最適化技術に基づく

LDPC符号の復号法について

和田山 正

wadayama@nitech.ac.jp

名古屋工業大学

自己紹介

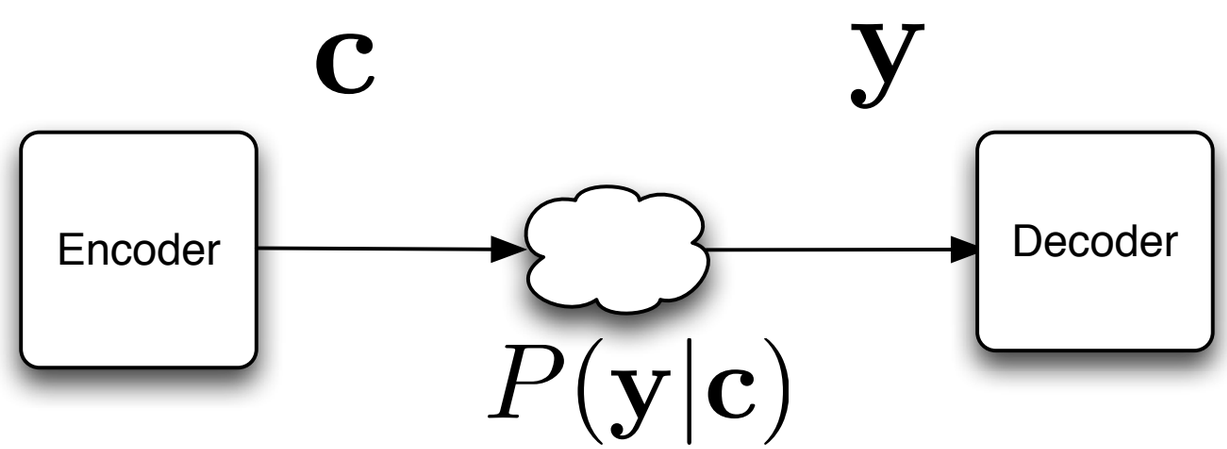
和田山 正

名古屋工業大学大学院工学研究科 教授
(情報工学専攻，情報工学科担当)

最近の研究上の興味: 誤り訂正符号，情報理論，
無線通信信号処理，深層学習技術

誤り訂正符号の本を書いています→

情報通信における復号問題



- 情報通信システムにおいて、復号器における「復号（推定）アルゴリズム」はシステムの要となる部分
- 望ましい復号アルゴリズム:
  - 誤り率が小さい
  - 計算量が少ない・高速動作可能である
  - 実装しやすい

2つのアプローチ

- **確率的アプローチ**：受信側で送信信号の事後確率を計算し、事後確率に基づき送信信号を推定する(ベイズ推定)→ビリーフプロパゲーション(BP)
- **最適化アプローチ**：復号問題を一種の組み合わせ最適化問題として定式化し、**数理最適化手法に基づくアルゴリズム**により、その最適化問題を解く(最尤推定問題を最適化問題として定式化)

典型的な問題

2元符号 $C \subseteq \{+1, -1\}^n$

- 興味がある符号長 n は数百〜数万
- $\log_2 |C| = Rn$ (R は 1 未満の正定数: 符号化率)

最短距離復号問題

与えられた $y \in \mathbb{R}^n$ に対して

$$\hat{x} = \arg\min_{x \in C} \|y - x\|_2^2$$

を見い出せ。

- 復号のために C は <u>適切な構造</u> を持つことが望まれる
- C が 2 元線形符号の場合、NP 困難性が示されている

復号問題への最適化アプローチ

凸緩和・LP 緩和

$$\hat{x} = \arg \min_{x \in conv(C)} \|y - x\|_2^2$$

- $conv(C)$ は C の凸包
- 上記の問題は凸計画問題になる
- 実際には、C の凸包を含む近似凸包を実行可能領域とする

非線形最適化

$$\hat{x} = \arg \min_{x} \|y - x\|_2^2 + p(x)$$

- p は符号語制約を与える非線形関数

本講演の目標

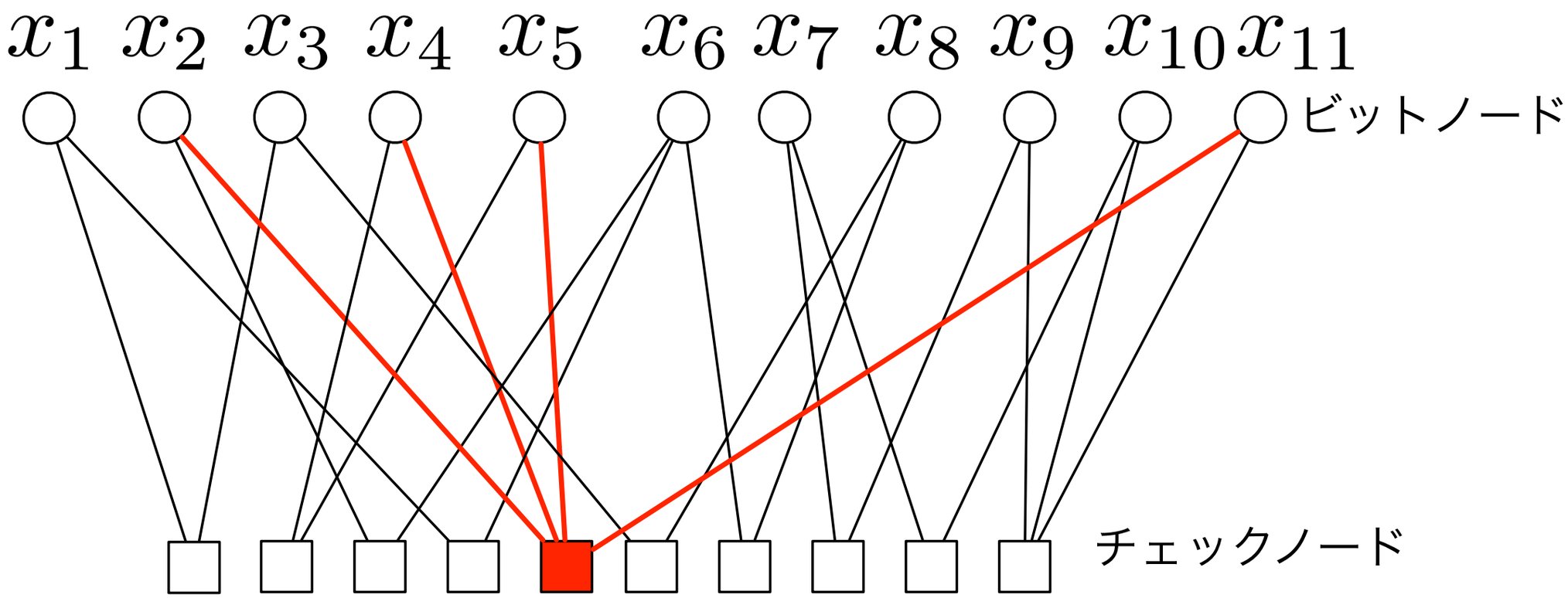
- LDPC符号の復号問題を例として、誤り復号問題への最適化アプローチを紹介する
- 実際に構成された過去のアルゴリズムを概観する→研究の流れを見る
- 新しい復号アルゴリズム・信号推定アルゴリズムを創り出すときに連続最適化・組み合わせ最適化分野で培われた知見が活かせる可能性があることを見ていく

本講演の構成

- 問題設定
  - LDPC符号の紹介
- ビットフリップ型復号法
  - ビットフリップ復号法の紹介
  - GDBF(勾配降下型ビットフリップ)復号法
- 凸緩和に基づく復号法
  - LP復号法
  - 射影勾配法に基づく復号法

LDPC符号の紹介

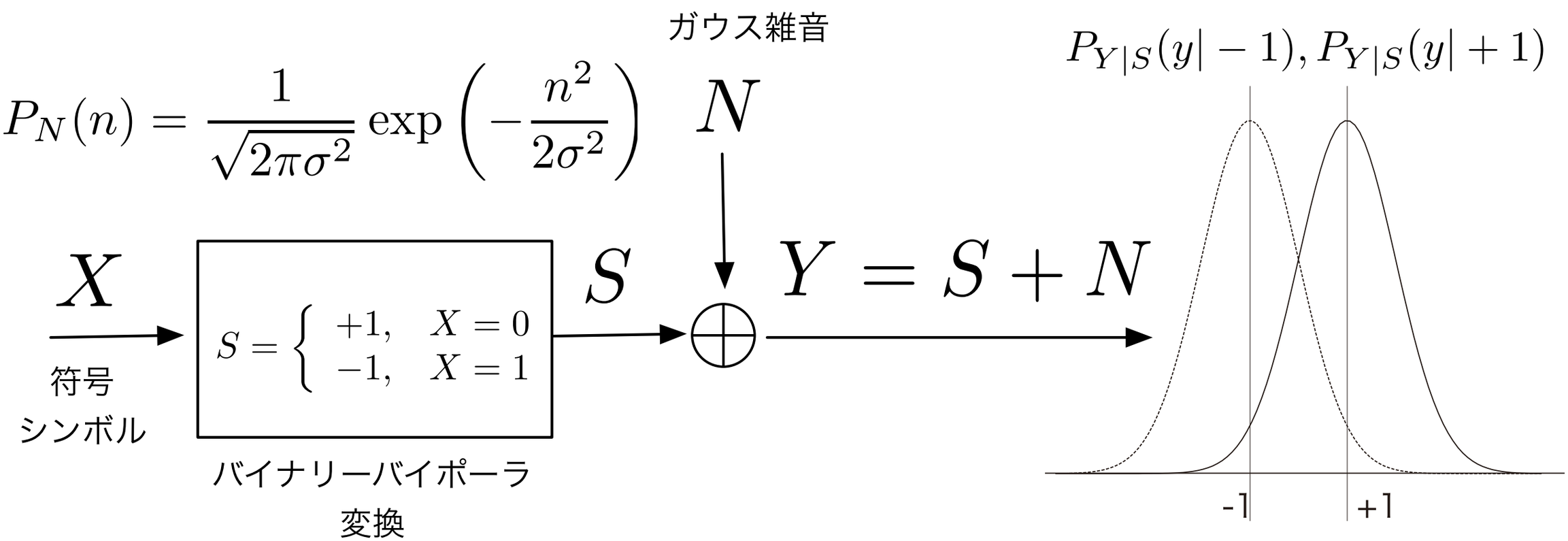
- ランダムに 2 部グラフを作成
- ビットノード側には、$(x_1, x_2, \ldots, x_n) \in \{0,1\}^n$ を割り当てる
- ひとつのチェックノードは、偶パリティ制約を表す。
- 充足解の集合 $=$ LDPC 符号 C



$$x_2 \oplus x_4 \oplus x_5 \oplus x_{11} = 0 (\mathrm{mod}\ 2)$$

AWGN通信路モデル

$$P_N(n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{n^2}{2\sigma^2}\right)$$



