離散最適化基礎論 (2025 年後学期)

高速指数時間アルゴリズム

第 13 回

強指数時間仮説と細粒度計算複雑性

岡本 吉央　(電気通信大学)

okamotoy@uec.ac.jp

2026 年 1 月 27 日

最終更新：2026 年 1 月 27 日　22:55

スケジュール (前半)

1. 高速指数時間アルゴリズムの考え方 (10/7)

＊ 休み (体育祭) (10/14)

2. 分枝アルゴリズム：基礎 (10/21)
3. 分枝アルゴリズム：高速化 (10/28)
4. 分枝アルゴリズム：測度統治法 (11/4)
5. 動的計画法：基礎 (11/11)
6. 動的計画法：例 (11/18)

スケジュール (後半)

7.	包除原理：原理	(11/25)
∗	休み (秋ターム試験)	(12/2)
8.	包除原理：例	(12/9)
9.	部分集合たたみ込み：原理	(12/16)
∗	休み (出張)	(12/23)
∗	休み (冬季休業)	(12/30)
10.	部分集合たたみ込み：例	(1/6)
11.	指数時間仮説：原理	(1/13)
12.	指数時間仮説：例	(1/20)
13.	**強指数時間仮説と細粒度計算複雑性**	(1/27)
∗	休み (修士論文発表会)	(2/3)

本日の内容

前回，前々回に示したこと

- 最大独立集合問題は $2^{o(n)}$ 時間で解けない
- (ただし，指数時間仮説が正しいという仮定のもとで)

言えていないこと

- 最大独立集合問題は $O^*(1.1^n)$ 時間で解けない

$\rightsquigarrow$ おそらく，指数時間仮説より強い仮定が必要

$\rightsquigarrow$ どんな仮定？

$\rightsquigarrow$ 強指数時間仮説

本日の内容

1. **強指数時間仮説**

2. 強指数時間仮説と細粒度計算複雑性

- R. Impagliazzo, R. Paturi, On the complexity of k-SAT. *Journal of Computer and System Sciences* 62 (2001) pp. 367–375.
- R. Impagliazzo, R. Paturi, F. Zane, Which problems have strongly exponential complexity? *Journal of Computer and System Sciences* 63 (2001) pp. 512–530.
- M. Cygan, H. Dell, D. Lokshtanov, D. Marx, J. Nederlof, Y. Okamoto, R. Paturi, S. Saurabh, M. Wahlström, On problems as hard as CNF-SAT. *ACM Transactions on Algorithms* 12 (2016) pp. 41:1–41:24.

定義：連言標準形

論理式 φ が **連言標準形** で表されているとは,
φ が「リテラルの OR の AND」で書かれていること

$$\varphi = \underbrace{\underbrace{(x_1 \vee \overline{x_2})}_{\text{リテラルの OR}} \wedge \underbrace{(x_1 \vee \overline{x_3} \vee x_4)}_{\text{リテラルの OR}}}_{\text{リテラルの OR の AND}}$$

連言標準形：conjunctive normal form (CNF)

$\boxed{\text{用語}}$：節 (clause) ＝ リテラルの OR

節 C のサイズ ＝ C が含むリテラルの数

[復習] CNF-SAT

問題：CNF-SAT

入力： 連言標準形で表された論理式 φ

出力： φ が充足可能である $\Rightarrow$ Yes

φ が充足可能ではない $\Rightarrow$ No

$$\varphi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned} \varphi[x_1 = 1, x_2 = 1, x_3 = 0, x_4 = 0] &= (1 \vee \overline{1}) \wedge (1 \vee \overline{0} \vee 0) \\ &= 1 \end{aligned}$$

$\leadsto$ Yes

注：CNF-SAT は NP 完全 (Cook '71; Levin '73)

[復習] k-SAT

$k \geq 1$ は正整数

問題：k-SAT

入力： 連言標準形で表された論理式 φ で，
各節のサイズが k 以下であるもの

出力： φ が充足可能である $\Rightarrow$ Yes
φ が充足可能ではない $\Rightarrow$ No

$$\varphi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3} \vee x_4)$$

<u>注</u>：$k \geq 3$ のとき，k-SAT は NP 完全 (Karp '72)

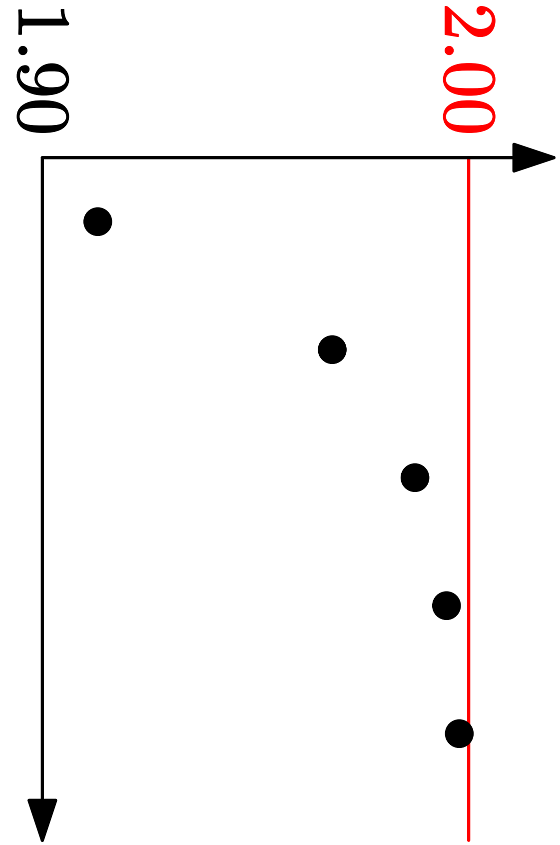
k-SAT のアルゴリズム

k	計算量	漸化式
$\mathbf{3}$	$O^*(1.9130^n)$	$T(n) \leq \mathbf{7}\,T(n-\mathbf{3})$

k-SAT のアルゴリズム

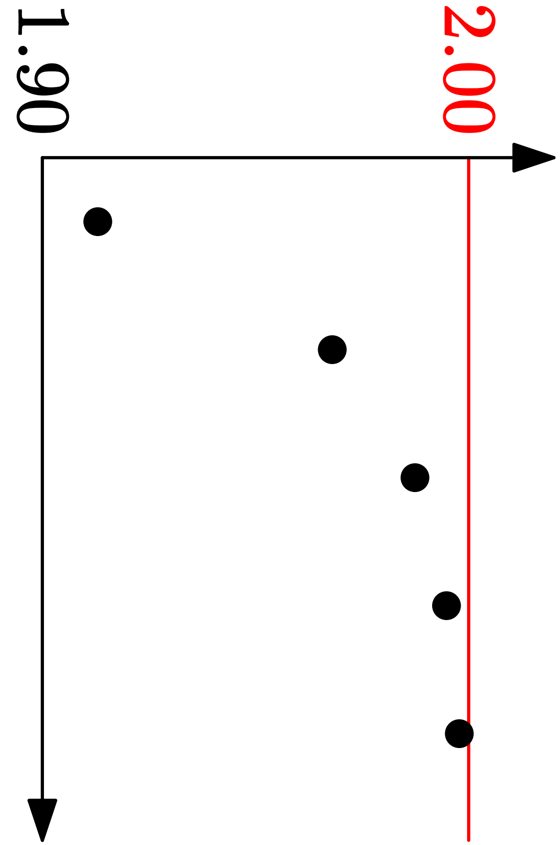
k	計算量	漸化式
3	$O^*(1.9130^n)$	$T(n) \le 7\,T(n-3)$
4	$O^*(1.9680^n)$	$T(n) \le 15\,T(n-4)$
5	$O^*(1.9874^n)$	$T(n) \le 31\,T(n-5)$
6	$O^*(1.9948^n)$	$T(n) \le 63\,T(n-6)$
7	$O^*(1.9978^n)$	$T(n) \le 127\,T(n-7)$

k	計算量	漸化式
3	$O^*(1.9130^n)$	$T(n) \leq 7\,T(n-3)$
4	$O^*(1.9680^n)$	$T(n) \leq 15\,T(n-4)$
5	$O^*(1.9874^n)$	$T(n) \leq 31\,T(n-5)$
6	$O^*(1.9948^n)$	$T(n) \leq 63\,T(n-6)$
7	$O^*(1.9978^n)$	$T(n) \leq 127\,T(n-7)$



