Johann Wolfgang Goethe-Universität
Frankfurt am Main

Institut für Informatik
Fachbereich Biologie und Informatik

# Minimizing Finite Automata is Computationally Hard

Andreas Malcher

Nr. 6/02

Frankfurter Informatik-Berichte

Institut für Informatik • Robert-Mayer-Straße 11–15 • 60054 Frankfurt am Main

# Interne Berichte am Fachbereich Informatik
# Johann Wolfgang Goethe-Universität Frankfurt

**1/1987** Risse, Thomas:
On the number of multiplications needed to evaluate the reliability of k-out-of-n systems

**2/1987** Roll, Georg [u.a.]:
Ein Assoziativprozessor auf der Basis eines modularen vollparallelen Assoziativspeicherfeldes

**3/1987** Waldschmidt, Klaus ; Roll, Georg:
Entwicklung von modularen Betriebssystemkernen für das ASSKO-Multi-Mikroprozessorsystem

**4/1987** Workshop über Komplexitätstheorie, effiziente Algorithmen und Datenstrukturen:
3.2.1987, Universität Frankfurt/Main

**5/1987** Seidl, Helmut:
Parameter-reduction of higher level grammars

**6/1987** Kemp, Rainer:
On systems of additive weights of trees

**7/1987** Kemp, Rainer:
Further results on leftist trees

**8/1987** Seidl, Helmut:
The construction of minimal models

**9/1987** Weber, Andreas ; Seidl, Helmut:
On finitely generated monoids of matrices with entries in N

**10/1987** Seidl, Helmut:
Ambiguity for finite tree automata

**1/1988** Weber, Andreas:
A decomposition theorem for finite-valued transducers and an application to the equivalence problem

**2/1988** Roth, Peter:
A note on word chains and regular languages

**3/1988** Kemp, Rainer:
Binary search trees for d-dimensional keys

**4/1988** Dal Cin, Mario:
On explicit fault-tolerant, parallel programming

**5/1988** Mayr, Ernst W.:
Parallel approximation algorithms

**6/1988** Mayr, Ernst W.:
Membership in polynomial ideals over Q is expotential space complete

**1/1989** Lutz, Joachim [u.a.]:
Parallelisierungskonzepte für ATTEMPO-2

**2/1989** Lutz, Joachim [u.a.]:
Die Erweiterung der ATTEMPO-2 Laufzeitbibliothek

**3/1989** Kemp, Rainer:
A One-to-one Correspondence between Two Classes of Ordered Trees

**4/1989** Mayr, Ernst W. ; Plaxton, C. Greg:
Pipelined Parallel Prefix Computations, and Sorting on a Pipelined Hypercube

**5/1989** Brause, Rüdiger:
Performance and Storage Requirements of Topology-conserving Maps for Robot Manipulator Control

**6/1989** Roth, Peter:
Every Binary Pattern of Length Six is Avoidable on the Two-Letter Alphabet

**7/1989** Mayr, Ernst W.:
Basic Parallel Algorithms in Graph Theory

**8/1989** Brauer, Johannes:
A Memory Device for Sorting

**1/1990** Vollmer, Heribert:
Subpolynomial Degrees in P and Minimal Pairs for L

**2/1990** Lenz, Katja:
The Complexity of Boolean Functions in Bound Depth Circuits over Basis $\{\wedge, \oplus\}$

**3/1990** Becker, Bernd ; Hahn R. ; Krieger, R. ; Sparmann, U.:
Structure Based Methods for Parallel Pattern Fault Simulation in Combinational Circuits

**4/1990** Goldstine, J. ; Kintala, C.M.R. ; Wotschke D.:
On Measuring Nondeterminism in Regular Languages

**5/1990** Goldstein, J. ; Leung, H. ; Wotschke, D.:
On the Relation between Ambiguity and Nondeterminism in Finite Automata

**1/1991** Brause, Rüdiger:
Approximator Networks and the Principles of Optimal Information Distribution

**2/1991** Brauer, Johannes ; Stuchly, Jürgen:
HyperEDIF: Ein Hypertext-System für VLSI Entwurfsdaten

**3/1991** Brauer, Johannes:
Repräsentation von Entwurfsdaten als symbolische Ausdrücke

**4/1991** Trier, Uwe:
Additive Weights of a Special Class of Nonuniformly Distributed Backtrack Trees

**5/1991** Dömel, P. [u.a.]:
Concepts for the Reuse of Communication Software

**6/1991** Heistermann, Jochen:
Zur Theorie genetischer Algorithmen

**7/1991** Wang, Alexander [u.a.]:
Embedding complete binary trees in faulty hypercubes

**1/1992** Brause, Rüdiger:
The Minimum Entropy Network

**2/1992** Trier, Uwe:
Additive Weights Under the Balanced Probability Model

**3/1992** Trier, Uwe:
(Un)expected path lengths of asymetric binary search trees

**4/1992** Coen Alberto ; Lavazza, Luigi ; Zicari, Roberto:
Assuring type-safety of object oriented languages

**5/1992** Coen, Alberto ; Lavazza, Luigi ; Zicari, Roberto:
Static type checking of an object-oriented database schema

**6/1992** Coen, Alberto ; Lavazza, Luigi ; Zicari, Roberto:
Overview and progress report of the ESSE project : Supporting object-oriented database schema analysis and evolution

**7/1992** Schmidt-Schauß, Manfred:
Some results for unification in distributive equational theories

**8/1992** Mayr, Ernst W. ; Werchner, Ralph:
Divide-and-conquer algorithms on the hypercube

**1/1993** Becker, Bernd ; Drechsler, Rolf ; Hengster, Harry:
Local circuit transformations preserving robust path-delay-fault testability

**2/1993** Krieger, Rolf ; Becker, Bernd ; Sinković, Robert:
A BDD-based algorithmen for computation of exact fault detection probabilities

**3/1993** Mayr, Ernst W. ; Werchner, Ralph:
Optimal routing of parentheses on the hypercube

**4/1993** Drechsler, Rolf ; Becker, Bernd:
Rapid prototyping of fully testable multi-level AND/EXOR networks

**5/1993** Becker, Bernd ; Drechsler, Rolf:
On the computational power of functional decision diagrams

**6/1993** Berghoff, P. ; Dömel, P. ; Drobnik, O. [u.a.]:
Development and management of communication software systems

**7/1993** Krieger, Rolf ; Hahn, Ralf ; Becker Bernd:
test_circ : Ein abstrakter Datentyp zur Repräsentation von hierarchischen Schaltkreisen (Benutzeranleitung)

**8/1993** Krieger, Rolf ; Becker, Bernd ; Hengster, Harry:
lgc++ : Ein Werkzeug zur Implementierung von Logiken als abstrakte Datentypen in C++ (Benutzeranleitung)

**9/1993** Becker, Bernd ; Drechsler, Rolf ; Meinel, Christoph:
On the testability of circuits derived from binary decision diagrams

**10/1993** Liu, Ling ; Zicari, Roberto ; Liebherr, Karl ; Hürsch, Walter:
Polymorphic reuse mechanism for object-oriented database specifications

**11/1993** Ferrandina, Fabrizio ; Zicari, Roberto:
Object-oriented database schema evolution: are lazy updates always equivalent to immediate updates ?

**12/1993** Becker, Bernd ; Drechsler, Rolf ; Werchner, Ralph:
On the Relation Between BDDs and FDDs

**13/1993** Becker, Bernd ; Drechsler, Rolf:
Testability of circuits derived from functional decision diagrams

**14/1993** Drechsler, R. ; Sarabi, A. ; Theobald, M. ; Becker, B. ; Perkowski, M.A.:
Efficient repersentation and manipulation of switching functions based on ordered Kronecker functional decision diagrams

**15/1993** Drechsler, Rolf ; Theobald, Michael ; Becker, Bernd:
Fast FDD based Minimization of Generalized Reed-Muller Forms

**1/1994** Ferrandina, Fabrizio ; Meyer, Thorsten ; Zicari, Roberto:
Implementing lazy database updates for an object database system

**2/1994** Liu, Ling ; Zicari, Roberto ; Hürsch, Walter ; Liebherr, Karl:
The Role of Polymorhic Reuse mechanism in Schema Evolution in an Object-oriented Database System

**3/1994** Becker, Bernd ; Drechsler, Rolf ; Theobald, Michael:
Minimization of 2-level AND/XOR Expressions using Ordered Kronecker Functional Decision Diagrams

**4/1994** Drechsler, R. ; Becker, B. ; Theobald, M. ; Sarabi, A. ; Perkowski, M.A.:
On the computational power of Ordered Kronecker Functional Decision Diagrams

**5/1994** Even, Susan ; Sakkinen, Marku:
The safe use of polymorphism in the O2C database language

**6/1994** GI/ITG-Workshop:
Anwendungen formaler Methoden im Systementwurf : 21. und 22. März 1994

**7/1994** Zimmermann, M. ; Mönch, Ch. [u.a.]:
Die Telematik-Klassenbibliothek zur Programmierung verteilter Anwendungen in C++

**8/1994** Zimmermann, M. ; Krause, G.:
Eine konstruktive Beschreibungsmethodik für verteilte Anwendungen

**9/1994** Becker, Bernd ; Drechsler, Rolf:
How many Decomposition Types do we need ?

**10/1994** Becker, Bernd ; Drechsler, Rolf:
Sympathy: Fast Exact Minimization of Fixed Polarity Reed-Muller Expression for Symmetric Functions

**11/1994** Drechsler, Rolf ; Becker, Bernd ; Jahnke, Andrea:
On Variable Ordering and Decompostion Type Choice in OKFDDs

**12/1994** Schmidt-Schauß:
Unification of Stratified Second-Order Terms

**13/1994** Schmidt-Schauß:
An Algorithmen for Distributive Unification

**14/1994** Becker, Bernd ; Drechsler, Rolf:
Synthesis for Testability: Circuit Derived from ordered Kronecker Functional Decision Diagrams