最尤復号法

最尤復号法は、最小のブロック誤り率を与える。

最尤復号

受信ベクトル $y \in \mathbb{R}^n$ に対して

$$\hat{x} = \arg\min_{x \in C} \|y - b(x)\|_2^2$$

を見い出せ。

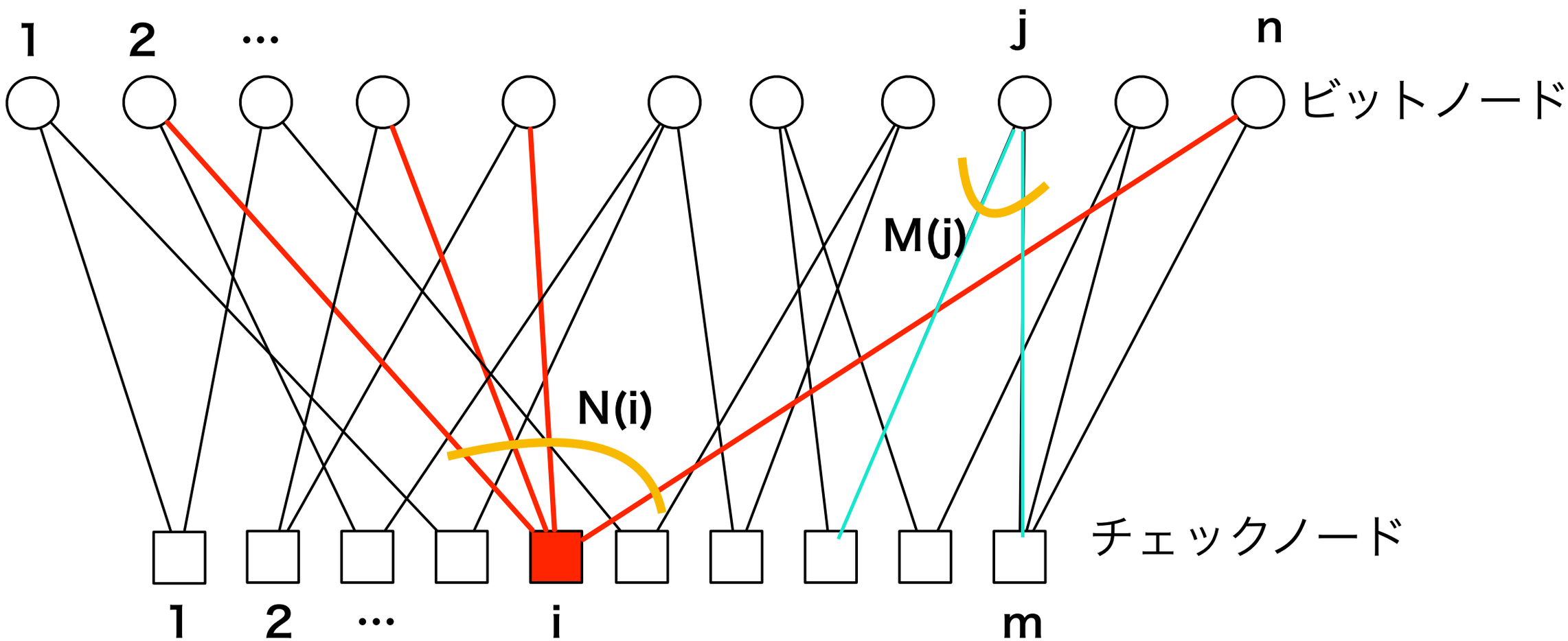
- C: LDPC 符号
- b: バイナリ-バイポーラ変換関数

残念ながら最尤復号法の正確な実行は、符号長に対して多項式時間では不可能（と信じられている）。

本講演の構成

- 問題設定
  - LDPC符号の紹介
- ビットフリップ型復号法
  - ビットフリップ復号法の紹介
  - GDBF(勾配降下型ビットフリップ)復号法
- 凸緩和に基づく復号法
  - LP復号法
  - 射影勾配法に基づく復号法

2部グラフにおける近傍ノード集合の表記



ビットフリップ型復号法

・記憶素子（フリップフロップ）の少ない復号器を目指して
・高速・低消費電力・小さいチップサイズを目指して

ステップ１

すべての受信シンボルについて硬判定を行う

$$x_j = \mathbf{sign}(y_j), \quad j \in [1,n]$$

ステップ2

パリティシンボルを計算する

$$s_i = \prod_{j \in N(i)} x_j, \quad j \in [1,m]$$

反復計算

ステップ3

ビットフリップ処理を行う $x_\ell := -x_\ell$

$$\ell = \arg\min_{k \in [1,n]} \underline{\Delta_k(\mathbf{x})}$$

反転関数(ビット信頼度)

ステップ２に戻る

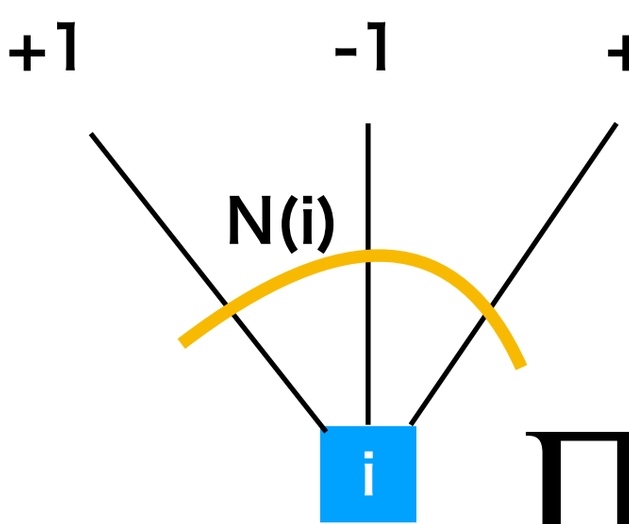
様々なビットフリップ型復号法

Gallager $$\Delta_k(\mathbf{x}) = \sum_{i \in M(k)} \prod_{j \in N(i)} x_j$$

WBF $$\Delta_k(\mathbf{x}) = \sum_{i \in M(k)} \beta_i \prod_{j \in N(i)} x_j, \quad \beta_i = \min_{j \in N(i)} |y_i|$$

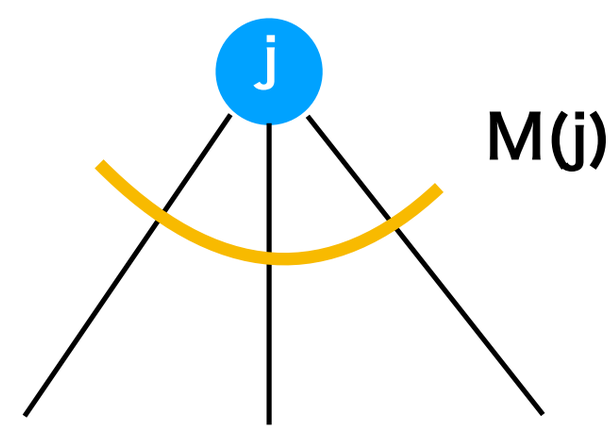
MWBF $$\Delta_k(\mathbf{x}) = \alpha |y_n| + \sum_{i \in M(k)} \beta_i \prod_{j \in N(i)} x_j, \quad \beta_i = \min_{j \in N(i)} |y_i|$$

バイポーラシンドローム値



$$\prod_{j \in N(i)} x_j$$

ビット信頼度



GDBF復号法の導出

Gradient Descent Bit Flipping Algorithm (Wadayama et al., 2010)

$\tilde{C} = \{b(x) \mid x \in C\}$: バイポーラ版 LDPC 符号

$$
\begin{aligned}
\hat{\mathbf{x}} &= \arg\min_{\mathbf{x}\in\tilde{C}} \|\mathbf{y}-\mathbf{x}\|_2^2 \\
&= \arg\min_{\mathbf{x}\in\tilde{C}} \sum_{i=1}^{n} (y_i - x_i)^2 \\
&= \arg\min_{\mathbf{x}\in\tilde{C}} \sum_{i=1}^{n} (y_i^2 - 2x_i y_i + 1) \\
&= \arg\max_{\mathbf{x}\in\tilde{C}} \sum_{i=1}^{n} x_i y_i .
\end{aligned}
$$

復号問題の定式化

目的関数

$$f(x) \triangleq \sum_{j=1}^{n} x_j y_j + \sum_{i=1}^{m} \prod_{j \in N(i)} x_j.$$

- 第 1 項: 受信ベクトルとバイポーラ符号語との相関
- 第 2 項: ペナルティ項 (x が符号語のときに最大となる)

対応する非線形最適化問題

$$\tilde{x} = \arg \max_{x \in [+1,-1]^n} \left(\sum_{j=1}^{n} x_j y_j + \sum_{i=1}^{m} \prod_{j \in N(i)} x_j \right).$$

