

Starry Starry Height

Colloquium, Ashoka University

Amaldev Manuel

School of Mathematics & Computer Science, IIT Goa

3 September 2026

Introduction

- Regular languages can be described by regular expressions.
 - $a^*b^* = \{a, b, aa, ab, bb, \dots\}$.
 - $(ab)^* + (ba)^* = \{\epsilon, ab, ba, ababbaba, \dots\}$.
- The operators used are union (+ or $\cup$), concatenation ($\cdot$), and the Kleene star ($*$).
- The **Kleene star** denotes zero or more repetitions of a pattern.
- **Question:** How deeply nested do these Kleene stars need to be to express a given regular language?
- This nesting depth is known as the **star height**.

Defining Star Height of an Expression

The star height $h(E)$ of a regular expression E over an alphabet Σ is defined inductively:

- $h(\emptyset) = 0$
- $h(\epsilon) = 0$
- $h(a) = 0$ for all $a \in \Sigma$
- $h(E \cup F) = \max(h(E), h(F))$
- $h(E \cdot F) = \max(h(E), h(F))$
- $h(E^*) = h(E) + 1$

Example: The expression $(a \cup b^*c)^*$ has a star height of 2.

Star Height of a Regular Language

- A regular language L can be represented by infinitely many different regular expressions ($e = e + e = e + e + e = \dots$).
- The **star height of a language** L , denoted $h(L)$, is the minimum star height among all regular expressions that define L :

$$h(L) = \min\{h(E) \mid L(E) = L\}$$

- **Example:** $L = (a^*b^*)^*$ has an expression of star height 2. However, the same language can be written as $(a \cup b)^*$, which has a star height of 1. Therefore, $h(L) = 1$.

Is Star Height Bounded?

- Early in the development of automata theory, it was unknown if there existed languages of arbitrary star height.
- Could every regular language be rewritten using a maximum of 1, 2, or 3 nested stars?

Is Star Height Bounded?

- **Eggan's Theorem (1963):** Star-height of a language = smallest *cycle rank* among all NFA accepting L .

Acyclic Graphs: If G contains no directed cycles, its cycle rank is zero:

$$r(G) = 0$$

Not Strongly Connected: If G is not strongly connected, its cycle rank is the maximum rank among its strongly connected components $G_1, G_2, \dots, G_k$:

$$r(G) = \max_{1 \leq i \leq k} r(G_i)$$

Strongly Connected: If G is strongly connected and contains at least one cycle, its cycle rank is determined by finding a vertex whose removal minimizes the rank of the remaining graph:

$$r(G) = 1 + \min_{v \in V} r(G - v)$$

- Eggan proved that over an alphabet of size $2^n - 1$, there exist languages of star height n .

Strict Hierarchy (1966)

- Eggen's proof relied on a growing alphabet size. What about fixed alphabets?
- **Dejean and Schützenberger (1966)** proved that the star height hierarchy is strict even for a **binary alphabet** (size 2).
- They demonstrated a family of languages L_n over $\{a, b\}$ such that $h(L_n) = n$.

Theorem

For $n > 0$, the language $L_n = \{w \in \{a, b\}^* : |w|_a \equiv |w|_b \pmod{2^n}\}$ has star height n .

- *Conclusion:* Star height is unbounded. There is no universal maximum nesting depth for all regular languages.

The Star Height Problem

The Core Question

Is the star height of a regular language **computable**?

- Suffices to answer: given $k > 0$ is $h(L) \leq k$?
- This became one of the most famous open problems in formal language theory for over two decades.

Hashiguchi's Theorem (1988)

- In 1988, Kosaburo Hashiguchi finally solved the Star Height Problem.
- **Result:** The star height of a regular language is computable.
- **Method:** Hashiguchi mapped the problem to the theory of finite semigroups and matrices.
- **Caveat:** The algorithm's complexity was astronomical (non-elementary), making it purely of theoretical interest and entirely impractical to compute even for very small automata.

Kirsten's Simplification (2005)

- In 2005, Daniel Kirsten provided a vastly simplified and more "automata-theoretic" proof.
- **Approach:** Reduced the star height problem to solving the limitedness problem for *nested distance desert automata* (NDDA).
- **Result:** Achieved an algorithm with $2^{2^{O(n)}}$ space complexity (double exponential space), a massive improvement over Hashiguchi's non-elementary complexity.
- Later works (e.g., Colcombet 2008, Toruńczyk 2011, Bojańczyk 2015) further refined these concepts.
- We will follow the exposition in [Bojańczyk 15].

