

# 非凸最適化に基づくアルゴリズムデザイン と深層展開

和田山 正

wadayama@nitech.ac.jp

名古屋工業大学

This work was partly supported by JSPS Grant-in-Aid for Scientific Research

(A) Grant Number 17H01280 and (B) 19H02138

# 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

研究上の興味:

誤り訂正符号(LDPC符号), 情報理論, 無線通信,  
深層学習, 信号処理

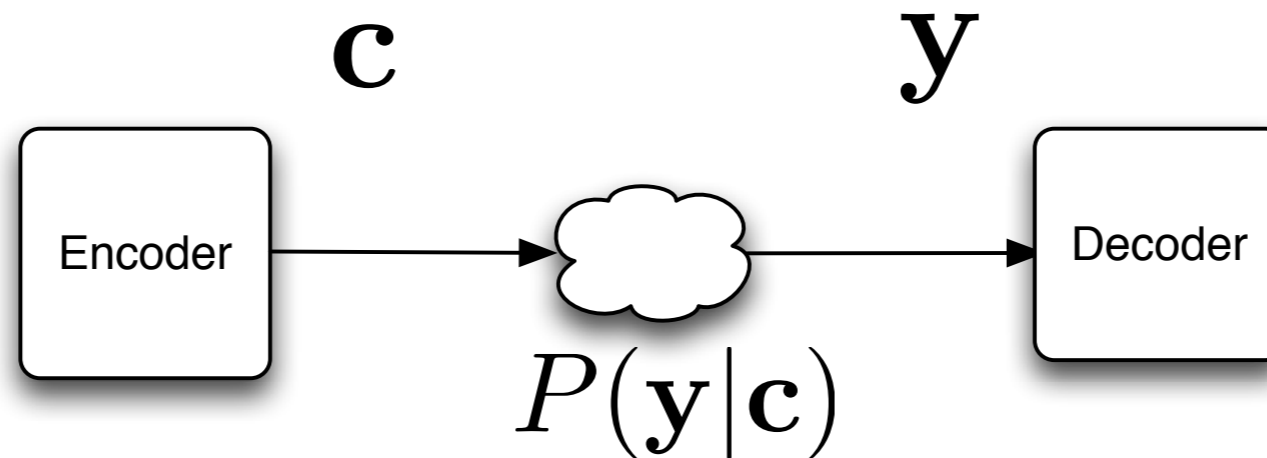
誤り訂正符号の本を書いています→



# はじめに

- 無線物理層・記憶デバイス分野における高度な信号処理の必要性の高まり
- 復号・信号検出問題への非凸最適化アプローチ：誤り訂正符号の復号問題、信号検出問題を非凸最適化問題として定式化し、勾配法、近接勾配法などの最適化技法を利用する
- 深層展開：反復型最適化アルゴリズムの信号流グラフを時間方向に展開したネットワークに対して、バックプロパゲーション・SGDなどの深層学習技術を適用する
- 次のトピックスをご紹介します：
  - GDBF復号法：LDPC符号の復号のためのビットフリップ型復号法（回路規模を小さくしやすい）・非凸目的関数+勾配法（的ビットフリップルール）
  - 射影勾配法を利用したLDPC復号法(ISIT2019): 深層展開を利用した復号法
  - 深層展開の考え方のご紹介
  - 圧縮センシング・MassiveMIMO検出問題への深層展開の適用

# 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

- ✓ **確率からのアプローチ**：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)
- ✓ **最適化からのアプローチ**：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 適切な構造 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

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

- ▶  $\text{conv}(C)$  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

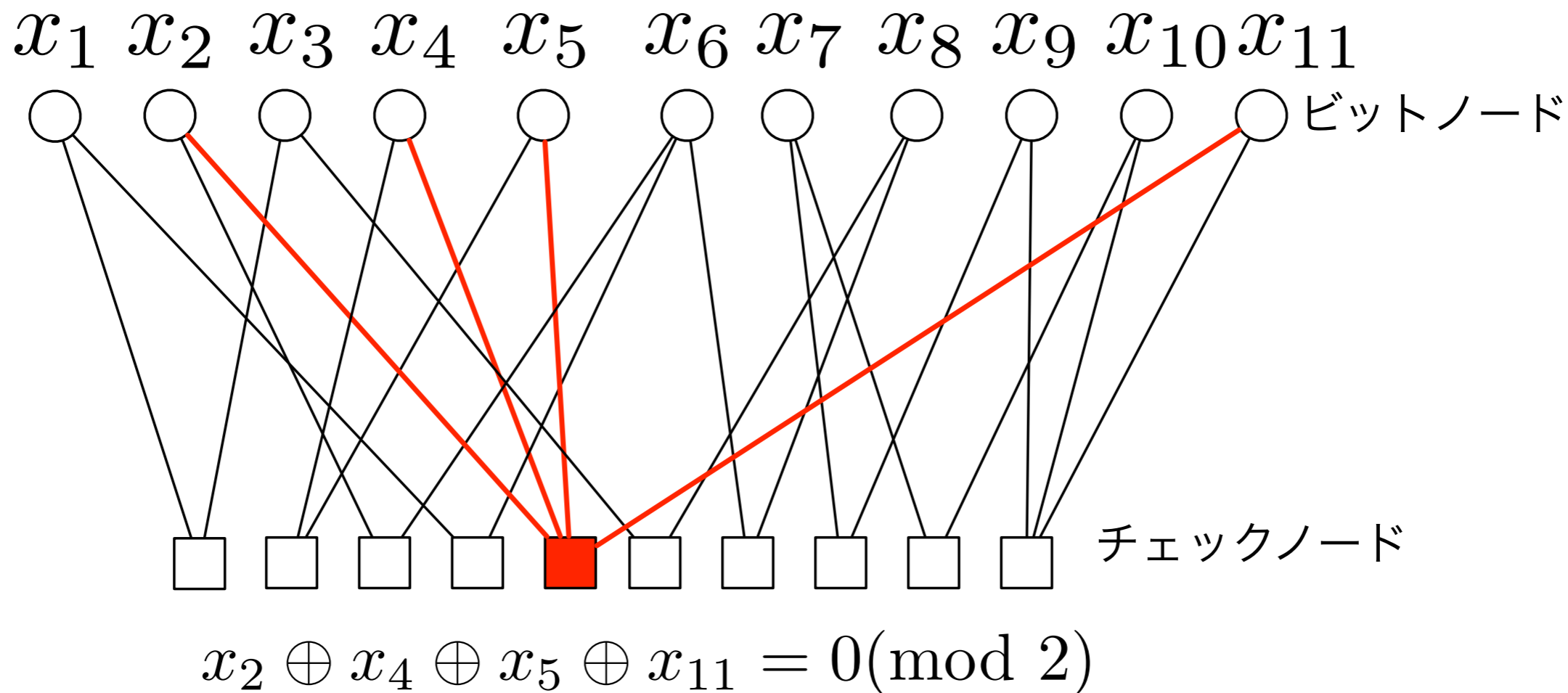
## 非線形最適化

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

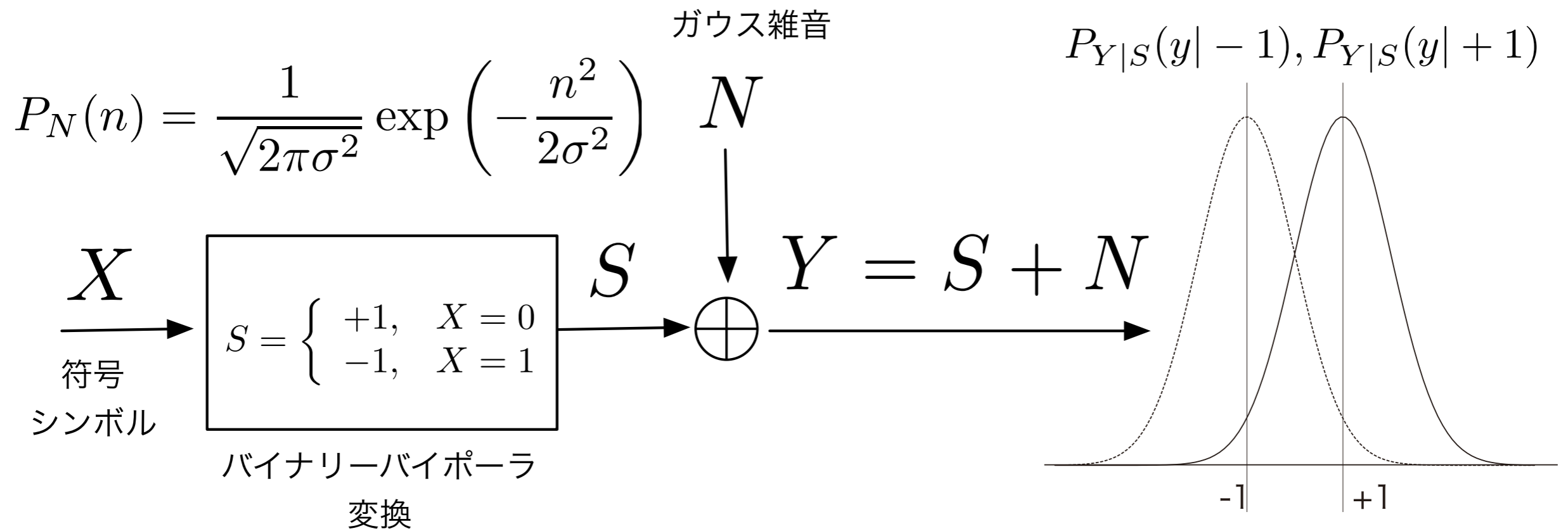
- ▶  $p$  は符号語制約を与える非線形関数

# 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、 $(x_1, x_2, \dots, x_n) \in \{0, 1\}^n$  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  $C$