反転関数の導出

変数 $x_k (k \in [1, n])$ に対応する偏導関数は

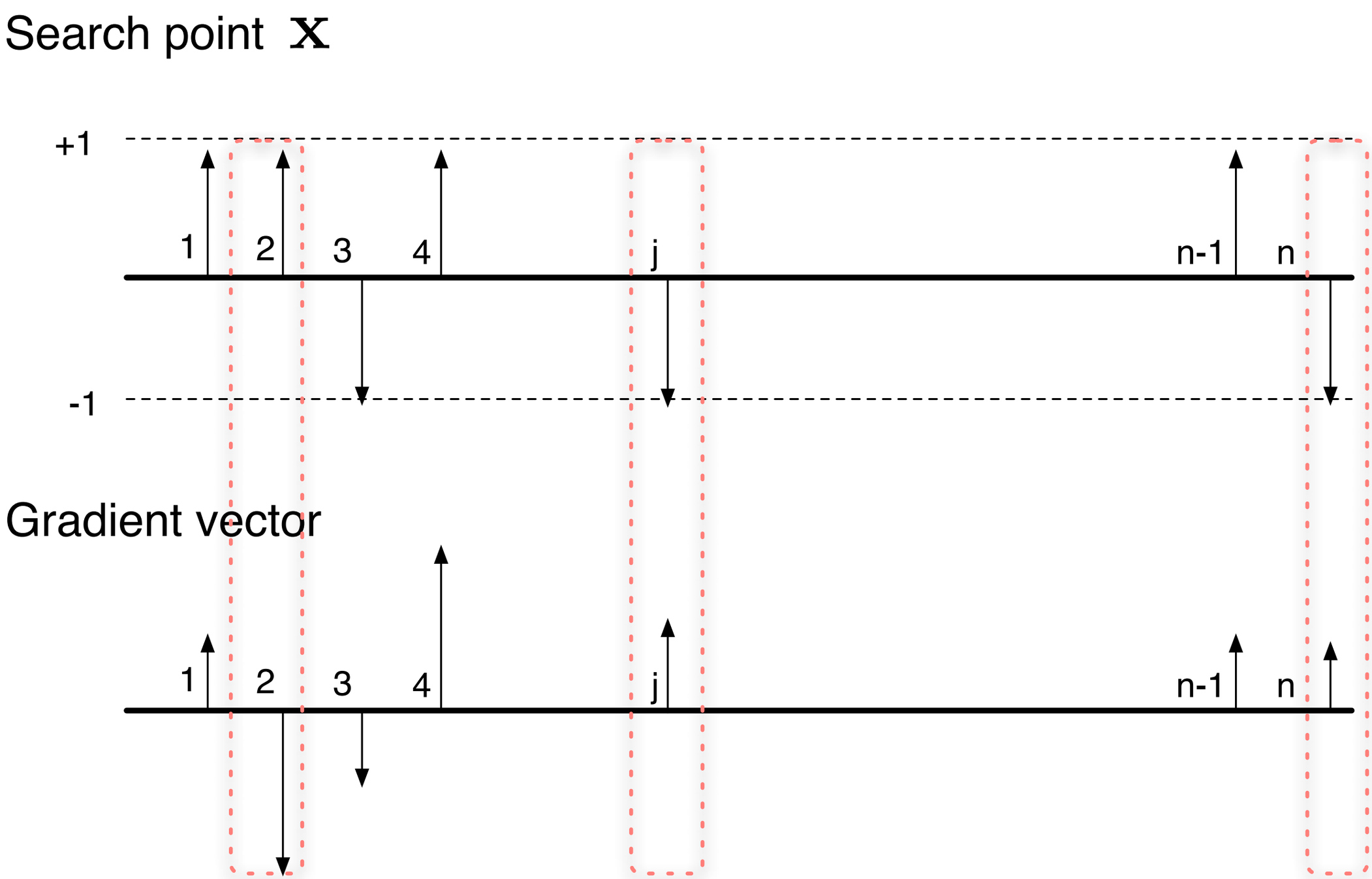
$$\frac{\partial}{\partial x_k} f(\mathbf{x}) = y_k + \sum_{i \in M(k)} \prod_{j \in N(i) \setminus k} x_j.$$

と与えられる。

反転関数

$$\begin{aligned} \Delta_k^{(GD)}(\mathbf{x}) &\triangleq x_k \frac{\partial}{\partial x_k} f(\mathbf{x}) \\ &= x_k y_k + \sum_{i \in M(k)} \prod_{j \in N(i)} x_j. \end{aligned}$$

探索ベクトルの要素の正負と勾配ベクトル



GDBFアルゴリズム

- Find $\ell := \arg\min_{k\in[1,n]} \Delta_k^{(GD)}(\mathbf{x})$
- Flip the bit: $x_\ell := -x_\ell$

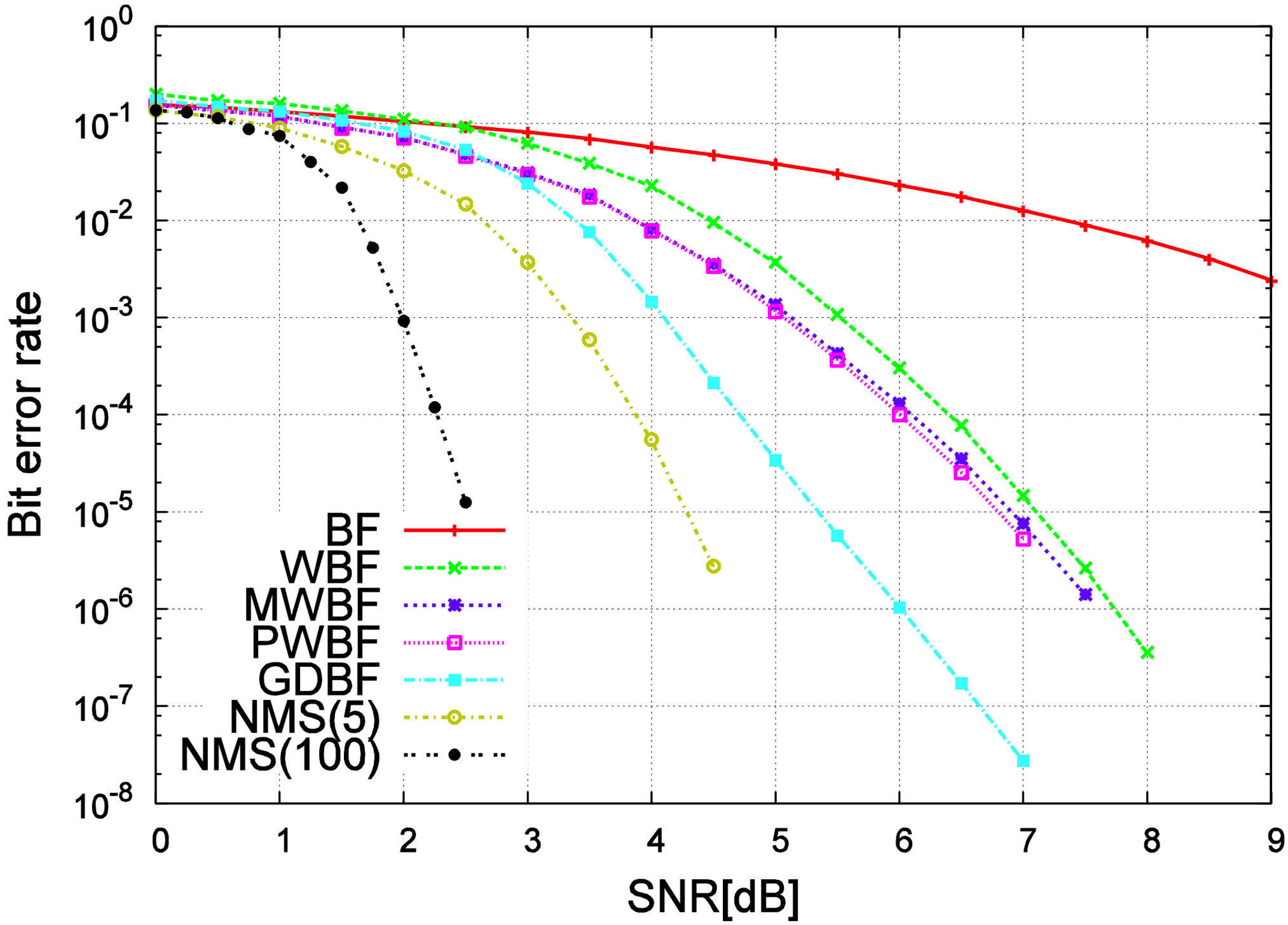
上記のプロセスは、ビットフリップ型勾配法と見なすことができる (coordinate decent/accent algorithm).

GDBF algorithm

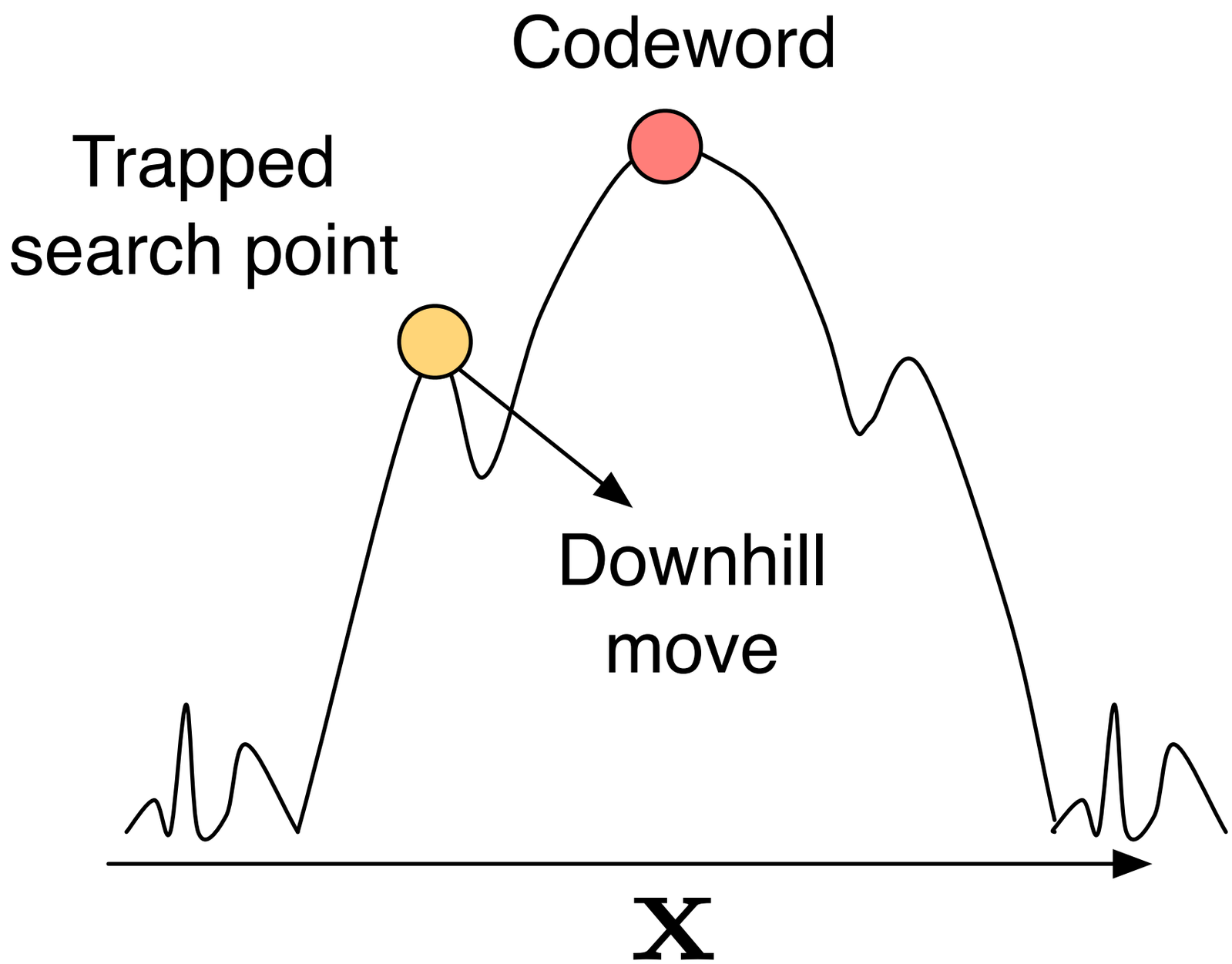
次の反転関数を持つシングルビット反転法を GDBF アルゴリズムと呼ぶ [4] :

$$\Delta_k^{(GD)}(\mathbf{x}) \triangleq x_k y_k + \sum_{i\in M(k)} \prod_{j\in N(i)} x_j$$