String Expressions [Cohen 70]

Let $h, m \in \mathbb{N}_0$. A **string expression** of star height at most h and degree at most m is defined inductively on h :

- **Base Case** ($h = 0$): Finite union of words of length at most m .
- **Inductive Step** ($h \geq 1$): Finite union of expressions of the form:

$$w_1 \cdot E_1^* \cdot w_2 \cdot E_2^* \cdots w_\ell \cdot E_\ell^*$$

where $\ell \leq m$, and $|w_i| \leq m$ and A_i is a string expression of star height at most $h - 1$ for $1 \leq i \leq \ell$.

Every regular expression is equivalent to a string expression of the same height and some degree (By induction on the structure).

String Expressions

- Therefore it suffices to check if a given language L is accepted by a string expression of height h .
- For each $h, m > 0$ there are only finitely many string expressions of height h and degree m (By induction on h).
- Let $C_{h,m}$ denote the set of string expressions of height h and degree m .
- $C_{h,m} \subseteq C_{h,n}$ if $m \leq n$ (By definition).
- Consider the function $f_h : L \rightarrow \mathbb{N}$ given by

$$f_h : w \mapsto \min\{m : \exists E \in C_{h,m}, w \in L(E) \subseteq L\} \quad (1)$$

- L is accepted by a string expression of height h if, and only if, f_h is bounded.
- Thus deciding $\text{star-height} \leq h$, reduces to checking boundedness of f_h .

Kirsten's Idea

- Formulate an automaton model that can express f_h , for $h > 0$.
- Show that boundedness is decidable for the automata.
- The automaton model is known by various names
 - Nested Distance Desert Automaton [Kirsten 05]
 - B-Automaton [Bojańczyk-Colcombet 05]
 - Cost Automaton [Colcombet 09]
 - R-Automaton [Abdulla-Krcal-Yi 08]
- We will use the name Cost Automata.

Cost Automaton

- A nondeterministic finite automaton equipped with a finite set of counters indexed $1, \dots, n$ storing natural numbers.
- Initially all counters are zero.
- **Counter Operations:** *increment c , reset c or no-op.*
- **Hierarchical Reset Rule:** incrementing a counter resets all lower-indexed counters $0, 1, \dots, c - 1$ to zero.
- **Cost of a Run:** The maximal value attained by any counter during the run.
- **Cost of a Word w :** The smallest cost of a successful run on w .
- Cost Automata define functions from Σ^* to $\mathbb{N} \cup \{\infty\}$.

Examples of Functions defined by Cost Automata

Functions are computed upto boundedness.

Let $w \in \{a, b\}^*$

- $w \mapsto |w|$ (increment always)
- $w \mapsto |w|_a$ (increment on a)
- $w \mapsto \max(|w|_a, |w|_b)$ equivalent to $w \mapsto |w|_a + |w|_b = |w|$ (increment always)
- $w \mapsto \min(|w|_a, |w|_b)$ (increment on precisely all a 's or precisely all b 's)

Let $w \in \{a, b, c\}^*$

- $w \mapsto \max(|w|_a, |w|_b)$ (one counter each for a and b)

Decidability of Boundedness of Cost Automata

- Using Stabilisation Monoids [Kirsten 05, Colcombet 09, Toruńczyk 11]
- By reduction to Church-Synthesis Problem [Bojańczyk 15] (or Solving a Gale-Stewart game).

The Reduction from Star-height to Boundedness of Cost Automaton

Computing $f_h : w \mapsto \min\{m : \exists E \in C_{h,m}, w \in L(E) \subseteq L\}$

By Induction on h

- **Base case** ($h = 0$) : Count the length of the word if it is in L .
- **Inductive case** : By I.H. assume there are cost automata computing degree for height $h - 1$ and each quotient of L . On each word w
 - check if $w \in L$,
 - count the smallest $m > 0$ such that there is a factorisation

$$w = w_1 v_1 w_2 v_2 \dots w_k v_k \quad (2)$$

such that $k \leq m$, $|w_i| \leq m$, and

- compute the smallest degree for which v_i is in the Kleene closure of a string-expression of height $h - 1$.

The Generalized Star Height Problem

- What if we allow **complementation** ($\bar{}$) and **intersection** ($\cap$) alongside union, concatenation, and star?
- This gives us *generalized regular expressions*.
- The nesting depth of stars in these expressions is the **generalized star height**.
- **Status:** This is currently an **open problem**.
- It is not even known if there exists any regular language with a generalized star height greater than 1! (Star-free languages have height 0; everything else might just be height 1).

Summary

- **Definition:** Minimum Kleene star nesting depth to express a language.
- **1960s:** Eggen, Dejean & Schützenberger prove star height is unbounded and forms a strict hierarchy.
- **1988:** Hashiguchi proves the star height is computable (resolving the main problem).
- **2005:** Kirsten drastically simplifies the algorithm and reduces complexity using distance automata.
- **Today:** The Generalized Star Height problem remains a fascinating open mystery in theoretical computer science.