# AWGN通信路モデル



# 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

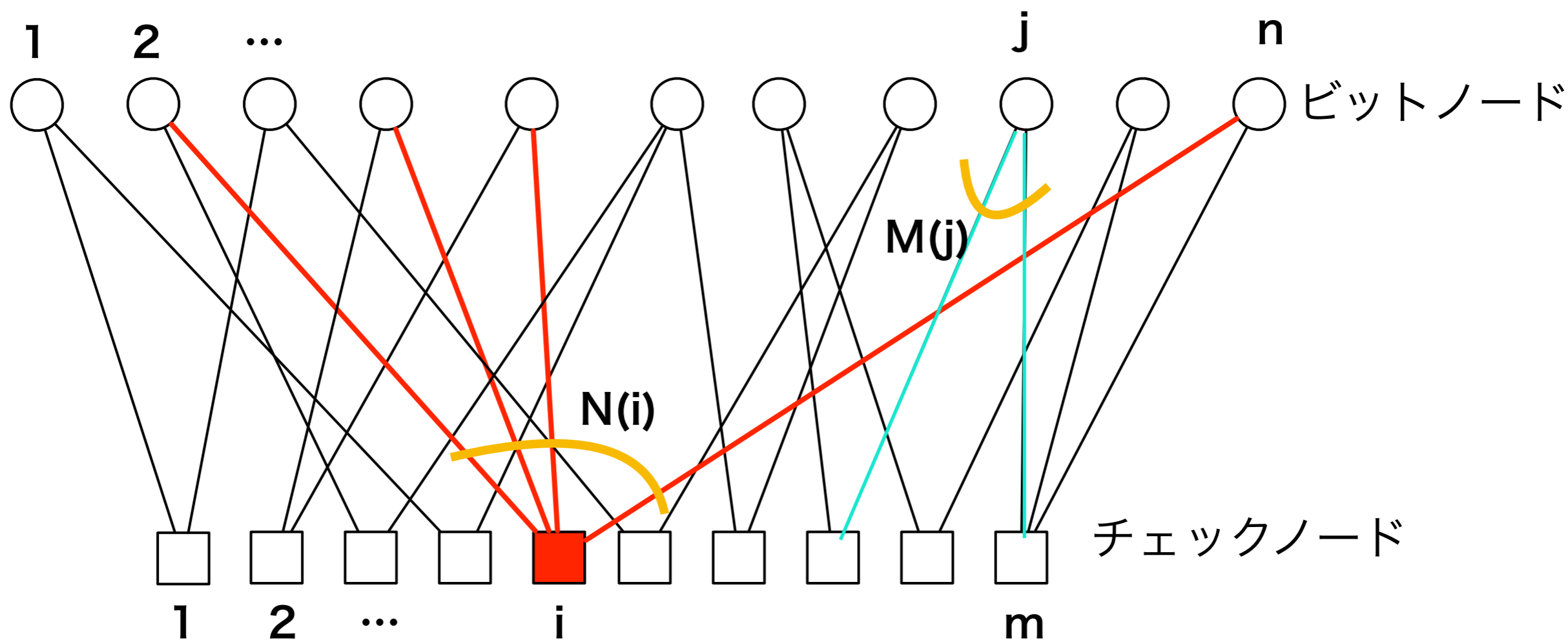
$$\hat{x} = \arg \min_{x \in C} \|y - b(x)\|_2^2$$

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）。

# 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

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

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  $x_\ell := -x_\ell$

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

ステップ2に戻る

反転関数(ビット信頼度)



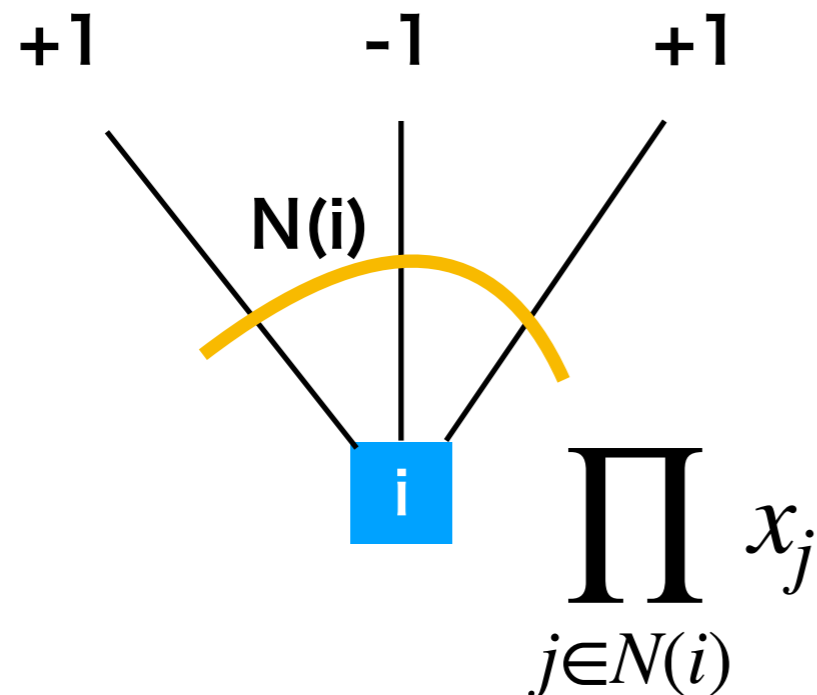
# 様々なビットフリック型復号法

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

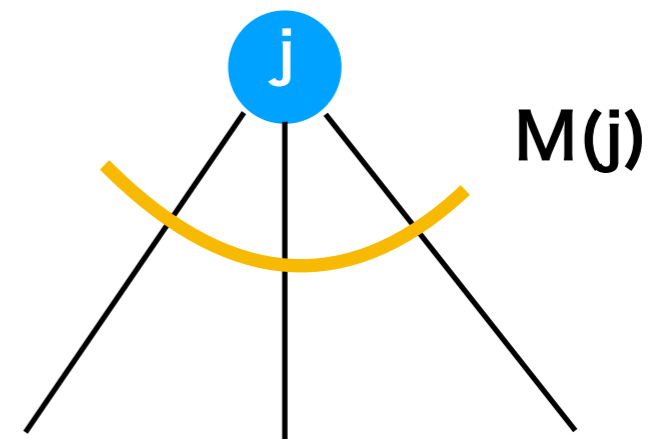
WBF  $\Delta_k(\mathbf{x}) = \sum_{i \in M(k)} \beta_i \prod_{j \in N(i)} x_j, \quad \beta_i = \min_{j \in N(i)} |y_i|$

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δροーム値



ビット信頼度



# GDBF復号法の導出

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

(Google Scholar にて、引用 1 2 3 件！)

$\tilde{C} = \{b(x) \mid x \in C\}$ : 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

