

# 形式语言与自动机理论

## 课程简介与基础知识

王春宇

chunyu@hit.edu.cn

计算学部

哈尔滨工业大学

2021 年 4 月

# 课程简介与基础知识

- 课程简介
- 基础知识



## 核心问题

计算机的基本能力和限制是什么？

- ① 究竟哪些问题, 可通过计算解决? — 可计算性理论
- ② 解决可计算的问题, 究竟需要多少资源? — 计算复杂性理论
- ③ 为了研究计算, 要使用哪些计算模型? — 形式语言与自动机理论

# 什么是自动机理论?

自动机理论: 研究抽象机器及其所能解决问题的理论.

- 图灵机
- 有限状态机
- 文法, 下推自动机

# 什么是形式语言?

形式语言: 经数学定义的语言.

		自然语言		形式语言	
		English	中文	化学分子式	C 语言
语言	字符	A,a,B,b,...	天, 地,...	A-Z,a-z,0-9...	A-Z,a-z,0-9...
	单词	apple	苹果	H <sub>2</sub> O	char
	句子	How're you?	早上好!	2H <sub>2</sub> +O <sub>2</sub> =2H <sub>2</sub> O	char a = 10;
	语法	Grammar	语法规则	精确定义的规则	

# 计算理论研究的意义

不同时间复杂度的计算时间比较 (假设每条指令  $1\ \mu\text{s}$ )

$T(n)$	$n = 10$	$n = 50$	$n = 100$	$n = 1000$
$\log_2 n$	$3.3\ \mu\text{s}$	$5.6\ \mu\text{s}$	$6.4\ \mu\text{s}$	$9.9\ \mu\text{s}$
$n$	$10.0\ \mu\text{s}$	$50.0\ \mu\text{s}$	$100.0\ \mu\text{s}$	$1.0\ \text{ms}$
$n^2$	$100.0\ \mu\text{s}$	$2.5\ \text{ms}$	$10.0\ \text{ms}$	$1.0\ \text{s}$
$2^n$	$1.0\ \text{ms}$	$35.8\ \text{y}$	$4.0\text{e}16\ \text{y}$	$3.4\text{e}287\ \text{y}$
$3^n$	$59.0\ \text{ms}$	$2.3\text{e}10\ \text{y}$	$1.6\text{e}34\ \text{y}$	$4.2\text{e}463\ \text{y}$
$n!$	$3.6\ \text{s}$	$9.7\text{e}50\ \text{y}$	$3.0\text{e}144\ \text{y}$	$1.3\text{e}2554\ \text{y}$

## 布尔可满足性 (SAT)

- 判断给定布尔公式是否可满足, 第一个被证明属于 NP 完全的问题
- 解决 3-SAT 的最好算法

---

$O(1.32793^n)$	Liu [2018] <sup>1</sup>
$O(1.3303^n)$	Makino, Tamaki and Yamamoto [2011, 2013]
$O(1.3334^n)$	Moser and Scheder [2011]
$O(1.439^n)$	Kutzkov and Scheder [2010]
$O(1.465^n)$	Scheder [2008]
$O(1.473^n)$	Brueggemann and Kern [2004]
$O(1.481^n)$	Dantsin, Goerdts, Hirsch, Kannan, <i>et al</i> [2002]
$O(1.497^n)$	Schiermeyer [1996]
$O(1.505^n)$	Kullmann [1999]
$O(1.6181^n)$	Monien and Speckenmeyer [1979, 1985]
$O(2^n)$	Brute-force search

---

<sup>1</sup>Sixue Liu. *Chain, Generalization of Covering Code, and Deterministic Algorithm for  $k$ -SAT*. In: 45th International Colloquium on Automata, Languages, and Programming.