**15/1994** Bär, Brigitte:
Konformität von Objekten in offenen verteilten Systemen

**16/1994** Seidel, T. ; Puder, A. ; Geihs, K. ; Gründer, H.:
Global object space: Modell and Implementation

**17/1994** Drechsler, Rolf ; Esbensen, Henrik ; Becker, Bernd:
Genetic algorithms in computer aided design of integrated circuits

**1/1995** Schütz, Marko:
The $G^{\#}$-Machine: efficient strictness analysis in Haskell

**2/1995** Henning, Susanne ; Becker, Bernd:
GAFAP: A Linear Time Scheduling Approach for High-Level-Synthesis

**3/1995** Drechsler, Rolf ; Becker, Bernd ; Göckel, Nicole:
A Genetic Algorithm for variable Ordering of OBDDs

**4/1995** Nebel, Markus E.:
Exchange Trees, eine Klasse Binärer Suchbäume mit Worst Case Höhe von $\log(n)$

**5/1995** Drechsler, Rolf ; Becker, Bernd:
Dynamic Minimization of OKFDDs

**6/1995** Breché, Philippe ; Ferrandina, Fabrizio ; Kuklok, Martin:
Simulation of Schema and Database Modification using Views

**7/1995** Breché, Philippe ; Wörner, Martin:
Schema Update Primitives for ODB Design

**8/1995** Schmidt-Schauß, Manfred:
On the Sematics and Interpretation of Rule Based Programs with Static Global Variables

**9/1995** Rußmann, Arnd:
Adding Dynamic Actions to LL($k$) Parsers

**10/1995** Rußmann, Arnd:
Dynamic LL($k$) Parsing

**11/1995** Leyendecker, Thomas ; Oehler, Peter ; Waldschmidt, Klaus:
Spezifikation hybrider Systeme

**12/1995** Cerone, Antonio ; Maggiolo-Schettini, Andrea:
Time-based Expressivity of Times Petri Nets

**1/1996** Schütz, Marko ; Schmidt-Schauß, Manfred:
A Constructive Calculus Using Abstract Reduction for Context Analysis (nicht erschienen)

**2/1996** Schmidt-Schauß, Manfred:
CPE: A Calculus for Proving Equivalence of Expressions in a Nonstrict Functional Language

**1/1997** Kemp, Rainer:
On the Expected Number of Nodes at Level $k$ in 0-balanced Trees

**2/1997** Nebel, Markus:
New Results on the Stack Ramification of Binary Trees

**3/1997** Nebel, Markus:
On the Average Complexity of the Membership Problem for a Generalized Dyck Language

**4/1997** Liebehenschel, Jens:
Ranking and Unranking of Lexicographically Ordered Words: An Average-Case Analysis

**5/1997** Kappes, Martin:
On the Generative Capacity of Bracketed Contextual Grammars

**1/1998** Arlt, B. ; Brause, R.:
The Principal Independent Components of Images. *Elektronisch publiziert unter URL http://www.informatik.uni-frankfurt.de/fbreports/fbreport1-98.ps.gz*

**2/1998** Miltrup, Matthias ; Schnitger, Georg:
Large Deviation Results for Quadratic Forms

**3/1998** Miltrup, Matthias ; Schnitger, Georg:
Neural Networks and Efficient Associative Memory

**4/1998** Kappes, Martin:
Multi-Bracketed Contextual Grammars

**5/1998** Liebehenschel, Jens:
Lexicographical Generation of a Generalized Dyck Language

**6/1998** Kemp, Rainer:
On the Joint Distribution of the Nodes in Uniform Multidimensional Binary Trees

**7/1998** Liebehenschel, Jens:
Ranking and Unranking of a Generalized Dyck Language

**8/1998** Grimm, Christoph ; Waldschmidt, Klaus:
Hybride Datenflußgraphen

**9/1998** Kappes, Martin:
Multi-Bracketed Contextual Rewriting Grammars

**1/1999** Kemp, Rainer:
On Leftist Simply Generated Trees

**2/1999** Kemp, Rainer:
A One-to-one Correspondence Between a Class of Leftist Trees and Binary Trees

**3/1999** Kappes, Martin:
Combining Contextual Grammars and Tree Adjoining Grammars

**4/1999** Kappes, Martin:
Descriptional Complexity of Deterministic Finite Automata with Multiple Initial States

**5/1999** Nebel, Markus E.:
New Knowledge on AVL-Trees

**6/1999** Manfred Schmidt-Schauß, Marko Schütz (editors):
$13^{\text{th}}$ International Workshop on Unification

**7/1999** Brause, R.; Langsdorf, T.; Hepp, M.:
Credit Card Fraud Detection by Adaptive Neural Data Mining. *Elektronisch publiziert unter URL http://www.informatik.uni-frankfurt.de/fbreports/fbreport7-99.ps.gz*

**8/1999** Kappes, Martin:
External Multi-Bracketed Contextual Grammars

**9/1999** Priese, Claus P.:
A Flexible Type-Extensible Object-Relational DataBase Wrapper-Architecture

**10/1999** Liebehenschel, Jens:
The Connection between Lexicographical Generation and Ranking

**11/1999** Brause, R.; Arlt, B.; Tratar, E.:
A Scale-Invariant Object Recognition System for Content-based Queries in Image Databases. *Elektronisch publiziert unter URL http://www.informatik.uni-frankfurt.de/fbreports/fbreport11-99.ps.gz*

**12/1999** Kappes, M.; Klemm, R. P.; Kintala, C. M. R.:
Determining Component-based Software System Reliability is Inherently Impossible

**13/1999** Kappes, Martin:
Multi-Bracketed Contextual Rewriting Grammars With Obligatory Rewriting

**14/1999** Kemp, Rainer:
On the Expected Number of Leftist Nodes in Simply Generated Trees

**1/2000** Kemp, Rainer:
On the Average Shape of Dynamically Growing Trees

**2/2000** Arlt, B.; Brause, R.; Tratar, E.:
MASCOT: A Mechanism for Attention-based Scale-invariant Object Recognition in Images. *Elektronisch publiziert unter URL http://www.cs.uni-frankfurt.de/fbreports/fbreport2-00.pdf*

**3/2000** Heuschen, Frank; Waldschmidt, Klaus:
Bewertung analoger und digitaler Schaltungen der Signalverarbeitung

**4/2000** Hamker, Fred H.; Paetz, Jürgen; Thöne, Sven; Brause, Rüdiger; Hanisch, Ernst:
Erkennung kritischer Zustände von Patienten mit der Diagnose „Septischer Schock" mit einem RBF-Netz. *Elektronisch publiziert unter URL http://www.cs.uni-frankfurt.de/fbreports/fbreport04-00.pdf*

**1/2001** Nebel, Markus E.:
A Unified Approach to the Analysis of Horton-Strahler Parameters of Binary Tree Structures

**2/2001** Nebel, Markus E.:
Combinatorial Properties of RNA Secondary Structures

**3/2001** Nebel, Markus E.:
Investigation of the Bernoulli-Model for RNA Secondary Structures

**4/2001** Malcher, Andreas:
Descriptional Complexity of Cellular Automata and Decidability Questions

**1/2002** Paetz, Jürgen:
Durchschnittsbasierte Generalisierungsregeln; Teil I: Grundlagen

**2/2002** Paetz, Jürgen; Brause, Rüdiger:
Durchschnittsbasierte Generalisierungsregeln Teil II: Analyse von Daten septischer Schock-Patienten

**3/2002** Nießner, Frank:
Decomposition of Deterministic $\omega$-regular Liveness Properties and Reduction of Corresponding Automata

**4/2002** Kim, Pok-Son:
Das $\mathcal{RSV}$-Problem ist $\mathcal{NP}$-vollständig

**5/2002** Nebel, Markus E.:
On a Statistical Filter for RNA Secondary Structures

**6/2002** Malcher, Andreas:
Minimizing Finite Automata is Computationally Hard

# Minimizing Finite Automata Is Computationally Hard

Andreas Malcher
Institut für Informatik, Johann Wolfgang Goethe-Universität
D-60054 Frankfurt am Main, Germany
E-Mail: malcher@psc.informatik.uni-frankfurt.de

**Abstract**

It is known that deterministic finite automata (DFAs) can be algorithmically minimized, i.e., a DFA $M$ can be converted to an equivalent DFA $M'$ which has a minimal number of states. The minimization can be done efficiently [6]. On the other hand, it is known that unambiguous finite automata (UFAs) and nondeterministic finite automata (NFAs) can be algorithmically minimized too, but their minimization problems turn out to be NP-complete and PSPACE-complete [8]. In this paper, the time complexity of the minimization problem for two restricted types of finite automata is investigated. These automata are nearly deterministic, since they only allow a small amount of nondeterminism to be used. On the one hand, NFAs with a fixed finite branching are studied, i.e., the number of nondeterministic moves within every accepting computation is bounded by a fixed finite number. On the other hand, finite automata are investigated which are essentially deterministic except that there is a fixed number of different initial states which can be chosen nondeterministically. The main result is that the minimization problems for these models are computationally hard, namely NP-complete. Hence, even the slightest extension of the deterministic model towards a nondeterministic one, e.g., allowing at most one nondeterministic move in every accepting computation or allowing two initial states instead of one, results in computationally intractable minimization problems.

## 1 Introduction

Finite automata are a well-investigated concept in theoretical computer science with a wide range of applications such as lexical analysis, pattern matching, or protocol specification in distributed systems. Due to time and space constraints it is often very useful to provide minimal or at least succinct descriptions of such automata. Deterministic finite automata (DFAs) and their corresponding language class, the set of regular languages, possess many nice properties such as, for example, closure under many language operations and many decidable questions. In addition, most of the decidability questions for DFAs, such as membership, emptiness, or equivalence, are efficiently solvable (cf. Section 5.2 in [15]). Furthermore, in [6] a minimization algorithm for DFAs is provided working in time $O(n \log n)$, where $n$ denotes the number of states of the given DFA.