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

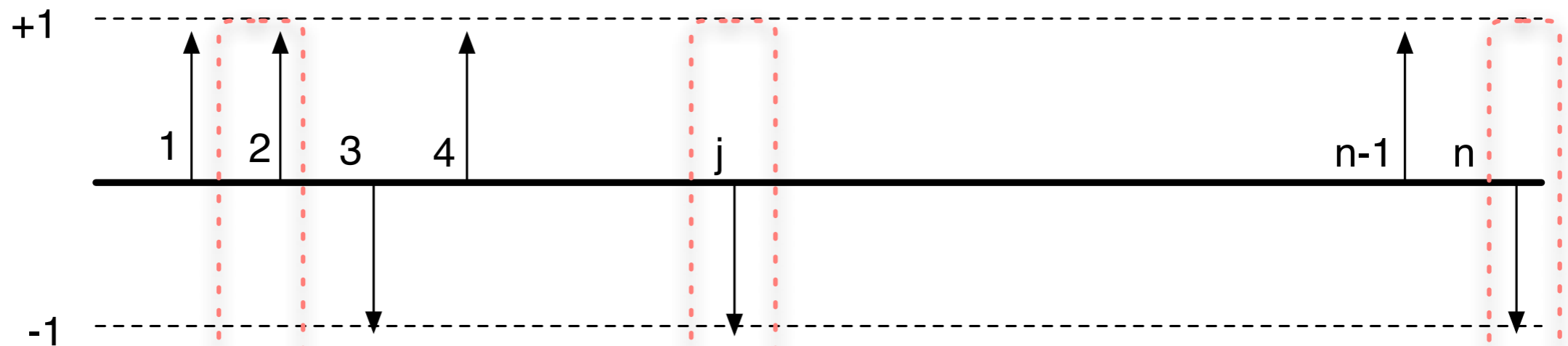
と与えられる。

## 反転関数

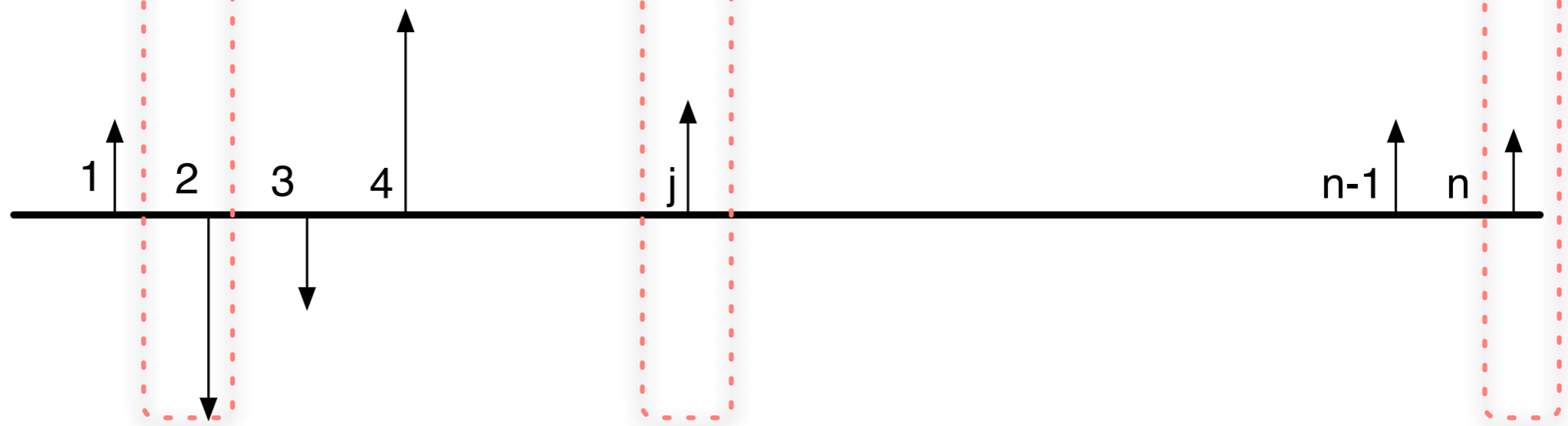
$$\begin{aligned} \Delta_k^{(GD)}(\mathbf{x}) &\triangleq x_k \frac{\partial}{\partial x_k} f(\mathbf{x}) \\ &= x_k y_k + \sum_{i \in M(k)} \prod_{j \in N(i)} x_j. \end{aligned}$$

# 探索ベクトルの要素の正負と勾配ベクトル

Search point  $\mathbf{X}$



Gradient vector



# GDBFアルゴリズム

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

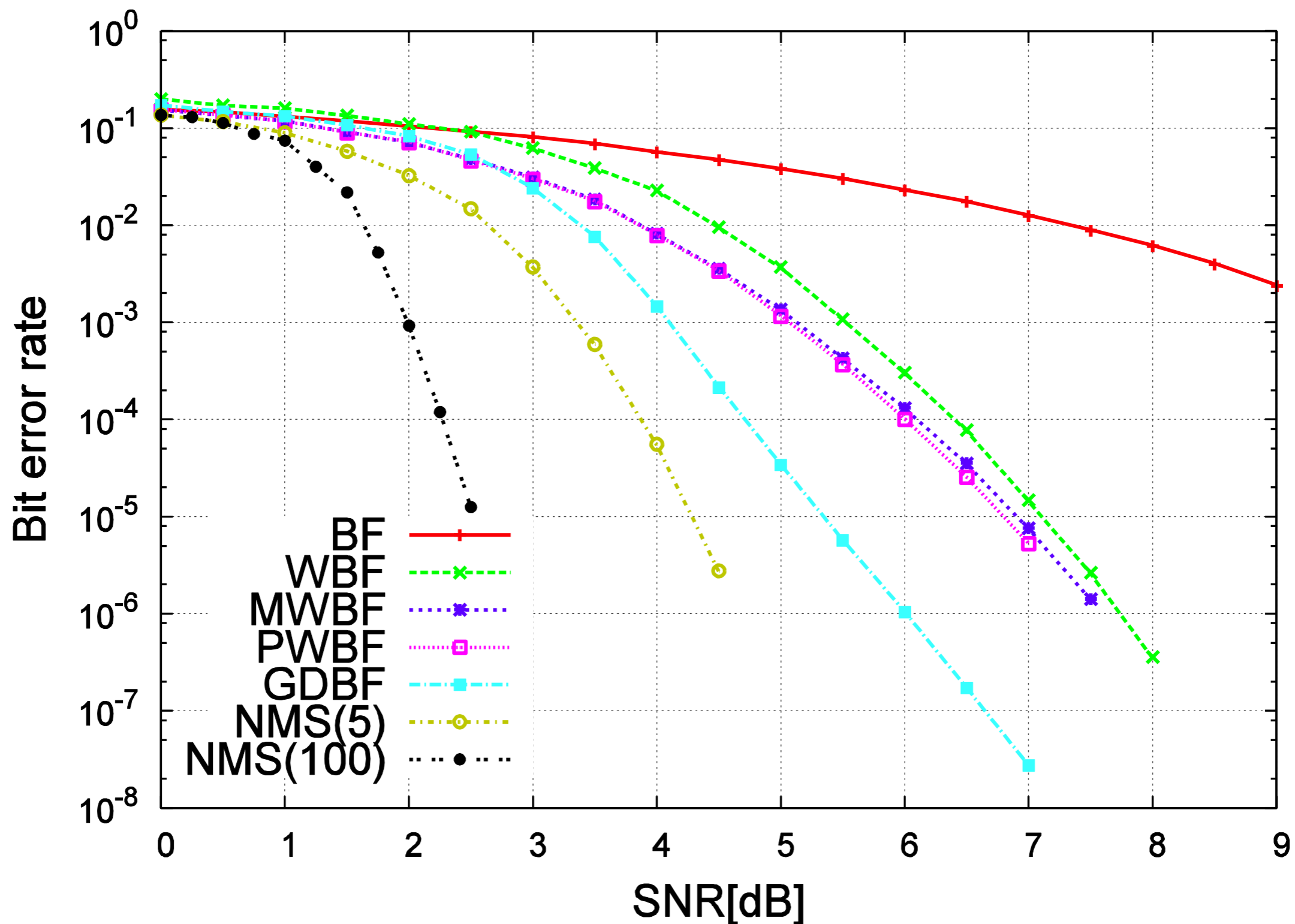
上記のプロセスは、ビットフリップ型勾配法と見なすことができる (coordinate decent/accnt algorithm).

## 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<sup>4</sup> :

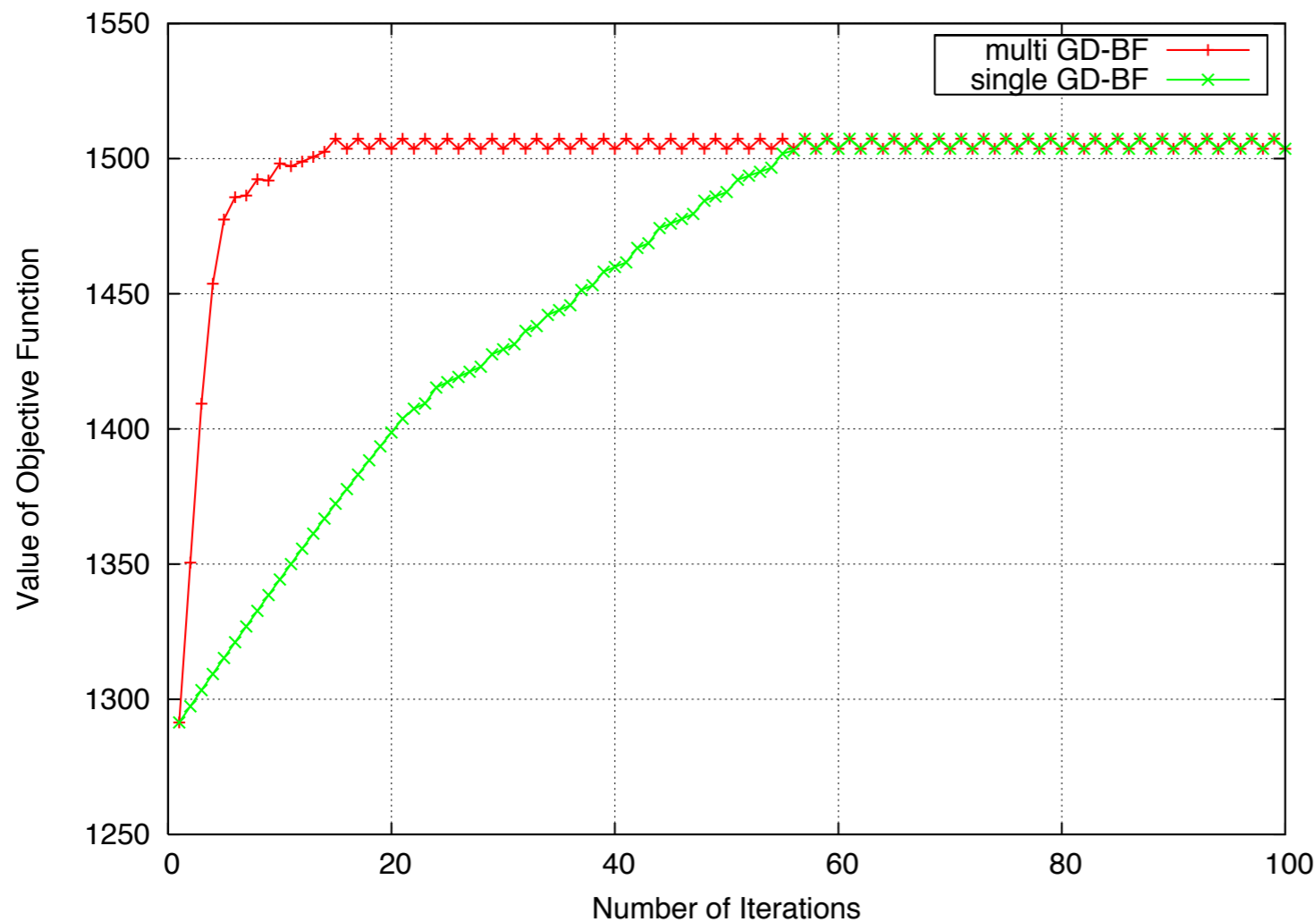
$$\Delta_k^{(GD)}(\mathbf{x}) \triangleq x_k y_k + \sum_{i \in M(k)} \prod_{j \in N(i)} x_j$$