## 运算最快的计算机 – [www.top500.org](http://www.top500.org)

- PFLOP: PetaFLOP, 每秒  $10^{15}$  次浮点运算

	Machine	PFLOPs
2020-11	Fugaku	442.01
2020-06	Fugaku	415.53
2019-06	Summit	148.60
2018-11	Summit	143.50
2018-06	Summit	122.30
2016-06	Sunway TaihuLight	93.01
2013-06	Tianhe-2	33.86
2012-06	Blue Gene/Q	16.32
2011-06	K computer	8.16



## 模拟计算 3-SAT 问题

- 计算机: 使用不同时期最快的
- 算法: 使用目前最快的 –  $O(1.32793^n)$

	Machine	PFLOPs	$n = 150$	$n = 200$	$n = 400$
2020-11	Fugaku	442.01	6.8 s	113.0 d	1.3e24 y
2020-06	Fugaku	415.53	7.2 s	120.2 d	1.4e24 y
2019-06	Summit	148.60	20.1 s	336.1 d	4.0e24 y
2018-11	Summit	143.50	20.8 s	348.1 d	4.1e24 y
2018-06	Summit	122.30	24.5 s	1.1 y	4.8e24 y
2016-06	Sunway TaihuLight	93.01	32.2 s	1.5 y	6.4e24 y
2013-06	Tianhe-2	33.86	1.5 m	4.1 y	1.7e25 y
2012-06	Blue Gene/Q	16.32	3.1 m	8.4 y	3.6e25 y
2011-06	K computer	8.16	6.1 m	16.8 y	7.3e25 y

## 模拟计算 3-SAT 问题

- 计算机: 使用目前最快的 – 442.01 PFLOPs
- 算法: 使用不同时期最快的

	Complexity	$n = 150$	$n = 200$	$n = 400$
2018	$T(1.32793^n)$	7.2 s	120.2 d	1.4e24 y
2013	$T(1.3303^n)$	9.4 s	172 d	2.9e24 y
2011	$T(1.3334^n)$	13.3 s	273.5 d	7.3e25 y
2010	$T(1.439^n)$	14.2 d	3.1e6 y	1.3e38 y
2008	$T(1.465^n)$	209.1 d	1.1e8 y	1.7e41 y
–	$T(2^n)$	1.1e20 y	1.3e35 y	2.0e95 y

# 课程内容

- 正则语言
  - 有穷自动机
  - 正则表达式
  - 正则语言的性质
- 上下文无关语言
  - 上下文无关文法
  - 下推自动机
  - 上下文无关语言的性质
- 计算导论
  - 图灵机及其扩展
  - 不可判定性

- *Introduction to Automata Theory, Languages, and Computation* – 3rd Ed.
  - 作者 John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman
  - <http://infolab.stanford.edu/~ullman/ialc.html>
  - 影印版《自动机理论、语言和计算导论》机械工业出版社

## 参考书

- *Models of Computation*, by Jeff Erickson — <http://algorithms.wtf>
- *Introduction to Theoretical Computer Science*, by Boaz Barak — <https://introtcs.org>
- *Introduction to the Theory of Computation*, 3ed, by Michael Sipser

# 课程简介与基础知识

- 课程简介
- 基础知识
  - 基本概念
  - 语言和问题
  - 形式化证明



# 基本概念

1. 字母表: 符号 (或字符) 的非空有穷集.

# 基本概念

1. 字母表: 符号 (或字符) 的非空有穷集.

$$\Sigma_1 = \{0, 1\},$$

$$\Sigma_2 = \{a, b, \dots, z\},$$

$$\Sigma_3 = \{x \mid x \text{ 是一个汉字}\}.$$



2. 字符串: 由某字母表中符号组成的有穷序列.

3. 空串: 记为  $\varepsilon$ , 有 0 个字符的串.

字母表  $\Sigma$  可以是任意的, 但都有  $\varepsilon \notin \Sigma$ .

4. 字符串的**长度**: 字符串中符号所占位置的个数, 记为  $|s|$ .