It is known that both nondeterministic finite automata (NFAs) and DFAs accept the set of regular languages, but NFAs can achieve exponentially savings in size when compared to DFAs [13]. Unfortunately, certain decidability questions, which are solvable in polynomial time for DFAs, are computationally hard for NFAs such as equivalence, inclusion, or universality [14, 15]. Furthermore, minimization of NFAs is proven to be PSPACE-complete in [8]. In the latter paper, it is additionally shown that unambiguous finite automata (UFAs) have an NP-complete minimization problem.

Therefore, we can summarize that determinism permits efficient solutions whereas the use of nondeterminism often makes solutions computationally intractable. Thus, one might ask what amount of nondeterminism is necessary to make things computationally hard, or, in other words, what amount of nondeterminism may be allowed so that efficiency is preserved.

Measures of nondeterminism in finite automata were first considered in [12] and [2] where the relation between the amount of nondeterminism of an NFA and the succinctness of its description is studied. Here, we look at computational complexity aspects of NFAs with a fixed finite amount of nondeterminism. In particular, these NFAs are restricted such that within every accepting computation at most a fixed number of nondeterministic moves is allowed to be chosen. It is easily observed that certain decidability questions then become solvable in polynomial time in contrast to arbitrary NFAs. However, the minimization problem for such NFAs is proven to be NP-complete.

We further investigate a model where the nondeterminism used is not only restricted to a fixed finite number of nondeterministic moves, but additionally is cut down such that only the first move is allowed to be a nondeterministic one. Hence we come to DFAs with multiple initial states (MDFAs) which were introduced in [5] and recently studied in [11] and [3]. The authors of the latter paper examine the minimization problem for MDFAs and prove its PSPACE-completeness. Their proof is a reduction from Finite State Automata Intersection [4] which states that it is PSPACE-complete to answer the question whether there is a string $x \in \Sigma^*$ accepted by each $A_i$, where DFAs $A_1, A_2, \ldots, A_n$ are given. As is remarked in [4], the problem becomes solvable in polynomial time when the number of DFAs is fixed. We would like to point out that the number of initial states is not part of the instance of the minimization problem for MDFAs discussed in [3]. Thus, one might ask whether minimization of MDFAs with a fixed number of initial states is possible in polynomial time. We will show in Section 3 that the minimization problem of such MDFAs is NP-complete even if only two initial states are given. In analogy to NFAs with fixed finite branching, certain decidability questions can be shown to be efficiently solvable.

The paper is organized as follows. In the next section we will provide and introduce the necessary definitions and notations. Section 3 contains the proof that it is NP-complete to minimize MDFAs with a fixed number of initial states. Some details of this proof will be helpful to prove the NP-completeness of the minimization problem for NFAs with fixed finite nondeterminism. A summary and short discussion of open problems conclude the paper.

## 2 Preliminaries and Definitions

Let $\Sigma^*$ denote the set of all strings over the finite alphabet $\Sigma$, $\epsilon$ the empty string, and $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$. By $|w|$ we denote the length of a string $w$ and by $|S|$ the cardinality of a set $S$. We assume that the reader is familiar with the common notions of formal language theory as presented in [7] as well as with the common notions of computational complexity theory that can be found in [4]. Let $L$ be a regular set; then $size(L)$ denotes the number of states of the minimal DFA accepting $L$. We say that two finite automata are equivalent if both accept the same language. The size of an automaton $M$, denoted by $|M|$, is defined to be the number of states. A state of a finite automaton will be called *trap state* when no accepting state can be obtained on every input.

Concerning the definitions of NFAs with finite branching and MDFAs we follow the notations introduced in [2] and [11].

A nondeterministic finite automaton over $\Sigma$ is a tuple $M = (Q, \Sigma, \delta, q_0, F)$, with $Q$ a finite set of states, $q_0 \in Q$ the initial state, $F \subseteq Q$ the set of accepting states, and $\delta$ a function from $Q \times \Sigma$ to $2^Q$. A move of $M$ is a triple $\mu = (p, a, q) \in Q \times \Sigma \times Q$ with $q \in \delta(p, a)$. A computation for $w = w_1 w_2 \dots w_n \in \Sigma^*$ is a sequence of moves $\mu_1 \mu_2 \dots \mu_n$ where $\mu_i = (q_{i-1}, w_i, q_i)$ with $1 \leq i \leq n$. It is an accepting computation if $q_n \in F$. The language accepted by $M$ is $T(M) = \{w \in \Sigma^* \mid \delta(q_0, w) \cap F \neq \emptyset\}$. $M$ is an (incomplete) deterministic finite automaton if $|\delta(q, a)| \leq 1$ for all pairs $(q, a)$. The branching $\beta_M(\mu)$ of a move $\mu = (q, a, p)$ is defined to be $\beta_M(\mu) = |\delta(q, a)|$. The branching is extended to computations $\mu_1 \mu_2 \dots \mu_n$, $n \geq 0$, by setting $\beta_M(\mu_1 \mu_2 \dots \mu_n) = \beta_M(\mu_1) \cdot \beta_M(\mu_2) \cdot \dots \cdot \beta_M(\mu_n)$. For each word $w \in T(M)$, let $\beta_M(w) = \min \beta_M(\mu_1 \mu_2 \dots \mu_n)$ where $\mu_1 \mu_2 \dots \mu_n$ ranges over all accepting computations of $M$ with input $w$. The branching $\beta_M$ of the automaton $M$ is $\beta_M = \sup \{\beta_M(w) \mid w \in T(M)\}$.

A DFA with multiple initial states (MDFA) is a tuple $M = (Q, \Sigma, \delta, Q_0, F)$ and $M$ is identical to a DFA except that there is a set of initial states $Q_0$. The language accepted by an MDFA $M$ is $T(M) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F \neq \emptyset\}$. An MDFA with $k = |Q_0|$ initial states is denoted by $k$-MDFA.

## 3 Minimizing MDFAs is computationally hard

In this section we are going to show that the minimization problem for $k$-MDFAs is NP-complete. Throughout this section, $k \geq 2$ denotes a constant integer.

| | |
|---|---|
| PROBLEM | $k$-MDFA $\longrightarrow$ $k$-MDFA |
| INSTANCE | A $k$-MDFA $M$ and an integer $l$. |
| QUESTION | Is there an $l$-state $k$-MDFA $M'$ such that $T(M') = T(M)$? |

**Theorem 1** *$k$-MDFA $\longrightarrow$ $k$-MDFA is* NP*-complete.*

**Proof:** The problem is in NP, since a $k$-MDFA $M'$ with $|M'| \leq l$ can be determined nondeterministically and the equality $T(M) = T(M')$ can be tested in polynomial time as is shown below. At first $M$ and $M'$ are converted to DFAs in the following manner. Let $M = (Q, \Sigma, \delta, \{q_0^1, q_0^2, \dots, q_0^k\}, F)$, $M_1 = (Q, \Sigma, \delta, q_0^1, F)$, $M_2 =$

$(Q, \Sigma, \delta, q_0^2, F), \ldots, M_k = (Q, \Sigma, \delta, q_0^k, F)$. Then $T(M_1) \cup T(M_2) \cup \ldots \cup T(M_k) = T(M)$ and we construct a DFA $\hat{M}$ as the Cartesian product of $M_1, M_2, \ldots, M_k$ accepting $T(M_1) \cup \ldots \cup T(M_k)$ in the usual way. A DFA $\hat{M}'$ can be constructed from $M'$ analogously. The time complexity of the inequivalence problem of two DFAs is in NLOGSPACE $\subseteq$ P [9]. Hence $T(\hat{M}) = T(\hat{M}')$ can be tested in polynomial time.

The NP-hardness of the problem will be shown by reduction from the Minimum Inferred DFA problem. In [8] the NP-hardness of the Minimum Inferred DFA problem is used to prove that the Minimum Union Generation problem is NP-complete. To obtain our result, we adapt the proof in [8] to our needs.

| | |
|---|---|
| PROBLEM | Minimum inferred DFA [1] |
| INSTANCE | Finite alphabet $\Sigma$, two finite subsets $S, T \subset \Sigma^*$, integer $l$. |
| QUESTION | Is there an $l$-state DFA that accepts a language $L$ such that $S \subseteq L$ and $T \subseteq \Sigma^* \setminus L$? |

Such an $l$-state DFA will be called *consistent* with $S$ and $T$.