k-SAT のアルゴリズム

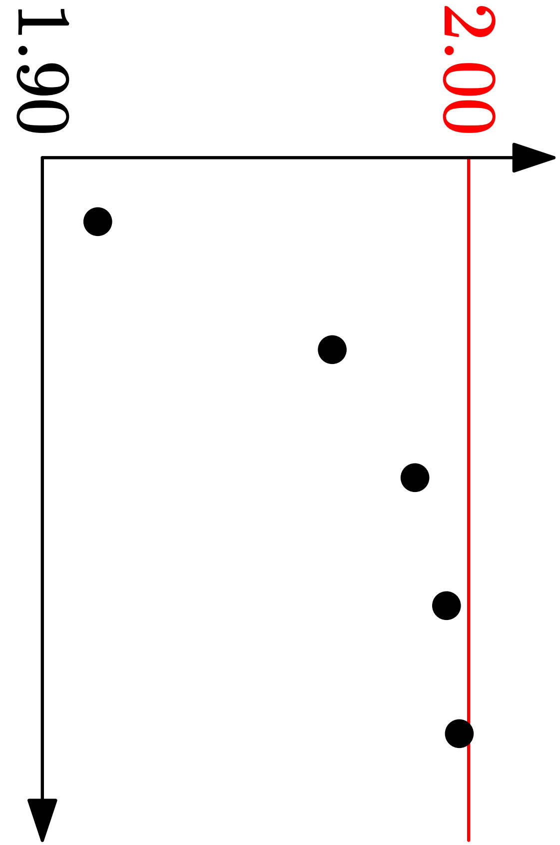
k	計算量	漸化式
3	$O^*(1.9130^n)$	$T(n) \leq 7\,T(n-3)$
4	$O^*(1.9680^n)$	$T(n) \leq 15\,T(n-4)$
5	$O^*(1.9874^n)$	$T(n) \leq 31\,T(n-5)$
6	$O^*(1.9948^n)$	$T(n) \leq 63\,T(n-6)$
7	$O^*(1.9978^n)$	$T(n) \leq 127\,T(n-7)$



k $\quad O^*\big(\underbrace{((2^k-1)^{1/k}}_{\to 2 \quad (k \to \infty)})^n\big)$

k-SAT のアルゴリズム

k	計算量	漸化式
3	$O^*(1.9130^n)$	$T(n) \le 7\,T(n-3)$
4	$O^*(1.9680^n)$	$T(n) \le 15\,T(n-4)$
5	$O^*(1.9874^n)$	$T(n) \le 31\,T(n-5)$
6	$O^*(1.9948^n)$	$T(n) \le 63\,T(n-6)$
7	$O^*(1.9978^n)$	$T(n) \le 127\,T(n-7)$



k $\quad O^*((\underbrace{(2^k-1)^{1/k}}_{\to 2 \quad (k \to \infty)})^n) = O^*((2^{1-c/(k2^k)})^n)$

現在最速の k-SAT アルゴリズムの計算量 $= O^*((2^{1-c/k})^n)$

次は未解決問題

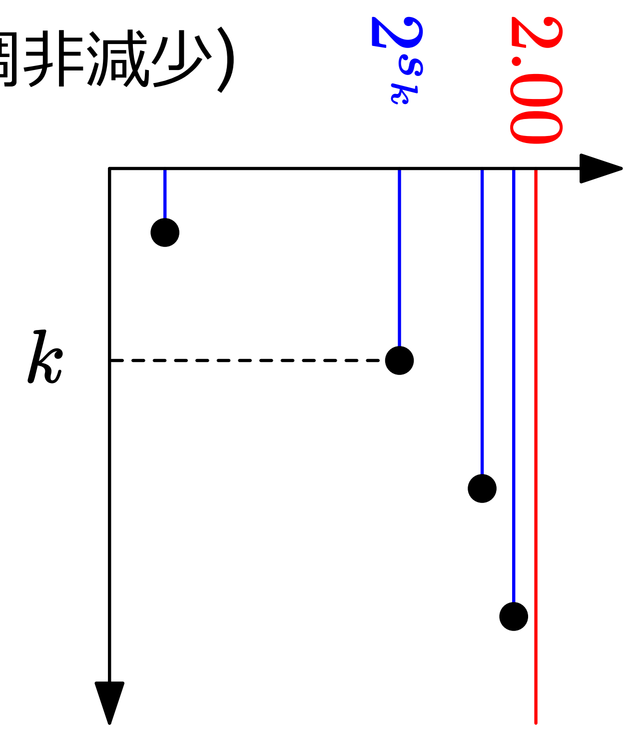
強指数時間仮説 (strong exponential-time hypothesis, SETH)

$s_k = \inf\{s \mid k$-SAT が $O^*(2^{sn})$ 時間で解ける$\}$ のとき

$$\lim_{k \to \infty} s_k = 1$$

注：この $\lim$ は存在する (数列 $\{s_k\}$ は単調非減少)

直感：CNF-SAT は $O^*((2-\delta)^n)$ 時間で解けない $(\delta > 0)$



CNF-SAT と同程度に難しい問題

性質：CNF-SAT と同程度に難しい問題

次は同値

- $\exists\, \delta > 0$：**CNF-SAT** が $O^*((2-\delta)^n)$ 時間で解ける ($n =$ 変数の数)
- $\exists\, \delta > 0$：**集合横断問題** が $O^*((2-\delta)^n)$ 時間で解ける ($n =$ 台集合の要素数)
- $\exists\, \delta > 0$：**集合分離問題** が $O^*((2-\delta)^n)$ 時間で解ける ($n =$ 台集合の要素数)

(Cygan, Dell, Lokshtanov, Marx, Nederlof, Okamoto, Paturi, Saurabh, Wahlström ’16)