4. 字符串的**长度**: 字符串中符号所占位置的个数, 记为  $|u|$ .  
若字母表为  $\Sigma$ , 可**递归定义**为:

$$|w| = \begin{cases} 0 & w = \varepsilon \\ |x| + 1 & w = xa \end{cases},$$

其中  $a \in \Sigma$ ,  $w$  和  $x$  是  $\Sigma$  中字符组成的字符串.

4. 字符串的**长度**: 字符串中符号所占位置的个数, 记为  $|u|$ .  
若字母表为  $\Sigma$ , 可**递归定义**为:

$$|w| = \begin{cases} 0 & w = \varepsilon \\ |x| + 1 & w = xa \end{cases},$$

其中  $a \in \Sigma$ ,  $w$  和  $x$  是  $\Sigma$  中字符组成的字符串.

★. 符号使用的一般约定:

- 字母表:  $\Sigma, \Gamma, \Delta, \dots$
- 字符:  $a, b, c, \dots$
- 字符串:  $\dots, w, x, y, z$
- 集合:  $A, B, C, \dots$

5. 字符串  $x$  和  $y$  的**连接**: 将首尾相接得到新串的运算, 记为  $x \cdot y$  或  $xy$ .

5. 字符串  $x$  和  $y$  的**连接**: 将首尾相接得到新串的运算, 记为  $x \cdot y$  或  $xy$ .  
同样, 可递归定义为

$$x \cdot y = \begin{cases} x & y = \varepsilon \\ (x \cdot z)a & y = za \end{cases},$$

其中  $a \in \Sigma$ , 且  $x, y, z$  都是字符串.

5. 字符串  $x$  和  $y$  的**连接**: 将首尾相接得到新串的运算, 记为  $x \cdot y$  或  $xy$ . 同样, 可递归定义为

$$x \cdot y = \begin{cases} x & y = \varepsilon \\ (x \cdot z)a & y = za \end{cases},$$

其中  $a \in \Sigma$ , 且  $x, y, z$  都是字符串.

对任何字符串  $x$ , 有  $\varepsilon \cdot x = x \cdot \varepsilon = x$ .

连接运算的符号 “ $\cdot$ ” 一般省略.

6. 字符串  $x$  的  $n$  次幂( $n \geq 0$ ), 递归定义为

$$x^n = \begin{cases} \varepsilon & n = 0 \\ x^{n-1}x & n > 0 \end{cases} .$$



7. 集合  $A$  和  $B$  的连接, 记为  $A \cdot B$  或  $AB$ , 定义为

$$A \cdot B = \{ w \mid w = x \cdot y, x \in A \text{ 且 } y \in B \}.$$

8. 集合  $A$  的  $n$  次幂( $n \geq 0$ ), 递归定义为

$$A^n = \begin{cases} \{\varepsilon\} & n = 0 \\ A^{n-1}A & n \geq 1 \end{cases} .$$

8. 集合  $A$  的  $n$  次幂( $n \geq 0$ ), 递归定义为

$$A^n = \begin{cases} \{\varepsilon\} & n = 0 \\ A^{n-1}A & n \geq 1 \end{cases} .$$

那么, 若  $\Sigma$  为字母表, 则  $\Sigma^n$  为  $\Sigma$  上长度为  $n$  的字符串集合.  
如果  $\Sigma = \{0, 1\}$ , 有

$$\Sigma^0 = \{\varepsilon\}$$

$$\Sigma^1 = \{0, 1\}$$

$$\Sigma^2 = \{00, 01, 10, 11\}$$

$$\Sigma^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}$$

$\vdots$

9. 克林闭包(*Kleene Closure*):

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i.$$

9. 克林闭包(*Kleene Closure*):

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i.$$

10. 正闭包(*Positive Closure*):

$$\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i.$$

显然,

$$\Sigma^* = \Sigma^+ \cup \{\varepsilon\}.$$

9. 克林闭包(Kleene Closure):

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i.$$

10. 正闭包(Positive Closure):

$$\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i.$$

显然,

$$\Sigma^* = \Sigma^+ \cup \{\varepsilon\}.$$

其他概念如有向图, 树, 字符串的前缀, 后缀等定义这里省略.

