符号と格子に関わるパラメータ化計算量

手塚 真徹　　田中 圭介

2024年5月10日

@CRESTクリプトマス 2024年度第1回全体会議

Version 2025/02/05

目次

・クラス**P**，**NP**と多項式時間帰着

・パラメータ化計算量

・クラス**FPT**，**para-NP**と**FPT**帰着

・回路，*Weight-SAT*問題とクラス**W[t]**

・暗号で用いられる計算問題に関連するパラメータ化計算量

・符号と格子に関するパラメータ化計算量

・γ-*NCP*からγ-*MDP*への乱択FPT帰着

クラス**P**，**NP**と多項式時間帰着

判定問題

判定問題 (Decision problem)

出力が 1 (Yes)，0(No) だけに限られる問題

判定問題の例（素数判定問題)

入力例　自然数 n

出力　　n が素数なら 1，そうでないならば 0 を出力する.

より一般的には,

Yesの入力例の言語 $L \subseteq \{0,1\}^*$ により判定問題を記述する.

例えば，上記の素数判定問題は $L = \mathit{PRIME} := \{\, n \mid n$ は素数$\}$ で表せる.

クラス**P**，**NP**は判定問題に関する計算量クラス

クラスP

クラスP

決定性多項式時間アルゴリズムで判定可能な判定問題のクラス

例 $\mathit{RELPRIMES} := \{(x, y) \mid x, y$ は互いに素$\}$

問題例 (x, y) に対して Euclid のアルゴリズムを用いることで多項式時間で判定可能.

クラスNP

クラス**NP**

答えが Yes である問題例に対して，多項式時間で検証できる証拠 (witness)が存在する判定問題のクラス

例　$COMPOSITES := \{\, x \mid$ 整数 $p, q > 1$ に対して $x = pq\}$

問題例 x に対して $w = (p, q)$ が証拠

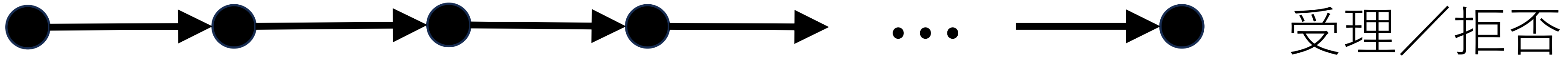
クラス**NP**は次のように言い換えられる．

非決定性多項式時間アルゴリズムで判定可能な判定問題のクラス

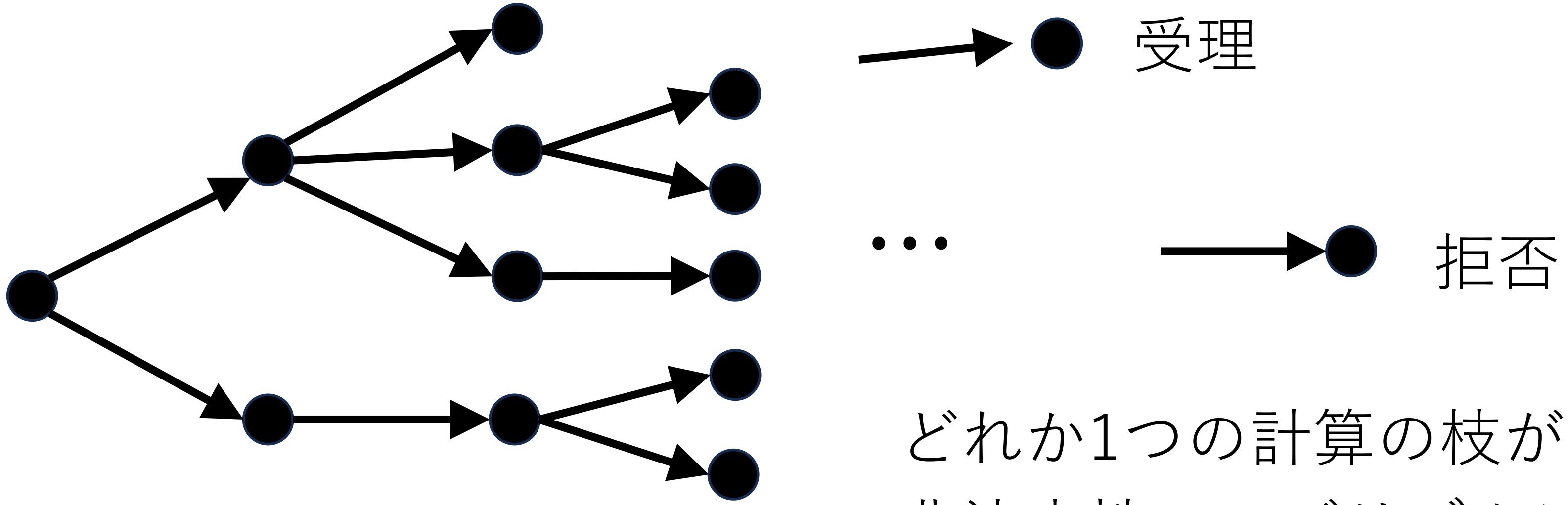
NPの**N**は非決定性 (Nondeterministic) に由来する．

決定性計算と非決定性計算の違い

決定性計算



非決定性計算



どれか1つの計算の枝が受理すれば，
非決定性アルゴリズムは受理する．

多項式時間帰着

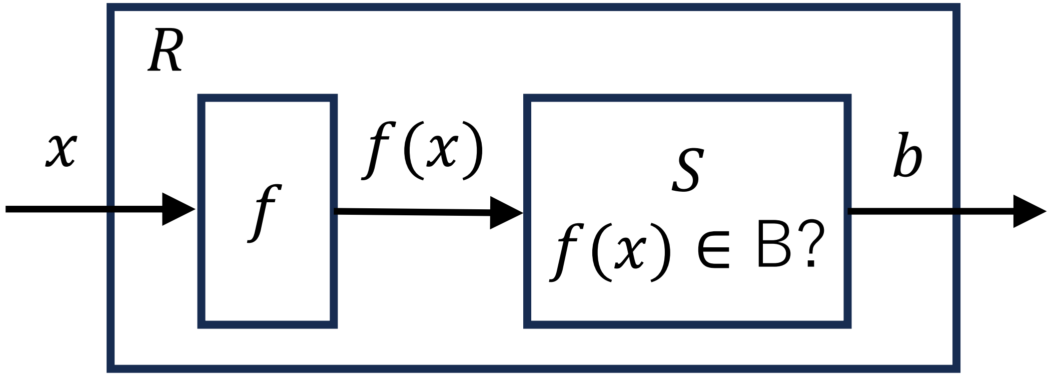
多項式時間帰着 f

言語A に対する問題を言語B に対する問題に変換する関数f で次を満たすもの.

・fは多項式時間アルゴリズムで計算できる

・$x \in \mathrm{A} \Longrightarrow f(x) \in \mathrm{B}$ かつ $x \notin \mathrm{A} \Longrightarrow f(x) \notin \mathrm{B}$　（すなわち，$x \in \mathrm{A} \Leftrightarrow f(x) \in \mathrm{B}$）

多項式時間帰着f と言語Bの判定問題を多項式時間で解くアルゴリズムS があれば，言語Aの判定問題を多項式時間で解くアルゴリズムR が構成可能.



言語Bの判定問題は少なくとも
言語Aの判定問題を解くこと以上に難しい.
問題の困難さを比較できる！

NP困難，NP完全

問題Pが**NP**困難

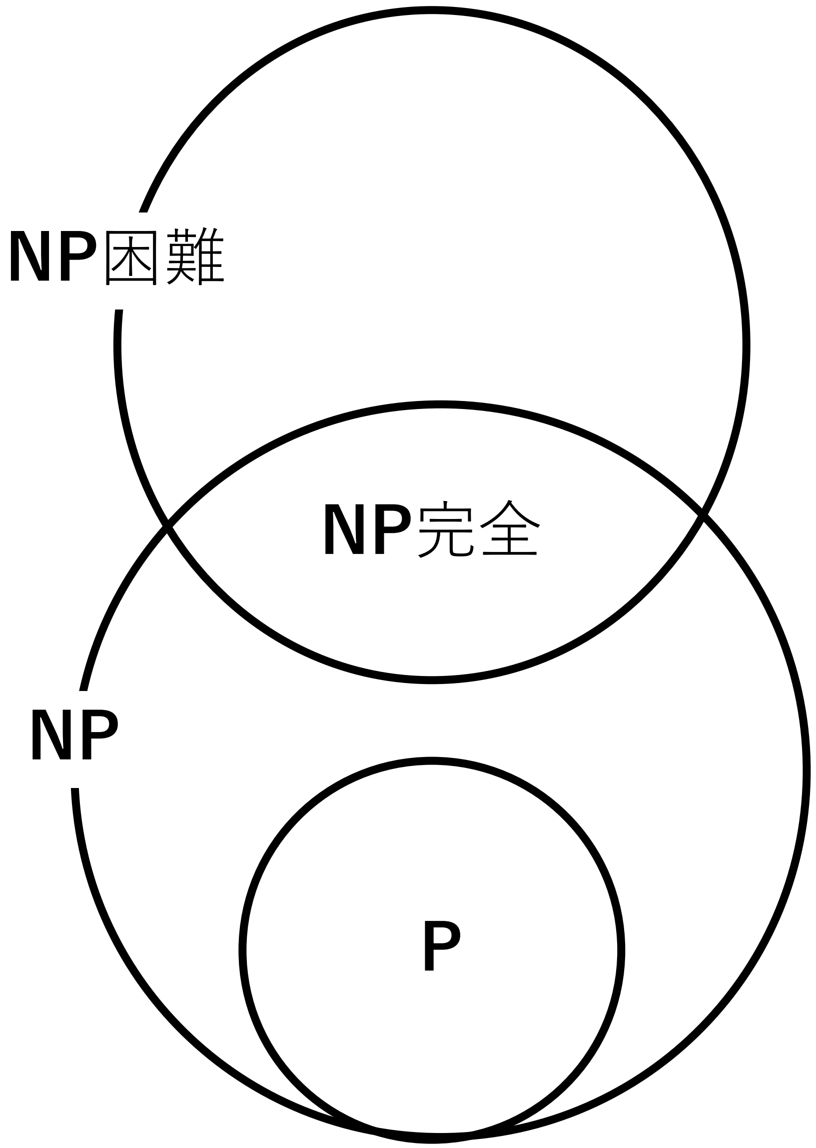
　NPに含まれる全ての問題が
　問題Pに多項式時間帰着する．

　NPに属するどの問題と比べても，
　少なくとも同等以上に難しい！

問題Pが**NP**完全

　問題Pが**NP**困難かつ**NP**に属する．

　NPで最も難しい！



NP完全問題の例 *CNF-SAT*, *3SAT*

CNF-SAT := { ϕ | ϕ は充足可能なCNF論理式}

CNF　全ての節 ( )が AND ∧ でつながれ,
全ての節の中身のリテラルをOR ∨ でつながれたもの.

問題例　$\phi = (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_3)$ 充足可能

証拠　$w = (x_1 = 1, x_2 = 0, x_3 = 1)$

3SAT := { ϕ | ϕ は充足可能な3CNF論理式}

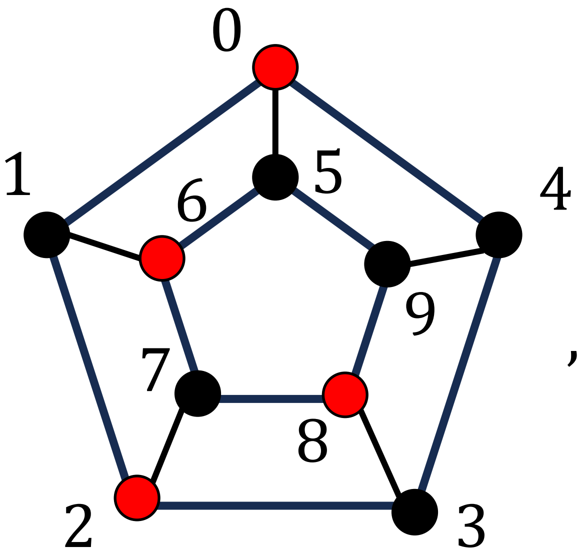
問題例　$\phi = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge \cdots \wedge (\neg x_5 \vee x_8 \vee \neg x_9)$

NP完全問題の例 *INDSET*

$INDSET := \{\ (G, k)\ |$ グラフGはサイズ k の独立集合をもつ$\}$

グラフG の独立集合 Iとは,
　グラフGの頂点の部分集合Iであり,
　Iの異なる2頂点間に辺が存在しないもの.

問題例

$(G =$  $,\ k = 4)$

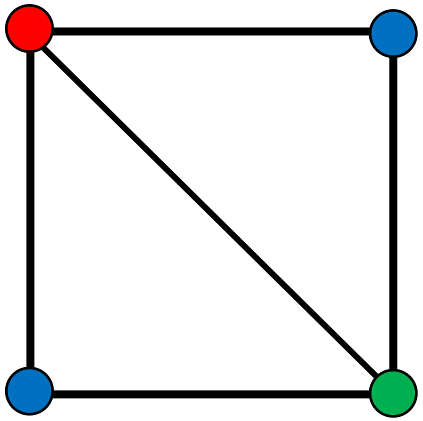
証拠　$w =$ 頂点 $0, 2, 6, 8$

NP完全問題の例 *3COLOR*

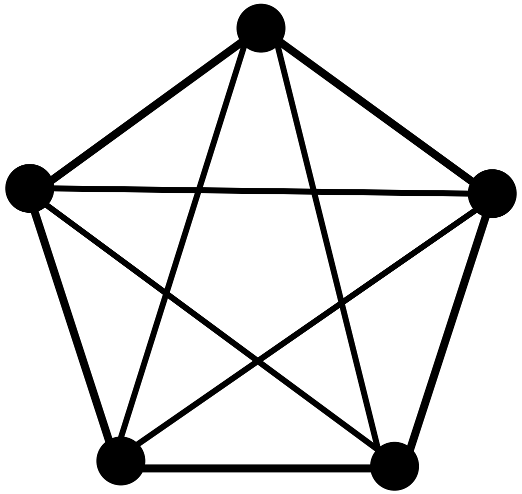
3COLOR := { G | グラフGは3彩色可能}

3彩色問題とは
　隣接するグラフの頂点どうしが同じ色にならないように,
　頂点を3つの色だけを用いて彩色する問題.

問題例（3彩色可能）

$G =$ 

問題例（3彩色不可能）

$G =$ 

パラメータ化計算量

問題がNP困難だったら？

解きたい問題が**NP**完全，**NP**困難であると証明されていたとしたら．．．

最悪ケースの問題を解こうとしても，
入力のサイズが大きくなってしまえば手に負えない．

この問題に対してどのように向き合うか？

・近似アルゴリズムの利用

・平均ケース計算量解析

・ヒューリスティック（発見的アルゴリズム）の利用

パラメータ化計算量はこの問題に対する別のアプローチ

パラメータ化計算量への動機

解きたい問題が**NP**完全，**NP**困難であれば，
最悪ケースの問題例に対して効率的なアルゴリズムが存在ない．

もし，実応用上特殊ケースの問題例だけを扱う場合には，
その特殊ケースで問題が速く解けることが保証できたら嬉しい．

↓

これを扱うための
フレームワークはあるか？

パラメータ化計算量
　計算量を入力のサイズ n のほかにパラメータ k を導入して評価する．

$CNF\text{-}SAT$にパラメータを導入する①

・各節におけるリテラルの最大数を $\boldsymbol{k}$ とする．

$$\phi = \underbrace{(\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc)}_{\bigcirc \text{ が最大 } \boldsymbol{k} \text{ 個}} \wedge \cdots \wedge (\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc)$$

$k = 2$ のときは，　$2SAT$なので多項式時間で解ける．

$k = 3$ のときは，　$3SAT$なので**NP**完全問題．

$k = 2$ から 3 に変化するだけで，劇的に難しくなる．

CNF-SAT にパラメータを導入する②

・論理式が k 変数以下，論理式の入力長を n とする.

$$\phi(x_1, \cdots, x_k) = \underbrace{(\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc)}_{} \wedge \cdots \wedge (\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc)$$

$\bigcirc$ は $x_1, \neg x_1, \cdots, x_k, \neg x_k$ のいずれか.