# ビット誤り率の比較



# 最適化プロセス中の目的関数値

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



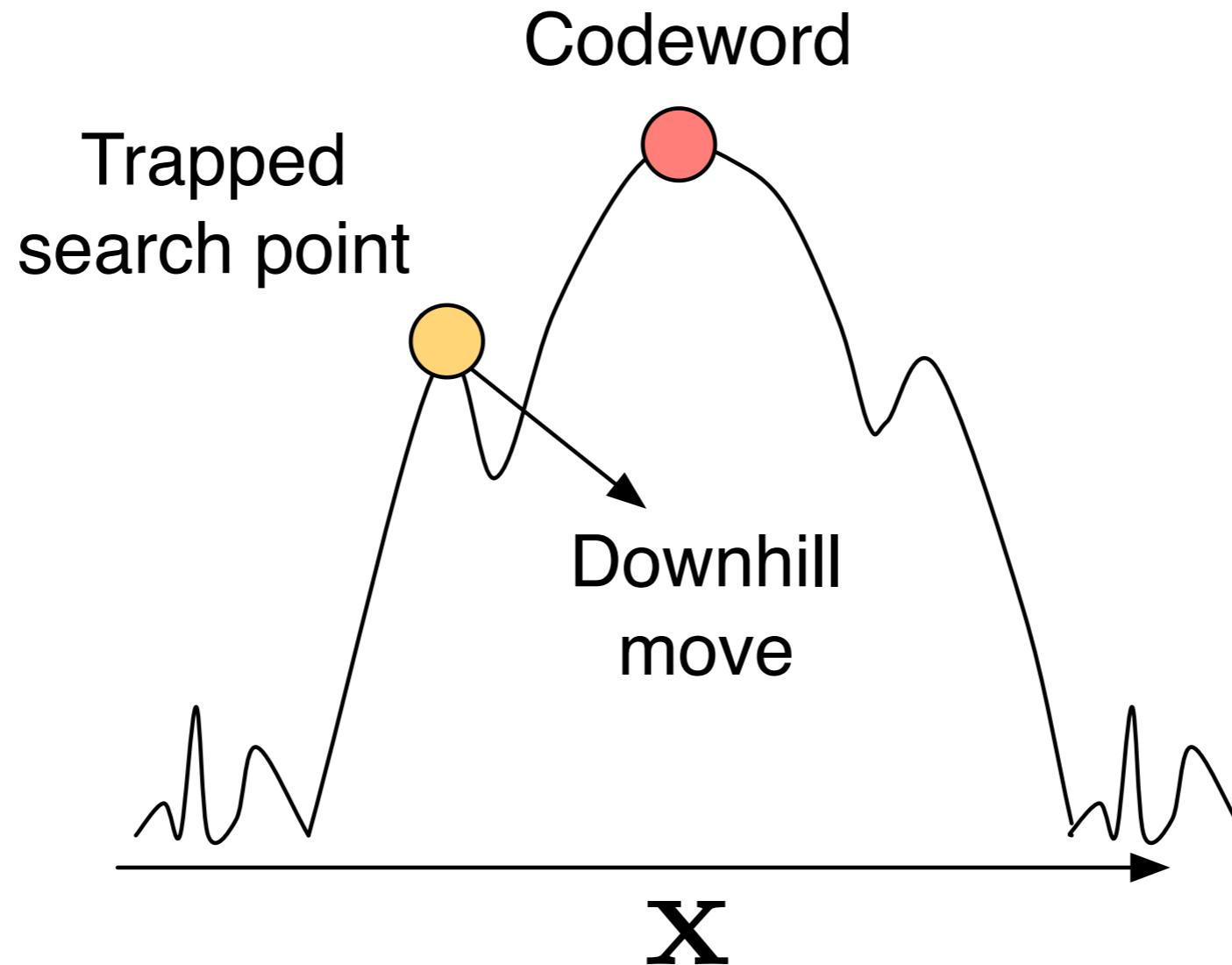
SNR=4 dB

ビットフリップ型復号アルゴリズムに最適化の視点が初めて持ち込まれた！



# 停留点からの脱出

目的関数は非凸関数→多数の停留点（鞍点・局所最大点）



Wadayama et. al (2010)では、停留点脱出のためにマルチビットフリップを利用

# Noisy GDBF アルゴリズム

## Noisy GDBF 反転関数 (Sundararajan et al. (2014))

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

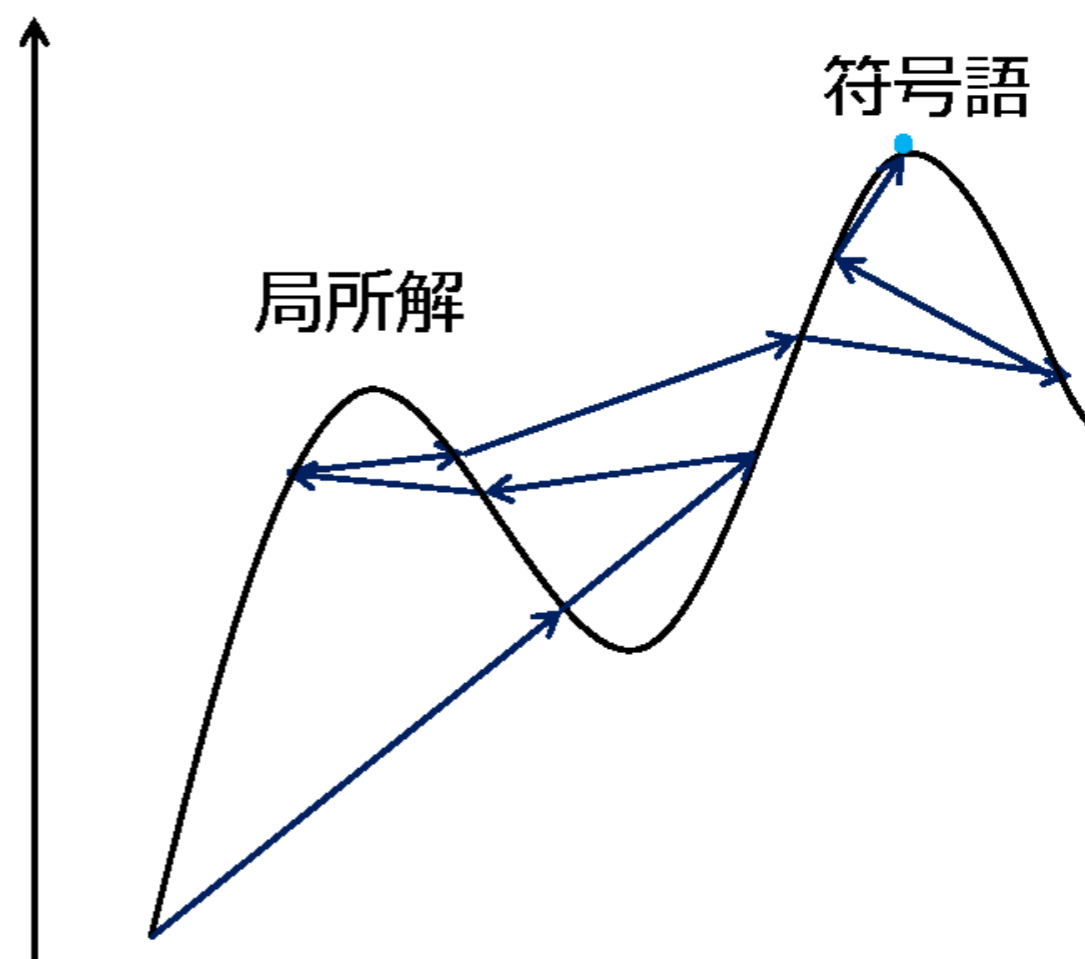
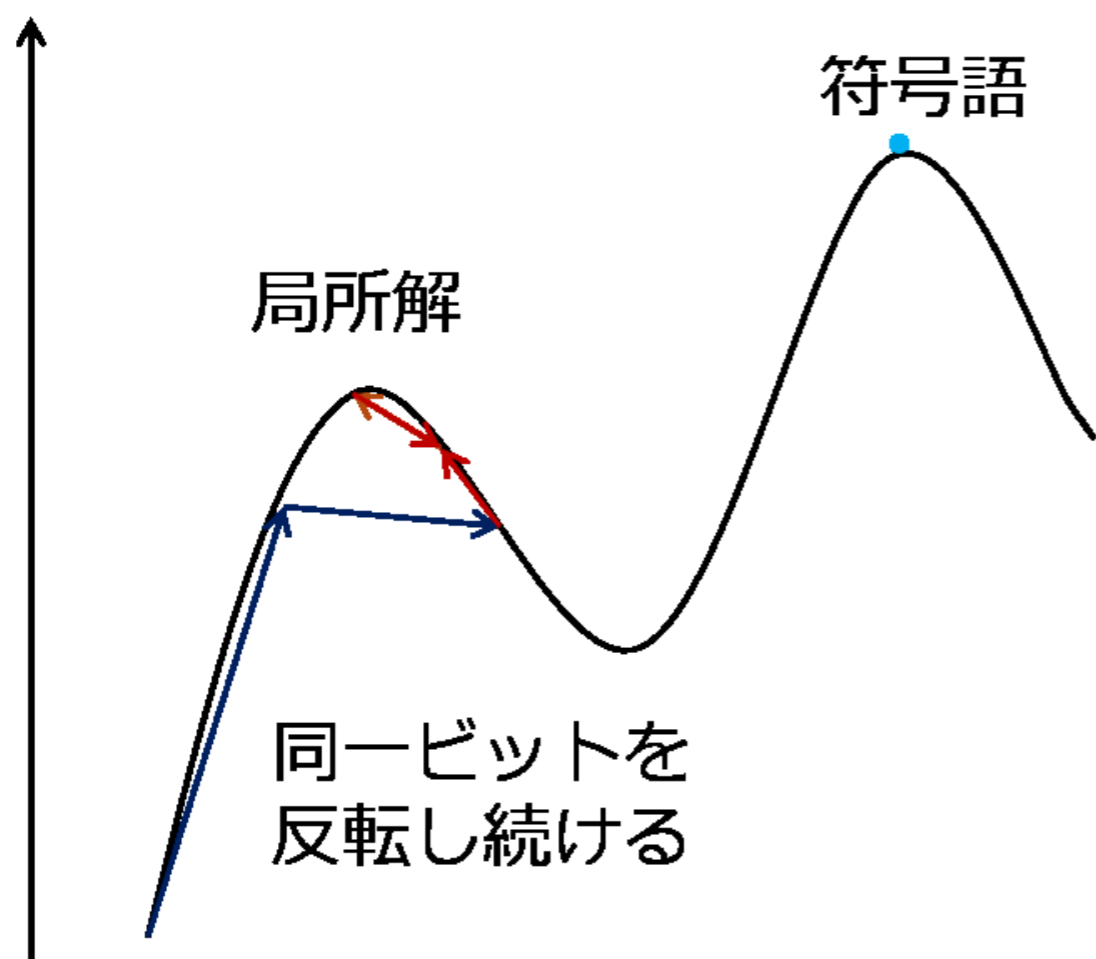
- ▶  $\alpha$ : 定数
- ▶  $q_i$ : ガウス乱数項

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。

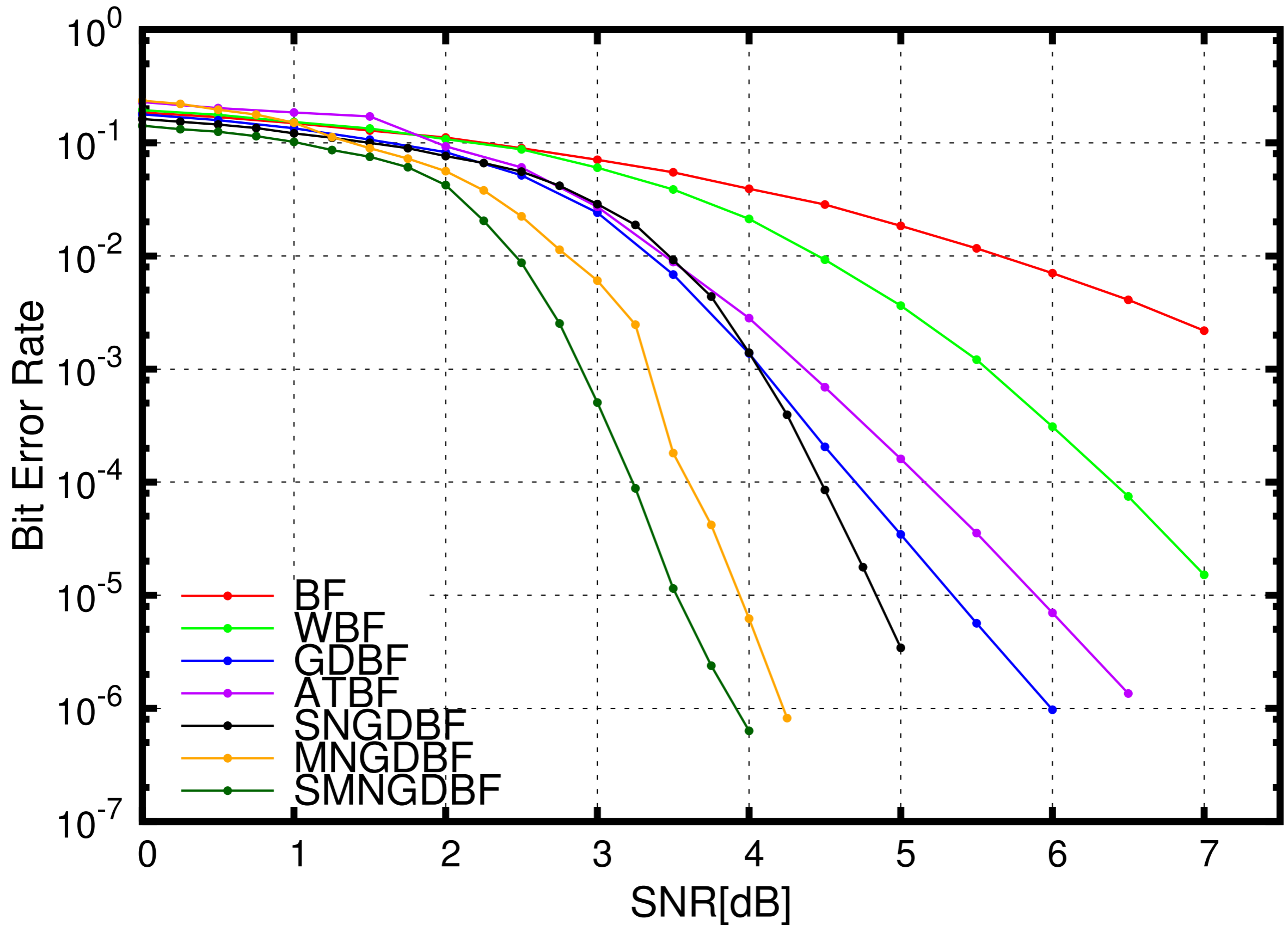
# 停留点からの脱出

G D B F

N G D B F



# Noisy GDBF のBER性能



# GDBF系アルゴリズムまとめ

- 実装時の回路規模・スループットなどの点で,最近広く興味を持たれている (光通信系・フラッシュメモリなど記録系)
- BP(min-sum)にかなり近いパフォーマンス
- 数多くの変種 (タブーリスト保持、適応的しきい値。。。)
- 停留点からの脱出が性能向上のためのキーポイント：勾配ベクトルへの雑音付加が有効
- いくつかのハイパーパラメータを含む→深層展開による調整が可能??

# ここで話がかわります

- 深層学習を無線物理層アルゴリズム、信号処理アルゴリズムに利用する動き
- 本グループでも2016年から取り組んでいます  
(Alpha Goのニュースがきっかけ)
- 以降はその関連のお話をします(特に無線分野の文脈で)。

# 無線物理層におけるDL+Comの動き

- ☑ ICC2018, Globecom2018, ICC2019 における機械学習関連のセッションとワークショップの盛り上がり
- ☑ 各論文誌の特集号(IEEE Access など)
- ☑ IEEE ComSoc

ComSoc Technology News (CTN)

# Nine Communications Technology Trends for 2019

## 1. Machine Learning. If You Are Not Doing It, People Will Still Shun You at Parties and Now Your Children Will Laugh at You Too.

Moving up to the number one spot for this year. Well it is actually happening now, especially in some network planning functions and radio resource management. **Everyone is now trying to use it all the way down to the physical layer.** Now that there is some real use, we expect this year will be when it turns and starts rolling back down the hype curve, such is human nature (in fact we expect AI to become disappointed in itself sometime this year and need some counselling). But in the meantime the activity in ML will continue to rise in comms, and the CTN EB will certainly milk it for a few articles. Implementation in the edge of the network might be a big topic for this year. Federated Learning, you heard it here first. Unless of course you already heard it somewhere else...