証明は省略，問題のみ紹介

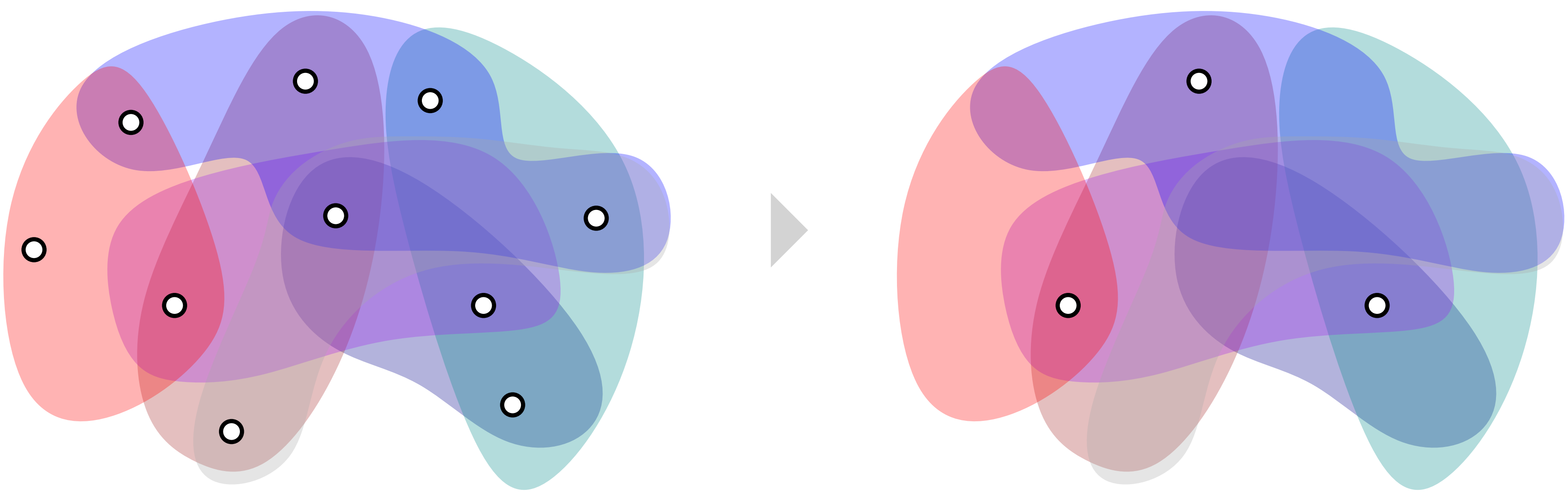
集合横断問題

問題：集合横断問題

入力： 有限集合 V，集合族 $\mathcal{S} \subseteq 2^V$

出力： 次の性質を満たす集合 $H \subseteq V$ で要素数最小のもの

- $\forall X \in \mathcal{S} : H \cap X \neq \emptyset$



注：しらみつぶし $\rightsquigarrow O^*(2^n)$ 時間で解ける $(n = |V|)$

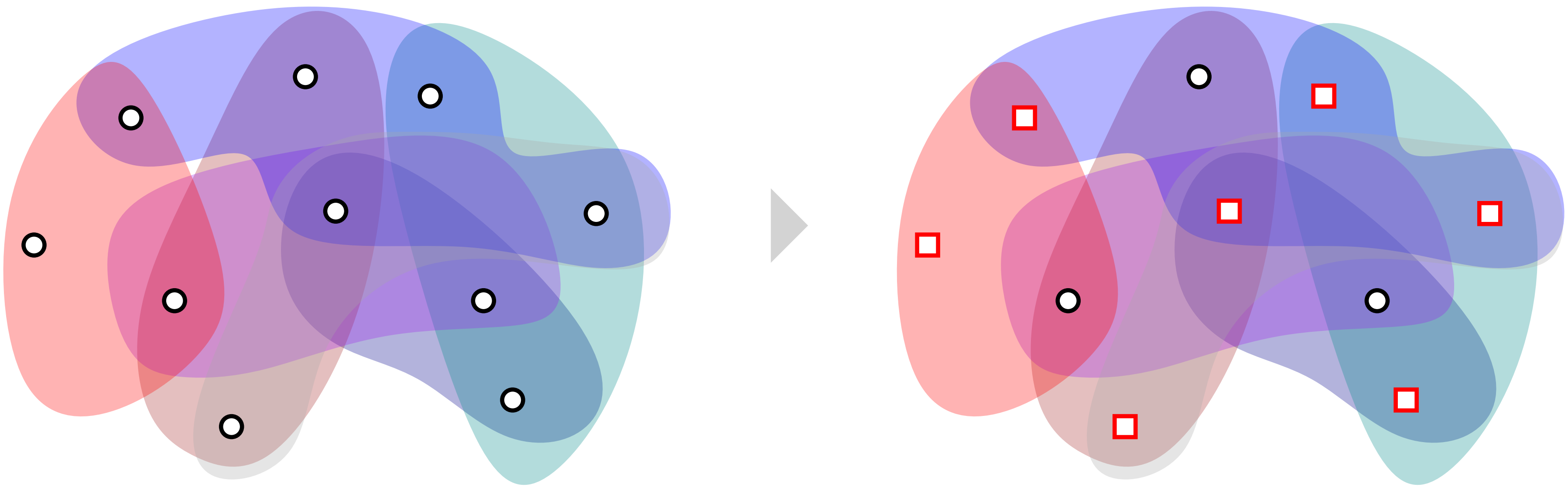
集合分離問題

問題：集合分離問題

入力： 有限集合 V，集合族 $\mathcal{S} \subseteq 2^V$

出力： 次の性質を満たす集合 $H \subseteq V$ の存在 (Yes/No)

- $\forall\, X \in \mathcal{S} : H \cap X \neq \emptyset$ かつ $(V - H) \cap X \neq \emptyset$



注：しらみつぶし $\rightsquigarrow O^*(2^n)$ 時間で解ける ($n = |V|$)

SETH $\Rightarrow$ ETH

k-SAT に対する疎化補題を使うと，次を証明できる

事実：強指数時間仮説と指数時間仮説

強指数時間仮説が正しい
(CNF-SAT は $O^*((2-\delta)^n)$ 時間で解けない)

$\Rightarrow$ 指数時間仮説が正しい
(3-SAT は $2^{o(n)}$ 時間で解けない)

k-SAT に対する疎化補題を使うと，次を証明できる

事実：強指数時間仮説と指数時間仮説

強指数時間仮説が正しい
(CNF-SAT は $O^*((2-\delta)^n)$ 時間で解けない)

$\Rightarrow$ 指数時間仮説が正しい
(3-SAT は $2^{o(n)}$ 時間で解けない)

$\Rightarrow$ P $\neq$ NP
(3-SAT は多項式時間で解けない)

k-SAT に対する疎化補題

定理：k-SAT に対する疎化補題 (sparsification lemma)

$\forall\, k \geq 3, \exists$ 関数 g：次のアルゴリズムが存在

入力： k-SAT の入力 φ (変数数 $= n$)，正整数 ℓ

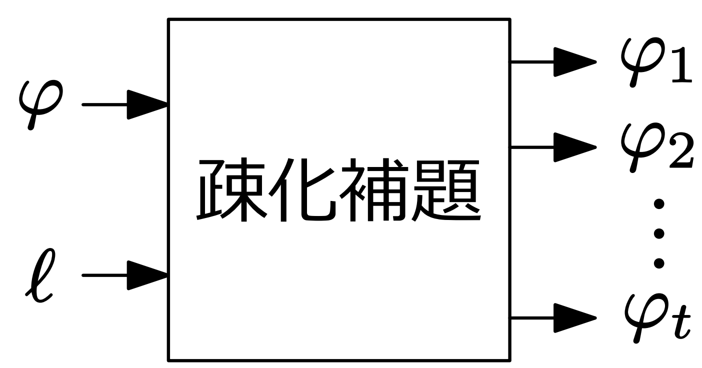
出力： t 個の k-SAT 入力 $\varphi_1, \varphi_2, \ldots, \varphi_t$

要請：
1. $t \leq 2^{n/\ell}$
2. φ が充足可能 $\Leftrightarrow$ ある φ_i が充足可能
3. φ_i の節は必ず φ の節
4. φ_i の中には各変数が $g(\ell)$ 回しか現れない

ここで，アルゴリズムの計算量は $O^*(2^{n/\ell})$ である

$$\varphi = \bigvee_{i=1}^{t} \varphi_i$$

節数 m $\leq g(\ell) n$



k-SAT に対する疎化補題：簡素版

定理：k-SAT に対する疎化補題 (簡素版)

$\forall\, k \geq 3$：次のアルゴリズムが存在

入力： k-SAT の入力 φ (変数数 $= n$)

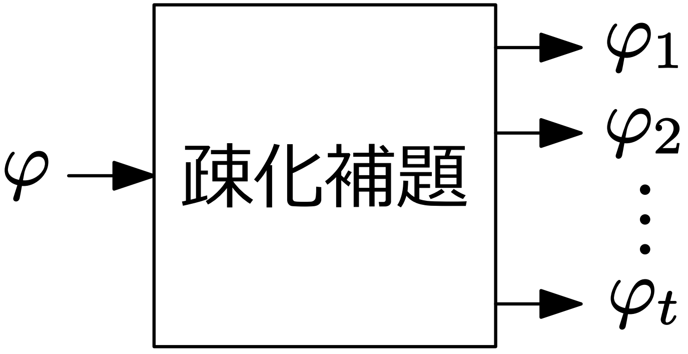
出力： t 個の k-SAT 入力 $\varphi_1, \varphi_2, \ldots, \varphi_t$

要請：
1. $t = 2^{o(n)}$
2. φ が充足可能 $\Leftrightarrow$ ある φ_i が充足可能
3. φ_i の節は必ず φ の節
4. φ_i の節の数 $= O(n)$

ここで，アルゴリズムの計算量は $2^{o(n)}$ である

$$\varphi = \bigvee_{i=1}^{t} \varphi_i$$

節数	m	$O(n)$

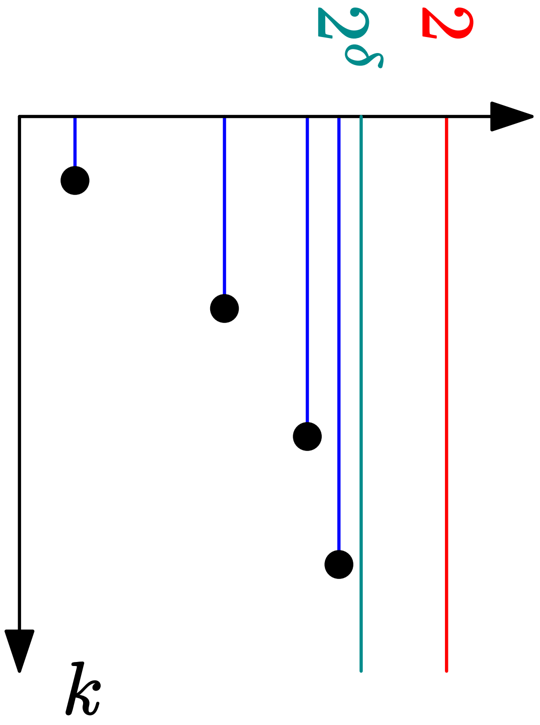


SETH $\Rightarrow$ ETH：証明 (1)