ビット誤り率の比較



局所解からの脱出



Noisy GDBF アルゴリズム

Noisy GDBF 反転関数 (Sundararajan at el. (2014))

$$\Delta_k^{(Noisy)}(x) \triangleq x_k y_k + \alpha \sum_{i \in M(k)} \prod_{j \in N(i)} x_j + q_i$$

- α: 定数
- q_i: ガウス乱数項

BP による復号性能にかなり近い値を達成しており、現時点での BF 型アルゴリズムにおけるベストパフォーマンスを出している。

本講演の構成

- 問題設定
  - LDPC符号の紹介
- ビットフリップ型復号法
  - ビットフリップ復号法の紹介
  - GDBF(勾配降下型ビットフリップ)復号法
- 凸緩和に基づく復号法
  - LP復号法
  - 射影勾配法に基づく復号法

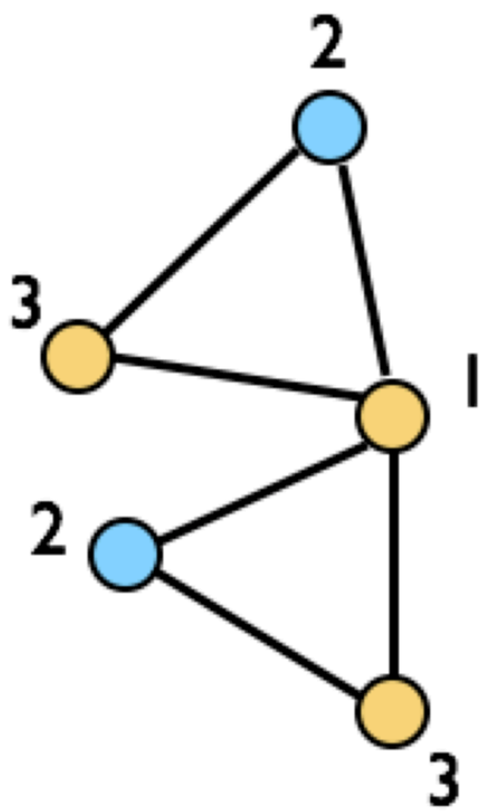
頂点被覆問題とそのLP緩和問題

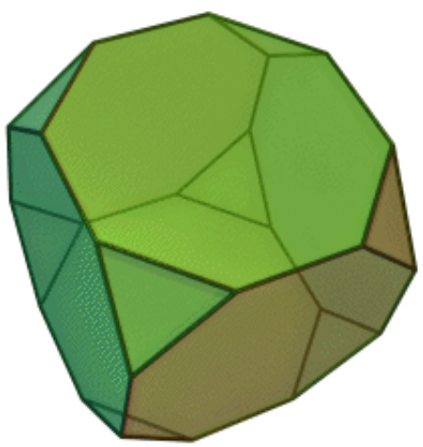
$$\text{minimize} \sum_i \gamma_i v_i$$

subject to

$$\forall i \in V, 0 \leq v_i \leq 1$$

$$\forall (i,j) \in E, v_i + v_j \geq 1$$

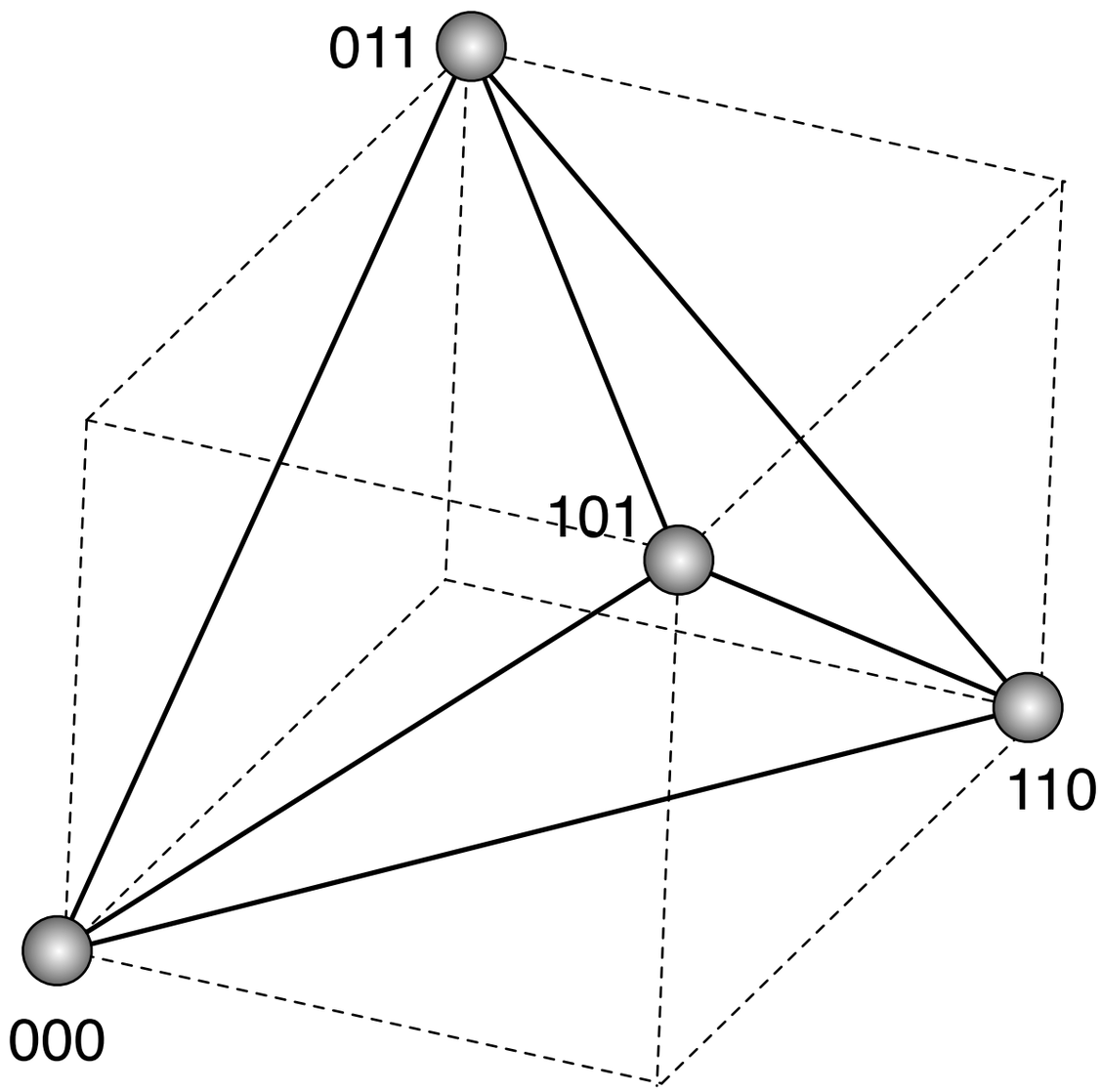




偶重み符号の凸包

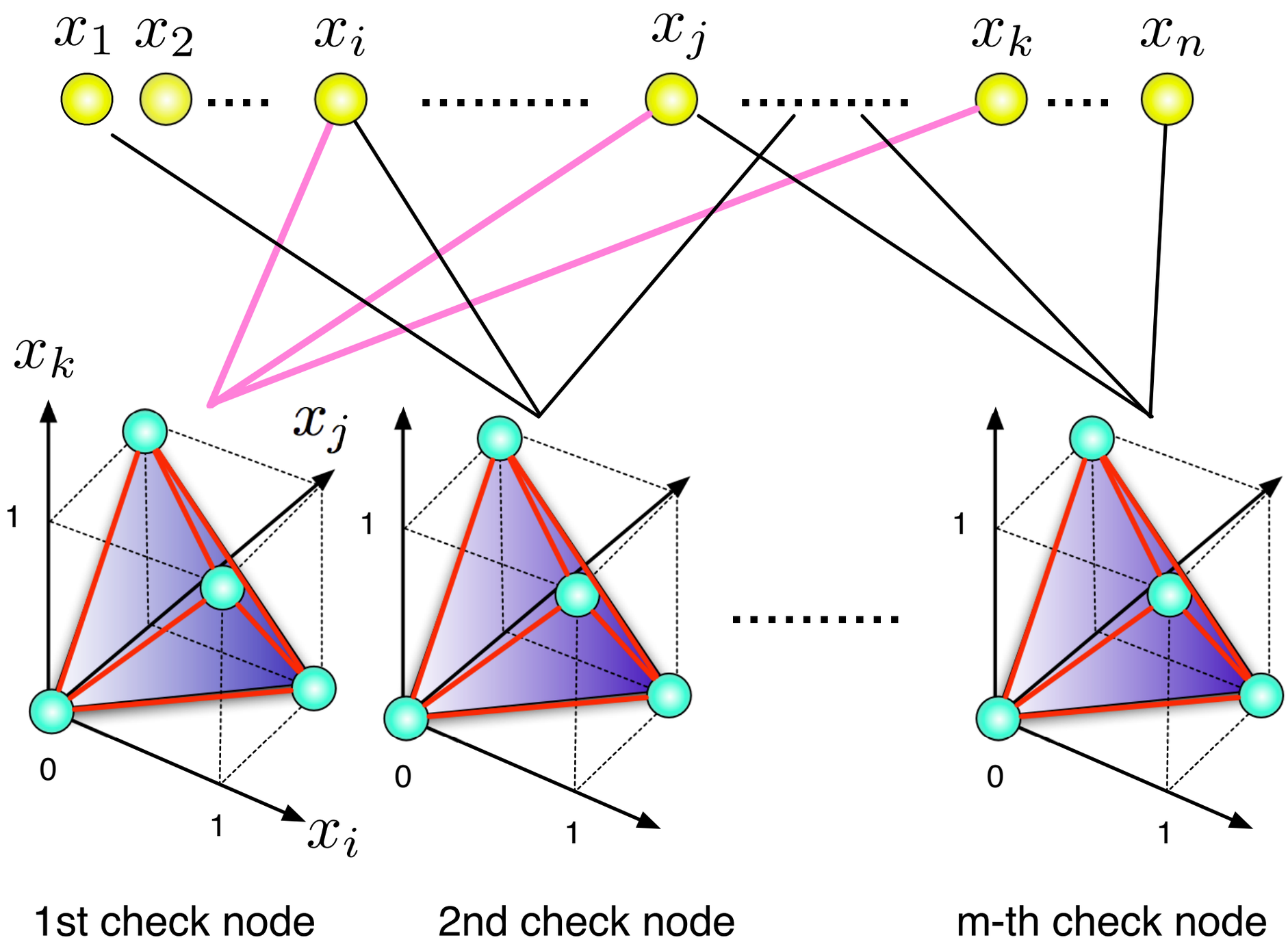
偶重み符号(n=3)