We follow the notations given in [8]. W.l.o.g. we may assume that $S \cap T = \emptyset$. Let $\#, \$$ and $\pounds$ be symbols not in $\Sigma$. Let $\Sigma' = \Sigma \cup \{\#, \$, \pounds\}$, $m = l + size(\overline{T} \cap \overline{S})$, and $t = \max(k, m)$.

$$\begin{aligned} L_1 &= \overline{T}, \\ L_2 &= \overline{T} \cap \overline{S}, \\ L_3 &= \{\$, \pounds\}\#^t L_2 \#^m (\pounds \#^t L_2 \#^m)^*, \\ L_4 &= \$\#^t \overline{T} \#^m, \\ L_5 &= L_3 \cup L_4. \end{aligned}$$

Following [8], it is easy to show the following lemma:

**Lemma 1** *Let $L$ be regular and $M'$ a DFA consistent with $S$ and $T$.*

*(a)* $size(\$\#^t L \#^m) = size((\$\#^t L \#^m)^+) = t + m + 1 + size(L)$

*(b)* $size(L_3) = t + m + 1 + size(L_2)$

*(c)* $\$\#^t L_1 \#^m = \$\#^t (L_2 \cup T(M'))\#^m$

**Proof:** The claims (a) and (c) can be shown similarly to the Claims 4.1. and 4.2. in [8]. Claim (b) can be shown similarly to (a). □

We now present the reduction. Let $M_1 = (Q_1, \Sigma', \delta_1, q_0^1, F_1)$, $M_2 = (Q_2, \Sigma', \delta_2, q_0^2, F_2)$ be two minimal DFAs such that $T(M_1) = L_3$ and $T(M_2) = L_4$. W.l.o.g. we may assume that $Q_1 \cap Q_2 = \emptyset$. We choose $k-2$ additional states $\{q_0^3, \ldots, q_0^k\}$ not in $Q_1 \cup Q_2$. Then we can construct a $k$-MDFA $M = (Q_1 \cup Q_2 \cup \{q_0^3, \ldots, q_0^k\}, \Sigma', \delta, \{q_0^1, q_0^2, \ldots, q_0^k\}, F_1 \cup F_2)$. For $\sigma \in \Sigma'$ we define $\delta(q, \sigma) = \delta_1(q, \sigma)$ if $q \in Q_1$, $\delta(q, \sigma) = \delta_2(q, \sigma)$ if $q \in Q_2$, and $\delta(q_0^i, \sigma) = \delta(q_0^1, \sigma)$ for $i \in \{3, \ldots, k\}$. Then $T(M) = L_5$. The instance $S, T, l$ has been transformed to $M, 3m + 2t + k$. Let $m' = |S| + |T| + l$ be the size of the instance of the Minimum Inferred DFA problem, then it is easily seen that $M$ can be constructed

from $S, T, l$ in time bounded by a polynomial in $m'$. We next show the correctness of the reduction.

Claim: There is an $l$-state DFA consistent with $S$ and $T$ if and only if $T(M) = L_5$ is accepted by a $k$-MDFA $M'$ having at most $3m + 2t + k$ states.

"$\Rightarrow$":
Let $M''$ be a DFA consistent with $S$ and $T$ and $|M''| \leq l$. Let $M_1$ and $M_2$ be the minimal DFAs with $T(M_1) = L_3$ and $T(M_2) = \$\#^t T(M'')\#^m$. Then we have $|M_1| = t+m+1+size(L_2) = t+2m+1-l$, $|M_2| \leq t+m+l+1$ and therefore $|M_1|+|M_2| \leq 3m+2t+2$. Considering the two symbols $\$, \pounds$ we can show analogously to [8] that $T(M_1) \cup T(M_2) = L_5$. Now we choose $k-2$ additional initial states $\{q_0^3, \ldots, q_0^k\} \not\subseteq Q_1 \cup Q_2$ and construct a $k$-MDFA $M' = (Q_1 \cup Q_2 \cup \{q_0^3, \ldots, q_0^k\}, \Sigma', \delta, \{q_0^1, q_0^2, \ldots, q_0^k\}, F_1 \cup F_2)$ in the above-mentioned manner. We thus obtain a $k$-MDFA such that $|M'| \leq 3m+2t+k$ and $T(M') = L_5$.

"$\Leftarrow$":
Let $M = (Q, \Sigma', \delta, \{q_0^1, q_0^2, \ldots, q_0^k\}, F)$ be a $k$-MDFA such that $T(M) = L_5$ and $|M| \leq 3m+2t+k$. We may assume that $M$ is minimal. We have to construct an $l$-state DFA $M'$ consistent with $S$ and $T$. To attain this goal we show that $M$ can be decomposed into two sub-DFA $M_1$ and $M_2$ such that $|M_1| + |M_2| \leq 3m$ and $T(M_1) \cup T(M_2) = \overline{T}\#^m \cup (\overline{T} \cap \overline{S}\#^m)^+$. But this situation is exactly the situation of the "if"-part in Claim 4.3 of [8]. Hence we can conclude that an $l$-state DFA $M'$ consistent with $S$ and $T$ can be constructed.

(a) W.l.o.g. $S \neq \emptyset$. If $S = \emptyset$, then any DFA accepting the empty set is a DFA consistent with $S$ and $T$. Hence there is a one-state DFA accepting the empty set, and there is in particular an $l$-state DFA $M'$ consistent with $S$ and $T$.

(b) Let $w = \$w_1$ with $w_1 \in \#^t S\#^m$ and $w' = w_1'w_2'$ with $w_1', w_2' \in \pounds\#^t L_2\#^m$ be two words in $L_5$. Then there are initial states $q_0^i$ and $q_0^j$ such that $\delta(q_0^i, w) \in F$ and $\delta(q_0^j, w') \in F$. We remark that $q_0^i$ and $q_0^j$ may be identical.

(c) $M$ contains exactly one waist, one tail and two distinct cores.

According to [8] a waist is defined as a sequence of states $q_1, q_2, \ldots, q_m$ such that $\delta(q_i, \#) = q_{i+1}$ for all $i \in \{1, 2, \ldots, m-1\}$ and $q_m$ is an accepting state and has an outgoing $\pounds$-edge. A tail is defined as a sequence of states $q_1, q_2, \ldots, q_m$ such that $\delta(q_i, \#) = q_{i+1}$ for all $i \in \{1, 2, \ldots, m-1\}$ and $q_m$ is an accepting state and has no outgoing edges. A core is defined as a sequence of states $q_1, q_2, \ldots, q_t$ such that $\delta(q_i, \#) = q_{i+1}$ for all $i \in \{1, 2, \ldots, t-1\}$ and $q_t$ is non-accepting and has outgoing edges, but no outgoing $\pounds$-edge.

Obviously, $M$ contains at least one waist, one tail, and one core. We observe that all initial states from which a word in $L_5$ can be accepted have a \$-edge or $\pounds$-edge or both to the first state of a core. Consider the above word $w = \$w_1$. If we have exactly one core, then $\delta(q_0^i, \$) = \delta(q_0^j, \pounds)$ and hence $\delta(q_0^j, \pounds w_1) = \delta(\delta(q_0^i, \$), w_1) = \delta(q_0^i, w) \in F$ which is a contradiction. If $M$ contains two cores which are not distinct, then there are initial states $q_0^i$, $q_0^j$, a state $q \in Q$, and $x \in S$ such that $\delta(q_0^i, \$\#^{i'}) = q = \delta(q_0^j, \pounds\#^{j'})$ with $1 \leq i', j' \leq t$ and $\delta(q, \#^{t-i'}x\#^m) \in F$. Then $\delta(q_0^j, \pounds\#^{j'}\#^{t-i'}x\#^m) \in F$ — contradiction.

If $M$ contains more than two cores, more than one waist, or more than one tail, then $|M|$ exceeds $3m+2t+k$, since $M$ requires at least $2m$ states for waist and tail, $2t$ states for two cores, $k$ initial states, and at least $m \leq t$ states for an additional waist, tail, or core. Hence at least one additional state is needed to realize $L_3$ and $L_4$.

W.l.o.g. we may assume that $w$ will be accepted from $q_0^i$ passing through $\text{core}_1$ and the tail and $w'$ will be accepted from $q_0^j$ passing through $\text{core}_2$ and the waist.

Let $q_t = \delta(q_0^i, w)$ and $q_w = \delta(q_0^j, w_1')$ denote the last states in the tail and the waist. Let $q^1 = \delta(q_0^i, \$), q^2 = \delta(q_0^j, £)$. By $q_c^1 = \delta(q_0^i, \$\#^t)$ and $q_c^2 = \delta(q_0^j, £\#^t)$ we denote the last states of $\text{core}_1$ and $\text{core}_2$. Since $w$ is accepted passing through $\text{core}_1$, we can conclude that $q^2 = \delta(q_w, £)$ is the starting state of the loop.

(d) All initial states have no incoming edges.

Let $q_0^p$ with $p \in \{1, 2, \ldots, k\}$ be an initial state. We may assume that from $q_0^p$ at least one word in $L_5$ can be accepted, otherwise all incoming edges can be removed without affecting the accepted language. Now, assume that $q_0^p$ has an incoming edge. Then this must be a #-edge. We have to show that $q_0^p \neq q_t$ and $q_0^p \neq q_w$. If $q_0^p = q_t$ or $q_0^p = q_w$, then $q_0^p \in F$ by definition of $q_t$ and $q_w$ and therefore $\epsilon \in L_5$ — contradiction.

(e) We claim that $\delta(q^1, \#^t S \#^m) \subseteq F$ and $\delta(q^2, \#^t S \#^m) \cap F = \emptyset$.

By way of contradiction we assume that there is a string $x \in \#^t S \#^m$ such that $\delta(q^1, x) \notin F$. Since $\$x \in L_5$, we then know that $\delta(q^2, x) \in F$ and therefore $\delta(q_0^j, £x) = \delta(q^2, x) \in F$ which is a contradiction. To show the second claim we assume that there is a string $x \in \#^t S \#^m$ such that $\delta(q^2, x) \in F$. Since $\delta(q_0^j, £) = q^2$, we have $\delta(q_0^j, £x) \in F$ — contradiction.

(f) We claim that $\delta(q^2, \#^t L_2 \#^m (£\#^t L_2 \#^m)^*) \subseteq F$.

For contradiction we assume that there is a word $x \in \#^t L_2 \#^m (£\#^t L_2 \#^m)^*$ such that $\delta(q^2, x) \notin F$. Since $£x \in L_5$, we then know that $\delta(q^1, x) \in F$. $\delta(q_0^j, w_1' £x) = \delta(q^2, x) \notin F$: then there must be an initial state $q_0'$ with $\delta(q_0', w_1' £x) = \delta(q^1, x) \in F$, in particular $\delta(q_0', w_1' £) = q^1$. Then we have $\delta(q_0', w_1' £ w_1) = \delta(q^1, w_1) \in F$ which is a contradiction.

(g) $M$ can be modified to the form depicted in Figure 1. (The initial states $q_0^3, \ldots, q_0^k$ are not included.)