対偶を証明する：3-SAT が $2^{o(n)}$ 時間で解けると仮定

対偶を証明する：3-SAT が $2^{o(n)}$ 時間で解けると仮定

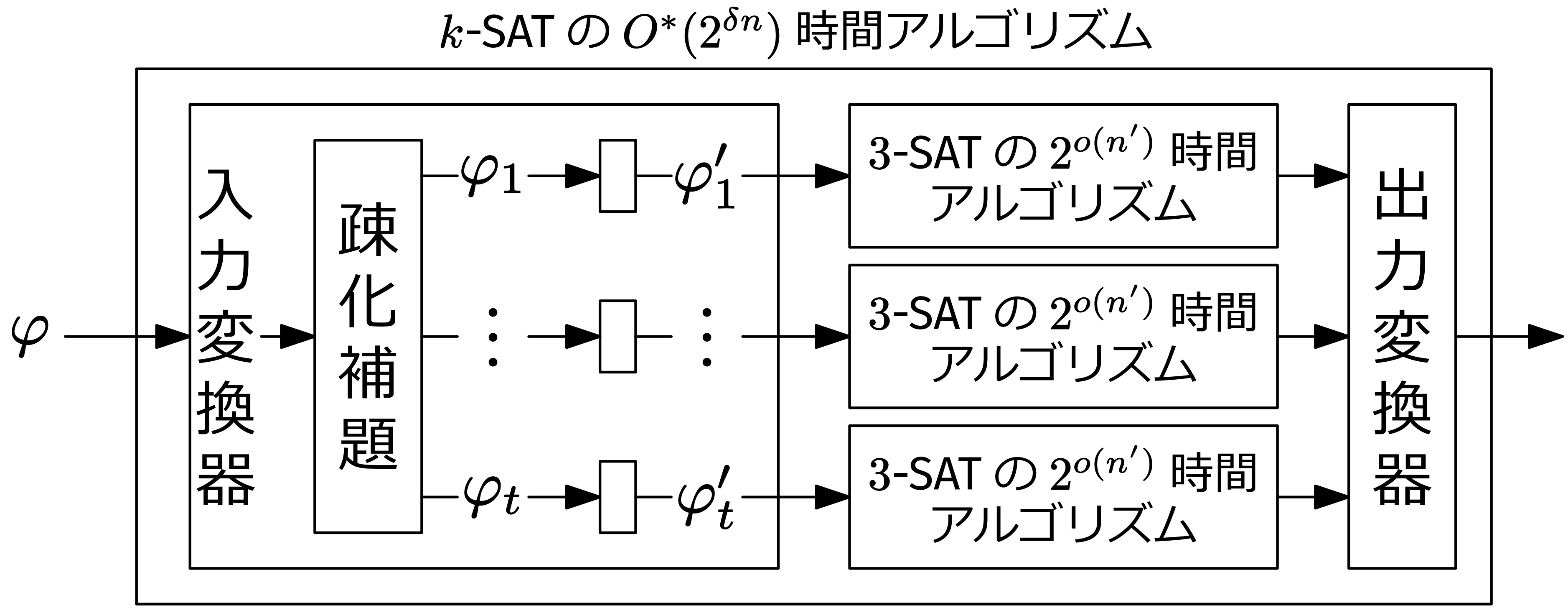
$\boxed{\text{目標}}$ $\exists\, \delta < 1, \forall\, k$: k-SAT が $O^*(2^{\delta n})$ 時間で解ける



SETH ⇒ ETH：証明 (1)

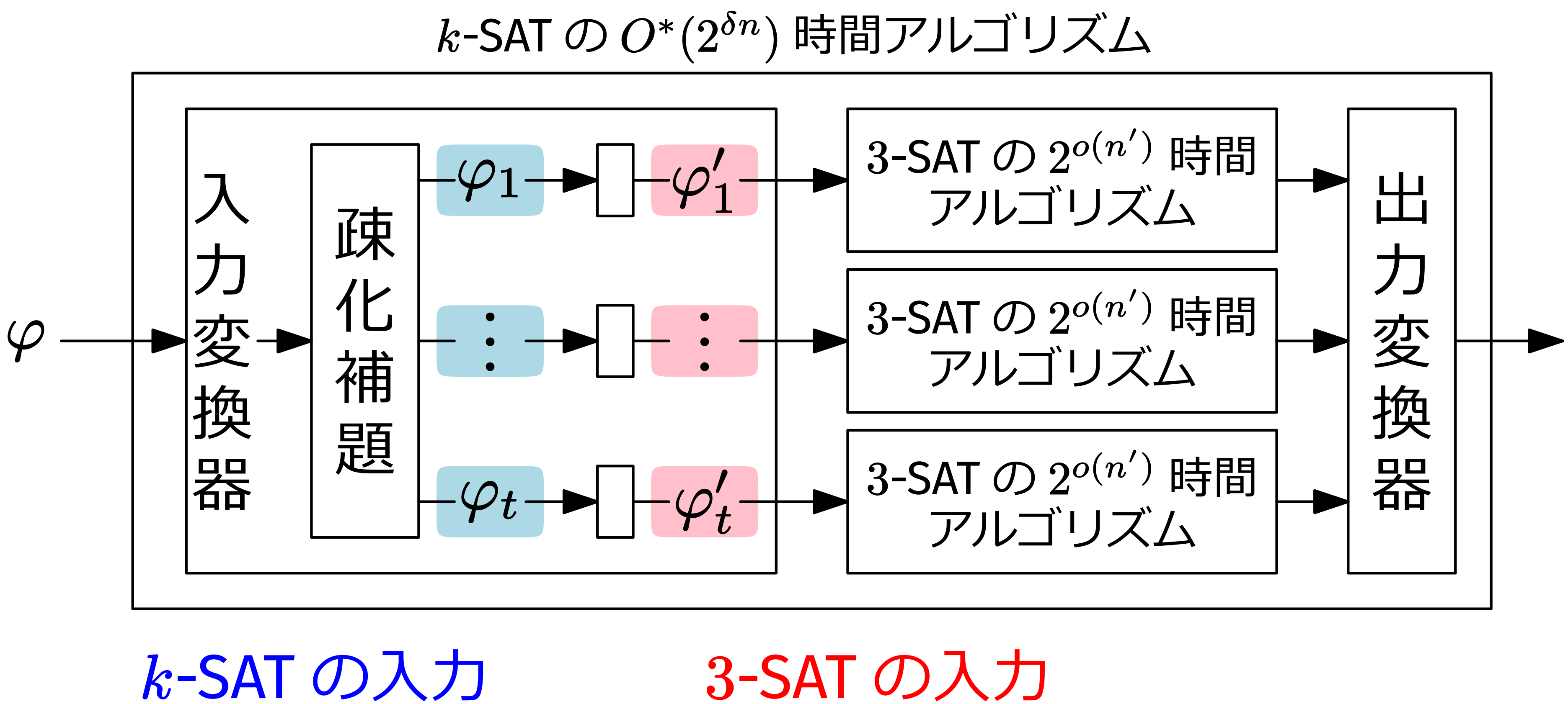
対偶を証明する：3-SAT が $2^{o(n)}$ 時間で解けると仮定

目標 $\exists\,\delta < 1, \forall\, k$: k-SAT が $O^*(2^{\delta n})$ 時間で解ける



対偶を証明する：3-SAT が $2^{o(n)}$ 時間で解けると仮定

$\boxed{\text{目標}}$ $\exists\, \delta < 1, \forall\, k$：$k$-SAT が $O^*(2^{\delta n})$ 時間で解ける



SETH $\Rightarrow$ ETH：証明 (2)

k-SAT の入力における各節に対して，次を実行

$k = 4$ のとき $\quad (\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4)$

$$\Leftrightarrow (\ell_1 \vee \ell_2 \vee y_1) \wedge (\overline{y_1} \vee \ell_3 \vee \ell_4)$$

SETH $\Rightarrow$ ETH：証明 (2)

k-SAT の入力における各節に対して，次を実行

$k=4$ のとき $(\ell_1 \lor \ell_2 \lor \ell_3 \lor \ell_4)$

$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor \ell_4)$$

$k=5$ のとき $(\ell_1 \lor \ell_2 \lor \ell_3 \lor \ell_4 \lor \ell_5)$

$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor \ell_4 \lor \ell_5)$$

$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor y_2) \land (\overline{y_2} \lor \ell_4 \lor \ell_5)$$