変数の割当てを全数探索するアルゴリズムの計算量は

$$2^k \cdot poly(n)$$

パラメータ k の部分を固定すれば，計算量は入力長の多項式と見なせる.

CNF-SAT にパラメータを導入する　まとめ

CNF-SAT にパラメータを導入することで，問題の違った側面が見えてくる．

パラメータ k 導入部分	$k = 2$	$k = 3$	$k = 4$
節中のリテラルの最大個数 k	Easy	Hard	Hard
論理式が k 変数以下	Easy	Easy	Easy

Easy:　k を固定すれば，入力長 n の多項式時間で解ける．

Hard:　k を固定すれば，n の多項式時間で解くことが難しいだろう．

クラス**FPT**，**para-NP**とFPT帰着

パラメータ化問題とFPTアルゴリズム

パラメータ化問題

入力例の言語 $L \subseteq \{0,1\}^* \times \mathbb{N}$ により判定問題を記述する.

FPTアルゴリズム

パラメータ化問題例 (x, k) を入力として受け取り,
ある計算可能関数 f を用いて

この部分は k に依存しない.

$$f(k) \cdot \underline{poly(|x|)}$$

で動作時間が上から抑えることができるアルゴリズムのこと.

$f(k)$ は計算可能関数であればよいので, $f(k)$ が指数関数でもよい.

FPTアルゴリズムの例，クラスFPT

para-CNF-SAT $:= \{\, (\phi, k) \mid \phi$ はk 変数以下の充足可能なCNF論理式$\}$

$$\phi = (\underbrace{\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc}) \wedge \cdots \wedge (\bigcirc \vee \bigcirc \vee \cdots \vee \bigcirc)$$

$\bigcirc$ は $x_1, \neg x_1, \cdots, x_k, \neg x_k$ のいずれか

変数の割当てを2^k 通り全数探索し，

論理式の充足可否をチェックするアルゴリズム

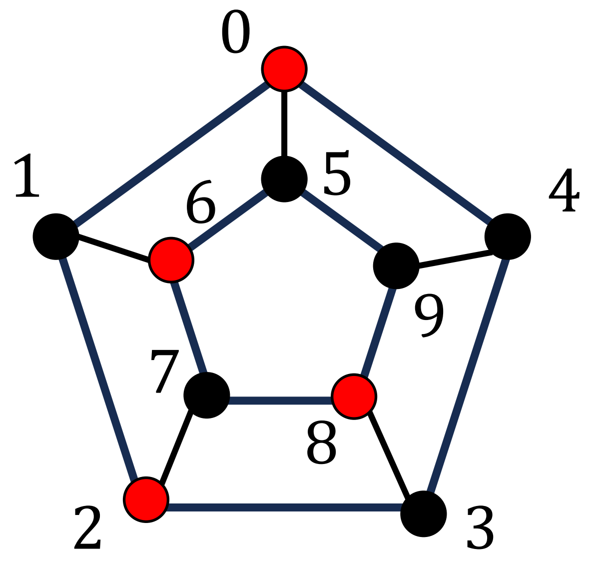
時間計算量　　$2^k \cdot O(n)$　　このアルゴリズムはFPTアルゴリズム

クラス**FPT**

パラメータ化問題に対して，決定性FPTアルゴリズムで判定可能な問題のクラス

FPTアルゴリズムではない例

para-INDSET := { (G,k) | グラフGはサイズ k の独立集合をもつ}



サイズ k の頂点部分集合をすべて全数探索し，
その部分集合が独立集合であるかチェックするアルゴリズム

時間計算量

k に依存している！

$$\binom{n}{k} \cdot O(k^2) = O(n^k) \cdot O(k^2) = O(k^2) \cdot O(n^k)$$

時間計算量が $f(k) \cdot poly(|x|)$ と表記できないので
このアルゴリズムはFPTアルゴリズムではない．

INDSET がクラス**FPT**に属さないと予想されている．（後ほど説明します）

クラスpara-NP

クラス**para-NP**

パラメータ化問題に対して，非決定性FPTアルゴリズムで
判定可能な問題のクラス

Fact　**P** = **NP** ⇔ **FPT** = **para-NP**

para-NPに属する問題の例

para-INDSET := { (G,k) | グラフGはサイズ k の独立集合をもつ}

para-COLOR = {(G,k) | グラフG は k 彩色可能}

FPT帰着

FPT帰着 f

言語 A に対する問題を言語 B に対する問題に変換する関数fで次を満たすもの.

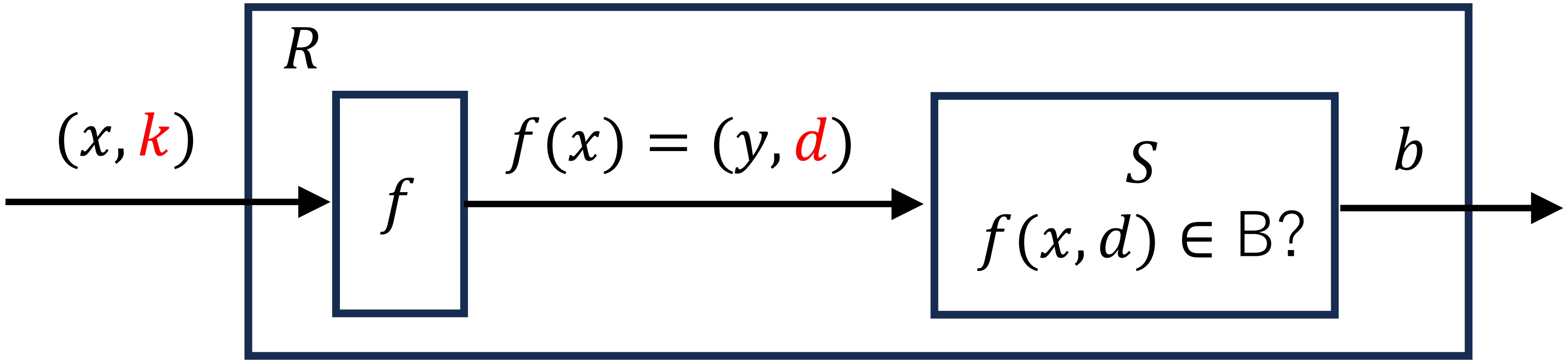
- fはFPTアルゴリズムで計算できる
- $(x,k) \in \mathrm{A} \Leftrightarrow f(x,k) = (y,d) \in \mathrm{B}$
- ある計算可能関数 g が存在し $d \leq g(k)$

FPT帰着f に

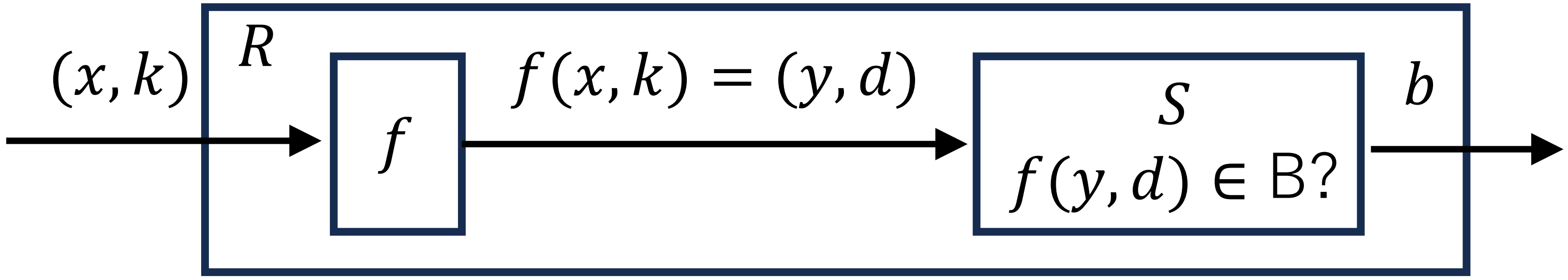
B ∈ **FPT** ならば A ∈ **FPT**

である性質をもたせるため.

FPT 帰着f と言語B に対する判定問題を解くFPTアルゴリズム S があれば,
言語A の判定問題を解くFPTアルゴリズム R が構成可能.



なぜ，「ある計算可能関数 g が存在し $d \leq g(k)$」？



・帰着f の時間計算量 $h_f(k) \cdot poly_f(|x|)$
・Bを解くFPTアルゴリズムS の時間計算量 $h_S(d) \cdot poly_S(|y|)$

Rの時間計算量

$$h_R(k) \cdot poly_R(|x|) + h_S(d) \cdot poly_S(|y|)$$

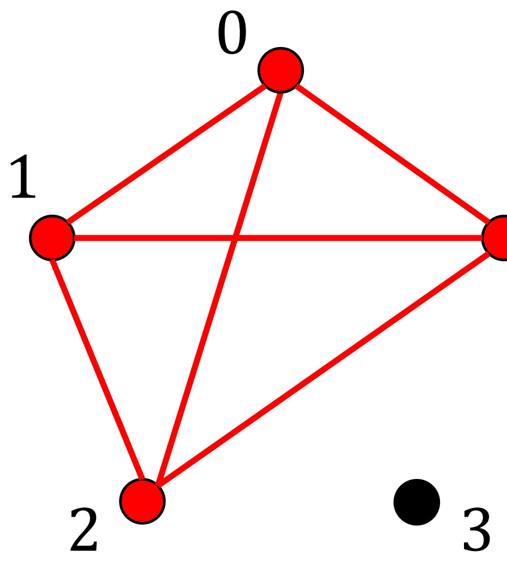
$h_S(d)$ は計算可能関数なので
指数関数である場合もある！

もしd が $n = |x|$に依存しない $g(k)$で抑えれれなければ，
RはFPTアルゴリズムでないことがあり，B ∈ **FPT** ⟹ A ∈ **FPT**を保証できない．

FPT帰着の例

para-CLIQUE $= \{(G, k) \mid G$ は
サイズ k 以上のクリークをもつ$\}$

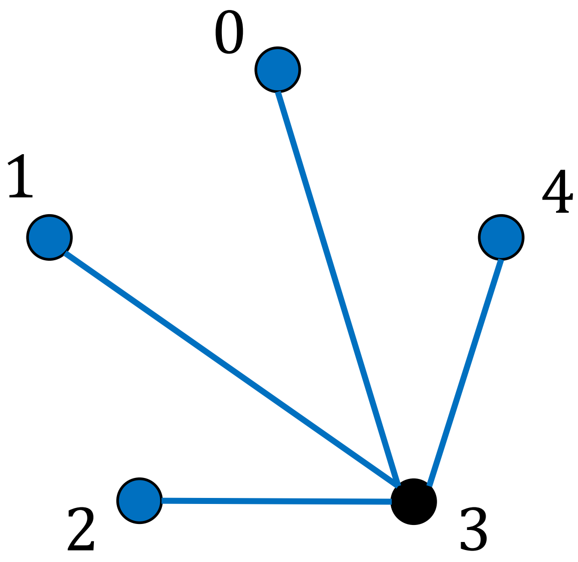
para-INDSET $:= \{(G', k') \mid$ グラフGは
サイズ k 以上の独立集合をもつ$\}$



FPT帰着f

$f: (G, k) \mapsto (\bar{G}, k)$

$\bar{G}$: G の補グラフ



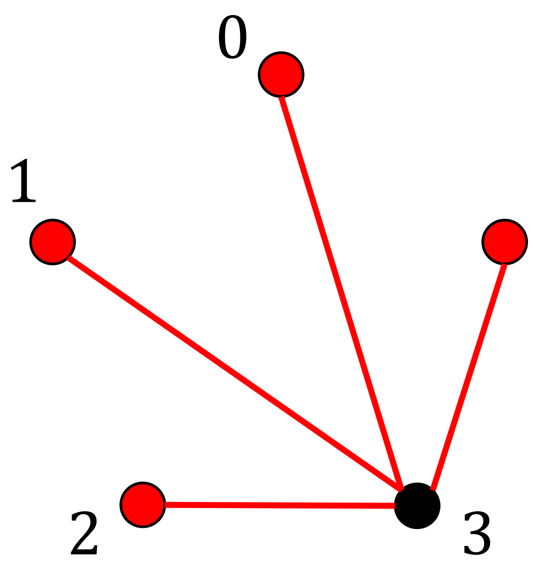
（非パラメータ） *CLIQUE* から *INDSET* の多項式時間帰着の手法を
para-CLIQUE から *para-INDSET* の帰着に適用できる．

しかし，FPT帰着では多項式帰着の手法をそのまま適用できない例もある．

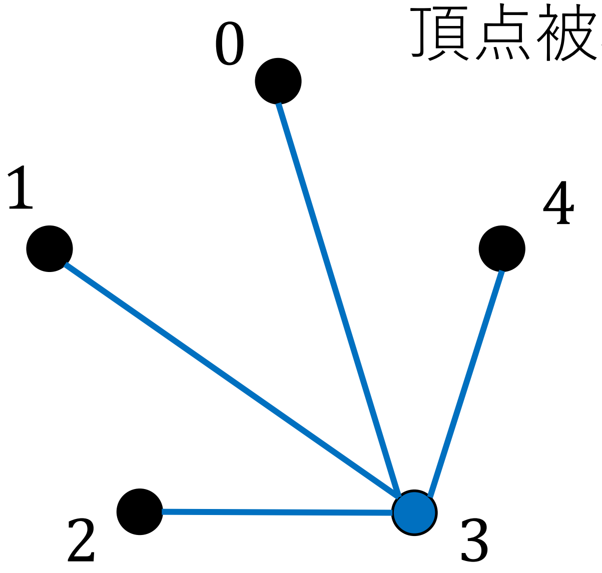
FPT帰着でない例

$para\text{-}INDSET := \{(G', k') \mid$ グラフGはサイズ k 以上の独立集合をもつ$\}$

$para\text{-}Vertex\text{-}Cover :=$
$\{(G', k') \mid$ グラフGはサイズ k 以下の頂点被覆をもつ$\}$



帰着f



$f : (G, k) \mapsto (G, |V| - k)$

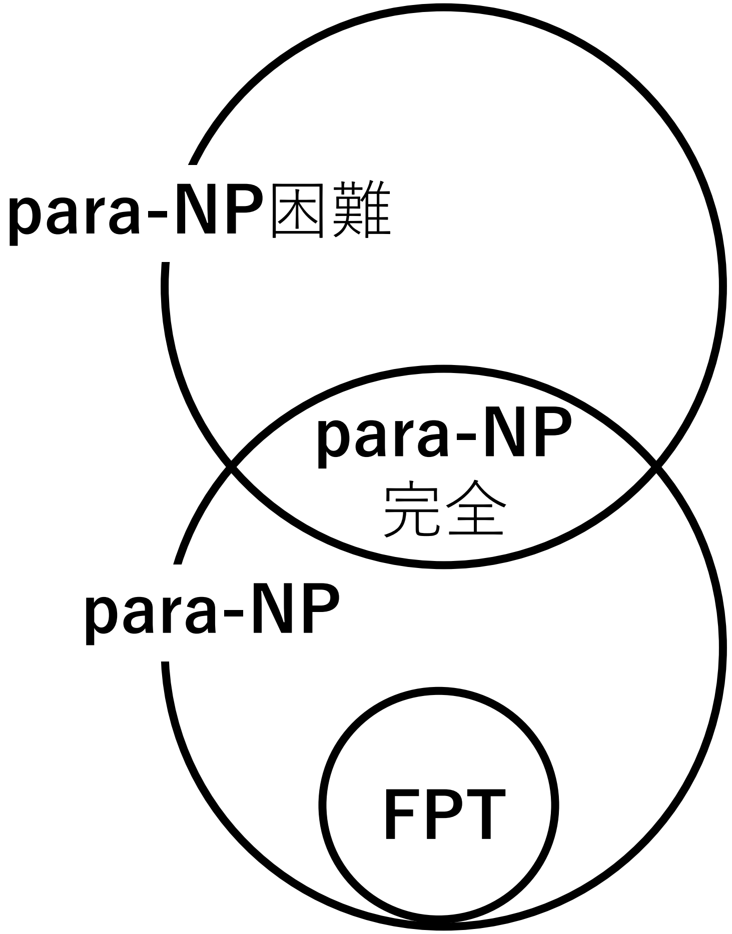
$g(k)$で抑えらないのでFPT帰着ではない.