C = {000, 110, 101, 011}



$$
\begin{aligned}
x + (1-y) + (1-z) &\leq 2 \\
y + (1-x) + (1-z) &\leq 2 \\
z + (1-x) + (1-y) &\leq 2 \\
x + y + z &\leq 2
\end{aligned}
$$

LDPC符号の基本多面体



LDPC符号の基本多面体

基本多面体を定義する不等式制約

<u>パリティ不等式</u>

$\forall i \in [1, m], \forall S \in T_i$

$$1 + \sum_{t \in S} (x_t - 1) - \sum_{t \in A_i \setminus S} x_t \leq 0$$

<u>ボックス不等式</u>

$\forall j \in [1, n], \forall S \in T_i$

$$0 \leq x_j \leq 1$$

これらの制約を満たす点の集合を基本多面体 $\mathcal{P}$ と呼ぶ。
(Feldman et al. 2002, Koetter and Vontobel)

LP復号法

最尤推定の LP 定式化

$$\hat{x} = \arg \min_{x \in conv(C)} (y^T x)$$

残念ながら不等式数が符号長 n に対して指数的 $\rightarrow$ 実行困難

LP 復号法

$$\hat{x} = \arg \min_{x \in \mathcal{P}} (y^T x)$$

Feldman et al. 2002

（注）基本多面体は符号の凸包を包含する凸多面体であり、その整数点は符号語と一致する

LP復号法 に関する研究の流れ

- 高速化
  - 適応型 LP 復号: Taghavi et al. (2008, IT)
  - 定計算量 LPD: Vontobel and Koetter (2006, Turbo symposium)
  - 線形時間 LPD: Burshtein (2008, ISIT)
- 内点法
  - 内点法に基づく LPD: Vontobel (2008, ITA)
  - 内点法に基づく LPD(ベクトル線形通信路): Wadayama (2008, ISIT)
  - 主双対内点法に基づく LPD: Taghavi (2008, Allerton)
  - 主パス追跡 LPD : Wadayama (2009, ISIT)
- ADMM 法
  - ADMM based LPD (Xiu and Draper, 2012)

射影勾配法に基づくLDPC復号法

Deep Learning-Aided Trainable Projected Gradient Decoding for LDPC Codes, Wadayama and Takabe, 2019, arXiv:1901.04630

- **目的関数**：受信語を含む線形項と基本多面体制約に対応するペナルティ関数の和（非凸関数）
- **射影勾配法**：勾配ステップと射影ステップを交互に実行
- **データ駆動調整**：深層学習技術に基づいて、ステップ係数・ペナルティ係数などを調整

目的関数とペナルティ関数

Deep Learning-Aided Trainable Projected Gradient Decoding for LDPC Codes, Wadayama and Takabe, 2019, arXiv:1901.04630

受信語
LLR

ペナルティ係数

$$\text{minimize}_{\boldsymbol{x}\in\{0,1\}^n} \boldsymbol{\lambda}\boldsymbol{x}^T + \beta P(\boldsymbol{x}),$$

$$P(\boldsymbol{x}) := \frac{1}{2}\sum_{i\in[m]}\sum_{S\in O_i}\left[\nu\left(1+\sum_{t\in S}(x_t-1)-\sum_{t\in A_i\setminus S}x_t\right)\right]^2$$

ReLU関数

多面体制約

射影勾配法

制約付き最適化において，しばしば用いられる最小化技法

勾配ステップ

学習可能パラメータ

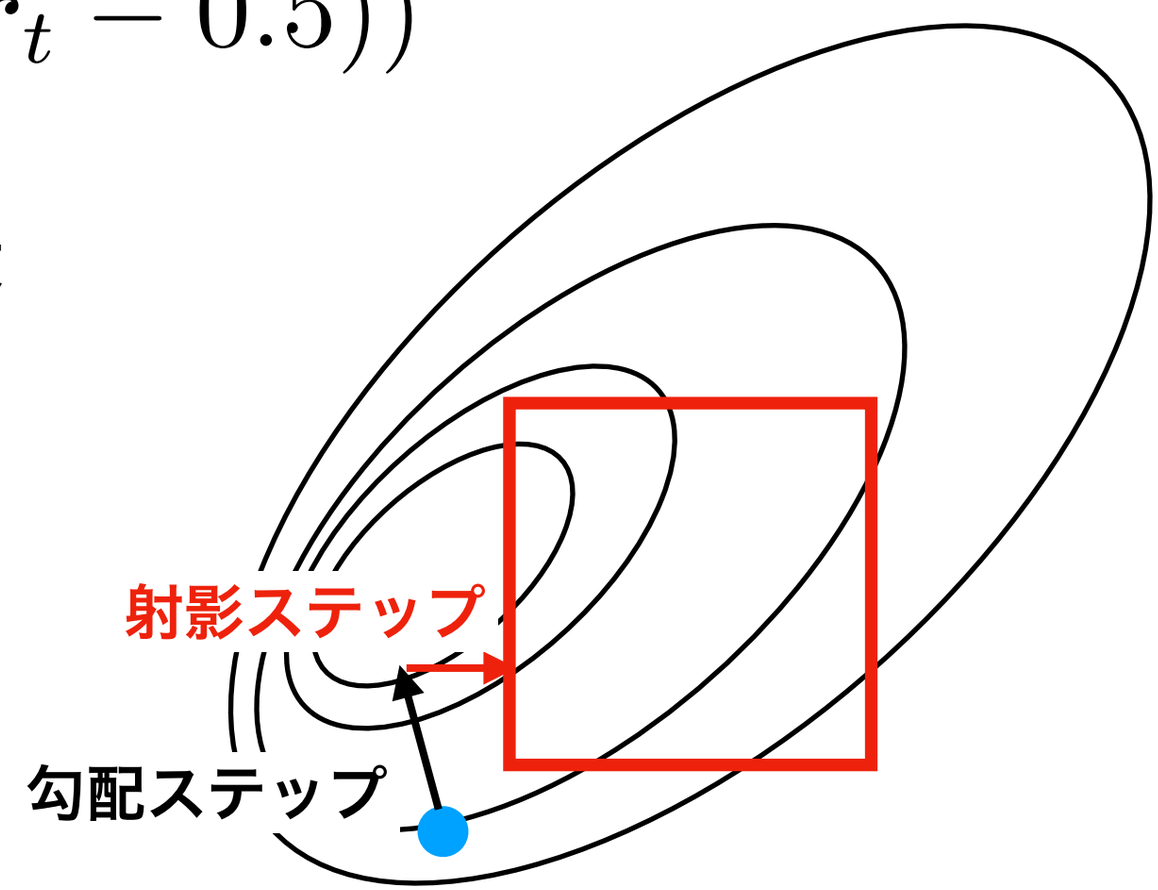
$$\boldsymbol{r}_t := \boldsymbol{s}_t - \gamma_t \left(\underline{\boldsymbol{y} + \beta_t \nu(1 + (\boldsymbol{s}_t - 1)Q - \boldsymbol{s}_t R)D^T}\right)$$

勾配ベクトル

射影ステップ

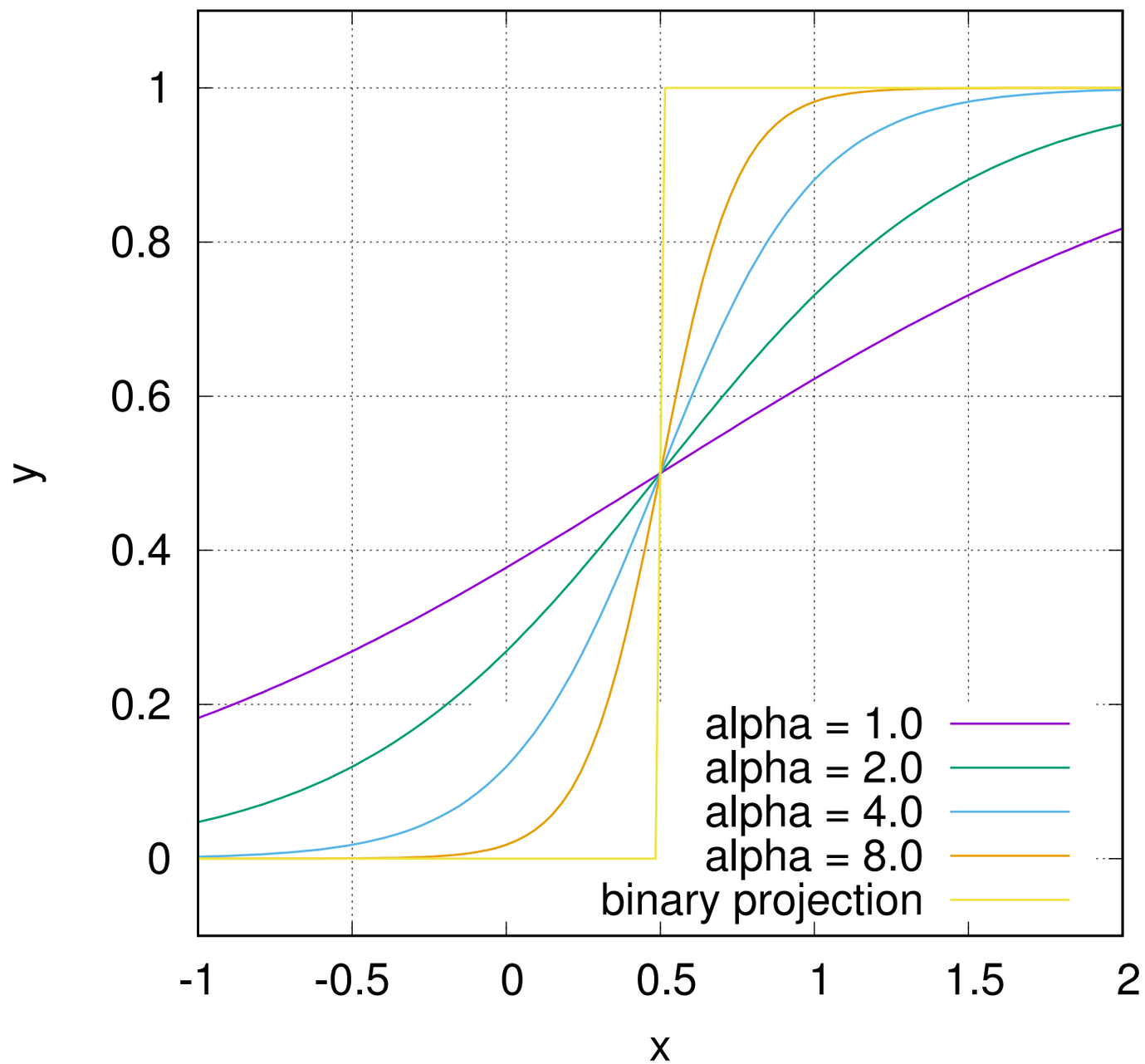
$$\boldsymbol{s}_{t+1} := \xi\left(\alpha(\boldsymbol{r}_t - 0.5)\right)$$

