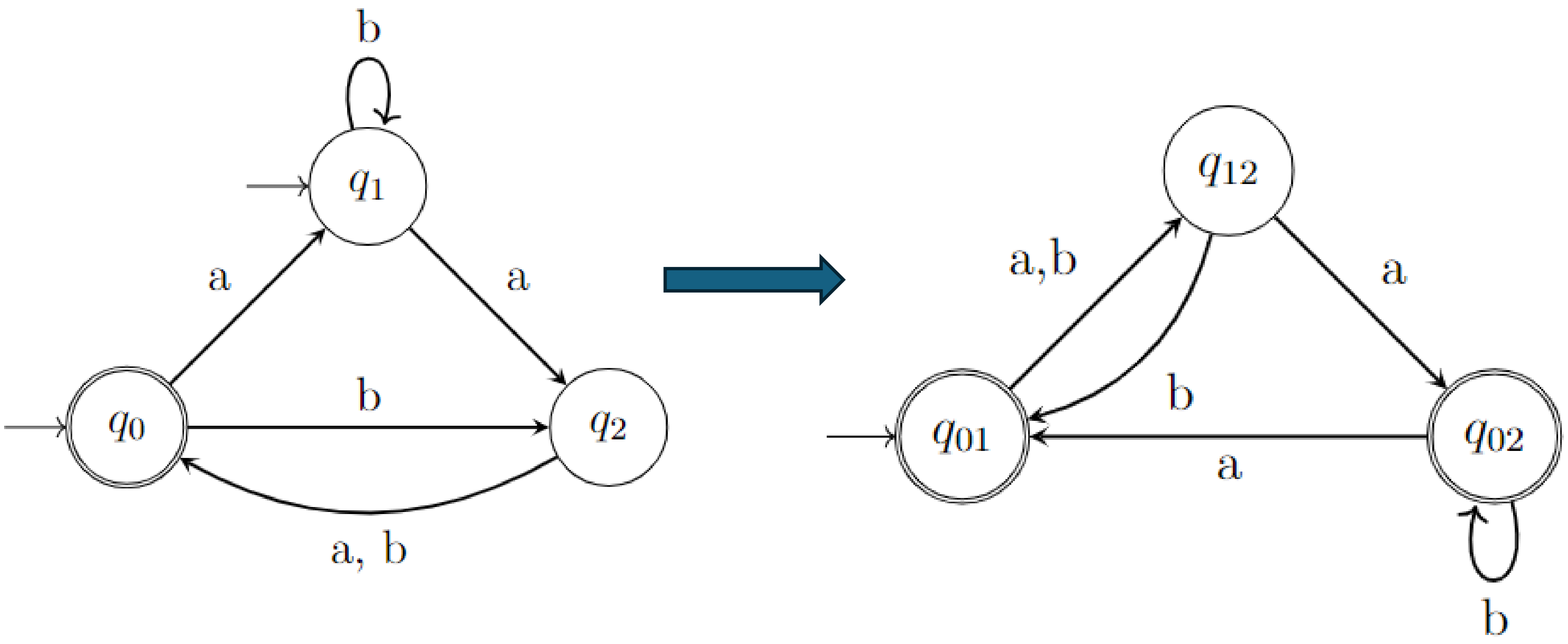


Worst-case state complexity