SETH ⇒ ETH：証明 (2)

k-SAT の入力における各節に対して，次を実行

$k = 4$ のとき　$(\ell_1 \lor \ell_2 \lor \ell_3 \lor \ell_4)$

$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor \ell_4)$$

$k = 5$ のとき　$(\ell_1 \lor \ell_2 \lor \ell_3 \lor \ell_4 \lor \ell_5)$

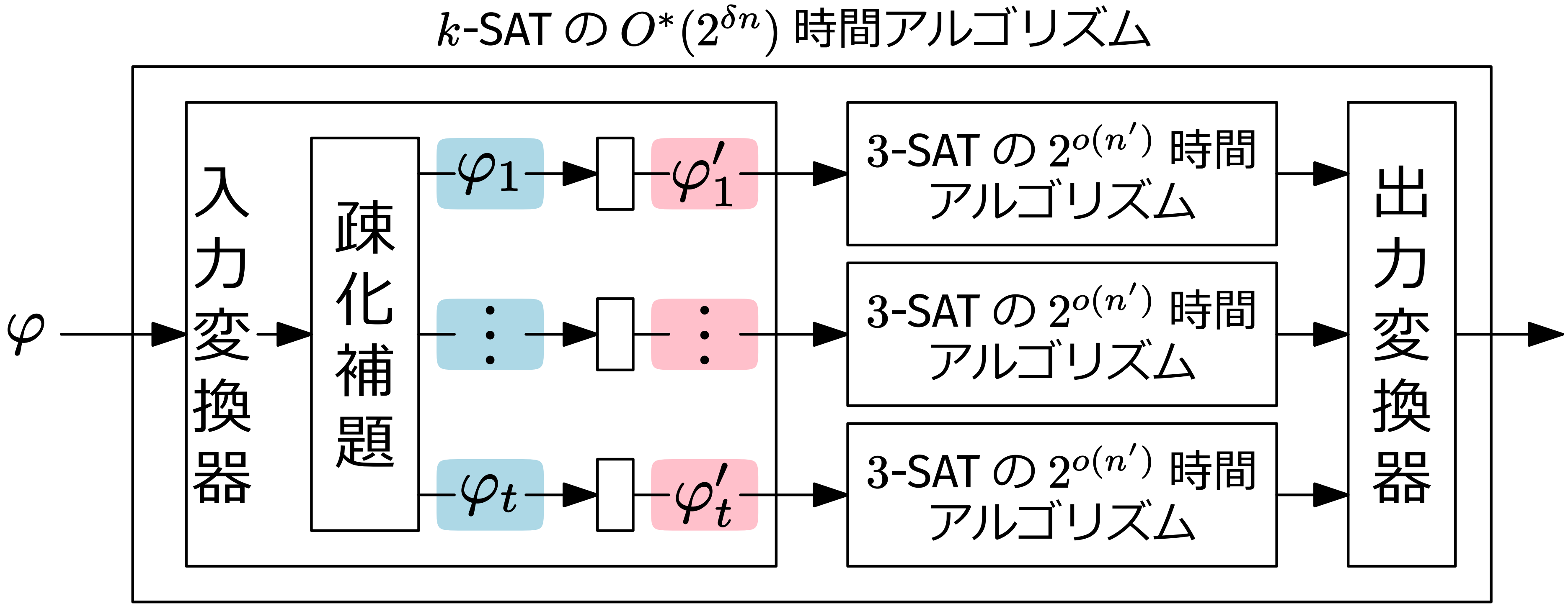
$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor \ell_4 \lor \ell_5)$$

$$\Leftrightarrow (\ell_1 \lor \ell_2 \lor y_1) \land (\overline{y_1} \lor \ell_3 \lor y_2) \land (\overline{y_2} \lor \ell_4 \lor \ell_5)$$

一般に，リテラルを k 個持つ節 1 個

$\rightsquigarrow$ リテラルを 3 個持つ節 $k-2$ 個，追加変数 $k-3$ 個

SETH ⇒ ETH：証明 (3)



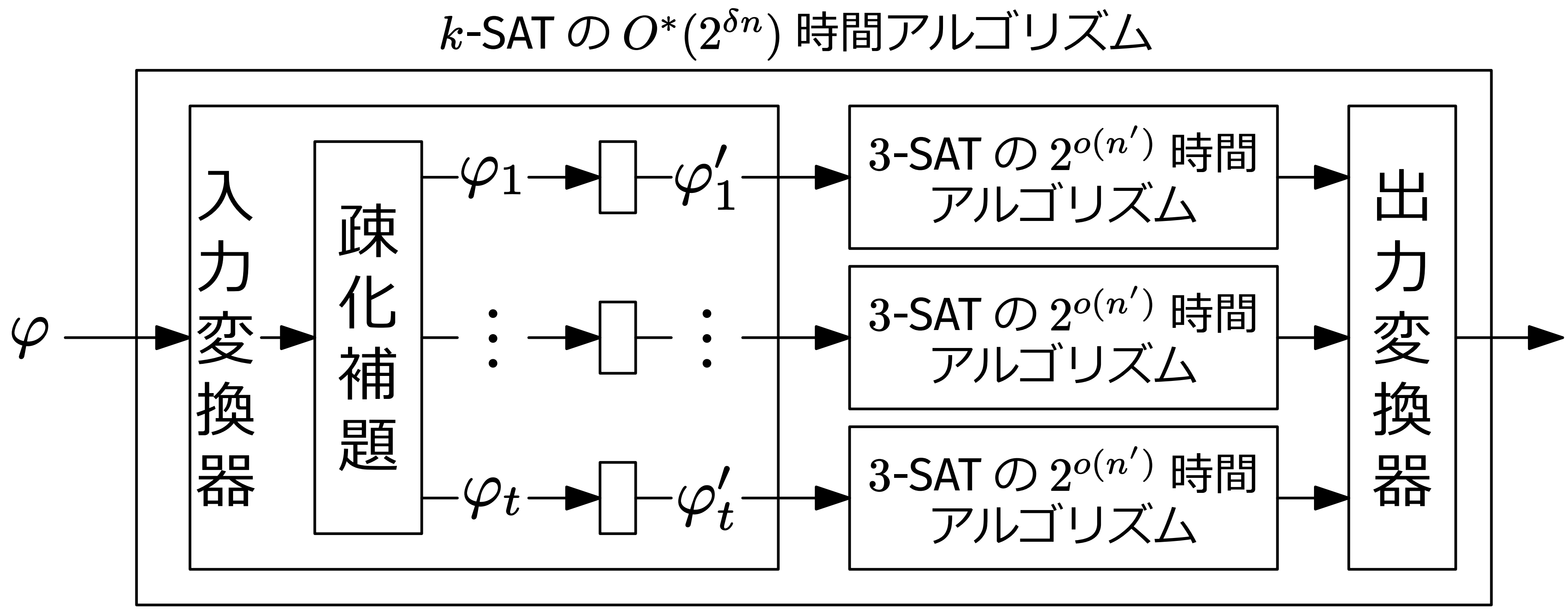
k-SAT の入力

- 変数の数 $\leq n$
- 節の数 $\leq m = O(n)$

3-SAT の入力

- 変数の数 $\leq n + (k-3)m = O(n)$
- 節の数 $\leq (k-2)m = O(n)$

SETH $\Rightarrow$ ETH : 証明 (3)



$2^{o(n)}$ 多項式 $2^{o(n)}$ $2^{o(n)}$

$t = 2^{o(n)}$

$\therefore$ 全体の計算量 $= 2^{o(n)} = O^*(2^{\delta n})$ $\square$

可能と不可能：指数時間仮説

問題	可能	不可能
最大独立集合問題	$O^*(1.2228^n)$	$2^{o(n)}$
3-SAT	$O^*(1.8393^n)$	$2^{o(n)}$
巡回セールスマン問題	$O^*(2^n)$	$2^{o(n)}$
最小被覆問題	$O^*(2^n)$	$2^{o(n)}$
彩色問題	$O^*(2^n)$	$2^{o(n)}$
最小シュタイナー木問題	$O^*(2^{\|K\|})$	$2^{o(\|K\|)}$

指数時間仮説
の成立を仮定

可能と不可能：強指数時間仮説

問題	可能	不可能
最大独立集合問題	$O^*(1.2228^n)$	???
3-SAT	$O^*(1.8393^n)$	???
巡回セールスマン問題	$O^*(2^n)$	???
最小被覆問題	$O^*(2^n)$	???
彩色問題	$O^*(2^n)$	???
最小シュタイナー木問題	$O^*(2^{\|K\|})$	???

強指数時間仮説
の成立を仮定

本日の内容

1. 強指数時間仮説
2. **強指数時間仮説と細粒度計算複雑性**

- R. Williams, A new algorithm for optimal 2-constraint satisfaction and its implications. *Theoretical Computer Science* 348 (2005) pp. 357–365,
- L. Roditty, V. Vassilevska Williams, Fast approximation algorithms for the diameter and radius of sparse graphs. *Proceedings of STOC 2013* (2013) pp. 515–524.
- A. Backurs, P. Indyk, Edit distance cannot be computer in strongly subquadratic time (unless SETH is false). *Proceedings of STOC 2015* (2015) pp. 51–58.