シグモイド関数



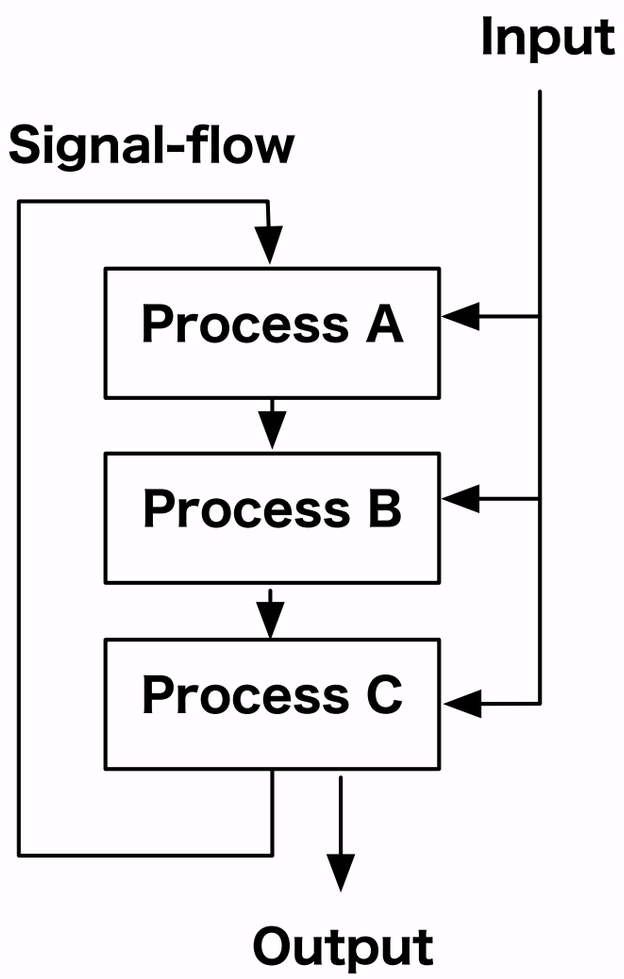
射影関数

$$\boldsymbol{s}_{t+1} := \xi\left(\alpha\left(\boldsymbol{r}_t - 0.5\right)\right)$$

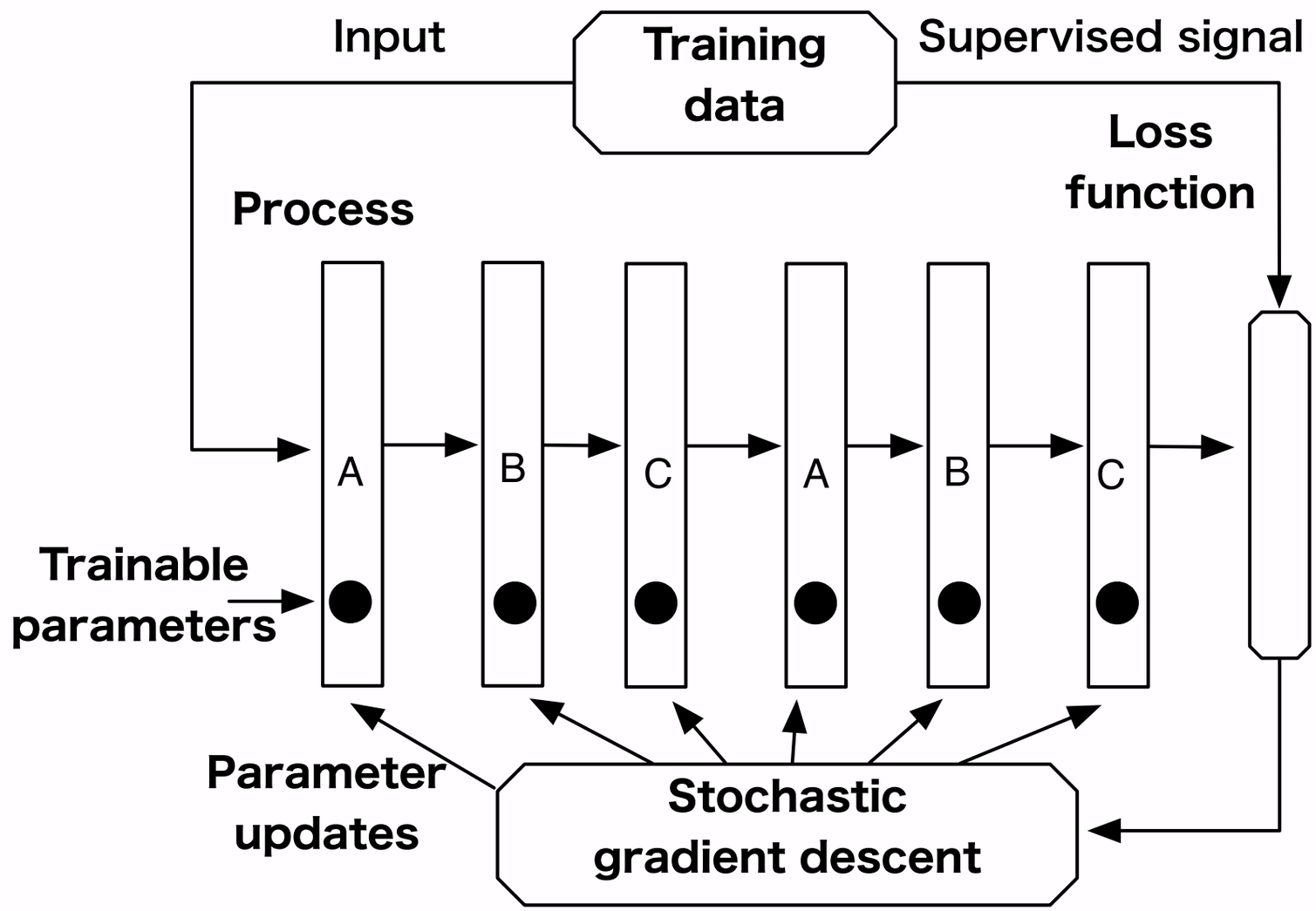


データ駆動調整

バックプロパゲーション＋確率的勾配法を利用して
学習可能パラメータを適切に調整する

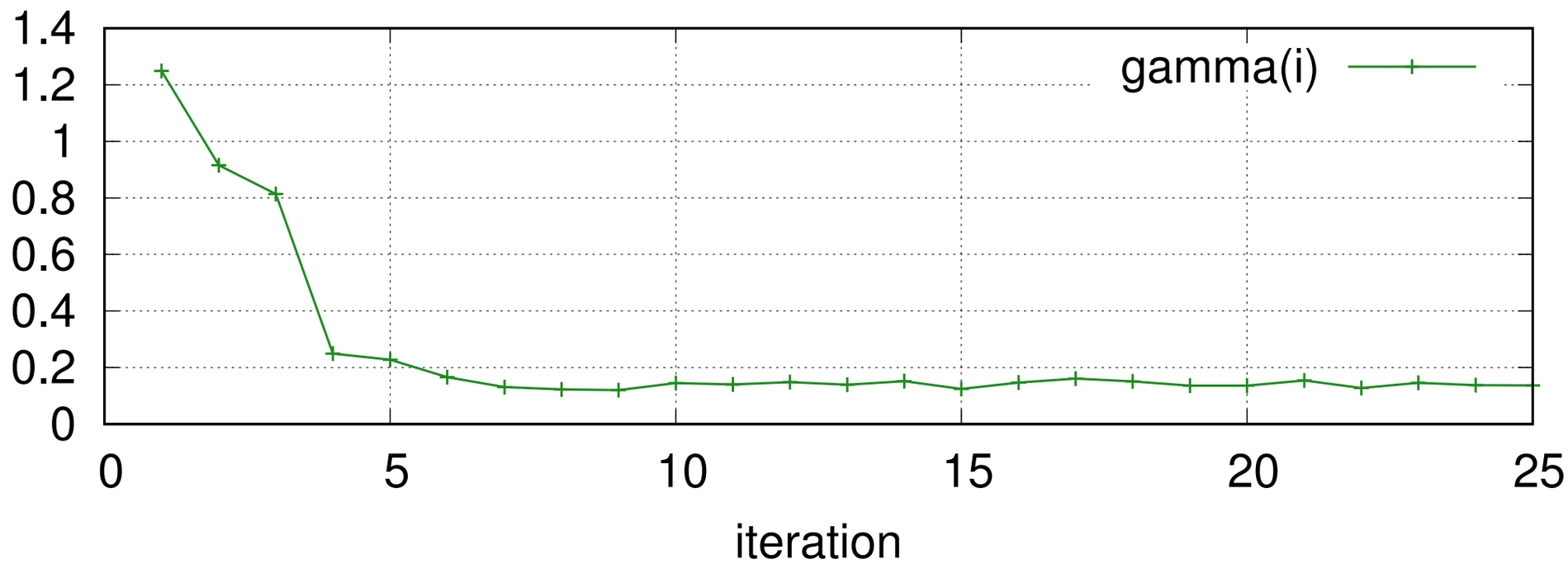


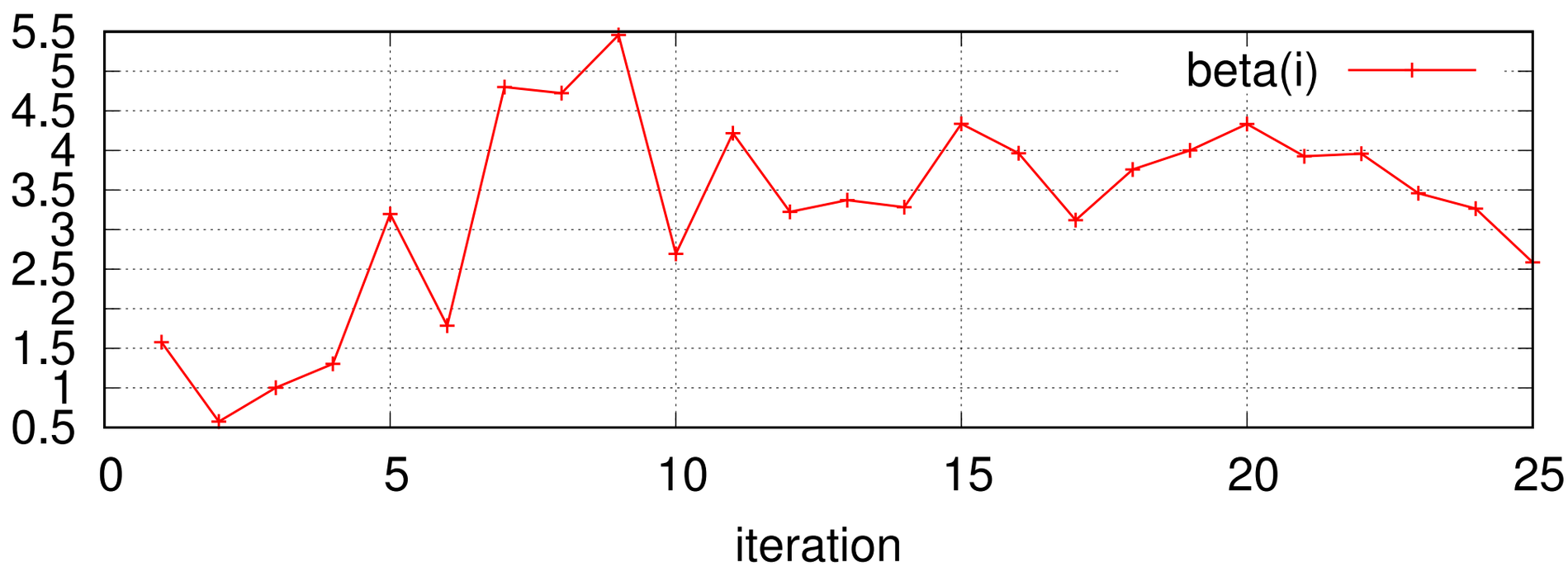
(a) Signal-flow diagram of an iterative algorithm