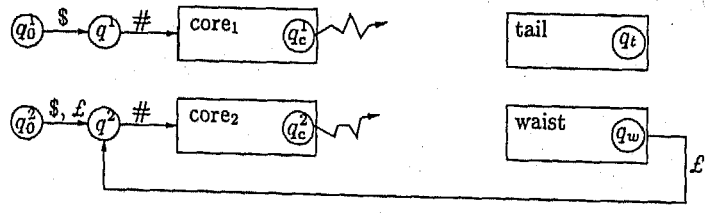


Figure 1: The modified $k$-MDFA $M$

At first we remove all edges from initial states to any other states. We choose two different initial states $q_0^1$ and $q_0^2$ and then insert the following edges: $q_0^1 \xrightarrow{\$} q^1$,

$q_0^2 \xrightarrow{\$,\pounds} q^2$, and $q_0^i \xrightarrow{\$,\pounds} q^2$ for $i \in \{3,\ldots,k\}$. We observe that due to (d), (e), and (f) the modified automaton still recognizes $L_5$. In particular, $L_3$ is accepted from $q_0^2$ and all words in $\$\#^t S\#^m$ are accepted only from $q_0^1$.

(h) We now look at the two DFAs obtained when considering only one initial state in $M$. We define the set of reachable states as follows: $\mathcal{E}(q_0^1) = \{q \in Q \mid \exists x, x' \in (\Sigma')^* : \delta(q_0^1, x) = q \wedge \delta(q, x') \in F\}$. $\mathcal{E}(q_0^2)$ is defined analogously.

We first claim that there is no edge from $p \in \mathcal{E}(q_0^2)$ to a state $q$ from which $q_t$ can be obtained. Assume by way of contradiction that there are $p \in \mathcal{E}(q_0^2)$, $q \in Q$, $s \in \Sigma'$, and $u \in (\Sigma')^*$ such that $\delta(p, s) = q$ and $\delta(q, u) = q_t \in F$. Since $p \in \mathcal{E}(q_0^2)$, there are strings $x, x' \in (\Sigma')^*$ such that $\delta(q_0^2, x) = p$ and $\delta(p, x') \in F$. Due to (g), we may assume that $x$ starts with $\pounds$. We then know that $\delta(q_0^2, xsu) = w_t \in F$, but $\delta(q_0^2, xsuxsu) \notin F$, because $q_t$ has no outgoing edges. Moreover, $\delta(q_0^1, xsuxsu) \notin F$, since $q_0^1$ has no outgoing $\pounds$-edge. Hence $xsuxsu \notin L_5$ which is a contradiction, because $xsu \in L_3$ and therefore $xsuxsu \in L_3 \subset L_5$.

Furthermore, we observe that all edges from states in $\mathcal{E}(q_0^1)$ to states in $\mathcal{E}(q_0^2)$ can be removed. If we have such an edge, all words passing this edge will be accepted in the waist and therefore are in $L_3$. Hence these words can already be accepted from $q_0^2$ due to (f) and (g). So, removing such edges does not affect the accepted language. We observe that this modification yields $\mathcal{E}(q_0^1) \cap \mathcal{E}(q_0^2) = \emptyset$.

(i) Since the sets of reachable states are distinct, we obtain two DFAs $M_1' = (Q_1', \Sigma', \delta_1', q_0^1, F_1')$ and $M_2' = (Q_2', \Sigma', \delta_2', q_0^2, F_2')$ after having minimized the two DFAs $(\mathcal{E}(q_0^1), \Sigma', \delta, q_0^1, F)$ and $(\mathcal{E}(q_0^2), \Sigma', \delta, q_0^2, F)$. Due to (e) and (g), we know that $L_4 \supseteq T(M_1') \supseteq \$\#^t S\#^m$ and $T(M_2') = L_3$. Furthermore, $|M_1'| + |M_2'| \leq 3m + 2t + 2$, since $Q_1' \cap Q_2' = \emptyset$.

(j) Starting from $M_1'$ we define another DFA $M_1$ by removing $q_0^1$, $q^1$ and the first $t-1$ states of $\text{core}_1$. We define $q_c^1$ as new initial state and observe that $\overline{T}\#^m \supseteq T(M_1) \supseteq S\#^m$. Starting from $M_2'$ we define another DFA $M_2$ by removing $q_0^2$, $q^2$ and the first $t-1$ states of $\text{core}_2$. We define $q_c^2$ as new initial state. The $\pounds$-edge from $q_w$ to $q^2$ is replaced by the following edges: if $\delta_2'(q_c^2, \sigma) = q$ for $\sigma \in \Sigma$, we add a $\sigma$-edge from $q_w$ to $q$. It is easy to see that $T(M_2) = (\overline{T} \cap \overline{S}\#^m)^+$. Hence we have $T(M_1) \cup T(M_2) = \overline{T}\#^m \cup (\overline{T} \cap \overline{S}\#^m)^+$. Moreover, $|M_1| + |M_2| \leq 3m$.

(k) We have $|M_2| = size((\overline{T} \cap \overline{S}\#^m)^+) = 2m - l$ and therefore $|M_1| \leq 3m - |M_2| = 3m - 2m + l = m + l$. Removing the tail in $M_1$ yields an $l$-state DFA $M'$ consistent with $S$ and $T$.

□

**Corollary 1** *Let $k, k' \geq 2$ be two constant numbers. Then DFA $\longrightarrow$ $k$-MDFA and $k$-MDFA $\longrightarrow$ $k'$-MDFA are* NP*-complete.*

The following theorem is a simple observation of the fact that $k$-MDFAs can be efficiently converted to DFAs whose size is bounded by a polynomial in $k$, and that the below-mentioned decidability questions are efficiently solvable for DFAs.

**Theorem 2** *Let $M$ be a $k$-MDFA and $M'$ be a $k'$-MDFA. Then the following problems are solvable in polynomial time. Is $T(M) = T(M')$? Is $T(M) \subseteq T(M')$? Is $T(M) \subset T(M')$? Is $T(M) = \Sigma^*$?*

# 4 Minimizing NFAs with fixed finite branching is computationally hard

In this section we are going to show that the minimization problem for NFAs with branching $\beta = k$ (NFA($\beta = k$)) is NP-complete for $k \geq 3$.

| | |
|---|---|
| PROBLEM | NFA($\beta = k$) $\longrightarrow$ NFA($\beta = k$) |
| INSTANCE | An NFA $M$ with branching $\beta = k$ and an integer $l$. |
| QUESTION | Is there an $l$-state NFA $M'$ with branching $\beta = k$ such that $T(M') = T(M)$? |

**Lemma 2** *Let $M$ be an NFA and $k \geq 2$ be a constant integer. Then the problem whether $M$ has branching $k$ can be solved in polynomial time.*

**Proof:** We consider the language

$$T_i(M) = \{w \in \Sigma^* \mid \text{there is an accepting computation } \pi \text{ of } M \text{ of } w \text{ with } \beta(\pi) \leq i\}.$$

In [2] it is shown that a DFA $M_i$ accepting $T_i(M)$ can be effectively constructed. We observe that the construction can be done in time polynomially bounded in $|M|$ and the resulting DFA has size $O(|M|^k)$. A detailed discussion may be found in the appendix.

Since $T_k(M) \subseteq T(M)$, we have: $T(M) \setminus T_k(M) = \emptyset \Leftrightarrow \beta_M \leq k$. Since $M_k$ is a DFA, we can simply construct a DFA $M'_k$ accepting the complement $\Sigma^* \setminus T_k(M)$.

$$\beta_M \leq k \Leftrightarrow T(M) \setminus T_k(M) = \emptyset \Leftrightarrow T(M) \cap \overline{T_k(M)} = \emptyset \Leftrightarrow T(M) \cap T(M'_k) = \emptyset$$

Since $M$ is an NFA and $M'_k$ is a DFA, we can construct, in polynomial time, an NFA $\hat{M}$ of size $O(|M| \cdot |M|^k)$ as the Cartesian product of $M$ and $M'_k$ accepting $T(M) \cap T(M'_k)$. The non-emptiness of $T(\hat{M})$ can be tested in NLOGSPACE $\subseteq$ P [10]. If $T(\hat{M}) \neq \emptyset$, then $\beta_M > k$. If $T(\hat{M}) = \emptyset$, then we know that $\beta_M \leq k$. To find out whether $\beta_M = k$, we construct $T_{k-1}(M)$ if $k - 1 \geq 1$. This can be done in polynomial time as well as the test for inequivalence of $T_{k-1}(M)$ and $T_k(M)$. If both sets are inequivalent, then $\beta_M = k$; otherwise $\beta_M < k$. □

**Theorem 3** *NFA($\beta = k$) $\longrightarrow$ NFA($\beta = k$) is* NP*-complete for $k \geq 3$.*

**Proof:** We first show that the problem is in NP. To this end we determine nondeterministically an NFA $M'$ with $|M'| \leq l$. Due to Lemma 2, we can test whether $M'$ has branching $k$ in polynomial time. We next convert $M$ and $M'$ to $k$-MDFAs $\hat{M}$ and $\hat{M}'$ with at most $k|M|+1$ and $k|M'|+1$ states applying the construction presented in [11]. The equality of $T(\hat{M})$ and $T(\hat{M}')$ can then be tested in polynomial time analogous to the considerations of Theorem 1. Hence the above problem is in NP.

The NP-hardness of the problem will be shown by reduction from the Minimum Inferred DFA problem similar to the proof for MDFAs.