細粒度計算複雑性

細粒度計算複雑性とは？
(fine-grained computational complexity)

多項式時間アルゴリズムの計算量 $O(n^c)$ の次数 c を気にする計算複雑性における考え方

問題	可能	不可能
グラフ：最小全域木問題	$O(m \log n)$ Kruskal 法	???
グラフ：最短路問題 (非負重み)	$O(m + n \log n)$ Dijkstra 法	???
文字列：編集距離	$O(n^2)$ 動的計画法	???

直交ベクトル問題

入力： d 次元 0/1 ベクトルの集合 $A, B \subseteq \{0,1\}^d$

出力： $\exists\, a \in A, b \in B : a \bullet b = 0 \Rightarrow$ Yes

$\forall\, a \in A, b \in B : a \bullet b \neq 0 \Rightarrow$ No

$$A = \{(0,1,0,0,1),(1,0,1,0,1),(0,0,1,1,0)\} \qquad d = 5$$

$$B = \{(1,1,0,1,0),(0,1,0,0,1),(0,1,1,0,1)\}$$

直交ベクトル問題

直交ベクトル問題

入力： d 次元 0/1 ベクトルの集合 $A, B \subseteq \{0,1\}^d$

出力： $\exists\, a \in A, b \in B : a \bullet b = 0 \Rightarrow$ Yes

$\forall\, a \in A, b \in B : a \bullet b \neq 0 \Rightarrow$ No

$$A = \{(0,1,0,0,1), (1,0,1,0,1), \underbrace{(0,0,1,1,0)}_{=a}\} \qquad d = 5$$

$$B = \{(1,1,0,1,0), \underbrace{(0,1,0,0,1)}_{=b}, (0,1,1,0,1)\}$$

$$a \bullet b = \begin{array}{c}(0,0,1,1,0)\\ \bullet \\ (0,1,0,0,1)\end{array} = 0$$

直交ベクトル問題：簡単なアルゴリズム

直交ベクトル問題

入力： d 次元 $0/1$ ベクトルの集合 $A, B \subseteq \{0,1\}^d$
出力： $\exists\, a \in A, b \in B : a \bullet b = 0 \Rightarrow$ Yes
$\forall\, a \in A, b \in B : a \bullet b \neq 0 \Rightarrow$ No

アルゴリズム OV：

1. すべての $a \in A$，すべての $b \in B$ に対して，次を実行
   $a \bullet b = 0$ ならば，Yes を出力
2. ここまで進んだら，No を出力

計算量 $= O(n^2 d)$　（ただし，$n = |A| = |B|$）

特に，$d = O(\log n)$ のとき，$O(n^2 \log n)$

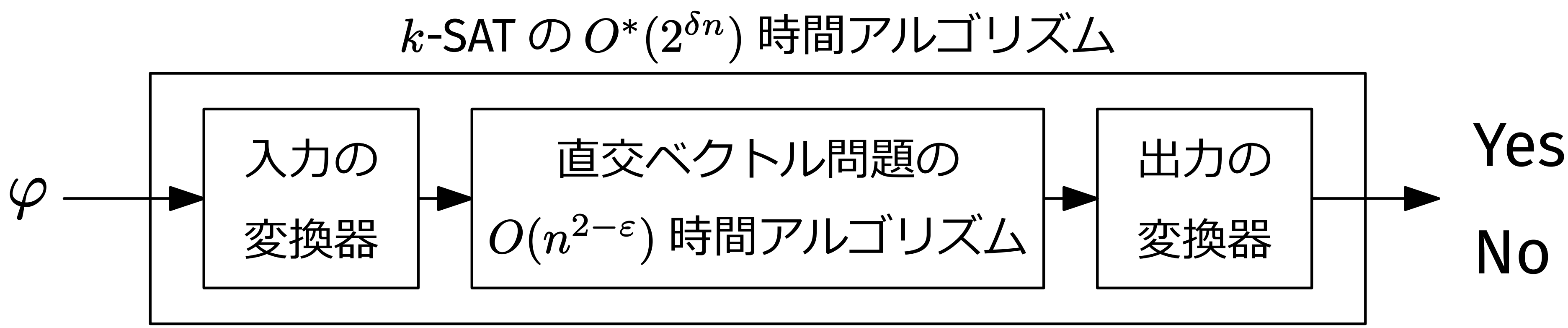
強指数時間仮説と直交ベクトル問題

次の定理の証明を紹介したい

定理：直交ベクトル問題の困難性 (Williams '05)

強指数時間仮説が正しい

$\Rightarrow d = O(\log n)$ の直交ベクトル問題は
$O(n^{2-\varepsilon})$ 時間で解けない ($\forall\, \varepsilon > 0$)



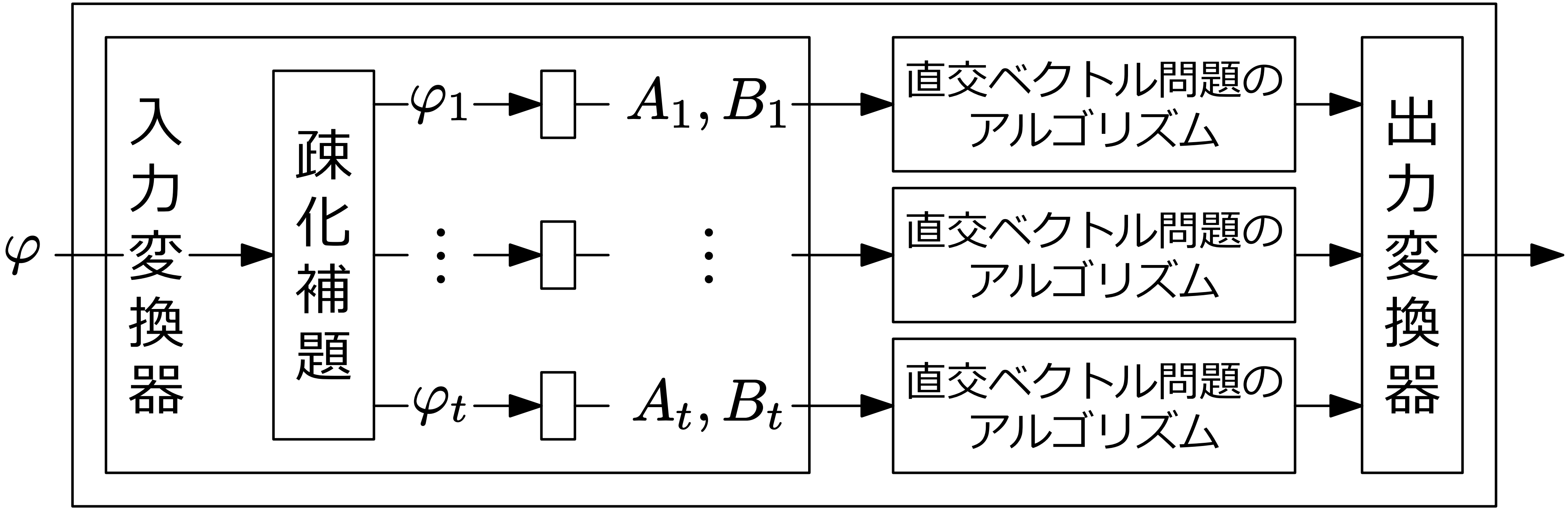
教訓

「特定の指数時間アルゴリズムの非存在」から
「特定の多項式時間アルゴリズムの非存在」が得られる

直交ベクトル問題の困難性：証明 (1)