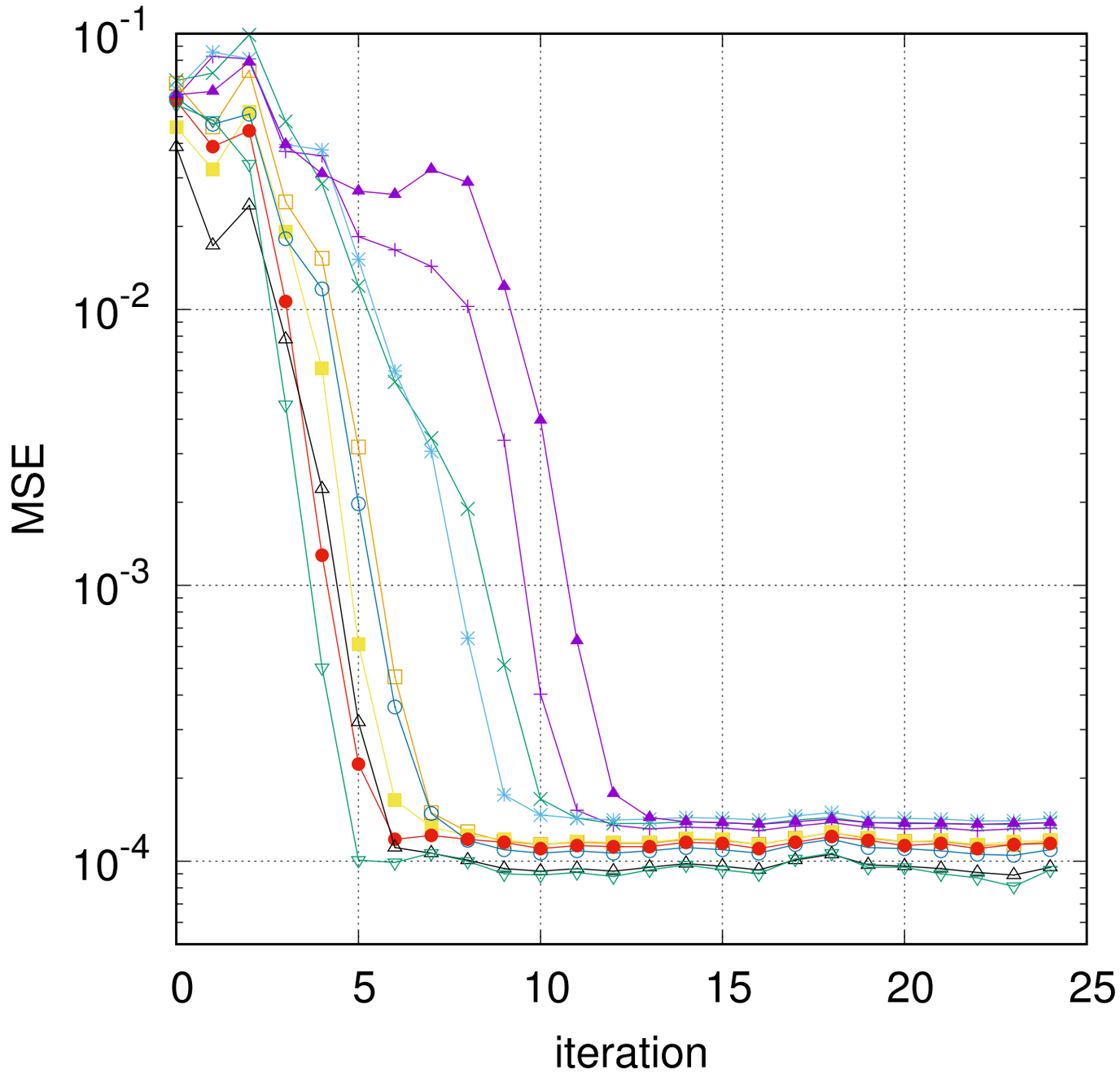
(b)Data-driven tuning

学習の結果


gamma(i)
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
5
10
15
20
25
iteration

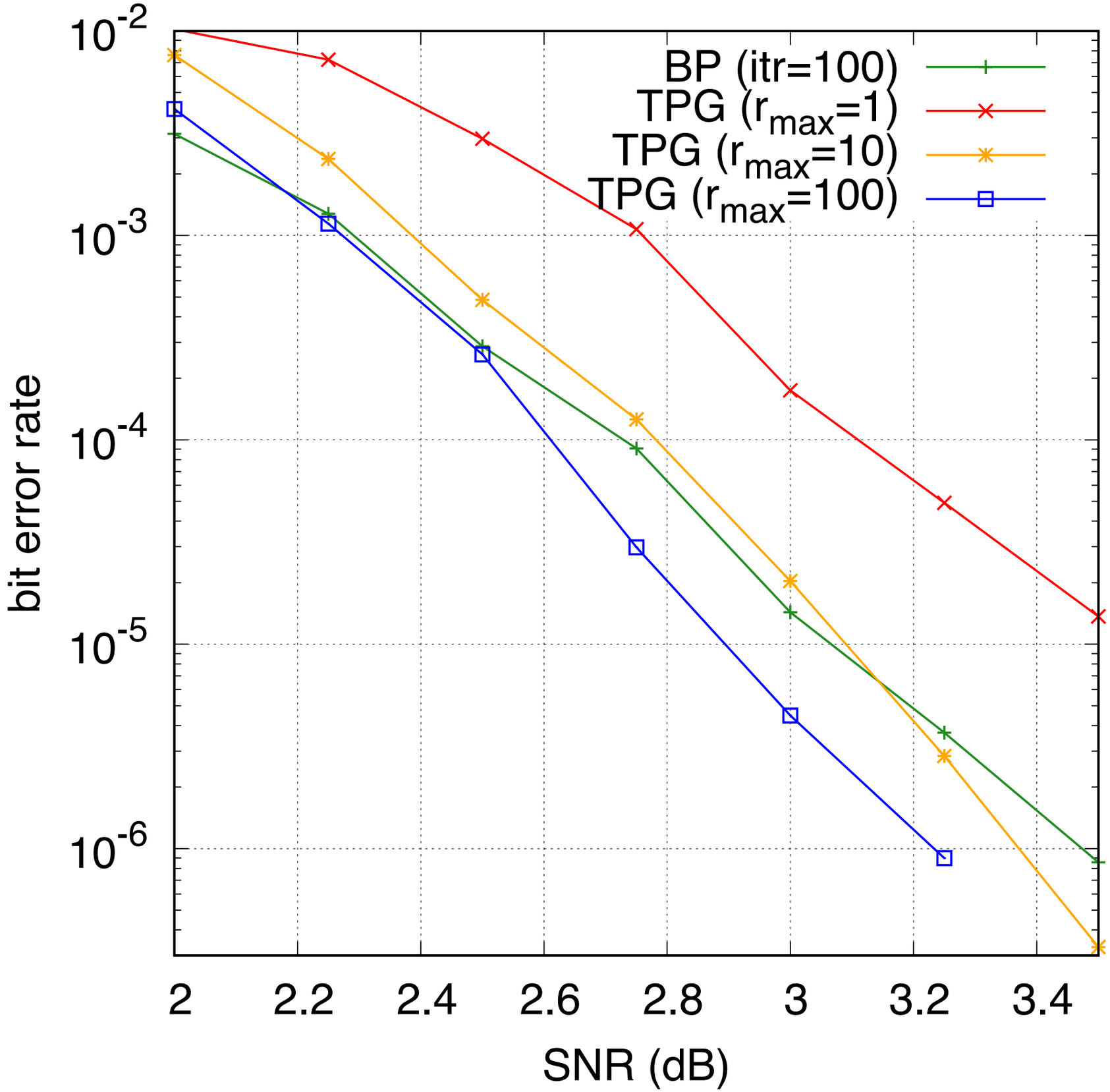

beta(i)
5.5
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
5
10
15
20
25
iteration