（非パラメータ）問題Aから問題Bへの多項式時間が存在しても，
その問題をパラメータ化した場合，常にFPT帰着が存在するか明らかでない.

para-NP困難，para-NP完全

問題Pが**para-NP**困難，
　para-NPに含まれる全ての問題が問題PにFPT帰着する．

問題Pが**para-NP**完全とは，
　問題Pが**para-NP**困難かつ**para-NP**に属する．



para-NP完全問題の例

グラフの彩色問題 $COLOR$

$$para\text{-}COLOR = \{(G,k) \mid \text{グラフ}G\text{ は }k\text{ 彩色可能}\}$$

$k = 3$ で**NP**完全問題

問題のパラメータ化での比較

問題	*para-CNF-SAT*	*para-INDSET*	*para-COLOR*
パラメータk 導入部分	変数の最大数	独立集合のサイズ	彩色数
パラメータ化問題は **FPT**に属する？	✓	× ETHが正しいなら×	×
k を固定すれば 多項式時間で解ける？	✓ 多項式の次数は k に依存しない	✓ 多項式の次数は k に依存しそう	× k によって **NP**完全問題
パラメータ化問題は **para-NP**に属する？	✓	✓	✓ **para-NP**完全

para-INDSET の計算困難さをより正確に捉える計算量クラスはあるか？

回路，Weighted-Circuit SAT問題，クラス**W[t]**

回路のdepthとweft

回路のdepth

入力と出力を結ぶ最長パスが含む論理ゲートの個数のこと．

回路のweft

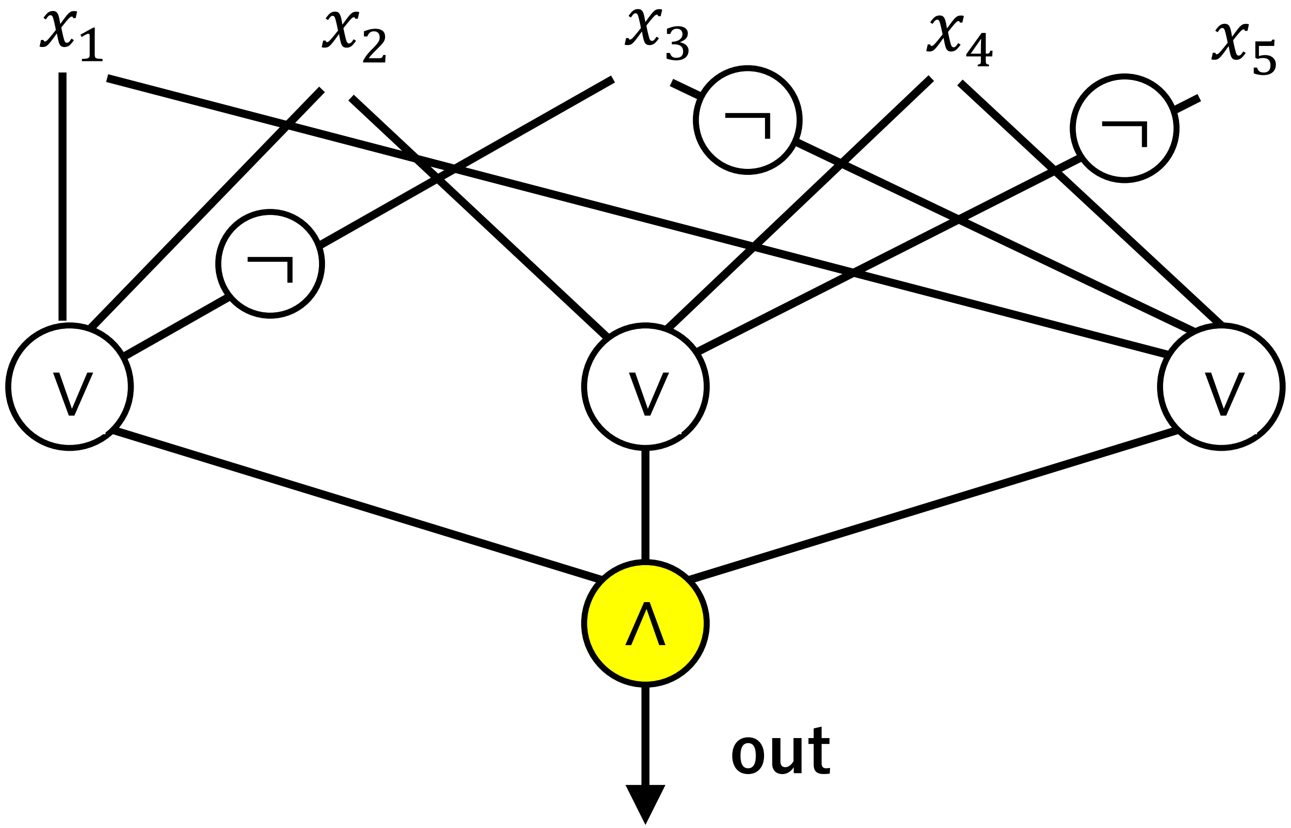
回路において最も多くのunbound fan-in ゲートを通る入力と出力を結ぶパスでのunbound fan-inのゲートの通過数.

Fan-inとは，論理ゲートの入力数のこと．

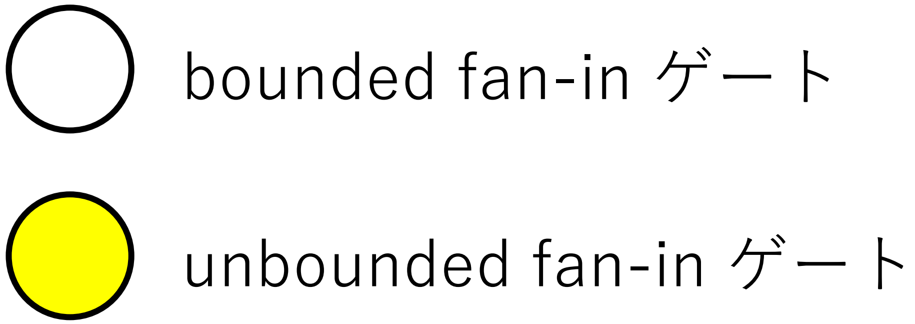
回路の例

論理式の集合 $\mathit{3SAT}$:= { ϕ | ϕ は充足可能な3CNF論理式}

例　$\phi = (x_1 \vee x_2 \vee \neg x_3) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_3 \vee x_4)$ の回路



任意の論理式 $\phi \in \mathit{3SAT}$ は，depth ≤ 3 (定数)， weft=1 で表現できる．



Weighted circuit SAT, クラスW[t]

パラメータ化問題 *Weighted weft* w *depth* d *circuit SAT*

$WCS\,(w,d) := \{(C_{w,d}, k) \mid$ 1がちょうど k 個割り当てられる入力で充足可能$\}$

$C_{w,d}$ はweft w以下，depth d以下の決定回路（出力が $\{0,1\}$）のこと

クラス$\mathbf{W}[t]$

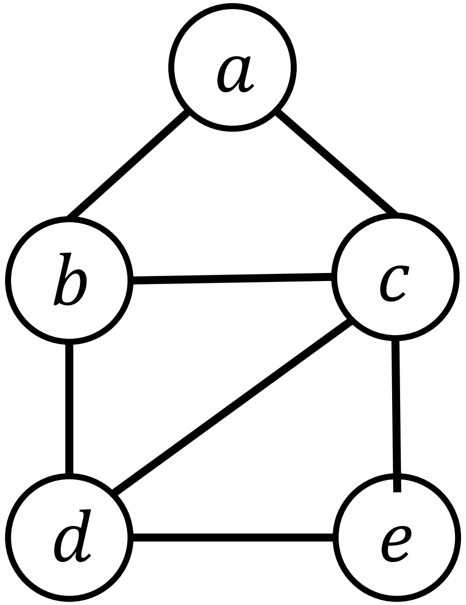
パラメータ化問題Pが $\mathbf{W}[t]$ に属するとは,

問題Pから定数深さ $d \geq 1$ の $WCS\,(t,d)$ 問題へのFPT帰着が存在すること.

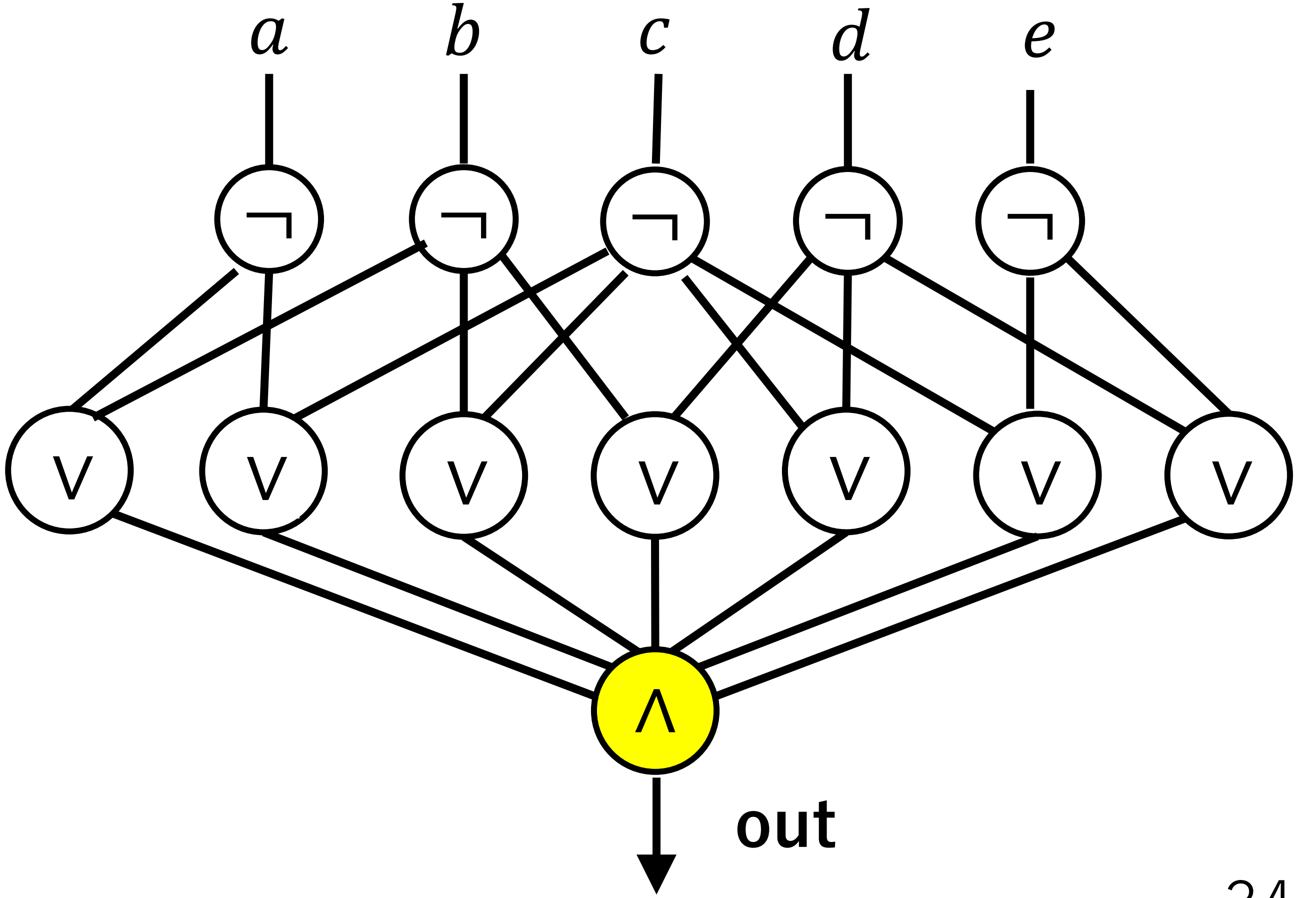
$INDSET \in$ W[1]

para-INDSET $:= \{(G,k) \mid$ グラフGは
サイズ k 以上の独立集合をもつ$\}$

weft 1，depth 3 回路 $C_{1,3}$

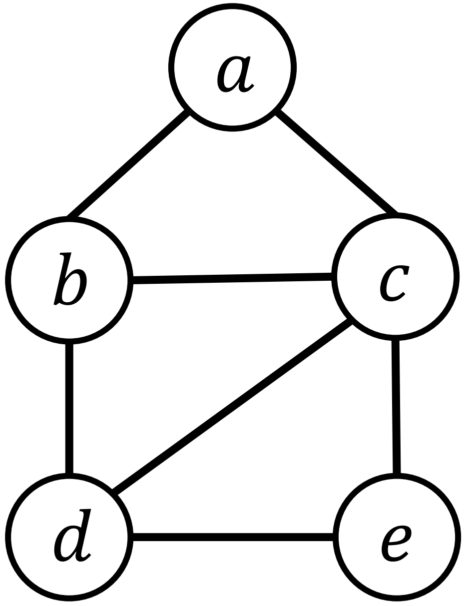


FPT帰着f



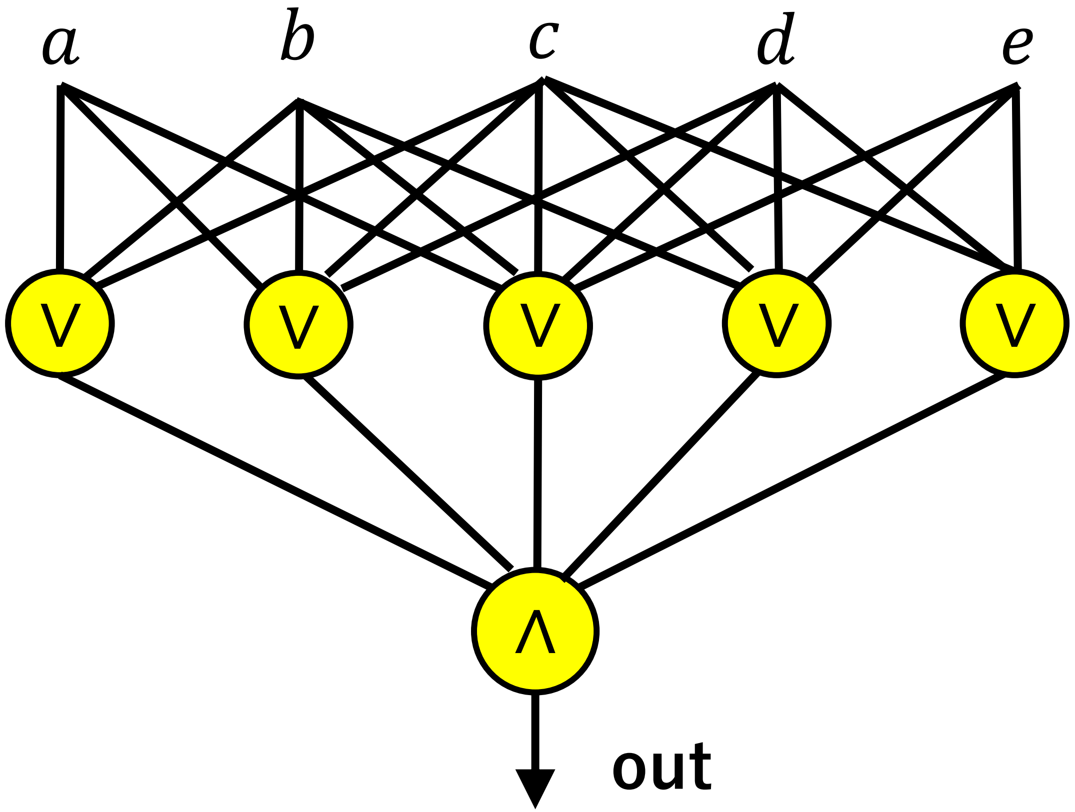
para-DOMINATING SET ∈ W[2]

para-DOMINATING SET ≔
$\{(G, k) \mid$ グラフGはサイズ k の支配集合をもつ$\}$



FPT帰着f

weft 2，depth 2 回路 $C_{2,2}$



G の頂点の部分集合 Dが支配集合とは
$\forall u \in V \setminus D$ に対し，$\exists v \in V$ s.t $(u, v) \in E$