証明：

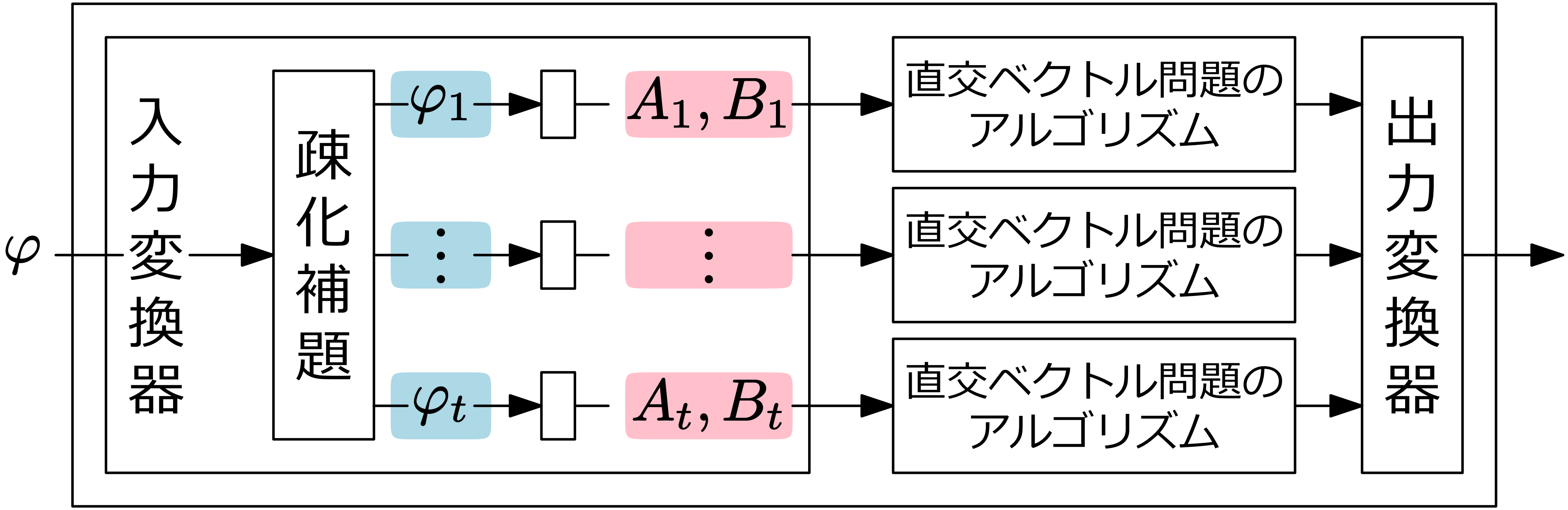
k-SAT の $O^*(2^{\delta n})$ 時間アルゴリズム



直交ベクトル問題の困難性：証明 (1)

証明：

k-SAT の $O^*(2^{\delta n})$ 時間アルゴリズム



k-SAT の入力

変数の数 $\leq n$

節の数 $\leq m = O(n)$

直交ベクトル問題の入力

$|A_i|, |B_i| \leq 2^{n/2}$

$d \leq m = O(n) = O(\log |A_i|)$

直交ベクトル問題の困難性：証明 (2)

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

直交ベクトル問題の困難性：証明 (2)

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

$$\begin{array}{lllcc}
 & & C_1\ C_2\ C_3 & & \\
A_i = & \{ & (1,0,0), & 0 & 0 \\
 & & (0,1,1), & 0 & 1 \\
 & & (0,0,1), & 1 & 0 \\
 & & (0,1,1)\ \} & 1 & 1 \\
 & & & x_1 & x_2
\end{array}
\qquad
\begin{array}{lllcc}
 & & C_1\ C_2\ C_3 & & \\
B_i = & \{ & (0,1,0), & 0 & 0 \\
 & & (0,0,0), & 0 & 1 \\
 & & (1,0,0), & 1 & 0 \\
 & & (1,0,1)\ \} & 1 & 1 \\
 & & & x_3 & x_4
\end{array}$$

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

$$\begin{array}{lllcc} & C_1\,C_2\,C_3 & & & \\ A_i = \{ & (1,0,0), & & 0 & 0 \\ & (\boxed{0},1,1), & & 0 & 1 \\ & (0,0,1), & & 1 & 0 \\ & (0,1,1) & \} & 1 & 1 \\ & & & x_1 & x_2 \end{array} \qquad \begin{array}{lllcc} & C_1\,C_2\,C_3 & & & \\ B_i = \{ & (0,1,0), & & 0 & 0 \\ & (0,0,0), & & 0 & 1 \\ & (1,0,0), & & 1 & 0 \\ & (1,0,1) & \} & 1 & 1 \\ & & & x_3 & x_4 \end{array}$$

$x_1 = 0, x_2 = 1$ とすると $C_1 = 1$ になる

直交ベクトル問題の困難性：証明 (2)

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

$$\begin{array}{rlcc} & \;C_1\,C_2\,C_3 & & \\ A_i = \{ & (1,0,0), & 0 & 0 \\ & (0,1,1), & 0 & 1 \\ & (0,0,1), & 1 & 0 \\ & (0,1,1)\ \} & 1 & 1 \\ & & x_1 & x_2 \end{array} \qquad \begin{array}{rlcc} & \;C_1\,C_2\,C_3 & & \\ B_i = \{ & (0,1,0), & 0 & 0 \\ & (0,0,0), & 0 & 1 \\ & (1,0,0), & 1 & 0 \\ & (1,0,1)\ \} & 1 & 1 \\ & & x_3 & x_4 \end{array}$$

$x_1 = 0, x_2 = 1$ とすると $C_2 = 1$ にならない

直交ベクトル問題の困難性：証明 (2)

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

$$\begin{array}{lccc} & C_1\,C_2\,C_3 & & \\ A_i = \{ & (1,0,0), & 0 & 0 \\ & (0,1,1), & 0 & 1 \\ & \boxed{(0,0,1),} & 1 & 0 \\ & (0,1,1)\ \} & 1 & 1 \\ & & x_1 & x_2 \end{array} \qquad \begin{array}{lccc} & C_1\,C_2\,C_3 & & \\ B_i = \{ & \boxed{(0,1,0),} & 0 & 0 \\ & (0,0,0), & 0 & 1 \\ & (1,0,0), & 1 & 0 \\ & (1,0,1)\ \} & 1 & 1 \\ & & x_3 & x_4 \end{array}$$

$$(0,0,1) \bullet (0,1,0) = 0$$

$\Leftrightarrow$ $x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 0$ は φ_i の充足割当

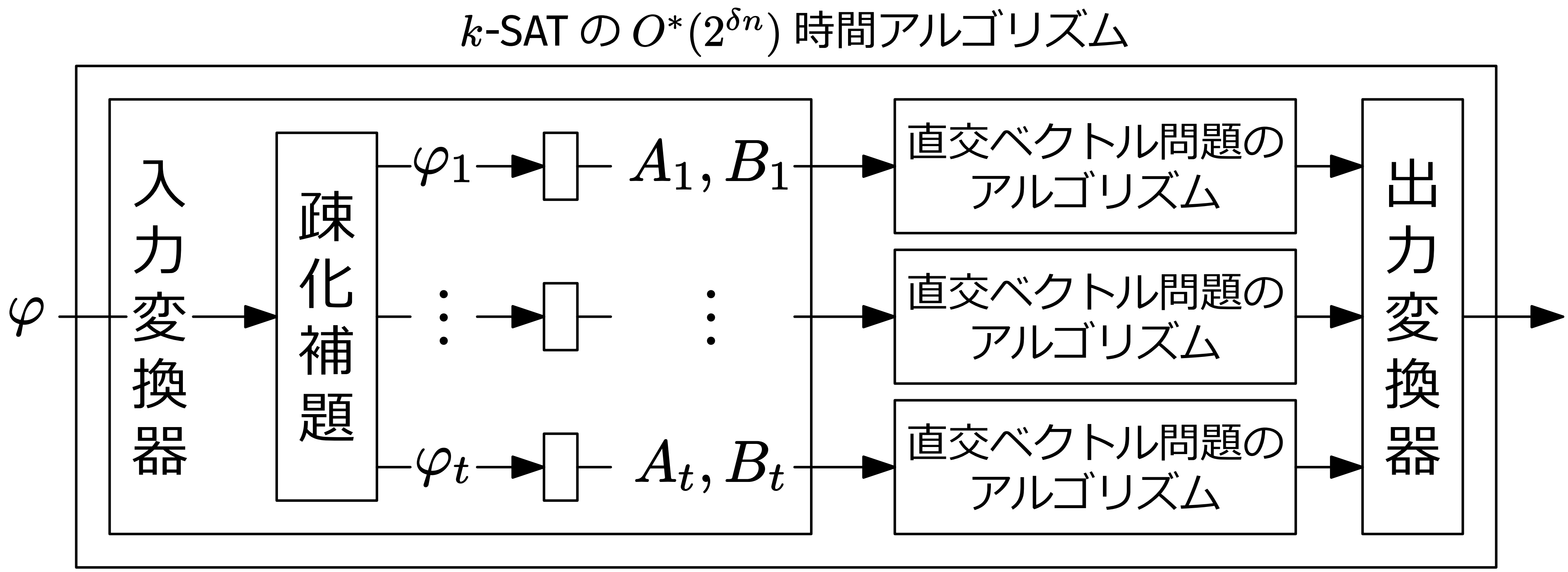
直交ベクトル問題の困難性：証明 (2)

$$\varphi_i = \underbrace{(x_1 \vee x_2 \vee \overline{x_3})}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee x_4)}_{C_2} \wedge \underbrace{(\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})}_{C_3}$$

$$A_i = \{\ \begin{array}{c} C_1\,C_2\,C_3 \\ (1,0,0), \\ (0,1,1), \\ (0,0,1), \\ (0,1,1)\ \} \end{array} \quad \begin{array}{cc} & \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ x_1 & x_2 \end{array} \qquad B_i = \{\ \begin{array}{c} C_1\,C_2\,C_3 \\ (0,1,0), \\ (0,0,0), \\ (1,0,0), \\ (1,0,1)\ \} \end{array} \quad \begin{array}{cc} & \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ x_3 & x_4 \end{array}$$