# 语言

## 定义

若  $\Sigma$  为字母表且  $\forall L \subseteq \Sigma^*$ , 则  $L$  称为字母表  $\Sigma$  上的语言.

# 语言

## 定义

若  $\Sigma$  为字母表且  $\forall L \subseteq \Sigma^*$ , 则  $L$  称为字母表  $\Sigma$  上的语言.

- 自然语言, 程序设计语言等
- $\{0^n 1^n \mid n \geq 0\}$
- The set of strings of 0's and 1's with an equal number of each:

$\{\epsilon, 01, 10, 0011, 0101, 1100, \dots\}$

- $\emptyset$ ,  $\{\epsilon\}$  和  $\Sigma^*$  都是任意字母表  $\Sigma$  上的语言, 但注意  $\emptyset \neq \{\epsilon\}$



# 语言

## 定义

若  $\Sigma$  为字母表且  $\forall L \subseteq \Sigma^*$ , 则  $L$  称为字母表  $\Sigma$  上的语言.

- 自然语言, 程序设计语言等
- $\{0^n 1^n \mid n \geq 0\}$
- The set of strings of 0's and 1's with an equal number of each:

$\{\epsilon, 01, 10, 0011, 0101, 1100, \dots\}$

- $\emptyset$ ,  $\{\epsilon\}$  和  $\Sigma^*$  都是任意字母表  $\Sigma$  上的语言, 但注意  $\emptyset \neq \{\epsilon\}$

## 关于语言

唯一重要的约束就是所有字母表都是有穷的.

# 问题

## 典型问题

判断给定的字符串  $w$  是否属于某个具体的语言  $L$ ,

$$w \in L?$$

- 任何所谓问题, 都可以转为语言成员性的问题
- 语言和问题其实是相同的

## 形式化证明: 演绎法, 归纳法和反证法

例 1. 若  $x$  和  $y$  是  $\Sigma$  上的字符串, 请证明  $|xy| = |x| + |y|$ .

证明: 通过对  $|y|$  的归纳来证明

① 基础: 当  $|y| = 0$ , 即  $y = \varepsilon$

$$\begin{aligned}|x\varepsilon| &= |x| \\ &= |x| + |\varepsilon|\end{aligned}$$

连接的定义  
长度的定义

② 递推: 假设  $|y| = n$  ( $n \geq 0$ ) 时命题成立,  
那么当  $|y| = n + 1$ , 即  $y = wa$

$$\begin{aligned}|x(wa)| &= |(xw)a| \\ &= |xw| + 1 \\ &= |x| + |w| + 1 \\ &= |x| + |wa|\end{aligned}$$

连接的定义  
长度的定义  
归纳假设  
长度的定义



# 形式化证明: 演绎法, 归纳法和反证法

例 1. 若  $x$  和  $y$  是  $\Sigma$  上的字符串, 请证明  $|xy| = |x| + |y|$ .

证明: 通过对  $y$  的结构归纳来证明

① 基础:  $y = \varepsilon$  时

$$\begin{aligned}|x\varepsilon| &= |x| \\ &= |x| + |\varepsilon|\end{aligned}$$

连接的定义  
长度的定义

② 递推: 假设  $y = w$  ( $w \in \Sigma^*$ ) 时命题成立,  
那么当  $y = wa$  时

$$\begin{aligned}|x(wa)| &= |(xw)a| \\ &= |xw| + 1 \\ &= |x| + |w| + 1 \\ &= |x| + |wa|\end{aligned}$$

连接的定义  
长度的定义  
归纳假设  
长度的定义





# 哈爾濱工業大學

HARBIN INSTITUTE OF TECHNOLOGY

chunyu@hit.edu.cn  
<http://nclab.net/~chunyu>