Let $m = l + size(\overline{T} \cap \overline{S})$ and $n = 5m + 1$. In addition to the previous definitions we define:

$$\begin{aligned} L'_3 &= \{\$, \pounds\}\#^m(\#^{m+1})^* L_2 \#^m (\pounds \#^m (\#^{m+1})^* L_2 \#^m)^*, \\ L'_4 &= \$\#^m(\#^{m+1})^* \overline{T} \#^m, \\ L^i &= \{(\$\#^{in^k-1})^+\}\ (1 \leq i \leq k-2), \\ L'_5 &= L^1 \cup L^2 \cup \ldots \cup L^{k-2}, \\ L'_6 &= L'_3 \cup L'_4 \cup L'_5. \end{aligned}$$

**Lemma 3** *Let $L$ be regular and $M'$ a DFA consistent with $S$ and $T$.*

*(a)* $size(\$\#^m(\#^{m+1})^* L \#^m) = size((\$\#^m(\#^{m+1})^* L \#^m)^+) = 2m + 1 + size(L)$

*(b)* $size(L'_3) = 2m + 1 + size(L_2)$

*(c)* $\$\#^m(\#^{m+1})^* L_1 \#^m = \$\#^m(\#^{m+1})^*(L_2 \cup T(M'))\#^m$

*(d)* $size(L^i) = in^k + 1$

*(e)* $size(\{\$\#^m(\#^{m+1})^*\} \cup L'_5) \geq n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + 1 + (m+1)$

**Proof:** The claims (a), (b), and (c) can be shown analogously to those of Lemma 1. Claim (d) is obvious. The proof of (e) is not difficult, but lengthy and will be shown in the appendix. □

We now present the reduction. Let $M_1 = (Q_1, \Sigma', \delta_1, q_0^1, F_1)$, $M_2 = (Q_2, \Sigma', \delta_2, q_0^2, F_2)$ be two minimal DFAs such that $T(M_1) = L'_3$ and $T(M_2) = L'_4$. Furthermore, let $M_i = (Q_i, \Sigma', \delta_i, q_0^i, F_i)$, $3 \leq i \leq k$ be $k-2$ minimal DFAs accepting $L^1, L^2, \ldots, L^{k-2}$. W.l.o.g. we may assume that $Q_1$, $Q_2$, ..., and $Q_k$ are pairwise distinct. We observe that for $3 \leq i \leq k$ the states $q_0^i$ have no incoming edges and only one outgoing edge to a non-trap state, namely a \$-edge. Moreover, $q_0^1$ has no incoming edges and only two outgoing edges to non-trap states, namely a \$-edge and a £-edge. We remove $q_0^1$ from $M_1$ and $q_0^i$ from $M_i$ for $3 \leq i \leq k$ and construct an NFA $M = ((Q_1 \setminus \{q_0^1\}) \cup Q_2 \cup (Q_3 \setminus \{q_0^3\}) \cup \ldots \cup (Q_k \setminus \{q_0^k\}), \Sigma', \delta, q_0^2, F_1 \cup F_2 \cup \ldots \cup F_k)$. For $\sigma \in \Sigma'$ and $1 \leq i \leq k$ we define $\delta(q, \sigma) = \delta_i(q, \sigma)$ if $q \in Q_i$. Furthermore, $\delta(q_0^2, \$) = \delta_1(q_0^1, \$)$, $\delta(q_0^2, \pounds) = \delta_1(q_0^1, \pounds)$, and $\delta(q_0^2, \$) = \delta_i(q_0^i, \$)$ for $3 \leq i \leq k$. Then $T(M) = L'_6$ and $M$ is an NFA with branching $k$.

The instance $S, T, l$ has been transformed to $M, 5m+1+\sum_{i=1}^{k-2} in^k$. Let $m' = |S|+|T|+l$ be the size of the instance of the Minimum Inferred DFA problem, then it is easily seen that $M$ can be constructed from $S, T, l$ in time bounded by a polynomial in $m'$. We next show the correctness of the reduction.

Claim: There is an $l$-state DFA consistent with $S$ and $T$ if and only if $T(M) = L'_6$ is accepted by an NFA $M'$ with branching $\beta_M = k$ that has at most $5m + 1 + \sum_{i=1}^{k-2} in^k$ states.

"$\Rightarrow$":
Let $M''$ be a DFA consistent with $S$ and $T$ and $|M''| \leq l$. Let $M_1$ and $M_2$ be the minimal DFAs with $T(M_1) = L'_3$, $T(M_2) = \$\#^m(\#^{m+1})^*T(M'')\#^m$. Furthermore, $M_3, \ldots, M_k$ are minimal DFAs accepting $L^1, \ldots, L^{k-2}$. Analogous to the proof of Theorem 1 and the above considerations we can construct an NFA $M'$ with branching $\beta_{M'} = k$ such that $T(M') = L'_6$ and $|M'| \leq 5m + 1 + \sum_{i=1}^{k-2} in^k$.

"$\Leftarrow$":
Let $M = (Q, \Sigma', \delta, q_0, F)$ be an NFA with branching $\beta_M = k$ such that $T(M) = L'_6$ and $|M| \leq 5m + 1 + \sum_{i=1}^{k-2} in^k$. We may assume that $M$ is minimal. We have to construct an $l$-state DFA $M'$ consistent with $S$ and $T$. Due to the definition of $L'_6$, we can show that the nondeterministic moves of $M$ have to start in $q_0$. Then, $M$ can be converted to a 2-MDFA $M''$ such that $|M''| \leq 3m + 2t + 2$, setting $t = m$, and $T(M'') = L_5$. Due to the proof of Theorem 1, we then can conclude that an $l$-state DFA $M'$ consistent with $S$ and $T$ can be constructed.

(a) W.l.o.g. $S \neq \emptyset$. Let $w = \$\#^m w_1 \#^m$ with $w_1 \in S$ and $w' = w'_1 w'_2$ with $w'_1, w'_2 \in \mathcal{L}\#^m L_2 \#^m$ be two words in $L'_6$.

(b) $M$ contains exactly one waist, one tail, two distinct loop-cores, and $k-2$ \$-#-loops of length $n^k, 2n^k, \ldots, (k-2)n^k$.

A loop-core is defined as a sequence of states $q_1, q_2, \ldots, q_m, q_{m+1}$ such that $\delta(q_i, \#) = q_{i+1}$ for all $i \in \{1, 2, \ldots, m\}$ and $q_{m+1}$ is non-accepting, has outgoing edges, in particular a #-edge to $q_1$, but no outgoing $\mathcal{L}$-edge.

A \$-#-loop of length $jn^k$ with $1 \leq j \leq k-2$ is defined as a sequence of states $q_1, q_2, \ldots, q_{jn^k}$ such that $\delta(q_i, \#) = q_{i+1}$ for all $i \in \{1, 2, \ldots, jn^k - 1\}$ and $q_{jn^k}$ is accepting and has an outgoing \$-edge to $q_1$.

Obviously, $M$ contains at least one waist, one tail, and one loop-core. Consider the above word $w \in L_5$. If we have exactly one loop-core, then there is a state $q \in \delta(q_0, \$) \cap \delta(q_0, \mathcal{L})$ and $\delta(q, \#^m w_1 \#^m) \cap F \neq \emptyset$. Hence we have that $\delta(q_0, \mathcal{L}\#^m w_1 \#^m) \cap F \neq \emptyset$ which is a contradiction. If $M$ contains two loop-cores which are not distinct, then there is a state $q \in \delta(q_0, \$\#^i) \cap \delta(q_0, \mathcal{L}\#^j)$ with $1 \leq i, j \leq m$ and $\delta(q, \#^{m-i} w_1 \#^m) \cap F \neq \emptyset$. Then $\delta(q_0, \mathcal{L}\#^j \#^{m-i} w_1 \#^m)$ $\cap F \neq \emptyset$ — contradiction.

It is easy to see that the states of the tail and the waist are distinct from those of a \$-#-loop. Furthermore, the states of a loop-core and a \$-#-loop are distinct. By way of contradiction we assume that there exist $1 \leq j \leq k-2$ and a state $q \in \delta(q_0, \$\#^i) \cap \delta(q_0, \$\#^{jn^k-1}\$\#^{j'})$ and $\delta(q, \#^{i'} w_1 \#^m) \cap F \neq \emptyset$ with $1 \leq i \leq m$, $1 \leq j' \leq jn^k - 1$. Then $\delta(q_0, \$\#^{jn^k-1}\$\#^j \#^{i'} w_1 \#^m) \cap F \neq \emptyset$ which is a contradiction. We now show that \$-#-loops of different length have distinct states; hence $M$ contains $k-2$ \$-#-loops of length $n^k, 2n^k, \ldots, (k-2)n^k$. Assume by way of contradiction that there is a state $q \in \delta(q_0, \$\#^{in^k-1}\$\#^{i'}) \cap \delta(q_0, \$\#^{jn^k-1}\$\#^{j'}) \neq \emptyset$ with $i \neq j$, $i' \leq in^k - 1$, $j' \leq jn^k - 1$, and $\delta(q, \#^{jn^k-1-j'}\$\#^{jn^k-1}) \cap F \neq \emptyset$. Then it follows that $\$\#^{in^k-1}\$\#^{i'+jn^k-1-j'}\$\#^{jn^k-1} \in L'_6$ which is a contradiction.

If $M$ contains more than two loop-cores, more than one waist, more than one tail, or more than one \$-#-loop of the same length, then $|M|$ exceeds $5m + 1 +$

$\sum_{i=1}^{k-2} in^k$, since $M$ requires at least $2m$ states for waist and tail, $2m$ states for two loop-cores, $\sum_{i=1}^{k-2} in^k$ states for the \$-#-loops, one initial state, and at least $m$ states for an additional waist, tail, loop-core, or \$-#-loop. Hence at least one additional state is needed to realize $L_3'$ and $L_4'$.

W.l.o.g. we may assume that $w$ will be accepted passing through loop-core$_1$ and the tail and $w'$ will be accepted passing through loop-core$_2$ and the waist.

Let $q_t \in \delta(q_0, w)$ and $q_w \in \delta(q_0, w_1')$ denote the las9.t states in the tail and the waist. By $q^1$ and $q^2$ we denote the states obtained after having read \$ and £ when $M$ passes through the accepting computations of $w$ and $w_1'$. Since $w$ is accepted passing through loop-core$_1$, we can conclude that $\{q^2\} = \delta(q_w, £)$ is the starting state of the loop in $L_3'$.