W[t]困難，W[t]完全

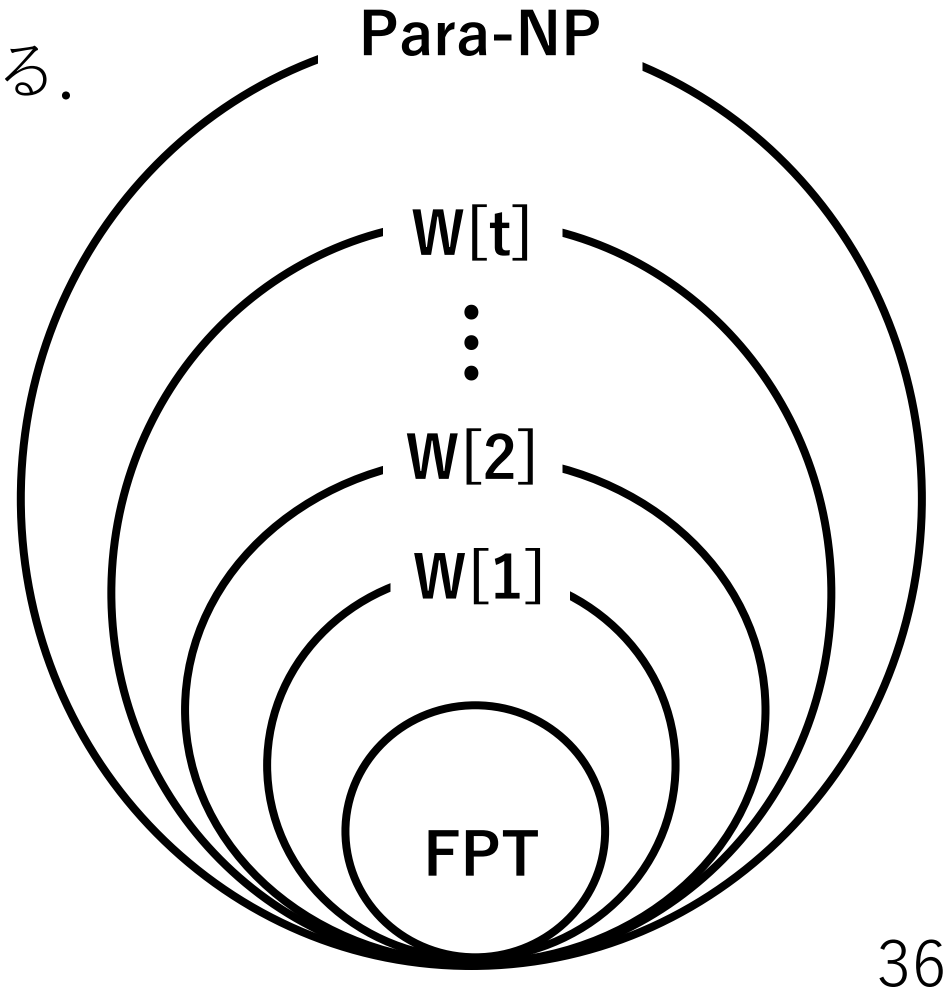
問題Pが**W[t]**困難
　W[t]に含まれる全ての問題が問題PにFPT帰着する.

問題Pが**W[t]**完全
　問題Pが**W[t]**困難かつ**W[t]**である.

W 階層

W[1] ⊂ **W**[2] ⊂ ・・・

のような階層構造をとる.



W[1]完全問題とW[2]完全問題の例

W[1]完全問題の例

para-INDSET $:= \{(G, k) \mid$ グラフGはサイズ k 以上の独立集合をもつ$\}$

para-CLIQUE $= \{(G, k) \mid G$ はサイズ k 以上のクリークをもつ$\}$

W[2]完全問題の例

para-DOMINATING SET $:= \{(G, k) \mid$ グラフGはサイズ k の支配集合をもつ$\}$

para-SET-COVER $= \{(U, \mathcal{S} = (S_1, \cdots S_{|\mathcal{S}|}), k) \mid U$ は $\mathcal{S}$ からk 個の部分集合を選び頂点被覆をつくれる．$\}$

すなわち，$\exists \Lambda$ s.t. $U = \bigcup_{i \in \Lambda} S_i \wedge |\Lambda| = k$

※ U は全体集合， S_i は U も部分集合.

W[1]と指数時間仮説(ETH)

指数時間仮説(ETH)

n 変数の *3SAT* は時間 $2^{o(n)}$ では解くことができない.

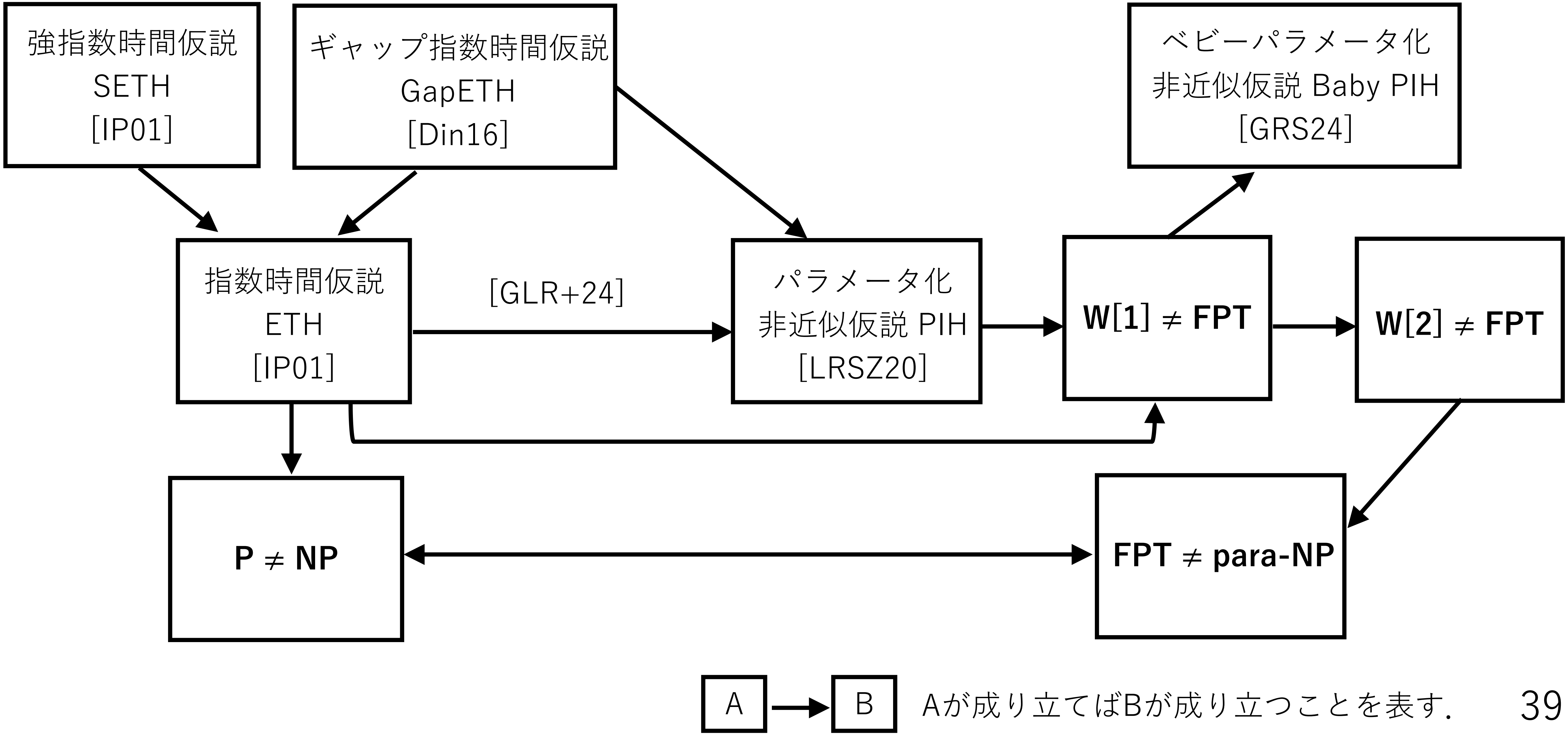
P ≠ NP予想よりも強い仮説

Fact　ETH が成り立てば k-*CLIQUE* は時間 $f(k) \cdot n^{O(1)}$ では解くことができない.

ETHが成り立てば **FPT ≠ W[1]**.

すなわち，ETHは **FPT ≠ W[1]** より強い仮説である！

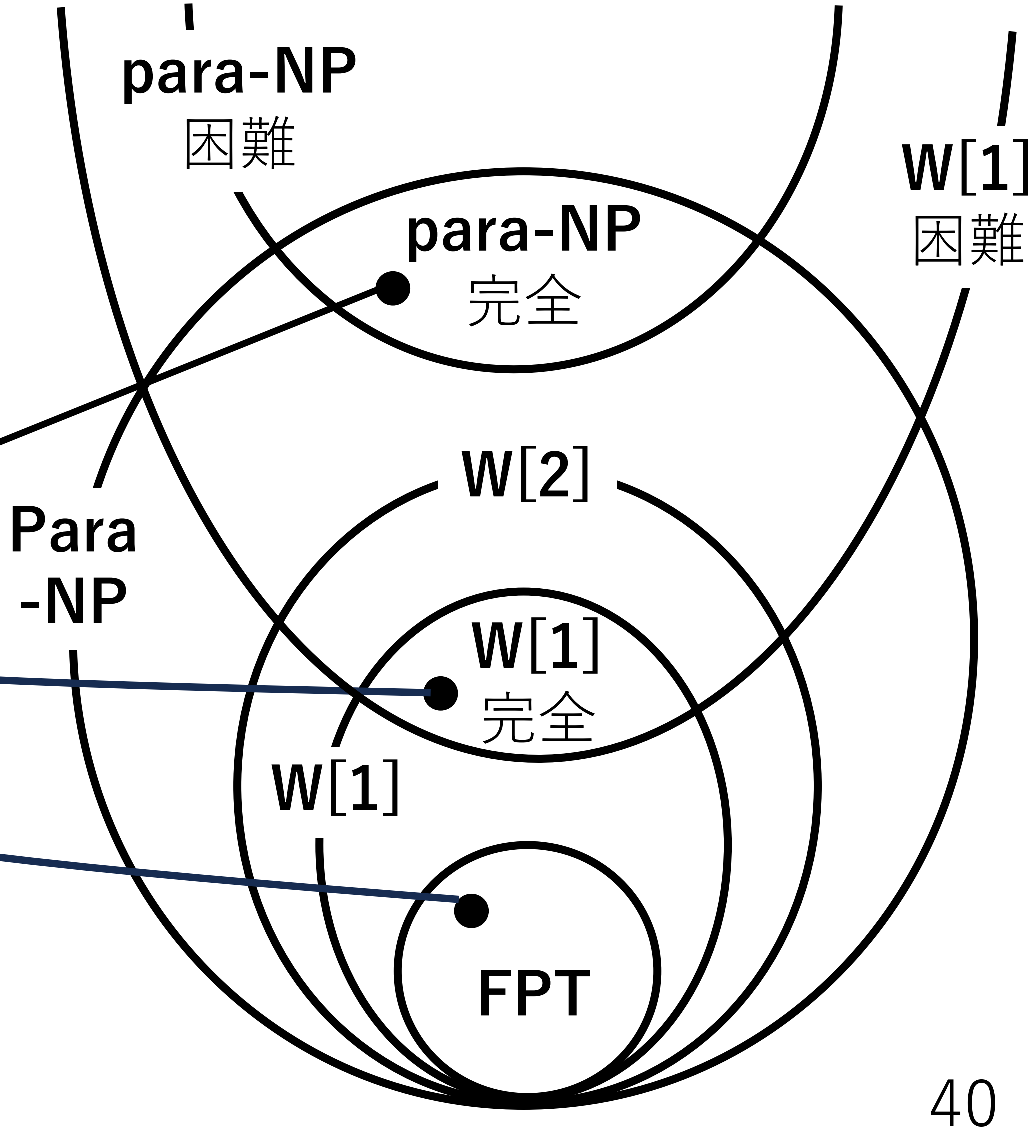
パラメータ化計算量に関連する仮説



Aが成り立てばBが成り立つことを表す.

登場した計算量クラスのまとめ

問題	*para-CNF-SAT*	*para-INDSET*	*para-COLOR*
パラメータk 導入部分	変数の 最大数	独立集合 のサイズ	彩色数
パラメータ化 問題クラス	**FPT**	**W[1]**完全	**para-NP** 完全



FPT ⊂ W[1] ⊂ W[2] ⊂ para-NP

暗号で用いられる計算問題に関連する
パラメータ化計算量

暗号で用いる計算問題とパラメータ化計算量

暗号で用いられる計算問題に関連するパラメータ化計算量

数論ベース	・離散対数問題	[FK93]（AAECC-10 1993） [Che04]（CRYPTO 2004）
	・素因数分解問題	[FK93]（AAECC-10 1993）
符号ベース	・近似最小距離問題	[BBE+21]（ICALP 2018, J.ACM 68 2021） [BCGR23]（STOC 2023）
格子ベース	・近似最近ベクトル問題	[BBE+21]（ICALP 2018, J.ACM 68 2021）
	・近似最短ベクトル問題	[BBE+21]（ICALP 2018, J.ACM 68 2021） [BCGR23]（STOC 2023）

素因数分解問題に関連するパラメータ化計算量

SMALL PRIME DIVISOR

判定版素因数分解問題．問題例 (N, k)　パラメータ k

n-bit 数 N は n^k 以下の非自明な約数をもつか？

[FK93]　*SMALL PRIME DIVISOR* はクラス**FPT**に属する

※ この結果はアルゴリズム動作における

計算時間の期待値が $f(k) \cdot \log(N)$ であることに注意

離散対数問題に関連するパラメータ化計算量

[FK93]　下記2つの問題はクラス**FPT**に属するか?