# 機械学習技術を利用した無線通信技術

- ✓ 無線通信技術 + 機械学習(ML)を組み合わせた研究が急増中→なぜ？

## 深層学習(ディープラーニング)の登場

深層学習は、画像認識・音声認識・自然言語処理などの分野ですでに基盤技術となっている。他分野への応用が広がり始めている

→深層学習技術の持つ普遍性

## より高度な信号処理への要求 (5G/6Gに向けて)

mmWave massive MIMO, ブラインド通信路推定, 誤り訂正, 端末位置推定, ビームフォーミング, NOMA(過負荷検出), 干渉制御

## 近未来におけるAI/MLと無線との融合

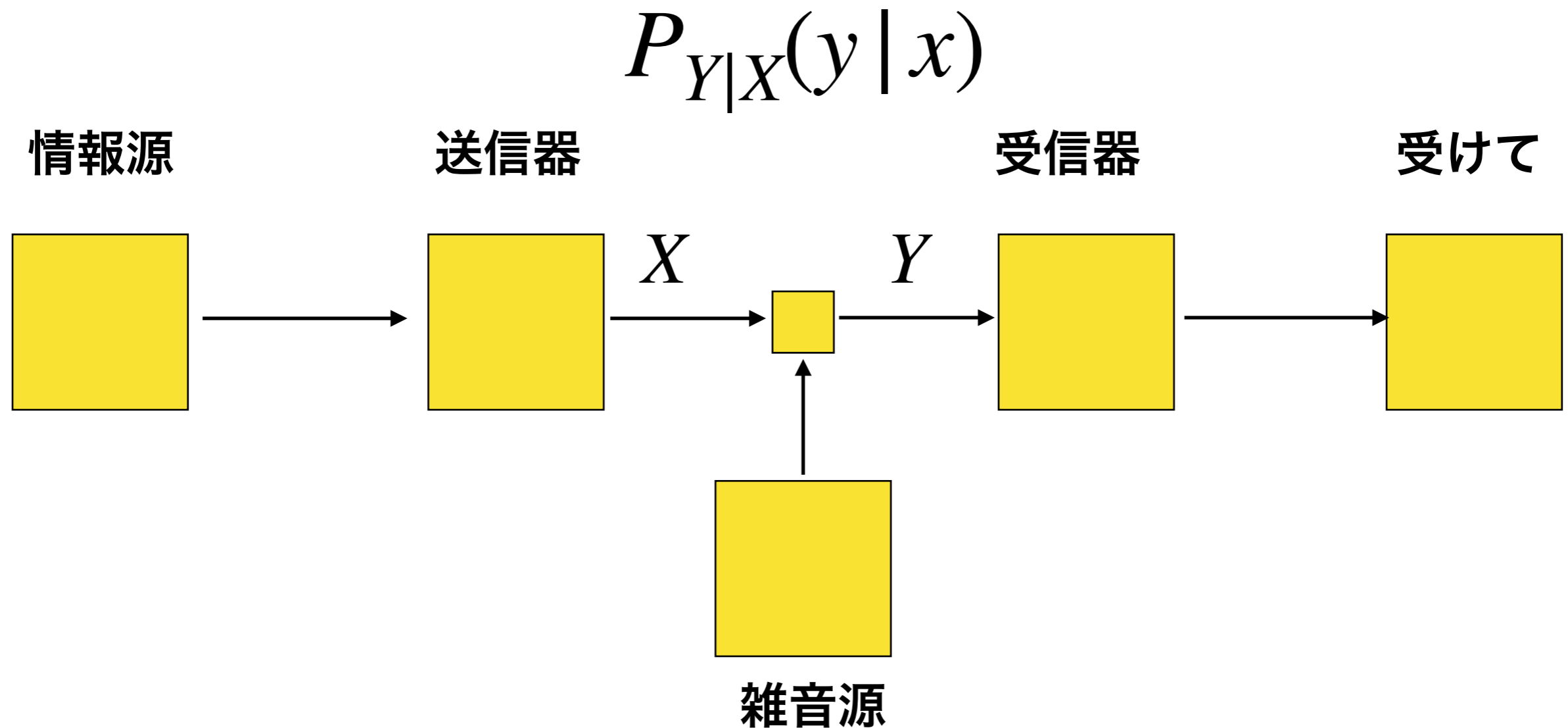
AI/MLが主たる無線通信通信のタスクになるのでは？

(分散学習のためのインフラ) また、高度無線ネットワークをAI/MLが支える

# 機械学習を利用した無線物理層技術の分類

タイプ	アイデア
ブラックボックスモデル	通信路全体・検出アルゴリズム・復号アルゴリズムを深層ニューラルネットワークを利用して表現。データ駆動型設計。事前知識・過去の知見必要なし。
深層展開	既知の反復型アルゴリズムに学習可能パラメータを埋め込み深層学習技術を利用してパラメータを学習。
最適化模擬	凸最適化・非凸最適化アルゴリズムのニューラルネットワークによる模擬。
分散学習	各端末における学習とその学習結果の統合
強化学習	報酬に基づく最適戦略の決定、マルコフ決定プロセス。