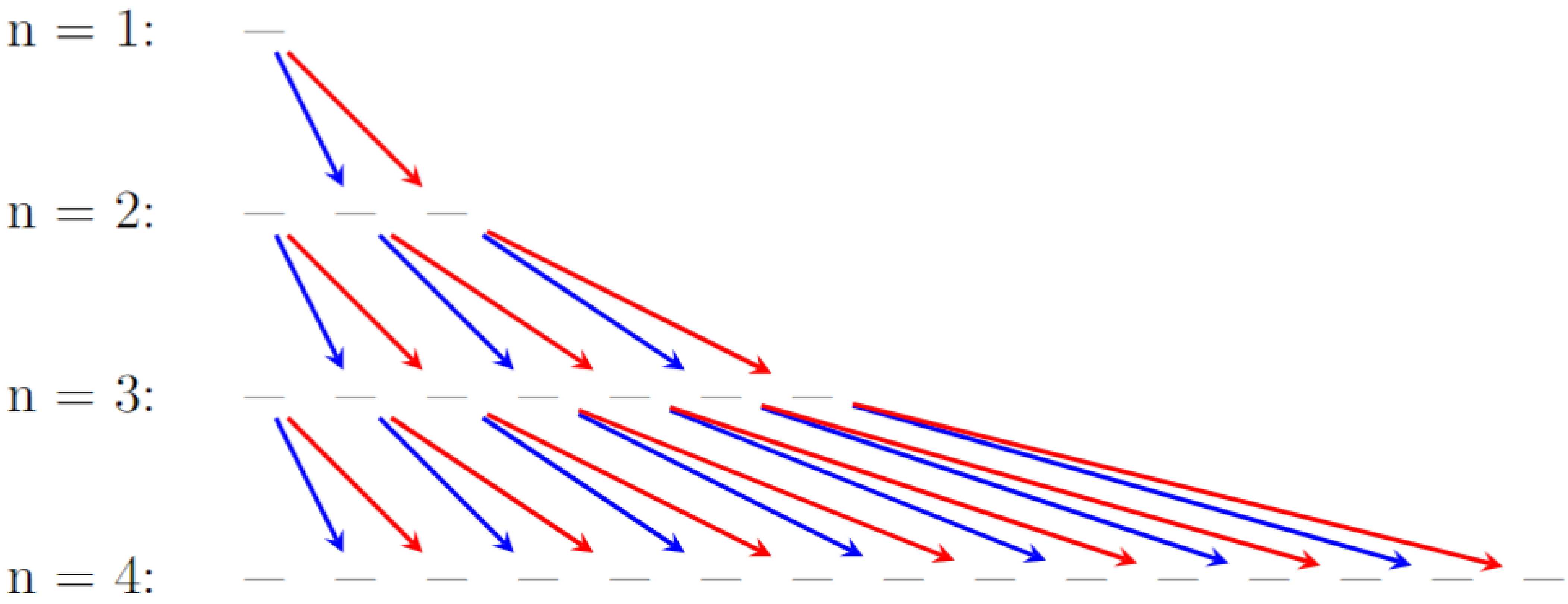




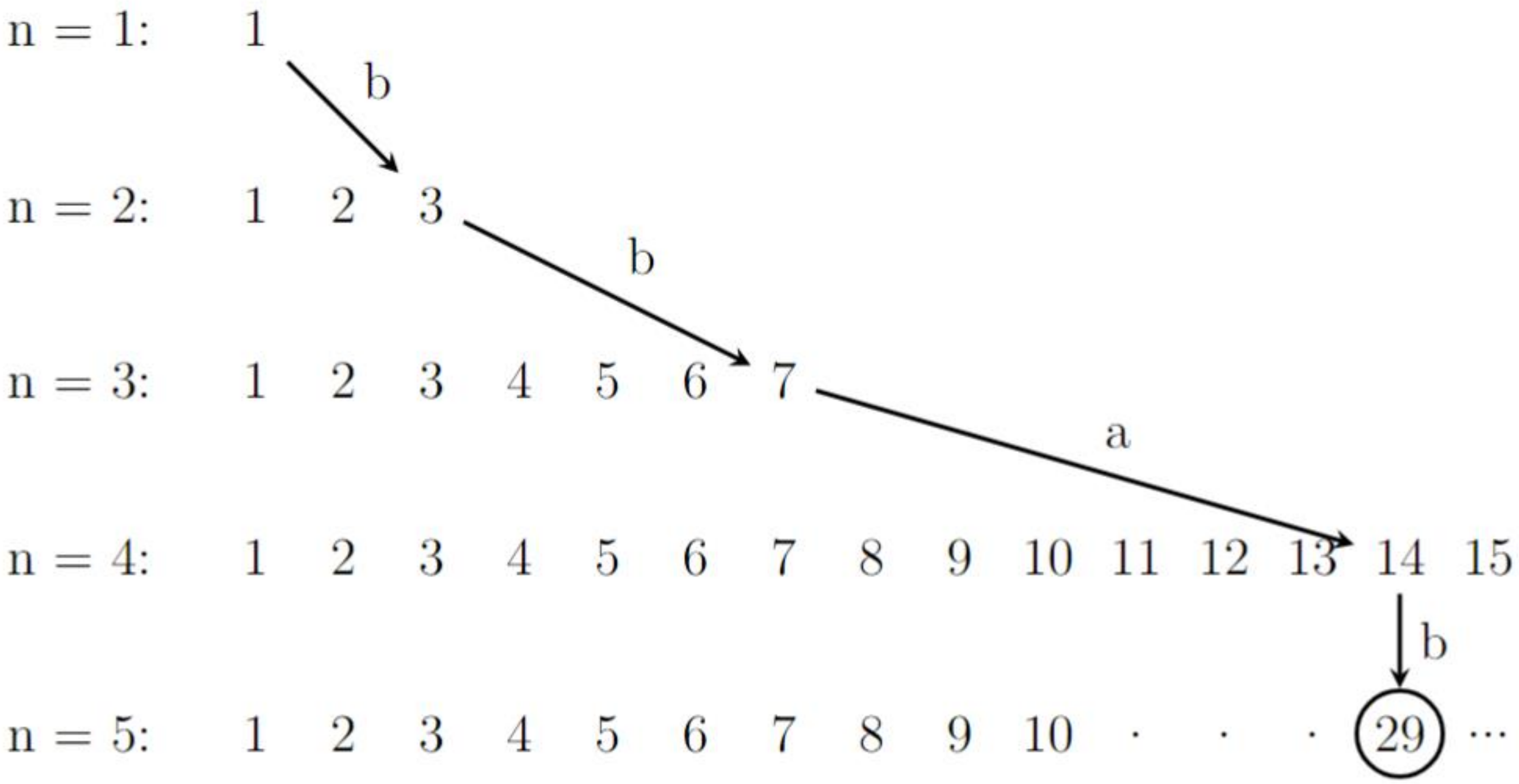
Magic numbers