BOUNDED HAMMING WEIGHT DISCRETE LOGARITHM

$\mathbb{F}_p$ 上での判定版離散対数問題．問題例 $(p, g,\ g^x, k)$　　パラメータ k

素数 p が n bit であるとき $x < n^k$ であるか？

BOUNDED HAMMING WEIGHT DISCRETE LOGARITHM

$\mathbb{F}_p$ 上での判定版離散対数問題．問題例 $(p, g,\ g^x, k)$　　パラメータ k

x を 2進数表記したときの Hamming Weight が k 以下であるか？

離散対数問題に関連するパラメータ化計算量

[Che04]　$\mathbb{F}_{q^n}$ 上での離散対数問題

BOUNDED SUM-OF-DIGIT DISCRETE LOGARITHM

$\mathbb{F}_{q^n}$ 上での探索版離散対数問題．問題例 $(q^n,\ g,\ g^x, k)$　パラメータ k

x の q 進数展開 $x = x_0 + x_1 q + \cdots x_{n-1} q^{n-1}$ に対し，$\sum x_i \le k$　に制限

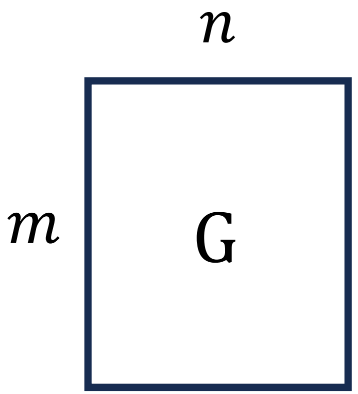
$\mathbb{F}_{q^{q-1}}$ 上の設定では上の問題は $O(f(k) \cdot log^4(q^{q-1}))$ で解ける．

符号と格子に関するパラメータ化計算量

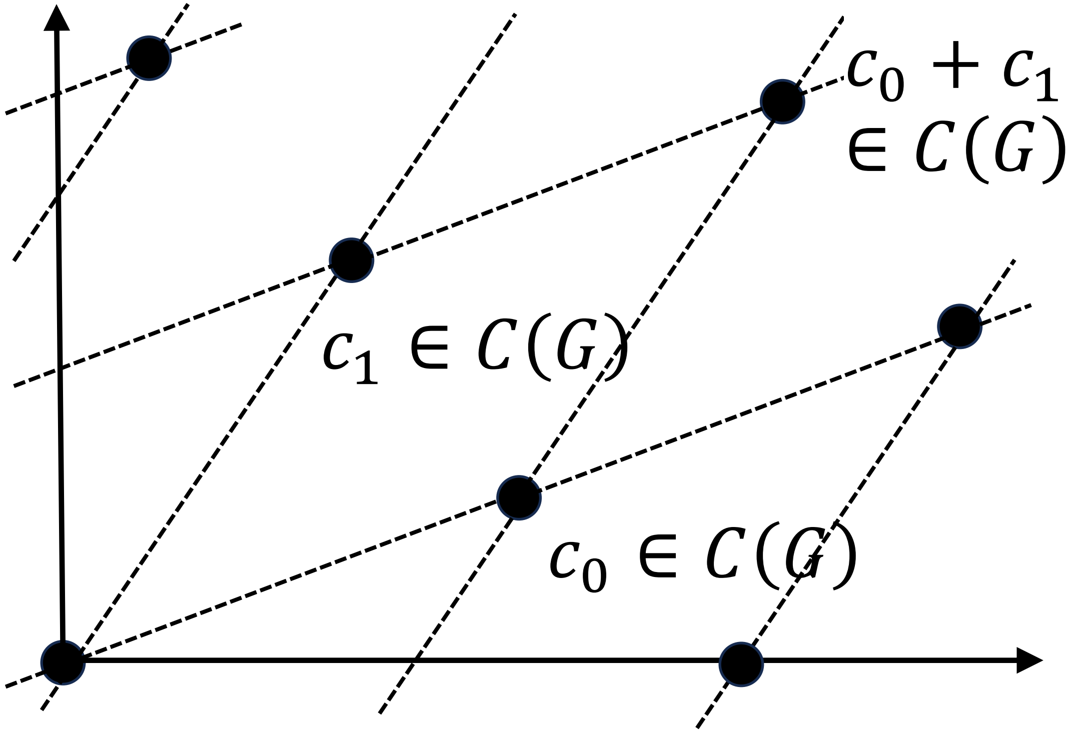
線形符号，ハミング距離

線形符号C $\mathbb{F}_q{}^m$ の部分ベクトル空間

生成行列



$$C(G) = \{Gx \mid x \in (\mathbb{F}_q)^n\}$$



Hamming 重み $\|c\|_0 :=$ ベクトル c の 0 でない成分の個数

Hamming 距離 $d^{(0)}(c, c') := \|c - c\|_0$

符号 C と $t \in (\mathbb{F}_q)^m$ の距離 $d(C, t) := \min d^{(0)}(c, t) \quad c \in C$