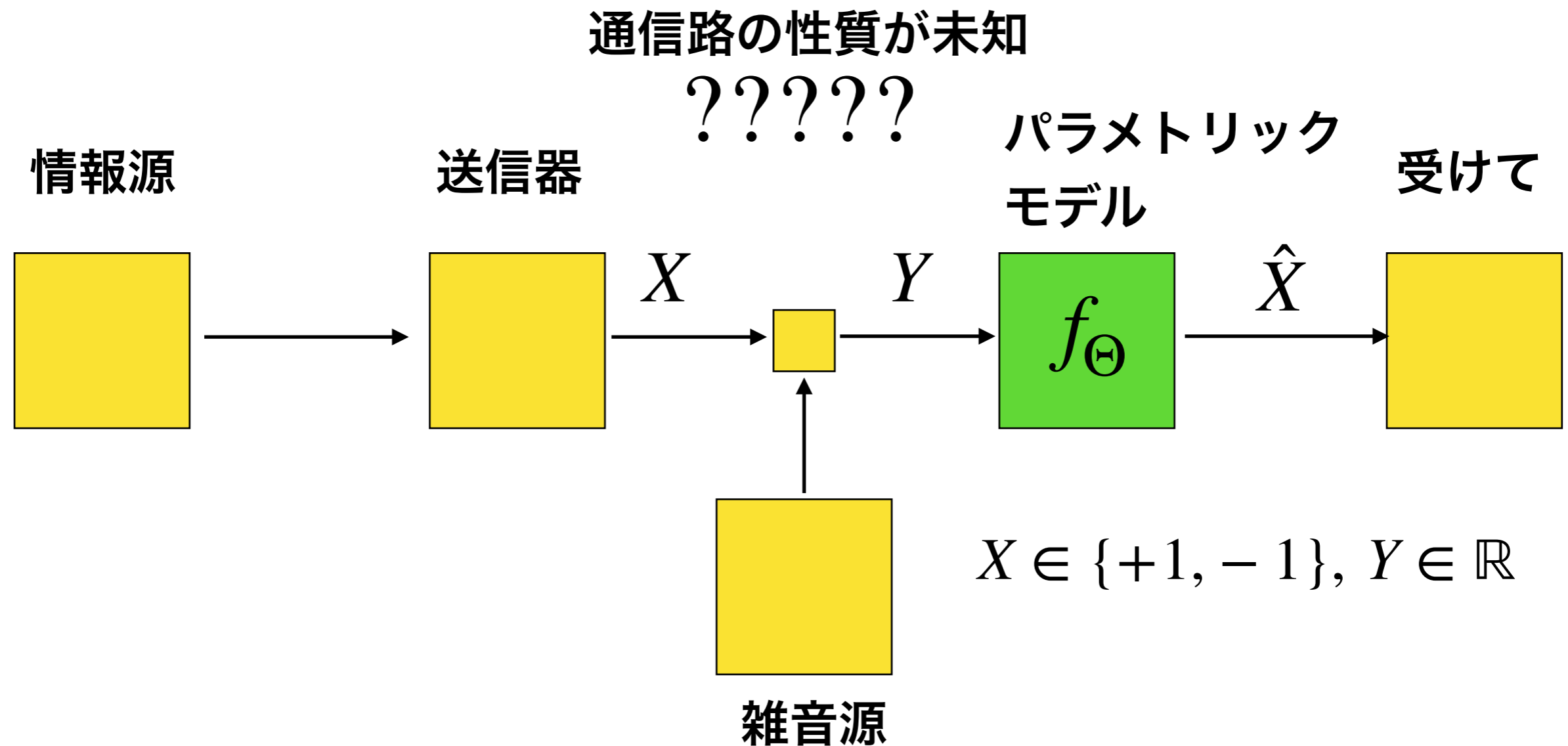
# シャノンの通信路モデル



C.E. Shannon: ``A mathematical theory of communications,’’ 1948

仮定：通信路の統計的性質が既知

# 通信路の統計的性質が未知な場合



データセット  $\{(x_1, y_1), (x_2, y_2), \dots, (x_K, y_K)\}$

$$y_i \sim P_{Y|X}(y|x)$$

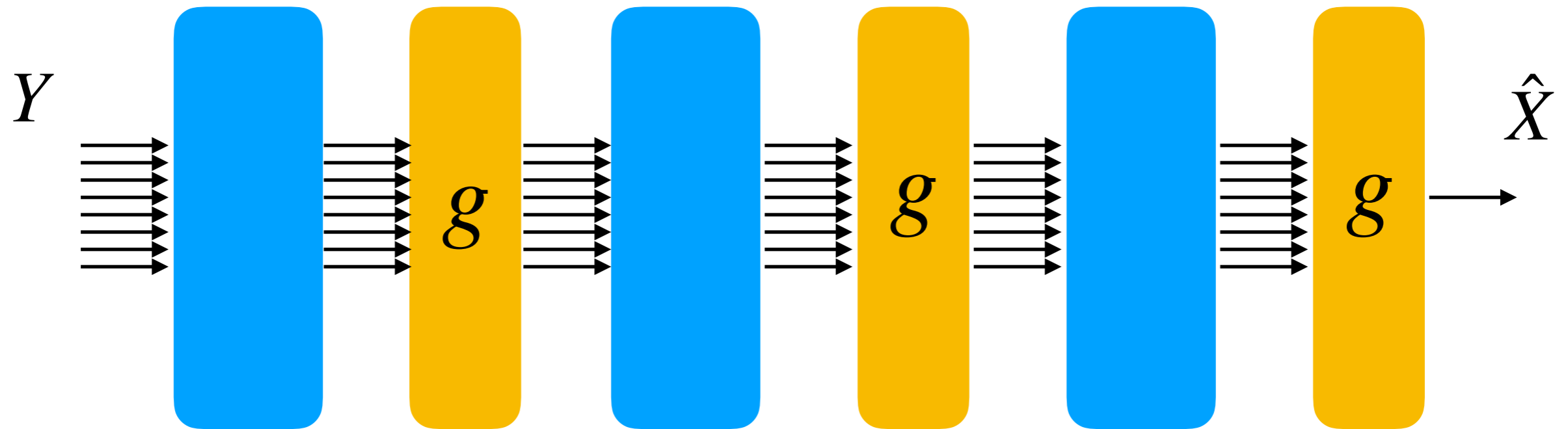
# 学習問題 (通信路) + 信号検出問題

- ✓ 通信路に関する事前知識を仮定しない
- ✓ 通信路の時間的変動に柔軟に適応
- ✓ 機械学習でいうところの“2クラス分類問題”
- ✓ 従前の“等化器トレーニング”ととても似ている

# 深層ニューラルネットワークモデル

$$f_{\Theta} : \mathbb{R}^n \rightarrow \mathbb{R}^m$$

$$\Theta = \{W_1, b_1, W_2, b_2, \dots\}$$



$$W_1 h_1 + b_1$$

線形層

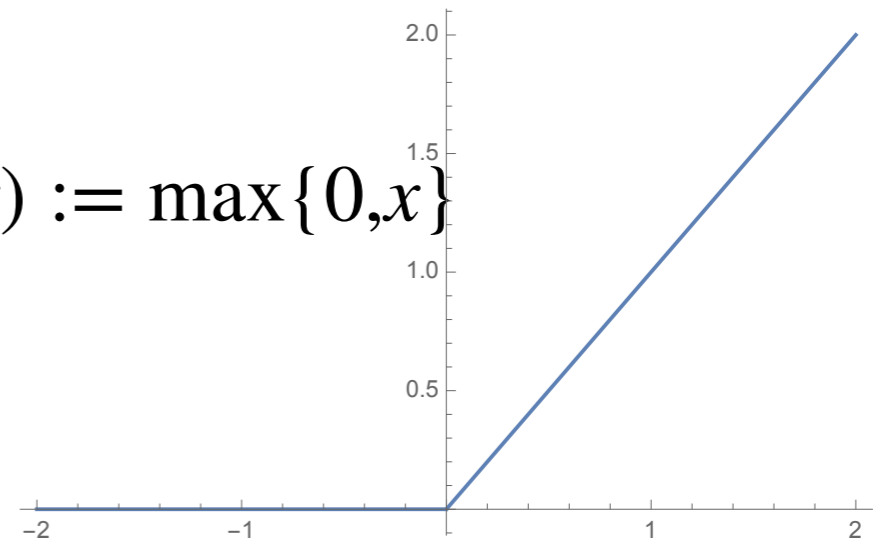
非線形活性化  
関数

$$ReLU(x) := \max\{0, x\}$$

$$h' = W h + b$$

↑

係数行列      バイアス



# アフィン変換

$$h' = W h + b$$

# 活性化関数

シグモイド関数

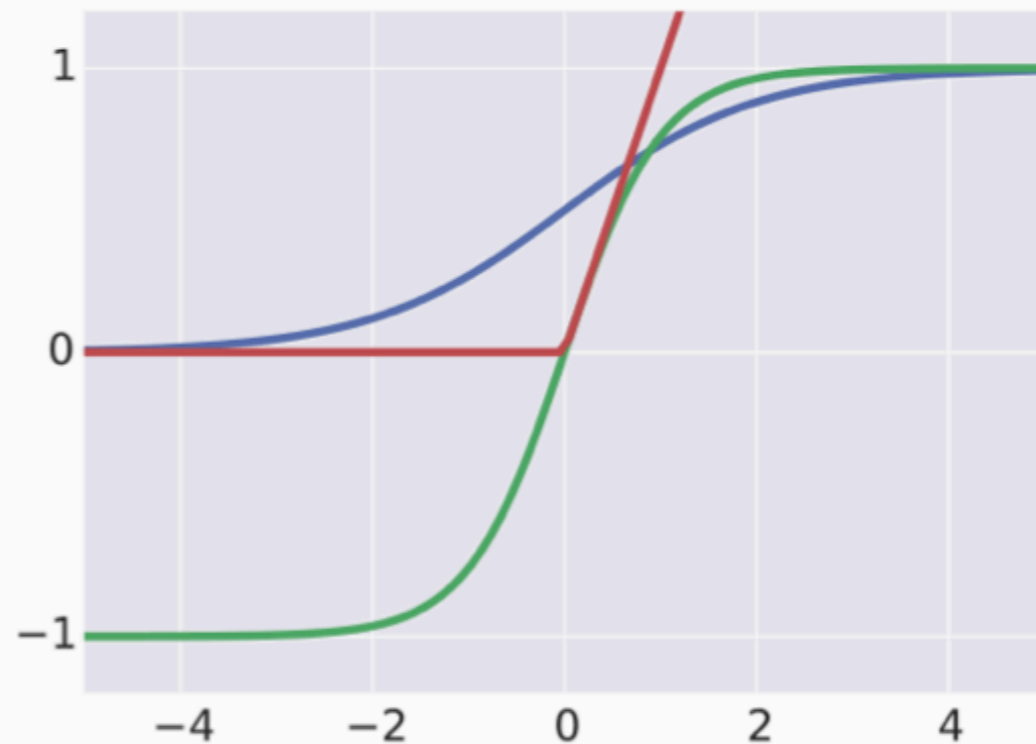
$$\sigma(u) = \frac{1}{1 + e^{-u}}$$

双曲線正接関数

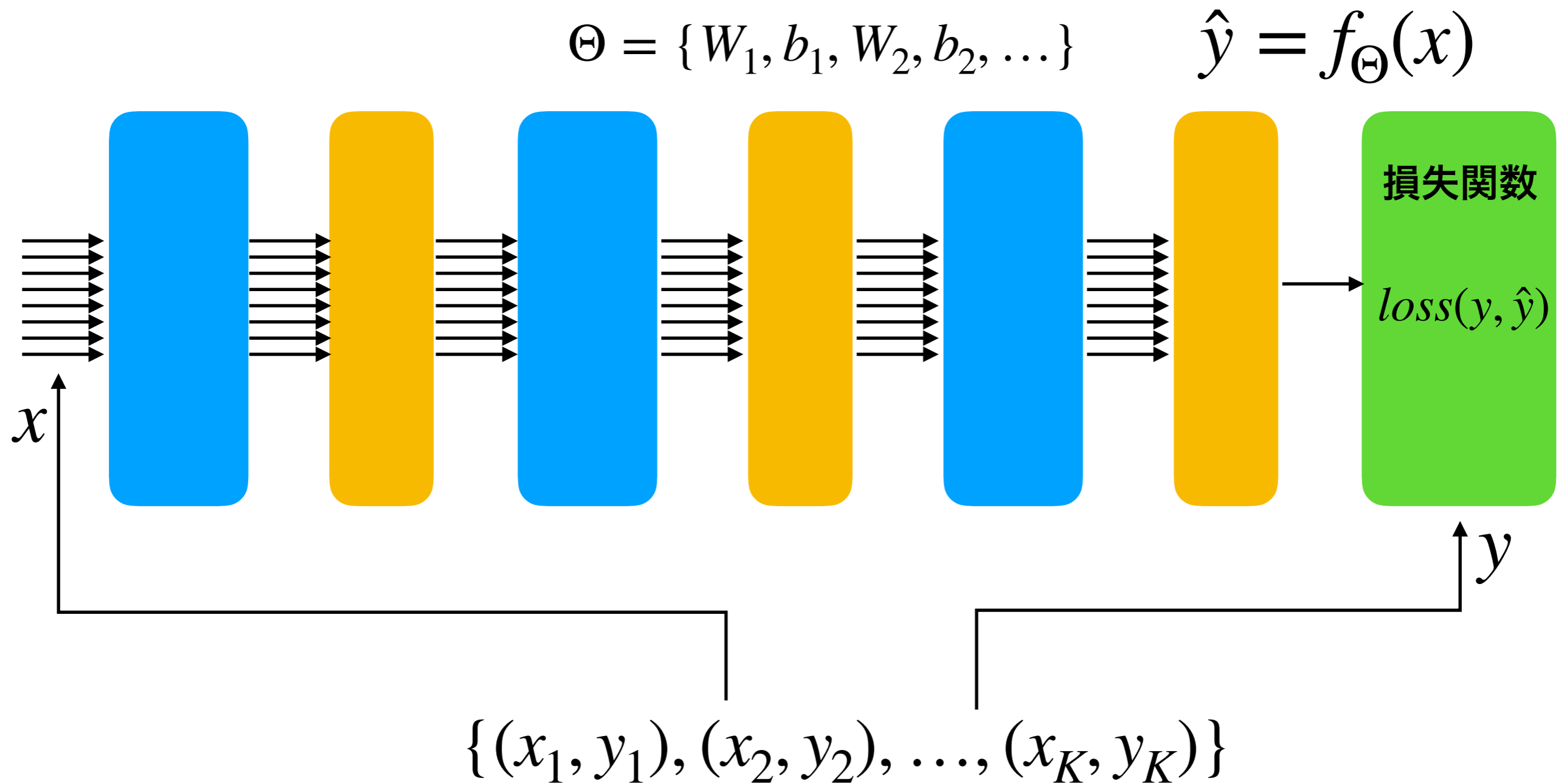
$$\tanh(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$$