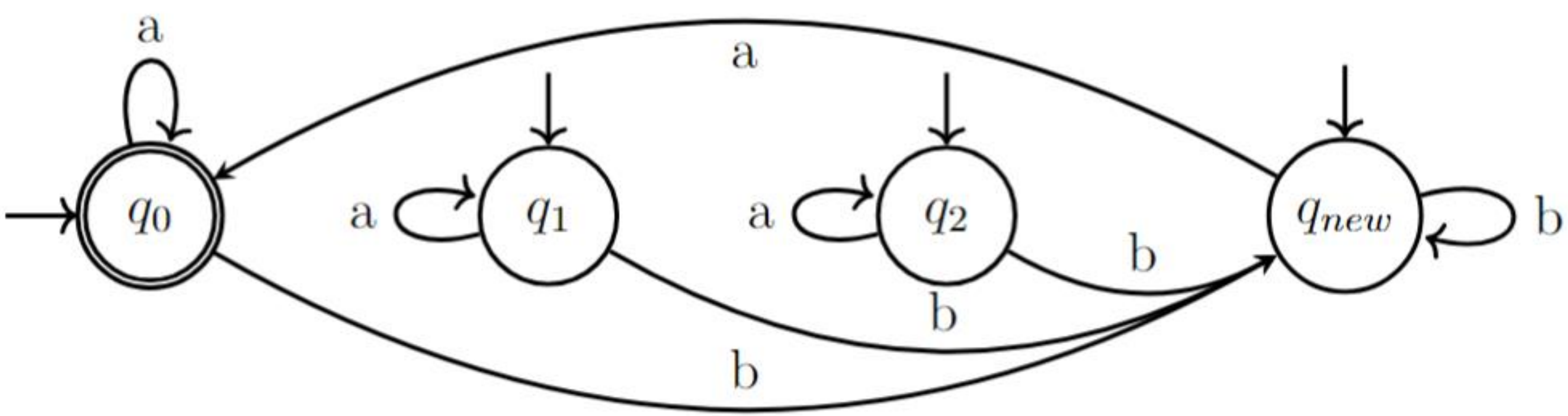
Magic numbers



Magic numbers



Magic numbers



Thank you!