符号 C の最小距離 $\lambda(C) := \min \|c\|_0 \quad c \in C \setminus \{0^m\}$

パラメータ化 γ-NCP_q

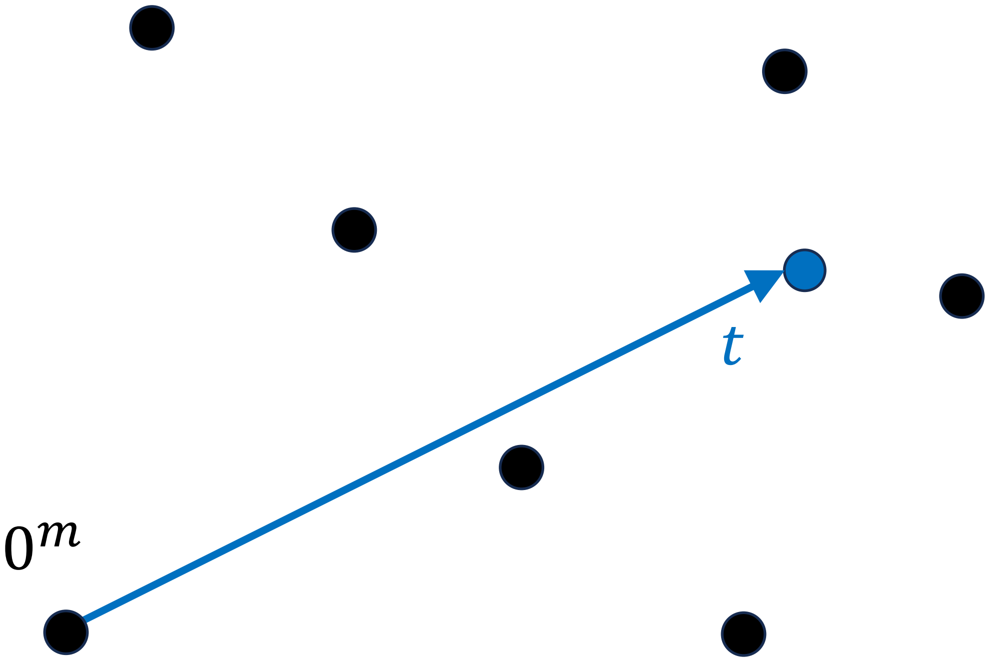
γ-近似最近符号語問題 γ-NCP_q

$\gamma\text{-}NCP_q = (\Pi_{Yes},\ \Pi_{No})$

$G \in \mathbb{F}_q^{\,m\times n}$: 符号の生成行列,　k: パラメータ

$$\Pi_{Yes} = \{((G,t),k)\ |\ d^{(0)}(C(G),t) \le k\}$$

$$\Pi_{No} = \{((G,t),k)\ |\ d^{(0)}(C(G),t) > \gamma k\}$$



● 符号語

● $t \in \mathbb{F}_q^{\,m}$: ターゲット

パラメータ化 γ-NCP_q

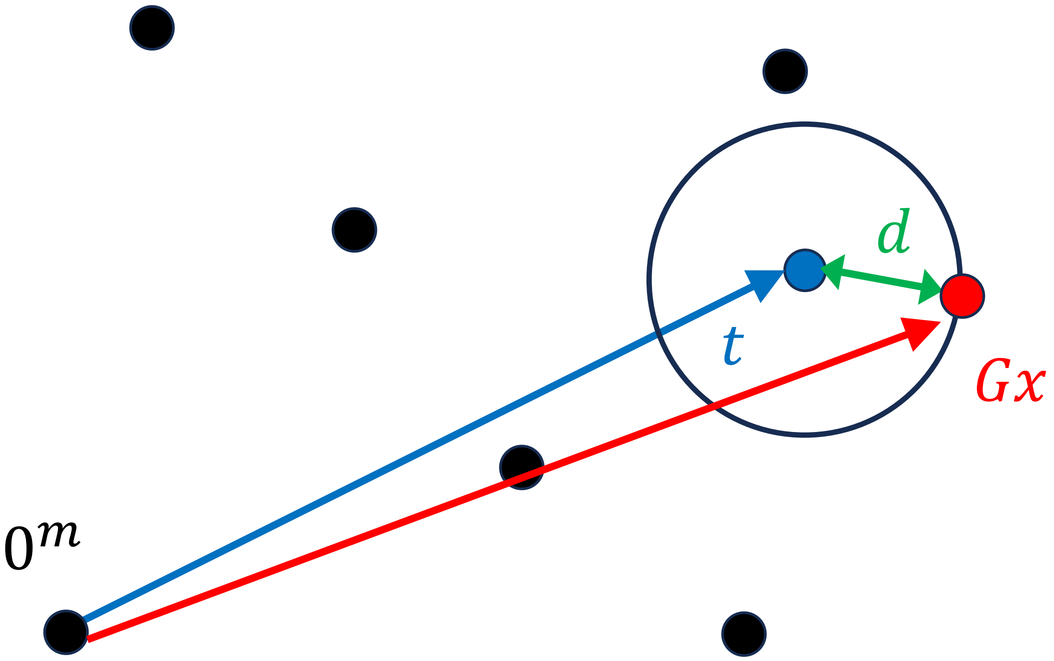
γ-近似最近符号語問題 γ-NCP_q

γ-$NCP_q = (\Pi_{Yes},\ \Pi_{No})$

$G \in \mathbb{F}_q{}^{m\times n}$: 符号の生成行列,　k: パラメータ

$$\Pi_{Yes} = \{((G,t),k) \mid d^{(0)}(C(G),t) \le k\}$$

$$\Pi_{No} = \{((G,t),k) \mid d^{(0)}(C(G),t) > \gamma k\}$$



- 符号語
- $t \in \mathbb{F}_q{}^m$: ターゲット
- Gx : 最近符号語

$$d = d^{(0)}(C(G),t) = \|Gx - t\|_0$$

パラメータ化 γ-MDP_q

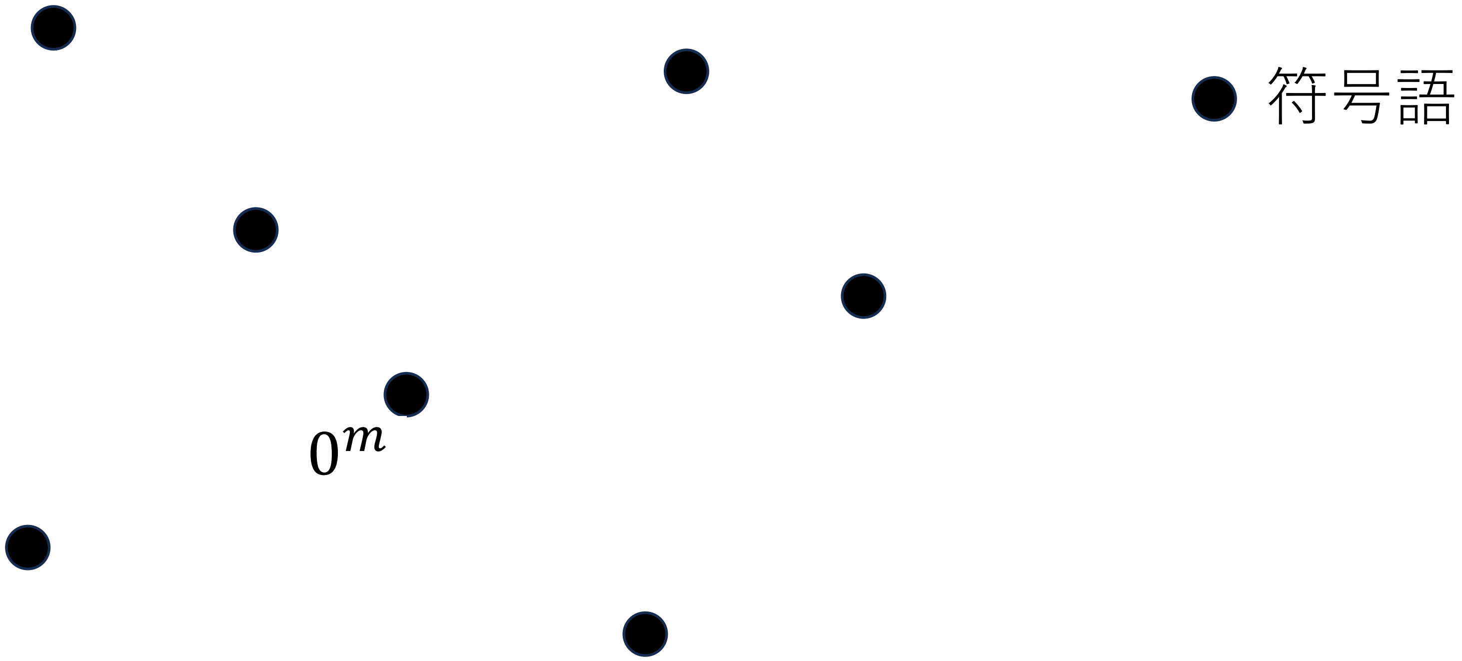
γ-近似最短距離問題 γ-MDP_q

γ-$MDP_q = (\Pi_{Yes}, \ \Pi_{No})$

$G \in \mathbb{F}_q{}^{m \times n}$: 符号の生成行列,　k: パラメータ

$\Pi_{Yes} = \{(G, k) \mid \lambda(C(G)) \leq k\}$

$\Pi_{No} = \{(G, k) \mid \lambda(C(G)) > \gamma k\}$



パラメータ化 γ-MDP_q

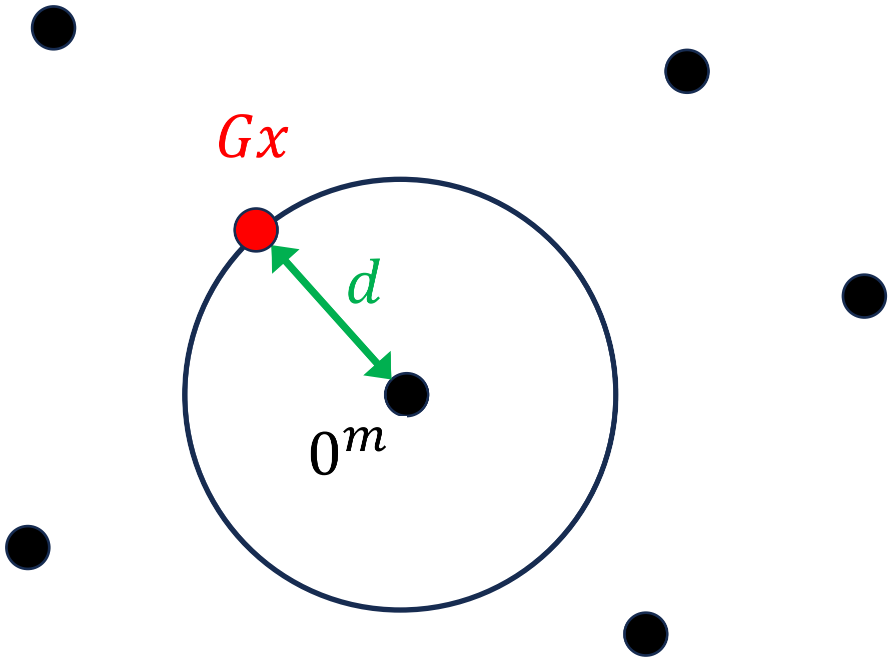
γ-近似最短距離問題 γ-MDP_q

$\gamma\text{-}MDP_q = (\Pi_{Yes},\ \Pi_{No})$

$G \in \mathbb{F}_q{}^{m\times n}$: 符号の生成行列,　k: パラメータ

$\Pi_{Yes} = \{(G,k)\ |\lambda(C(G)) \le k\}$

$\Pi_{No} = \{(G,k)\ |\lambda(C(G)) > \gamma k\}$



● 符号語

● Gx : 0^mに最も近い符号語

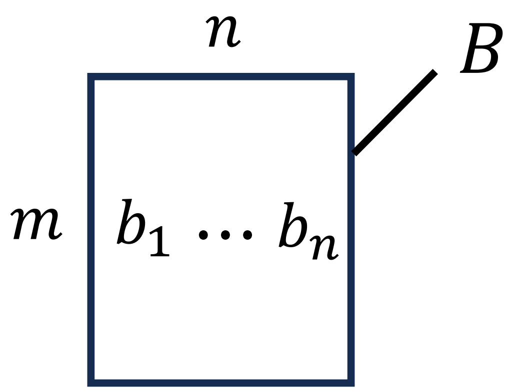
$d = \lambda(C(G)) = \|Gx\|_0$

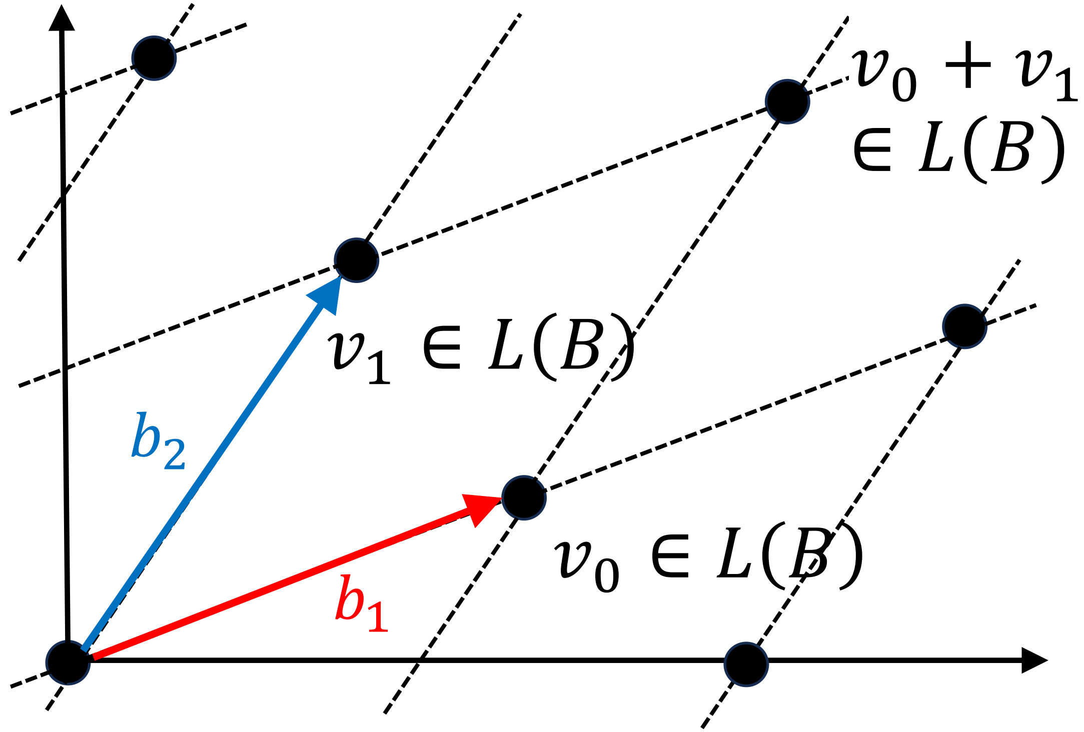
格子, ℓ_p-ノルム

格子L　$\mathbb{R}^m$ の離散加法部分群

$L(B) = \{\mathrm{B}x \mid x \in \mathbb{Z}^n\}$

基底





ℓ_p -ノルム　$\|v\|_p \coloneqq (\sum_{i=1}^{m} |v|^p)^{1/p}$

距離　$d^{(p)}(v, v') \coloneqq \|v - v\|_p$

格子 L と $t \in \mathbb{R}^m$ の距離　$d^{(p)}(L, t) \coloneqq \min\ d^{(p)}(v, t)$　$v \in L$

格子 L の最短ベクトル　$\lambda_1^{(p)}(L) \coloneqq \min \|v\|_p$　$v \in L \setminus \{0^m\}$