(c) All computations starting in $q^2 \in \delta(q_0, £)$ and leading to an accepting state, thus computations of words in $£\#^m(\#^{m+1})^* L_2 \#^m (£\#^m(\#^{m+1})^* L_2 \#^m)^*$, have branching 1. This is obvious, since even one move with a branching greater than one would imply that $M$ contains accepting computations with infinite branching due to the £-edge from $q_w$ to $q^2$.

(d) The loop-cores and the \$-#-loops contain no moves with branching greater than one, since due to their loops there would be computations with infinite branching.

(e) All computations starting in $\delta(q_0, \$)$ and leading to an accepting state, thus computations of words in $\$\#^m(\#^{m+1})^* S' \#^m$ with $S \subseteq S' \subseteq \overline{T}$, $L_5'$, and $\$\#^m(\#^{m+1})^* L_2 \#^m (£\#^m(\#^{m+1})^* L_2 \#^m)^*$, have branching 1.

Due to (c) and (d) the moves with branching greater than one have to be located either in the states before entering the loop-core and the \$-#-loops, or in the states recognizing $S'\#^m$.

First of all, we assume that all moves with branching greater than one start before entering the loop-core and the \$-#-loops. Then we can shift the branching to $q_0$: we remove any outgoing \$-edges from $q_0$ and insert $k-2$ \$-edges to the first states of the \$-#-loops and two \$-edges to loop-core$_1$ and loop-core$_2$. It follows that the modified automaton still recognizes $L_6'$, but there is at least one unnecessary state $q \in \delta(q_0, \$)$. Hence $M$ was not minimal which is a contradiction.

We now assume that there is at least one move with branching 2 within the states recognizing $S'\#^m$. Then $L = \$\#^m(\#^{m+1})^* \cup L_5'$ must be recognized by an NFA with a branching of at most $\lfloor \frac{k}{2} \rfloor$. Due to Lemma 3 we know that a DFA for $L$ needs at least $n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + 1 + (m+1)$ states. Analogous to the considerations in (b) one can see that every NFA accepting $L$ with finite branching contains $k-2$ different \$-#-loops of length $n^k, 2n^k, \ldots, (k-2)n^k$, a loop-core of length $m+1$ and an initial state. In comparison with the minimal DFA, an NFA with finite branching can therefore achieve savings in size only through nondeterministic moves that start in states which are not part of a loop. Subtracting the loop-states from $n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + 1 + (m+1)$, there remain $(k-2)n^k + 1$ states. In [2] it is shown that the best possible reduction of states that an NFA with branching $i$ can achieve in comparison with the corresponding minimal DFA is at most the $i$-th root of the size of the DFA.

Hence an NFA accepting $L$ with branching $\lfloor \frac{k}{2} \rfloor$ has at least $n^k + 2n^k + \ldots + (k-2)n^k + (m+1) + ((k-2)n^k+1)^{1/\lfloor \frac{k}{2} \rfloor}$ states. Since $((k-2)n^k+1)^{1/\lfloor \frac{k}{2} \rfloor} \geq n^2$, we have that $|M| \geq \sum_{i=1}^{k-2} in^k + n^2$ which is a contradiction to $|M| \leq 5m+1+\sum_{i=1}^{k-2} in^k = n + \sum_{i=1}^{k-2} in^k$.

It follows that $|\delta(q_0, \$)| > 1$. From $q_0$ we then have a \$-edge to $q^1$ and the first states of the $k-2$ \$-#-loops. Furthermore, we can assume to have a \$-edge to $q^2$. If there is no such edge, we can insert one without affecting the accepted language. We next remove the $k-2$ \$-#-loops and reduce the two loop-cores to cores by removing their #-loops. We then have an NFA with branching 2 with $3m + 2t + 1$ states ($t = m$) accepting $L_5$. Now, we remove the \$-edge from $q_0$ to $q^1$ and we insert an additional state $q_0'$ which has an outgoing \$-edge to $q^1$. Thus, we have a 2-MDFA with $3m+2t+2$ states accepting $L_5$. Due to Theorem 1 we can construct an $l$-state DFA $M'$ consistent with $S$ and $T$. □

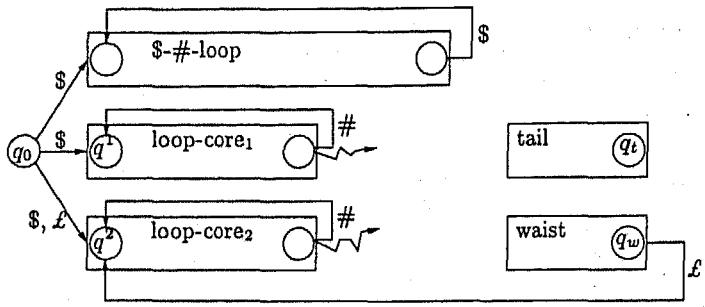


Figure 2: The NFA($\beta = 3$) $M$ accepting $L_6'$.

**Corollary 2** *Let $k \geq 2$ and $k' \geq 3$ be constant integers. Then DFA $\longrightarrow$ NFA($\beta = k'$) and NFA($\beta = k$) $\longrightarrow$ NFA($\beta = k'$) are* NP*-complete.*

**Theorem 4** *The following problems, which are* PSPACE*-complete when arbitrary NFAs are considered, are solvable in polynomial time.*

*(a) Given two NFAs $M, M'$ with $\beta_M = k$ and $\beta_{M'} = k'$. Is $T(M) = T(M')$? Is $T(M) \subseteq T(M')$? Is $T(M) \subset T(M')$? Is $T(M) = \Sigma^*$?*

*(b) Given an arbitrary NFA $M$ and an NFA $M'$ with $\beta_{M'} = k$. Is $T(M) \subseteq T(M')$?*

**Proof:** Claim (a) results from the fact that NFAs with branching $k$ can be efficiently converted to DFAs whose size is bounded by a polynomial in $k$, and that the decidability questions are efficiently solvable for DFAs. To prove (b) we observe that $T(M) \subseteq T(M') \Leftrightarrow T(M) \cap \overline{T(M')} = \emptyset$. $M'$ can be converted to a DFA of size $O(|M'|^k)$ and a DFA accepting $\overline{T(M')}$ has then $O(|M'|^k)$ states as well. Analogous to the construction of Lemma 2, we obtain an NFA $\hat{M}$ accepting $T(M) \cap \overline{T(M')}$ and test its emptiness. We observe that the construction and the test can be performed in polynomial time. □

# 5 Conclusions

In this paper, we have shown that the minimization of finite automata equipped with a very small and fixed amount of nondeterminism is computationally hard. In particular, the minimization problems for DFAs with a fixed number of initial states as well as for NFAs with fixed finite branching have been proven NP-complete. Hence, even the slightest amount of nondeterminism makes minimization computationally intractable whereas equivalence, inclusion, or universality questions preserve their efficient solutions. Hence the question arises whether there are extensions of the deterministic model at all that preserve polynomial time minimization algorithms. Two candidates result from our considerations. At first, the computational complexity of the problem $\mathrm{NFA}(\beta = k) \longrightarrow \mathrm{NFA}(\beta = 2)$ remains open. Obviously, the problem is in NP, but NP-hardness cannot be shown using the approach of Theorem 3. The two constructions in Theorem 1 and Theorem 3 present finite automata which are not unambiguous. It is currently unknown whether unambiguous $k$-MDFAs or unambiguous NFAs with branching $k$ provide efficient minimization algorithms.

# References


[1] E.M. Gold, "Complexity of automaton identification from given data", Information and Control 37(3): 302-320, 1978

[2] J. Goldstine, C.M.R. Kintala, D. Wotschke: "On measuring nondeterminism in regular languages", Information and Computation, 86(2): 179-194, 1990

[3] M. Holzer, K. Salomaa, S. Yu: "On the state complexity of $k$-entry deterministic finite automata", Journal of Automata, Languages and Combinatorics, 6(4): 453-466, 2001

[4] M.R. Garey, D.S. Johnson: "Computers and Intractability", W.H. Freeman and Co., San Francisco, 1979

[5] A. Gill, L. Kou, "Multiple-entry finite automata", Journal Computing and System Sciences 9: 1-19, 1974

[6] J.E. Hopcroft: "An $n \log n$ algorithm for minimizing states in a finite automaton", In Z. Kohavi (ed.): "Theory of machines and computations", 189-196, Academic Press, New York, 1971

[7] J.E. Hopcroft, J.D. Ullman: "Introduction to Automata Theory, Languages and Computation", Addison-Wesley, Reading MA, 1979

[8] T. Jiang, B. Ravikumar: "Minimal NFA problems are hard", SIAM Journal on Computing 22(6): 1117-1141, 1993

[9] N.D. Jones, E.Y. Lien, W.T. Laaser, "New problems complete for nondeterministic log space", Mathematical Systems Theory, 10(1): 1-17, 1976

[10] N.D. Jones, "Space-bounded reducibility among combinatorial problems", Journal of Computer and System Sciences, 11(1): 68-85, 1975

[11] M. Kappes: "Descriptional complexity of deterministic finite automata with multiple initial states", Journal of Automata, Languages and Combinatorics, 5(3): 269-278, 2000

[12] C.M.R. Kintala, D. Wotschke, "Amounts of nondeterminism in finite automata", Acta Informatica, 13(2): 199-204, 1980

[13] A.R. Meyer, M.J. Fischer: "Economy of descriptions by automata, grammars, and formal systems", IEEE Symposium on Foundations of Computer Science, 188-191, 1971

[14] L. Stockmeyer, A.R. Meyer, "Word problems requiring exponential time: preliminary report", Fifth Annual ACM Symposium on Theory of Computing, 1-9, 1973

[15] S. Yu: "Regular languages", In G. Rozenberg, A. Salomaa (Eds.): "Handbook of Formal Languages Volume 1", 41-110, Springer-Verlag, Berlin, 1997

# Appendix

**Claim:** Let $M = (Q, \Sigma, \delta, q_0, F)$ be an NFA. A DFA accepting $T_k(M)$ can be constructed in time polynomially bounded in $|M|$.