復号過程をみてみる

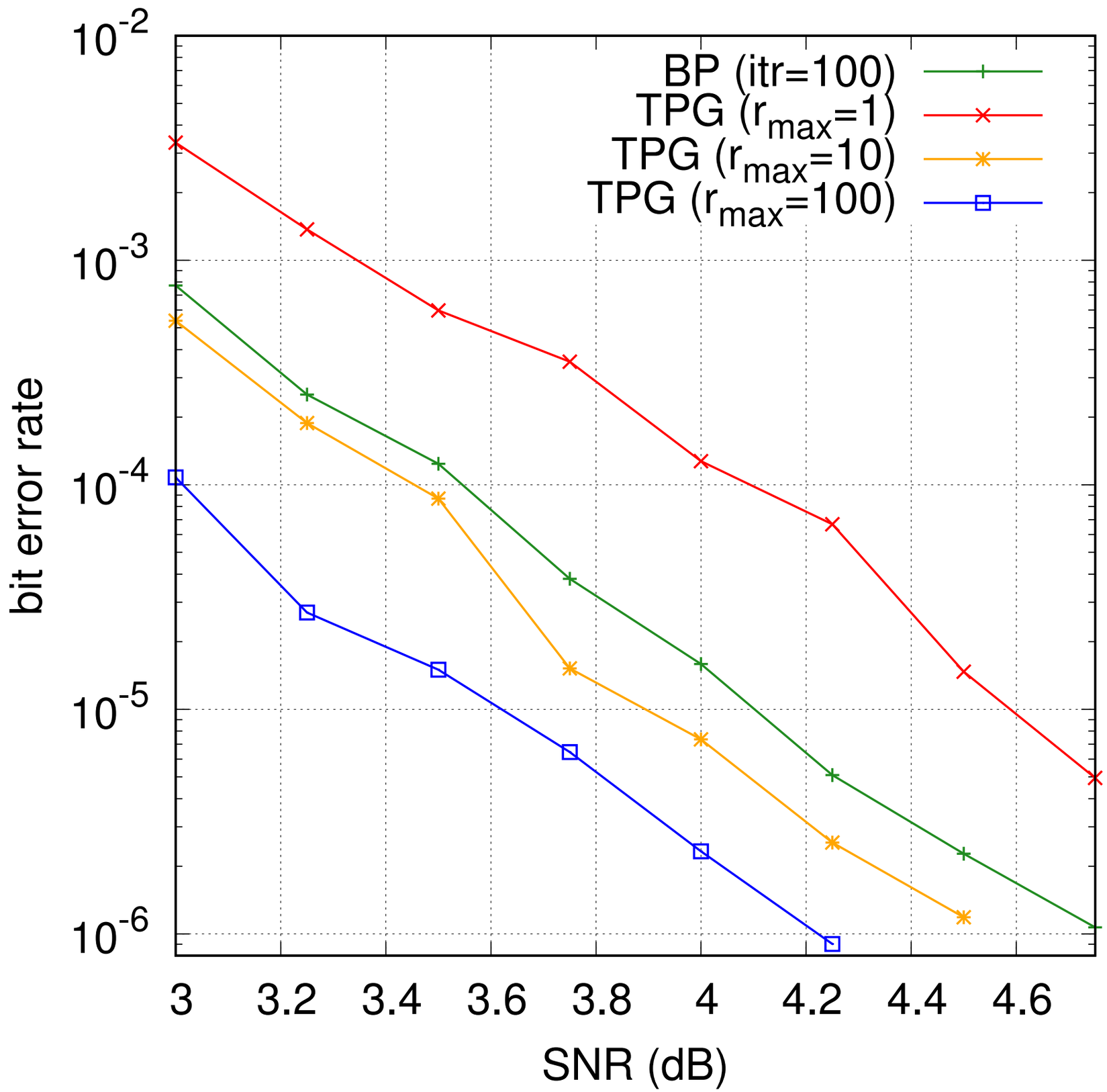


初期値を取り替えてリスタート（ランダムリスタート）の有効性

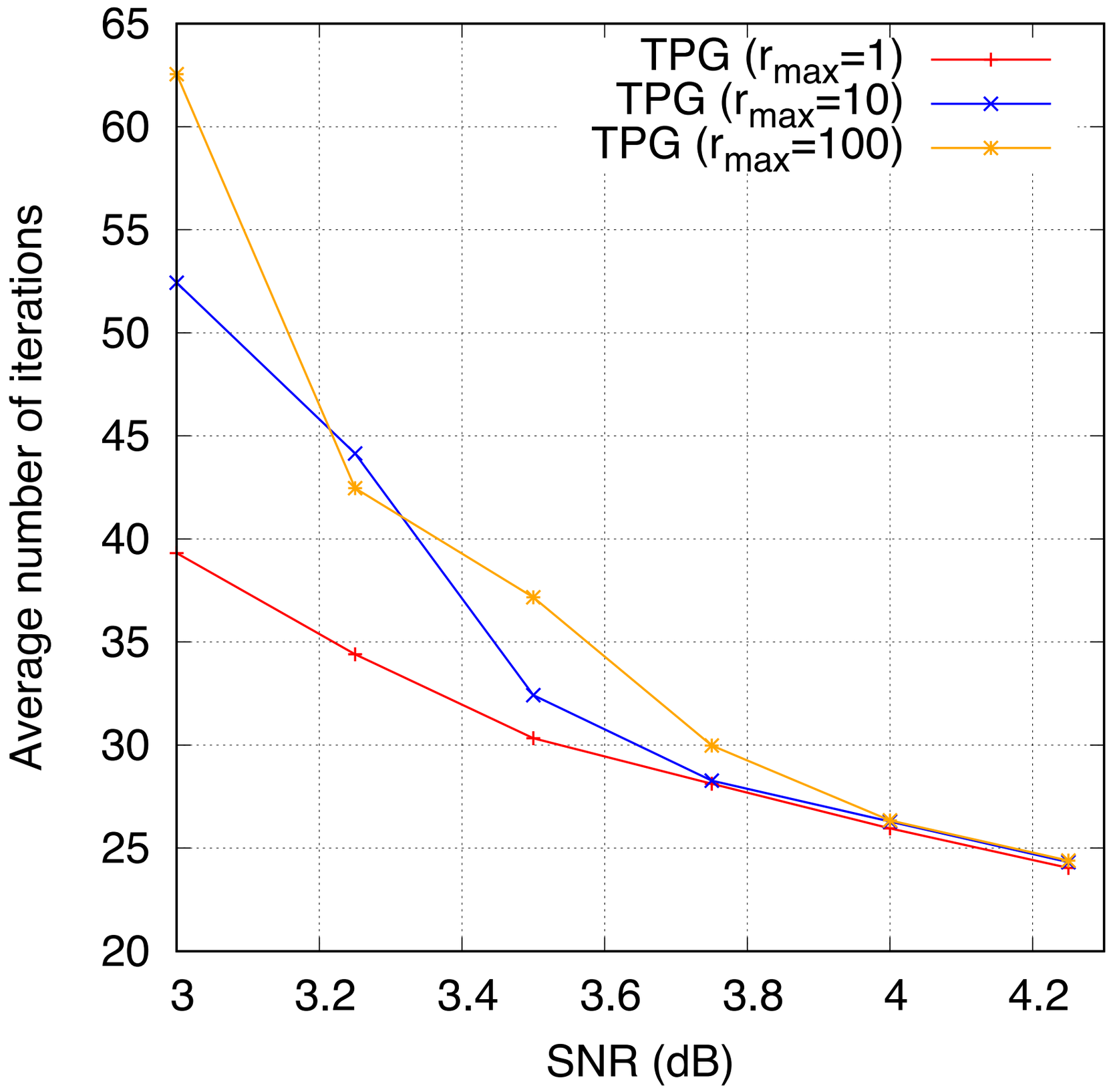
提案法のBER性能(n=504, m=252)



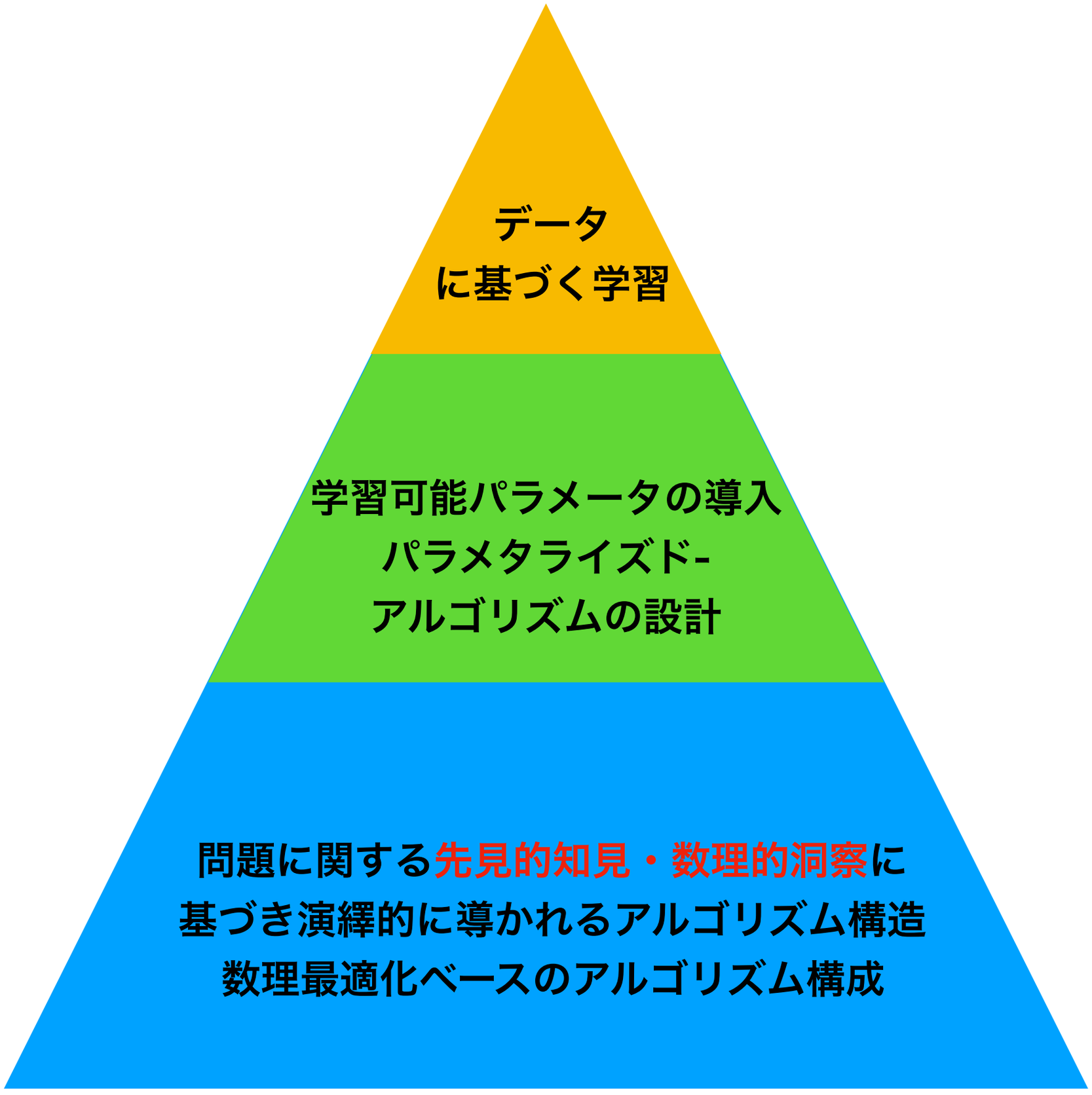
提案法のBER性能(n=204, m=102)



平均反復回数(n=204, m=102)



データ駆動アプローチに基づく反復アルゴリズムの設計



まとめ

- 最適化技法に基づくLDPC復号アルゴリズムを概観
- ビットフリップ型復号法においては、非線形目的関数の最小化を目指したアルゴリズム構成が有効
- 符号の凸緩和に基づく手法（組み合わせ最適化問題の連続緩和に影響を受けた手法）が有効
- 最適化に基づくアルゴリズム設計は、データ駆動調整（深層学習技術）に高い親和性を持つ