パラメータ化 γ-CVP_p

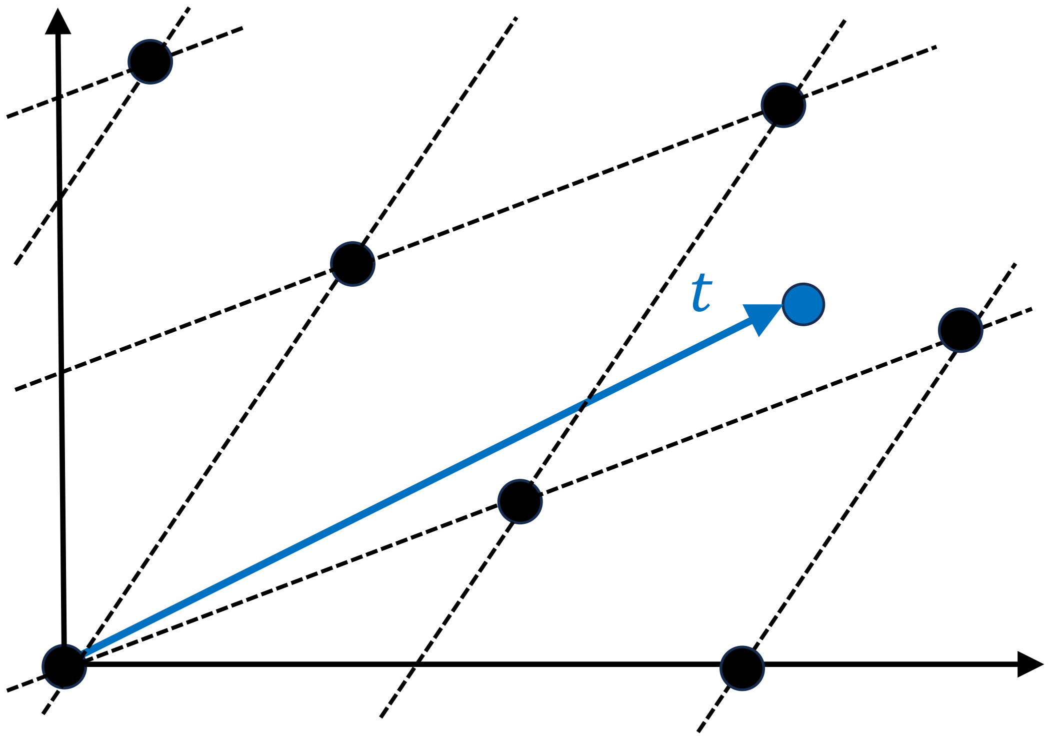
γ-近似最近ベクトル問題 γ-CVP_p

$$\gamma\text{-}CVP_p = (\Pi_{Yes},\ \Pi_{No})$$

$B \in \mathbb{R}^{m \times n}$: 格子の基底, k: パラメータ

$$\Pi_{Yes} = \{((B,t),k)\ |d^{(p)}(L(B),t)\ \leq k\}$$

$$\Pi_{No} = \{((B,t),k)\ |d^{(p)}(L(B),t) > \gamma k\}$$



格子点

$t \in \mathbb{Z}^m$: ターゲット

パラメータ化 γ-CVP_p

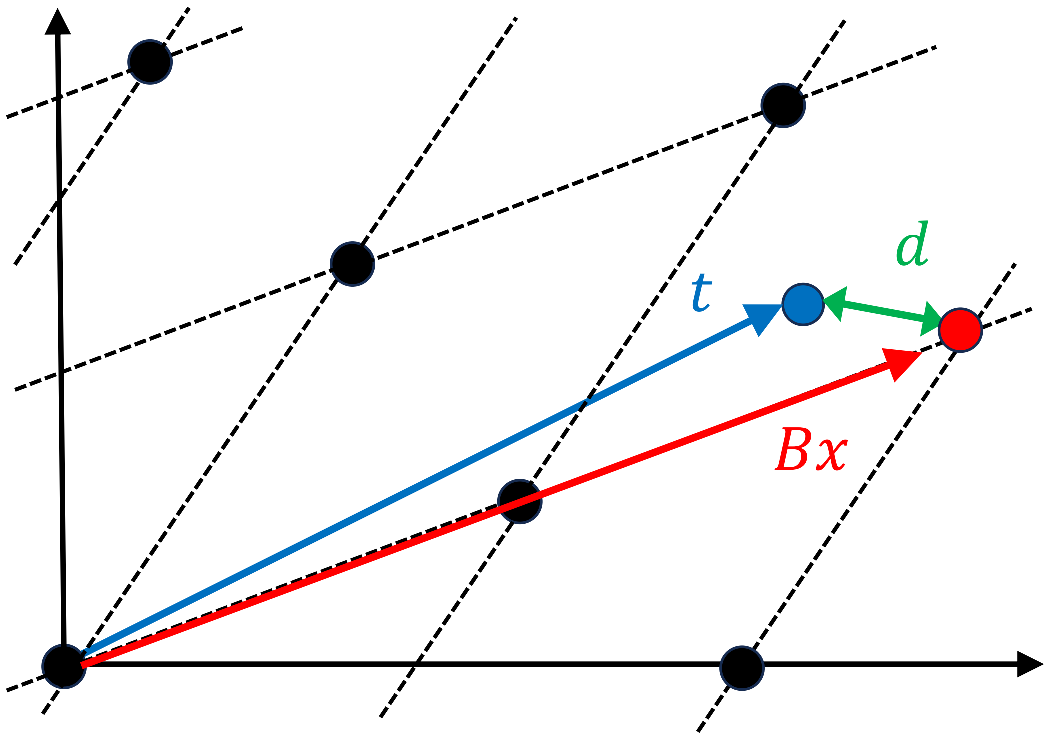
γ-近似最近ベクトル問題 γ-CVP_p

$$\gamma\text{-}CVP_p = (\Pi_{Yes},\ \Pi_{No})$$

$B \in \mathbb{R}^{m\times n}$: 格子の基底,　k: パラメータ

$$\Pi_{Yes} = \{((B,t),k)\ |d^{(p)}(L(B),t) \le k\}$$

$$\Pi_{No} = \{((B,t),k)\ |d^{(p)}(L(B),t) > \gamma k\}$$



格子点

$t \in \mathbb{Z}^m$: ターゲット

最近ベクトル

$d = d^{(p)}(L(B),t) = \|Bx - t\|_p$

パラメータ化 γ-SVP_p

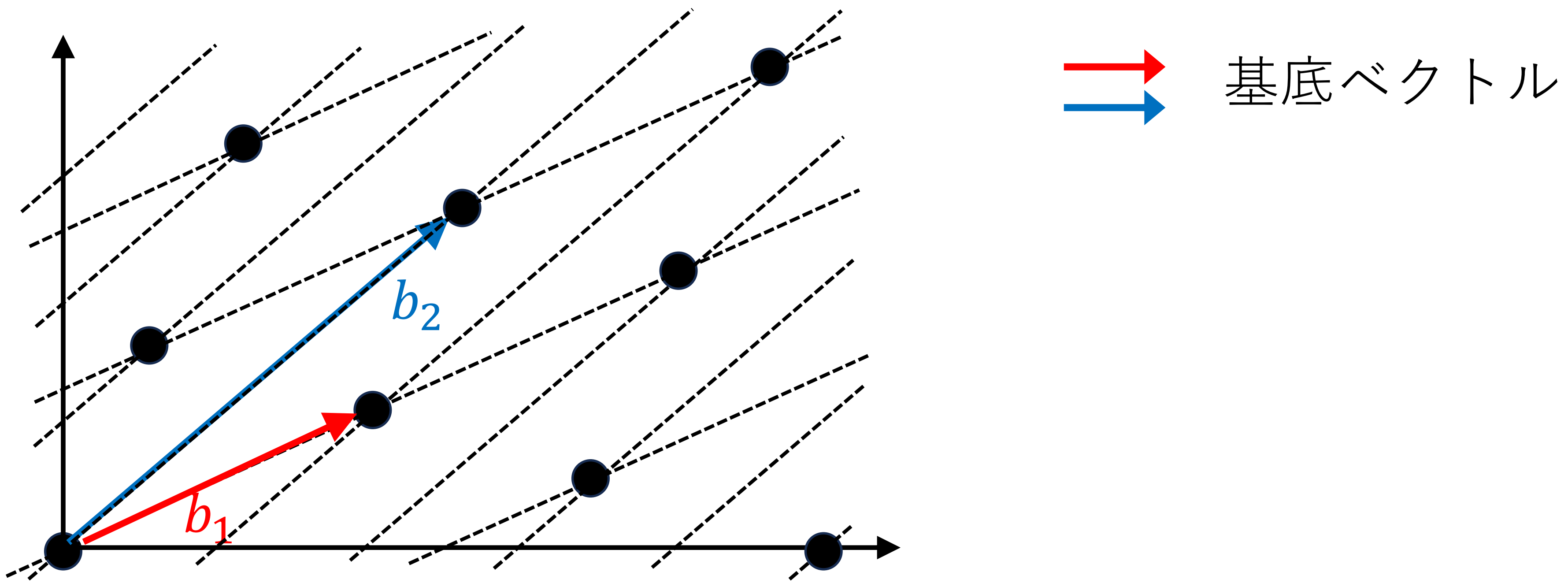
γ-近似最短ベクトル問題 γ-SVP_p

$\gamma\text{-}SVP_p = (\Pi_{Yes},\ \Pi_{No})$

$B \in \mathbb{R}^{m \times n}$: 格子の基底,　k: パラメータ

$\Pi_{Yes} = \{(B,k)\ |\lambda_1^{(p)}(L(B)) \leq k\}$

$\Pi_{No} = \{(B,k)\ |\lambda_1^{(p)}(L(B)) > \gamma k\}$



パラメータ化 γ-SVP_p

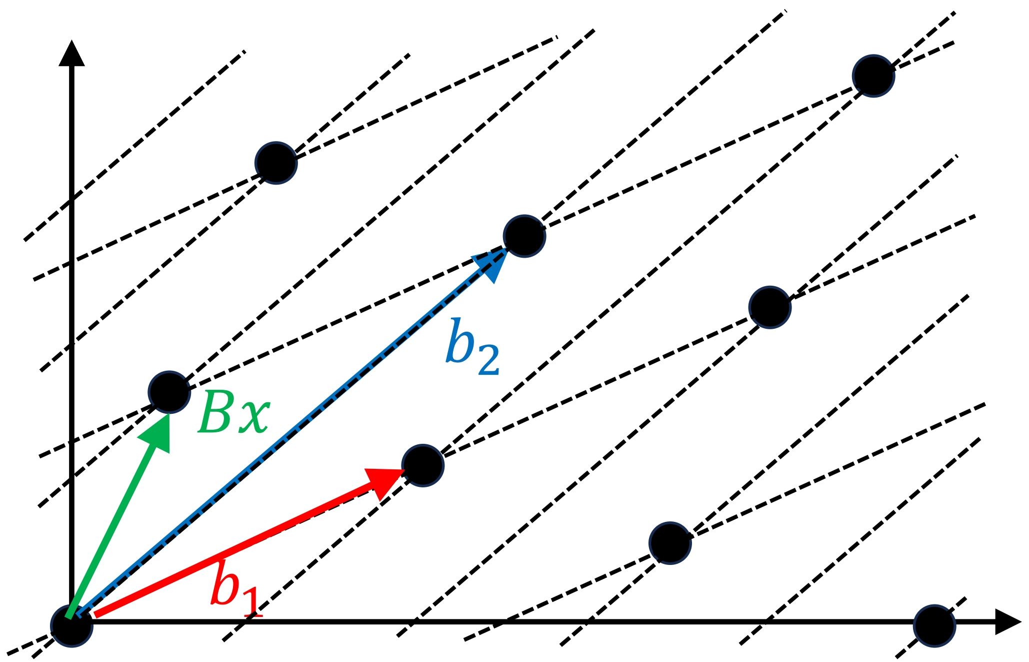
γ-近似最短ベクトル問題 γ-SVP_p

$$\gamma\text{-}SVP_p = (\Pi_{Yes},\ \Pi_{No})$$

$B \in \mathbb{R}^{m \times n}$: 格子の基底,　k: パラメータ

$$\Pi_{Yes} = \{(B, k)\ |\lambda_1^{(p)}(L(B)) \leq k\}$$

$$\Pi_{No} = \{(B, k)\ |\lambda_1^{(p)}(L(B)) > \gamma k\}$$



基底ベクトル

最短ベクトル

$$\lambda_1^{(p)}(L(B)) = \|Bx\|_p$$

符号と格子に関するパラメータ化計算量

$\gamma\text{-}NCP_q$	$\gamma \geq 1$
$q \geq 2$	**W[1]**困難 [BBE+21]

$\gamma\text{-}CVP_p$	$\gamma \geq 1$
$p \geq 1$	**W[1]**困難 [BBE+21]

$\gamma\text{-}MDP_q$	$\gamma \geq 1$
$q = 2$	**W[1]**困難 [BBE+21]
$q \geq 2$	**W[1]**困難 [BCGR23]

$\gamma\text{-}SVP_p$	$\gamma \in [1, 2^{1/p}]$	$\gamma \geq 1$
$p \in (1, \infty)$	**W[1]**困難 [BCGR23]	**W[1]**困難 [BCGR23]
$p \in [1, \infty)$	**W[1]**困難 [BCGR23]	Open problem

γ-NCP問題からγ-MDP問題への乱択FPT帰着[BCGR23]

符号と格子に関するパラメータ化計算量

γ-NCP_q	$\gamma \geq 1$
$q \geq 2$	**W[1]困難** [BBE+21]

γ-CVP_p	$\gamma \geq 1$
$p \geq 1$	**W[1]困難** [BBE+21]

γ-MDP_q	$\gamma \geq 1$
$q = 2$	**W[1]困難** [BBE+21]
$q \geq 2$	**W[1]困難** [BCGR23]

γ-SVP_p	$\gamma \in [1, 2^{1/p}]$	$\gamma \geq 1$
$p \in (1, \infty)$	**W[1]困難** [BCGR23]	**W[1]困難** [BCGR23]
$p \in [1, \infty)$	**W[1]困難** [BCGR23]	Open problem

γ-MDP_q のW[1]困難性の証明について

γ-NCP_q	$\gamma \geq 1$
$q \geq 2$	**W[1]**困難 [BBE+21]

γ-MDP_q	$\gamma \geq 1$
$q = 2$	**W[1]**困難 [BBE+21]
$q \geq 2$	**W[1]**困難 [BCGR23]

[BCGR23]

γ-NCP_q から γ-MDP_q への乱択FPT帰着により
γ-MDP_q の**W[1]**困難性を証明

証明に用いた技術

・Local dense code

・Affine to linear 帰着[Kho05]

時間の都合上，γ-CVP_p から γ-SVP_p への乱択FPT帰着
による**W[1]**困難性の証明[BCGR23]は省略.

証明のアイディアはγ-NCP_q から γ-MDP_q に似ている.
証明技術　Local dense code + Affine to linear 帰着

乱択FPT帰着

両側エラー乱択FPT帰着 f

　パラメータ約束問題 (Π_{Yes}, Π_{No}) をパラメータ約束問題 (Π'_{Yes}, Π'_{No}) に変換する関数 f で次を満たすもの.

・f はFPTアルゴリズムで計算できる

・ある計算可能関数 g が存在し $d \leq g(k)$

・$(x, k) \in \Pi_{Yes} \Rightarrow \Pr[(y, d) \in \Pi'_{Yes}] \geq 2/3$

・$(x, k) \in \Pi_{No} \Rightarrow \Pr[(y, d) \in \Pi'_{No}] \geq 2/3$

Locally dense code

$\mathcal{B}_m^{(0)}(r) = \{x \in \left(\mathbb{F}_q\right)^m | \|c\|_0 \leq r\}$

Locally dense code

$(A \in \mathbb{F}_q^{m \times n}, s \in \mathbb{F}_q^m)$ が (α, d, N)-locally dense codeとは,
次を２つを満たす符号のこと.

・$\lambda(A) \geq d$

・$|(\mathrm{C}(A) - s) \cap \mathcal{B}_m^{(0)}(\alpha d)| \geq N$

(α, d, N)-locally dense codeは BCH符号から構成できる.

$\boldsymbol{\gamma}$-*NCP* から $\boldsymbol{\gamma}$-*MDP* への帰着[BCGR23]

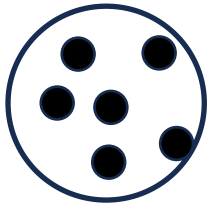
(G, s, k)： γ-*NCP*$_q$ の問題例

$(A \in \mathbb{F}_q^{m' \times n'}, s \in \mathbb{F}_q^{m'})$ ： (α, d, N)-locally dense code（$\gamma k = d > k'$）

Step 1：生成行列 A, G を組み合わせて新たな符号を作る.

$$G' \coloneqq \begin{pmatrix} G & 0 & -t \\ 0 & A & -s \end{pmatrix} \qquad k' \coloneqq k + \alpha d$$

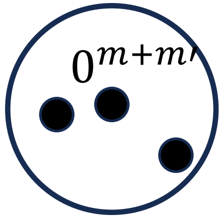
中心 $0^{m+m'}$ Hw半径k'
範囲内の符号語



(G, s, k) が γ-*NCP*$_q$ Yes問題例 → $|\mathrm{C}(G') \cap \mathcal{B}_{m+m'}^{(0)}(k')| \geq N$

(G', k') が γ'-*MDP*$_q$ Yes問題例である.

中心 $0^{m+m'}$ Hw半径γkの
範囲内の符号語



(G, s, k) が γ-*NCP*$_q$ No問題例 → $|\mathrm{C}(G') \cap \mathcal{B}_{m+m'}^{(0)}(\gamma k)| \leq |\mathcal{B}_m^{(0)}(\gamma k)|$

(G', k') が γ'-*MDP*$_q$ No問題例といえそうにない.

$\boldsymbol{\gamma}$-*NCP*から$\boldsymbol{\gamma}$-*MDP*への帰着[BCGR23]

$$G' \coloneqq \begin{pmatrix} G & 0 & -t \\ 0 & A & -s \end{pmatrix} \qquad k' \coloneqq k + \alpha d$$

(G, s, k) が γ-NCP_q のYes問題のとき

(G, s, k) がYes問題例なら$\|Gx - t\|_0 \leq k$ となる x が存在する.

(A, s) は(α, d, N)-locally dense code なので,

$\|Ay - s\|_0 \leq \alpha d$ を満たす y が N個以上存在する.

$$\|G'(x, y, 1)\|_0 = \|Gx + Ay - s - t\|_0 \leq \|Gx - t\|_0 + \|Ay - s\|_0 \leq k + \alpha d = k'.$$

$|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(k')| \geq N$ なので(G', k') は γ'-MDP_q の Yesの問題例である.

γ-NCP から γ-MDP への帰着[BCGR23]

$$G' = \begin{pmatrix} G & 0 & -s \\ 0 & A & -t \end{pmatrix} \times \begin{pmatrix} x \\ y \\ \beta \end{pmatrix} = \begin{pmatrix} Gx - \beta s \\ Ay - \beta t \end{pmatrix}$$

G' の構造により

$$\|G'(x,y,\beta)\|_0 \geq \|Gx - \beta s\|_0$$

$$\|G'(x,y,\beta)\|_0 \geq \|Ay - \beta t\|_0$$

(G,s,k) が γ-NCP_q のNo問題例 → $\forall x \in \mathbb{F}_q^n,\ \beta \in \mathbb{F}_q \setminus \{0\},\quad \|Gx\ - \beta s\|_0 \geq \gamma k = d$

↑ γ-NCP_q の定義に scaling argumentを適用すると示せる

$\|G'(x,y,\beta)\|_0$ について

$\beta \neq 0$ のとき， $\|G'(x,y,\beta)\|_0 \geq \|Gx - \beta s\|_0 > \gamma k$

$\beta = 0,\ y \neq 0$ のとき， $\|G'(x,y,\beta)\|_0 \geq \|Ay - \beta t\|_0 > d = \gamma k$

$\beta = 0,\ y = 0$ のとき， $\|G'(x,y,\beta)\|_0 = \|Gx\|_0$

$$|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k)| \leq |\mathcal{B}^{(0)}_m(\gamma k)|$$

$\boldsymbol{\gamma}$-*NCP* から $\boldsymbol{\gamma}$-*MDP* への帰着[BCGR23]

(G,s,k) が γ-NCP_q Yes問題例 → $|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(k')| \geq N$

(G,s,k) が γ-NCP_q No問題例 → $|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k)| \leq |\mathcal{B}^{(0)}_{m}(\gamma k)|$ （$\gamma k = d > k'$）

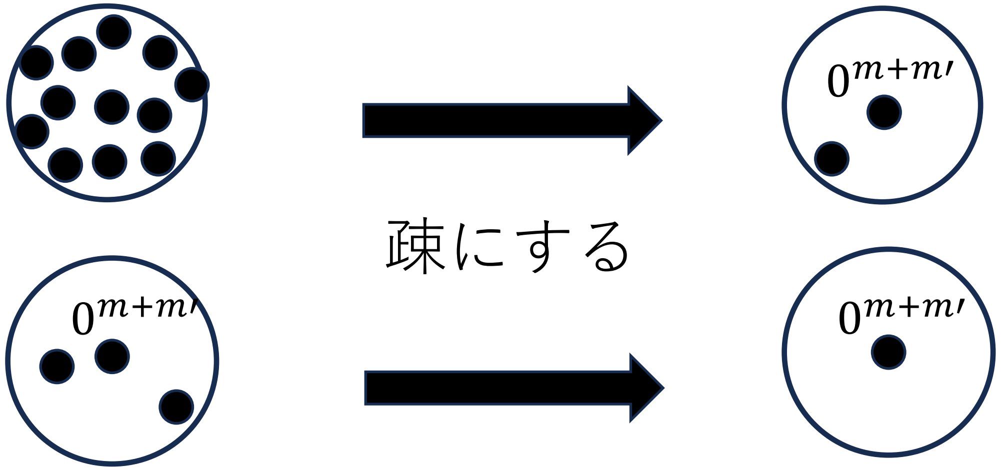
Step 2：$N > 100\,|\mathcal{B}^{(0)}_{m}(\gamma k)|$，ランダム符号と$C(G')$ の共通部分をとり．符号$C(G'')$ をつくる． おおよそ確率$1/N$ で$C(G')$の各符号が残るほどにパラメータ調節する．

γ-NCP_q Yes問題例

$\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(k')$

γ-NCP_q No問題例

$\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k)$



$\mathrm{C}(G'') \cap \mathcal{B}^{(0)}_{m+m'}(k')$

$\mathrm{C}(G'') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k)$

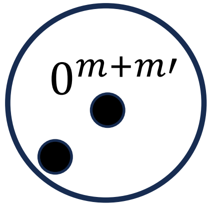
γ-NCP からγ-MDP への帰着[BCGR23]

(G,s,k) が γ-NCP_q Yes問題例 → $|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(k')| \geq N$

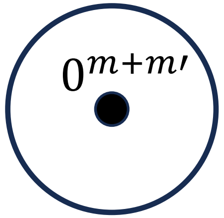
(G,s,k) が γ-NCP_q No問題例 → $|\mathrm{C}(G') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k)| \leq |\mathcal{B}^{(0)}_{m}(\gamma k)|$ （$\gamma k = d > k'$）

Step 2：$N > 100\,|\mathcal{B}^{(0)}_{m}(\gamma k)|$，ランダム符号と$C(G')$ の共通部分をとり．符号$C(G'')$ をつくる． おおよそ確率$1/N$ で$C(G')$の各符号が残るほどにパラメータ調節する．

(G,s,k) が γ-NCP_q Yes問題例 → $|\mathrm{C}(G'') \cap \mathcal{B}^{(0)}_{m+m'}(k')| \geq 2$ が
ある程度高い確率で期待できる．



(G,s,k) が γ-NCP_q No問題例 → $\mathrm{C}(G'') \cap \mathcal{B}^{(0)}_{m+m'}(\gamma k) = \{0^{m+m'}\}$ が
ある程度高い確率で期待できる．



参考文献

[AB09] Samjeev Arora and Boaz Barak, COMPUTATIONAL COMPLEXITY A Modern Approach, CAMBRIDGE University Press 2009.

[Ben23] The Complexity of the Shortest Vector Problem SIGATC News 54(1), 2023

[BBE+21] Arnab Bhattacharyya, Édouard Bonnet, László Egri, Suprovat Ghoshal, Karthik C. S., Bingkai Lin, Pasin Manurangsi, and Dániel Marx. Parameterized Intractability of Even Set and Shortest Vector Problem. ACM 68, 3, Article 16 2021.

[BCGR23] Huck Bennett, Mahdi Cheraghchi, Venkatesan Guruswami, and João Ribeiro. Parameterized Inapproximability of the Minimum Distance Problem over All Fields and the Shortest Vector Problem in All lp Norms. STOC 2023

[Che04] Qi Cheng. On the Bounded Sum-of-Digits Discrete Logarithm Problem in Finite Fields. CRYPTO 2004.