**Proof:** We reproduce the construction from [2] and observe that it can be performed in polynomial time.

Let $\Sigma_T = \{[p,a,q] \in Q \times \Sigma \times Q \mid q \in \delta(p,a)\}$ be the alphabet of triples corresponding to moves of $M$.

$$R = \{[q_0,a_1,q_1][q_1,a_2,q_2]\ldots[q_{n-1},a_n,q_n] \in \Sigma_T^* \mid n \geq 1, q_n \in F\} \cup \{\epsilon \mid q_0 \in F\}$$

is then the regular set of all accepting computations of $M$. Obviously, a DFA accepting $R$ is the "deterministic version" of $M$ with $size(R) = O(|M|)$ that can be constructed in time $O(|M| \cdot |\Sigma_T|) = O(|M|^3)$. Let $f : \Sigma_T^* \to \Sigma^*$ and $g : \Sigma_T^* \to \{c,d\}^*$ be homomorphisms such that $f([p,a,q]) = a$ and $g([p,a,q]) = \epsilon$ if $|\delta(p,a)| = 1$ and $g([p,a,q]) = c^{|\delta(p,a)|}d$ otherwise. Furthermore,

$$S_k = \{c^{j_1}d\ldots c^{j_t}d \mid t \geq 1, \text{ each } j_i \geq 2, j_1 \cdot j_2 \cdot \ldots \cdot j_t \leq k\} \cup \{\epsilon\}.$$

Since $k$ is a constant number, it follows that $size(S_k)$ and $size(g^{-1}(S_k))$ are in $O(1)$ and the corresponding DFAs can be constructed in constant time and $O(|\Sigma_T|) = O(|M|^2)$, respectively. Constructing the Cartesian product of $R$ and $g^{-1}(S_k)$, we obtain a DFA accepting $R \cap g^{-1}(S_k)$ of size $O(|M|)$ in time $O(|M|^3)$. The construction of an NFA $M'$ accepting $f(R \cap g^{-1}(S_k))$ can be done by relabeling of the edges of the DFA for $R \cap g^{-1}(S_k)$, and can be performed in time $O(|M|^3)$. We observe that $M'$ has branching $k$, $|M'| = O(|M|)$, and $T(M') = f(R \cap g^{-1}(S_k)) = f(\{\pi \in R \mid \beta(\pi) \leq k\}) = T_k(M)$. Applying the construction presented in [11], we can convert $M'$ to a $k$-MDFA with at most $k|M'|+1 = O(|M|)$ states in time $O(|M|)$. Then, this $k$-MDFA can be converted to a DFA with at most $O(|M|^k)$ states in time $O(|M|^k)$ analogous to the construction of Theorem 1. □

**Claim:** Let $L = \{\$\#^m(\#^{m+1})^*\} \cup L_5'$.
Then $size(L) \geq n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + 1 + (m+1)$.

**Proof:** We use the Nerode equivalence relation $\equiv_L$ on $L$ and show that the index $\text{index}(\equiv_L) \geq n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + 1 + (m+1)$. For $x, y \in \Sigma^*$, $\equiv_L$ is defined as:

$$x \equiv_L y :\Longleftrightarrow xz \in L \Leftrightarrow yz \in L \text{ for all } z \in \Sigma^*.$$

Let $1 \leq i \leq k-2$; we define the following sets of strings:

$$\begin{aligned}
A_i &= \{a_{i,0}, a_{i,1}, \ldots, a_{i,in^k-1}\} \text{ with } a_{i,j} = \$\#^{in^k-1}\$\#^j \text{ and } 0 \leq j \leq in^k - 1,\\
B &= \{b_1, b_2, \ldots, b_{m+1}\} \text{ with } b_j = \$\#^{(k-2)n^k-1}\#^j \text{ and } 1 \leq j \leq m+1,\\
C &= \{c_0, c_1, \ldots, c_{(k-2)n^k-1}\} \text{ with } c_j = \$\#^j \text{ and } 0 \leq j \leq (k-2)n^k - 1,\\
D &= \{\epsilon\}.
\end{aligned}$$

Obviously, $|A_i| = in^k$, $|B| = m+1$, $|C| = (k-2)n^k$, and $|D| = 1$. We have to show that each two words from $A_1 \cup A_2 \cup \ldots \cup A_{k-2} \cup B \cup C \cup D$ are not $\equiv_L$-equivalent.

(a) Claim: Let $x, y \in A_i$ such that $x \neq y$. Then $x \not\equiv_L y$.
Let $x = \$\#^{in^k-1}\$\#^{i'}$ and $y = \$\#^{in^k-1}\$\#^{j'}$ with $0 \leq i' < j' \leq in^k - 1$. We define $z = \#^{in^k-1-j'}\$\#^{in^k-1}$ and obtain that $xz \notin L$ and $yz \in L$.

(b) Claim: Let $x, y \in B$ such that $x \neq y$. Then $x \not\equiv_L y$.
Let $x = \$\#^{(k-2)n^k-1}\#^{i'}$ and $y = \$\#^{(k-2)n^k-1}\#^{j'}$ with $1 \leq i' < j' \leq m+1$. Then $j' = i' + r$ with $1 \leq r \leq m$. Let $i'' \geq 0$ be the minimal integer such that $(k-2)n^k - 1 + i' + i'' - m$ is a multiple of $m+1$. Then $(k-2)n^k - 1 + i' + i'' = m + t(m+1)$ with $t \geq 1$. We now set $z = \#^{i''}$ and observe that $xz = \$\#^{(k-2)n^k-1}\#^{i'}\#^{i''} = \$\#^{m+t(m+1)} \in L$, but $yz = \$\#^{(k-2)n^k-1}\#^{j'}\#^{i''} \notin L$, since $m + t(m+1) = (k-2)n^k - 1 + i' + i'' < (k-2)n^k - 1 + j' + i'' = (k-2)n^k - 1 + i' + r + i'' = m + t(m+1) + r < m + (t+1)(m+1)$.

(c) Claim: Let $x, y \in C$ such that $x \neq y$. Then $x \not\equiv_L y$.
Let $x = \$\#^{i'}$ and $y = \$\#^{j'}$ with $0 \leq i' < j' \leq (k-2)n^k - 1$. We set $z = \#^{(k-2)n^k-1-j'}\$\ \#^{(k-2)n^k-1}$ and obtain that $xz \notin L$ and $yz \in L$.

(d) Claim: Let $x \in A_i$ and $y \in A_j$ with $1 \leq j \leq k-2$ and $i \neq j$. Then $x \not\equiv_L y$.
Let $x = \$\#^{in^k-1}\$\#^{i'}$ and $y = \$\#^{jn^k-1}\$\#^{j'}$ with $0 \leq i' \leq in^k - 1$ and $0 \leq j' \leq jn^k - 1$. W.l.o.g. we may assume that $i < j$. We define $z = \#^{jn^k-1-j'}\$\#^{jn^k-1}$ and obtain that $xz \notin L$ and $yz \in L$.

(e) Claim: Let $x \in A_i$ and $y \in B$. Then $x \not\equiv_L y$.
Let $x = \$\#^{in^k-1}\$\#^{i'}$ and $y = \$\#^{(k-2)n^k-1}\#^{j'}$ with $0 \leq i' \leq in^k - 1$ and $1 \leq j' \leq m+1$. We set $z = \#^{in^k-1-i'}\$\#^{in^k-1}$ and obtain that $xz \in L$ and $yz \notin L$.

(f) Claim: Let $x \in A_i$ and $y \in C$. Then $x \not\equiv_L y$.
Let $x = \$\#^{in^k-1}\$\#^{i'}$ and $y = \$\#^{j'}$ with $0 \leq i' \leq in^k - 1$ and $0 \leq j' \leq (k-2)n^k - 1$. Let $j'' \geq 0$ be the minimal integer such that $j' + j'' - m$ is a multiple of $m+1$. Then $j' + j'' = m + t(m+1)$ with $t \geq 0$. We now set $z = \#^{j''+(m+1)(in^k-1)}$ and observe that $yz = \$\#^{j'+j''+(m+1)(in^k-1)} = \$\#^{m+(m+1)(t+in^k-1)} \in L$, but $xz = \$\#^{in^k-1}\$\#^{i'}\#^{j''+(m+1)(in^k-1)} \notin L$, since $i' + j'' + (m+1)(in^k - 1) > in^k - 1$.

(g) Claim: Let $x \in B$ and $y \in C$. Then $x \not\equiv_L y$.
Let $x = \$\#^{(k-2)n^k-1}\#^{i'}$ and $y = \$\#^{j'}$ with $1 \leq i' \leq m+1$ and $0 \leq j' \leq (k-2)n^k - 1$. We define $z = \#^{(k-2)n^k-1-j'}\$\#^{(k-2)n^k-1}$ and obtain that $xz \notin L$ and $yz \in L$.

(h) Claim: Let $x \in A_1 \cup \ldots \cup A_{k-2} \cup B \cup C$ and $y \in D$. Then $x \not\equiv_L y$.
Let $x \in A_1 \cup \ldots \cup A_{k-2} \cup B \cup C$, $y = \epsilon$, and $z = \$\#^{m+(m+1)((k-2)n^k-1)}$. Then $xz \notin L$, since $m + (m+1)((k-2)n^k - 1) > (k-2)n^k - 1$, and $yz = z \in L$. Hence, $x \not\equiv_L y$.

Thus, $\text{index}(\equiv_L) \geq |A_1| + \ldots + |A_{k-2}| + |B| + |C| + |D| = n^k + 2n^k + \ldots + (k-2)n^k + (k-2)n^k + (m+1) + 1$ and the claim is proven. □