一般に，$|A_i| = |B_i| \leq 2^{n/2}$，
次元 $d \leq m = O(n) = O(\log |A_i|)$

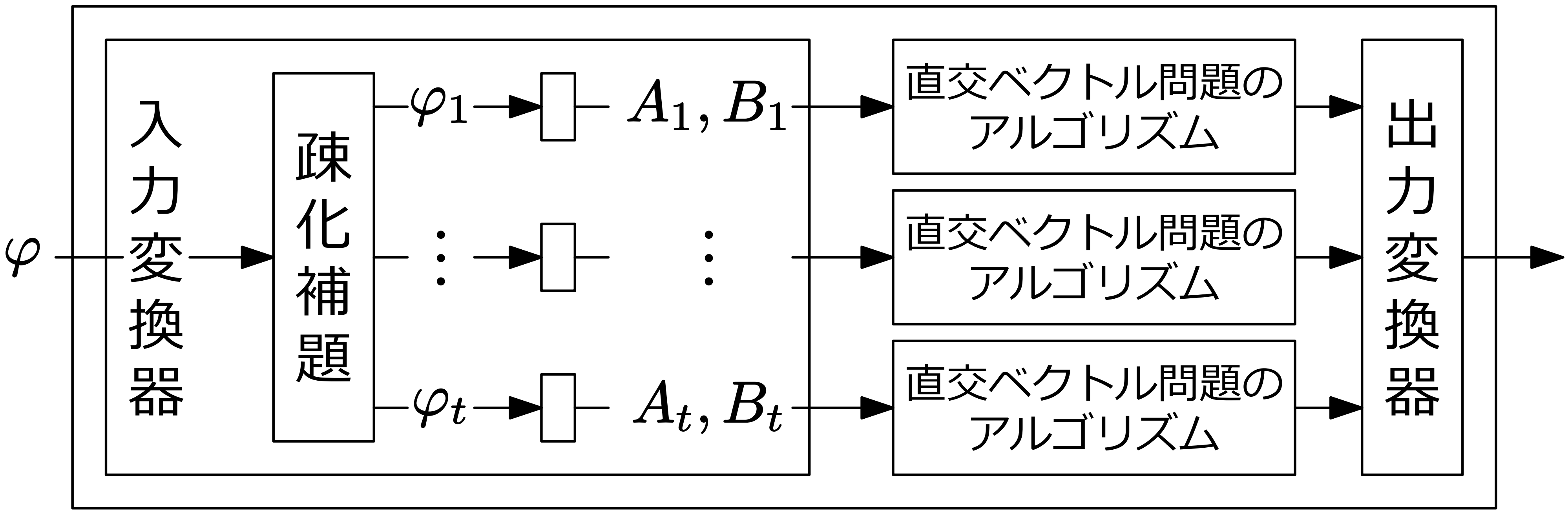
直交ベクトル問題の困難性：証明 (3)



$$|A_i|^{2-\varepsilon} \le (2^{n/2})^{2-\varepsilon} = 2^{(1-\varepsilon/2)n}$$

直交ベクトル問題の困難性：証明 (3)

k-SAT の $O^*(2^{\delta n})$ 時間アルゴリズム

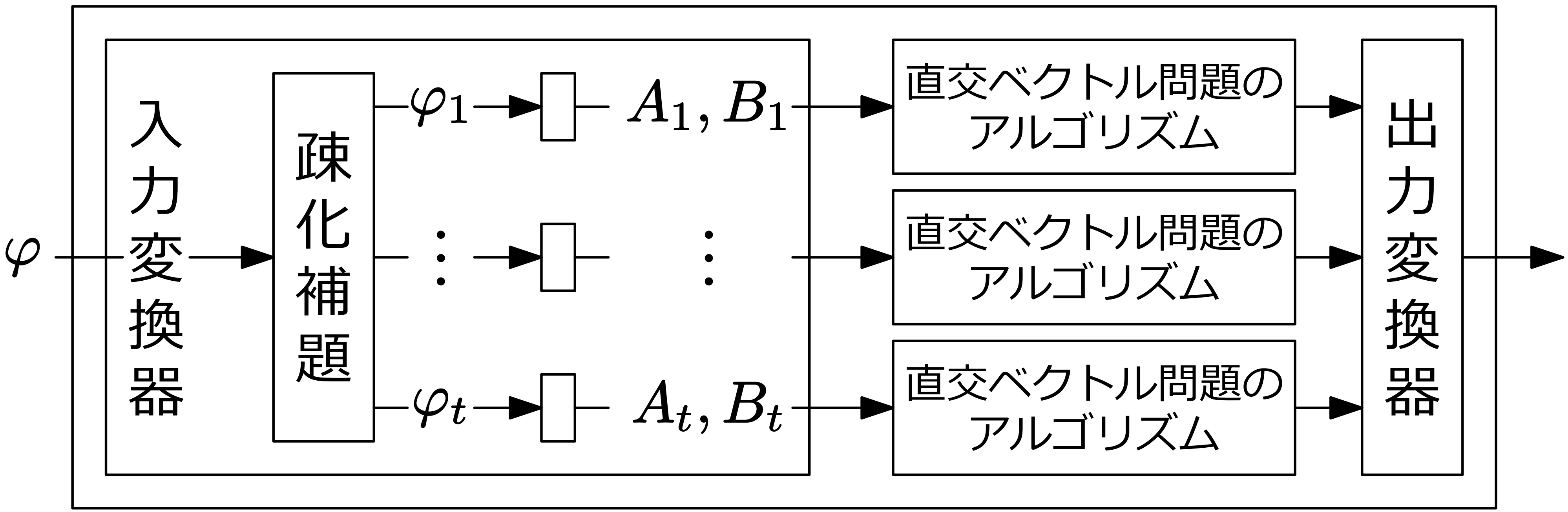


$2^{o(n)}$ $O^*(2^{n/2})$ $2^{o(n)}$

$$|A_i|^{2-\varepsilon} \leq (2^{n/2})^{2-\varepsilon}$$
$$= 2^{(1-\varepsilon/2)n}$$

直交ベクトル問題の困難性：証明 (3)

k-SAT の $O^*(2^{\delta n})$ 時間アルゴリズム



$2^{o(n)}$ $\quad O^*(2^{n/2})$ $\quad 2^{o(n)}$

$$\begin{aligned} |A_i|^{2-\varepsilon} &\leq (2^{n/2})^{2-\varepsilon} \\ &= 2^{(1-\varepsilon/2)n} \end{aligned}$$

全体の計算量 $= 2^{o(n)} \cdot 2^{(1-\varepsilon/2)n} = O^*(2^{(1-\varepsilon/2)n+o(n)})$ $\square$

多項式時間の困難性

問題	可能	不可能
直交ベクトル問題 (次元 $= O(\log n)$)	$O(n^2)$	$O(n^{2-\varepsilon})$
グラフ：直径計算問題 (辺数 $= O(n)$)	$\tilde{O}(n^2)$	
文字列：編集距離	$O(n^2)$	

強指数時間仮説の成立を仮定

多項式時間の困難性

問題	可能	不可能
直交ベクトル問題 (次元 $= O(\log n)$)	$O(n^2)$	$O(n^{2-\varepsilon})$
グラフ：直径計算問題 (辺数 $= O(n)$)	$\tilde{O}(n^2)$	$O(n^{2-\varepsilon})$
文字列：編集距離	$O(n^2)$	$O(n^{2-\varepsilon})$

強指数時間仮説の成立を仮定

(Roditty, Vassilevska Williams '13; Backurs, Indyk '15)

本日の内容

未解決：強指数時間仮説 (SETH)

CNF-SAT は $(2-\varepsilon)^n$ 時間で解けない

- 強指数時間仮説 $\Rightarrow$ 指数時間仮説
- 強指数時間仮説 $\Rightarrow$ 細粒度計算複雑性
  - 直交ベクトル問題の複雑性