[DF13] Rodney G. Downey and Michael R. Fellows. Fundamentals of Parameterized Complexity, Springer 2013.

[Din16] Irit Dinur. Mildly exponential reduction from gap 3SAT to polynomial-gap label-cover. ECCC, 23:128, 2016.

[FG06] Jörg Flum and Martin Grohe. Parameterized Complexity Theory. Springer 2006.

[FK93] Michael R. Fellows and Neal Koblitz. Fixed-parameter complexity and cryptography. AAECC-10, 1993.

[GLR+24] Venkatesan Guruswami and Bingkai Lin and Xuandi Ren and Yican Sun and Kewen Wu. Parameterized Inapproximability Hypothesis under Exponential Time Hypothesis. STOC 2024.

[GRS24] Venkatesan Guruswami and Xuandi Ren and Sai Sandeep. Baby PIH: Parameterized Inapproximability of Min CSP. CCC 2024

参考文献

[IP01] Russell Impagliazzo and Ramamohan Paturi. On the Complexity of k-SAT. J. Comput. Syst. Sci. 62. 2. 2001.
[Koh04] Subhash Khot. Lecture 6: Minimum distance of a linear code https://cs.nyu.edu/~khot/pcp- lecnotes/lec6.ps.
[LRSZ20] Daniel Lokshtanov and M. S. Ramanujan and Saket Saurabh and Meirav Zehavi. Parameterized Complexity and Approximability of Directed Odd Cycle Transversal. SODA 2020
[Man19] Pasin Manurangsi. Parameterized Inapproximability. https://pasin30055.github.io/ALGO19.pdf
[Mar21] Dániel Marx. Complexity of parameterized problems, https://www.mpi-inf.mpg.de/fileadmin/inf/d1/teaching/winter21/paraalg/Lectures/lecture_4.pdf
[Oka12] 岡本 吉央, **2**と**3**の違い. http://dopal.cs.uec.ac.jp/okamotoy/lect/2012/osakafu-u/difference2-3.pdf 2012.
[Oka14] 岡本 吉央, **2**と**3**の違い. http://dopal.cs.uec.ac.jp/okamotoy/PDF/2014/la2014winter.pdf 2014.
[Sip12] Michael Sipser. Introduction to the Theory of Computation 3rd edition. Cengage Learning 2012.

補足資料

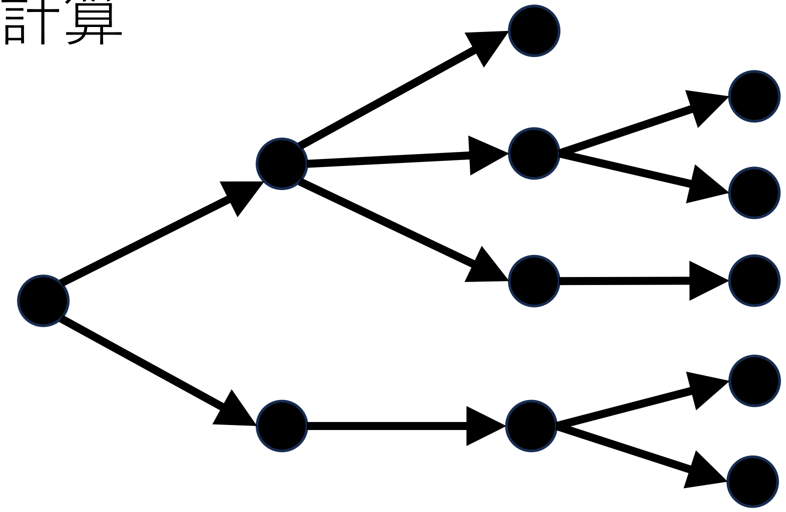
非パラメータ版 $\boldsymbol{\gamma}$-SVP の近似因子と困難さ

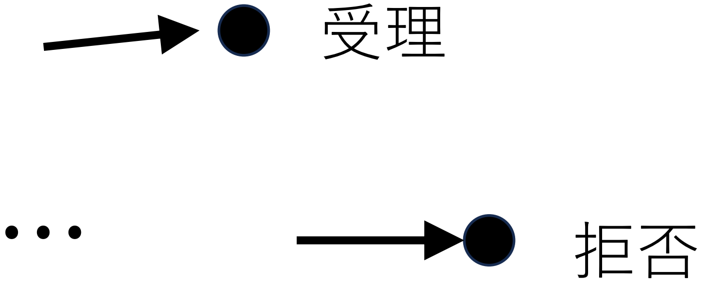
n 次元格子でのγ-SVP_2 の困難さ [Ben23]

	Hardness	Hardness barriers		Cryptography		Algorithms
近似因子 $\gamma(n)$	$O(1)$	$O(\sqrt{n/\log n})$	$O(\sqrt{n})$	$O(n)$	$O(n^2)$	$O(2^{n\mathrm{loglog}n/\mathrm{log}n})$
	NP $\not\subseteq$ **RP**	γ-SVP $\in$ **coAM**	γ-SVP $\in$ **coNP**	SIS が困難	LWE が困難	γ-$SVP \in$ **P**

決定性計算と非決定性計算の違い

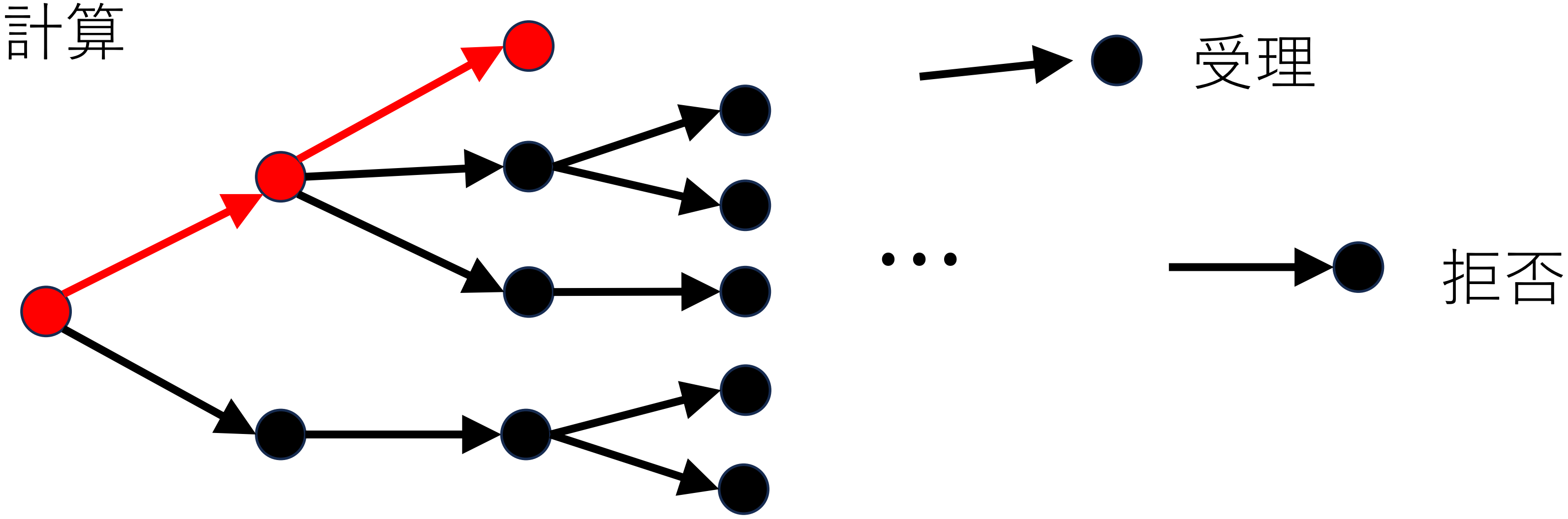
非決定性計算



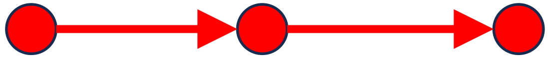


決定性計算と非決定性計算の違い

非決定性計算

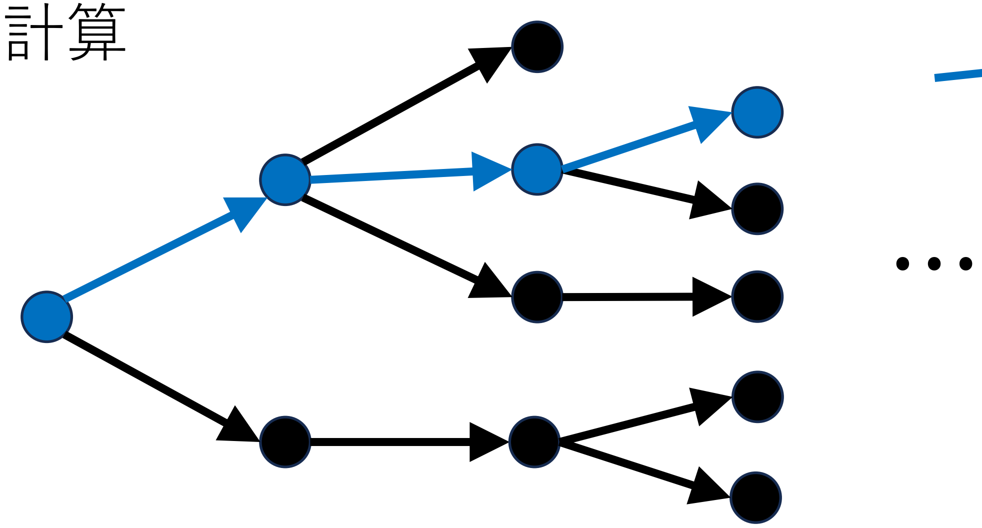


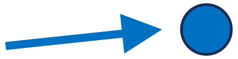
計算の枝を並べてみる



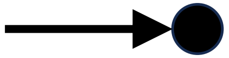
決定性計算と非決定性計算の違い

非決定性計算



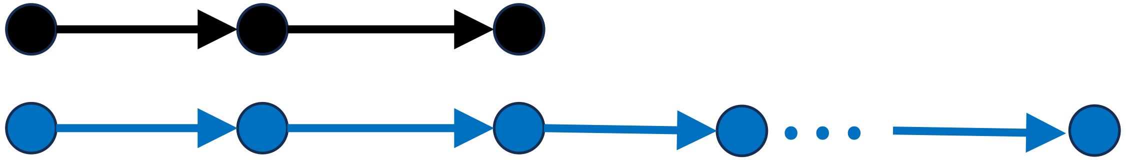


受理



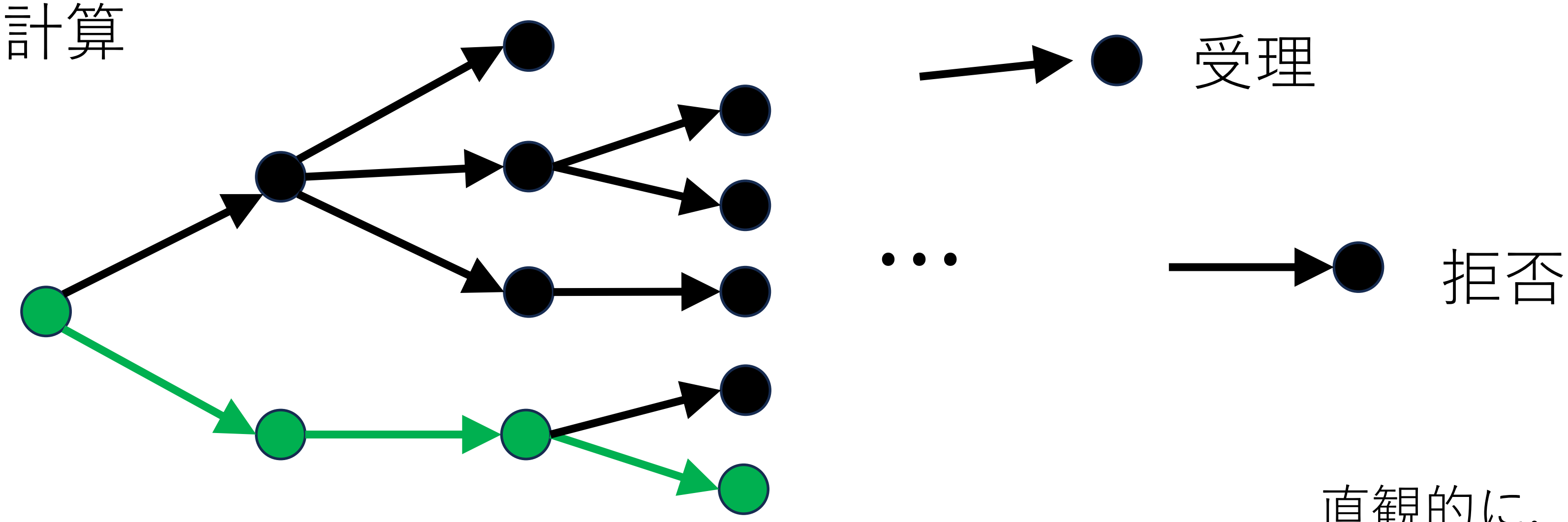
拒否

計算の枝を並べてみる

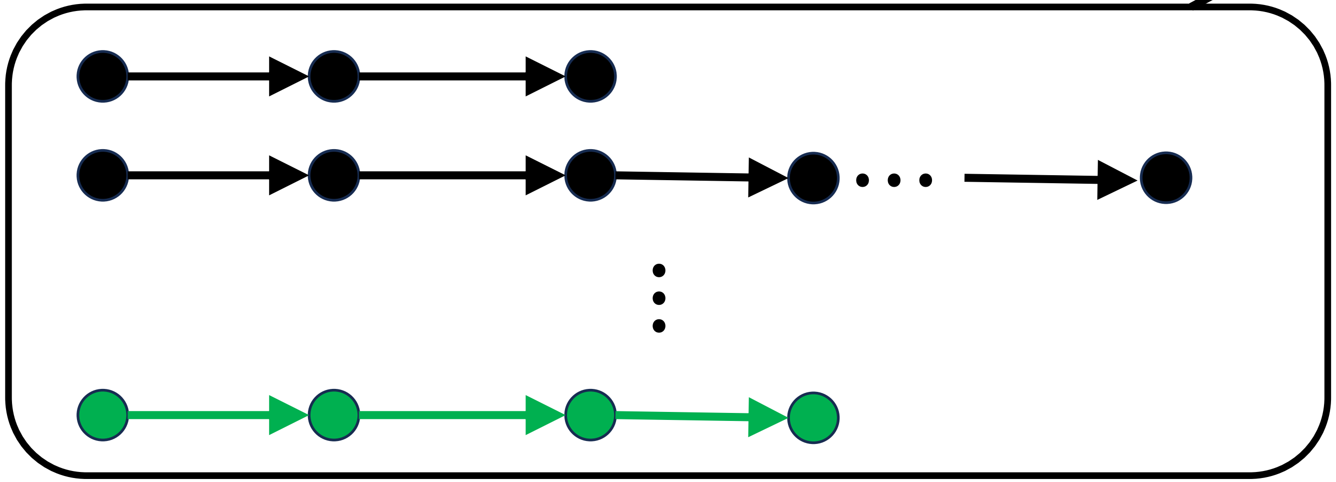


決定性計算と非決定性計算の違い

非決定性計算



計算の枝を並べてみる



直観的に，非決定性計算では，
決定性計算を並列に実行できる
程度の能力がある．