ランプ関数

$$\text{ReLU}(u) = \max(0, u)$$



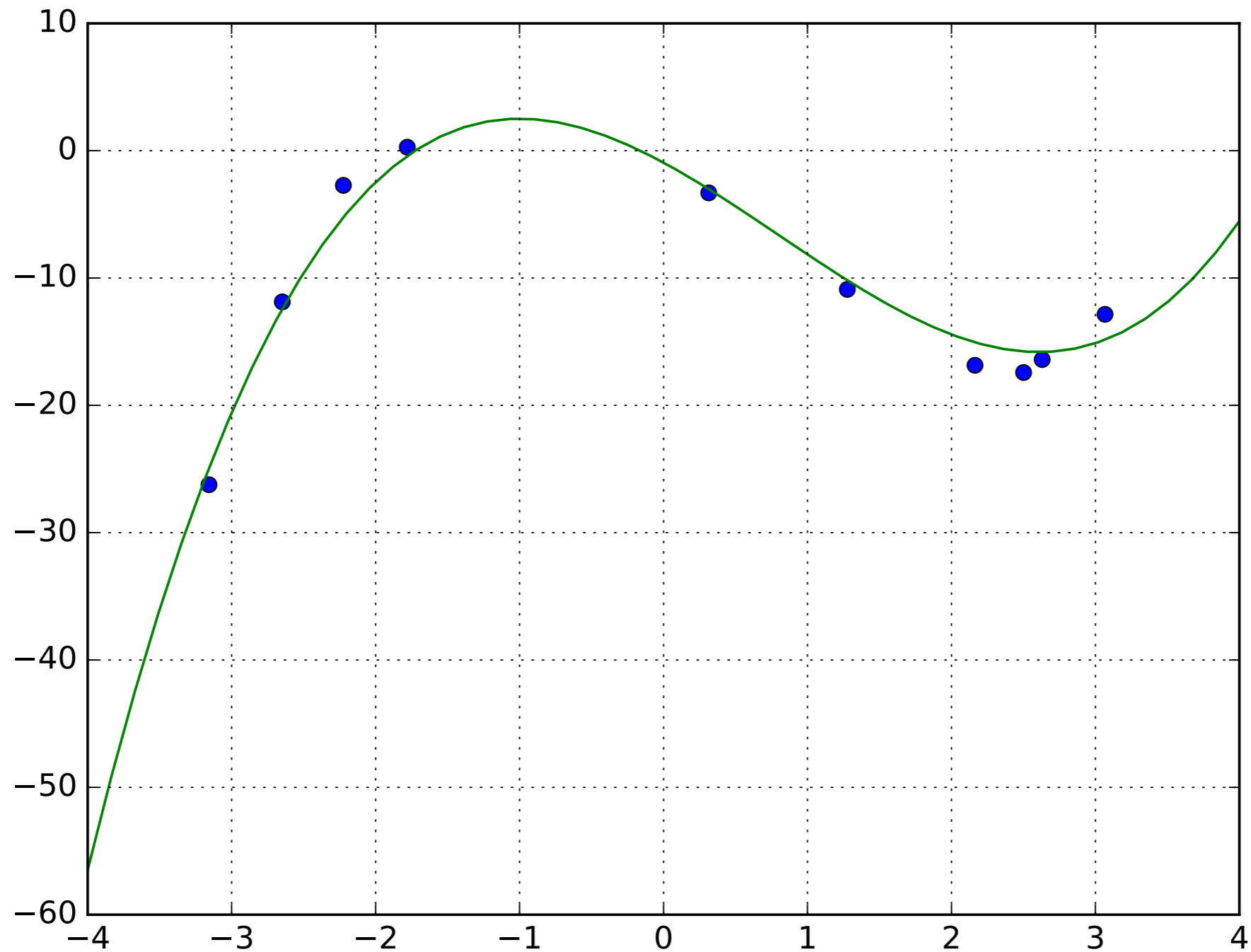
# 深層ネットワークの学習(訓練)



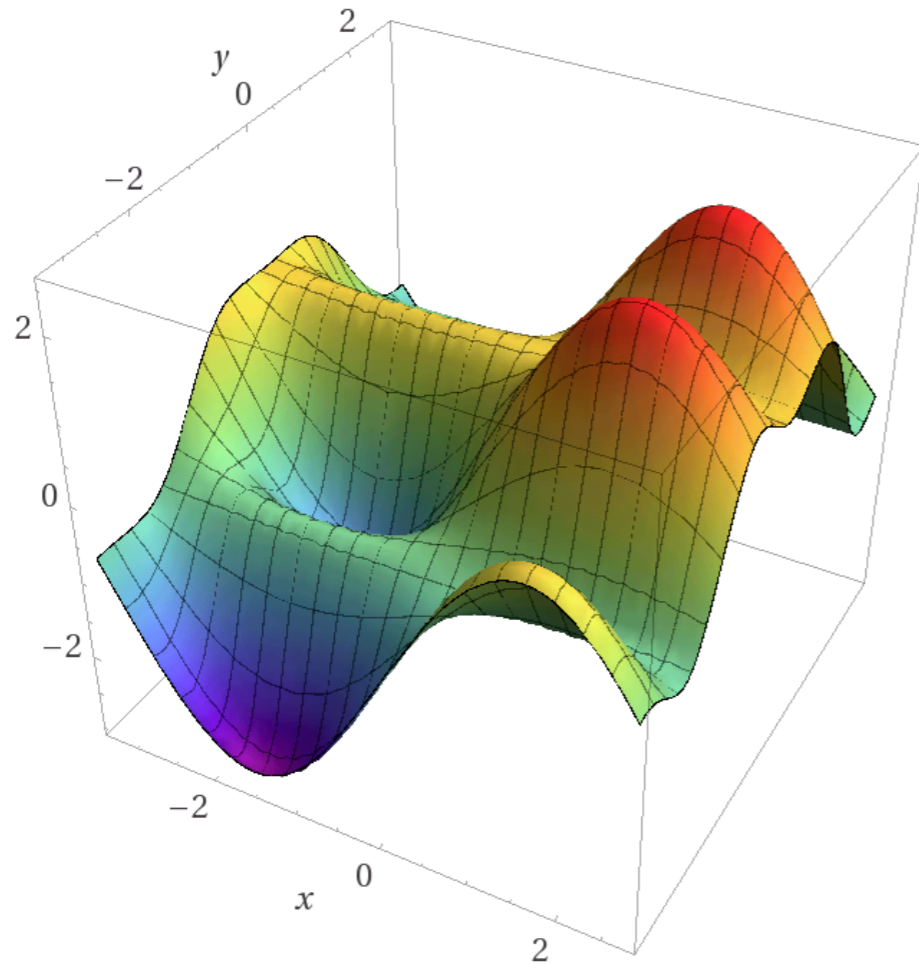
訓練・学習過程では、損失関数値を最小化するように  
パラメータを変更する



# 深層学習 = 最小二乗法の親玉？



# 非凸目的関数



- ✔ 複数の停留点（極値）
- ✔ 勾配法では局所解にしか到達できない
- ✔ 鞍点付近などでは，勾配法の収束速度が大幅にスローダウン

# 確率的勾配法 (SGD)

訓練データ  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_K, y_K)\}$

ミニバッチ  $B = \{(x_{b1}, y_{b1}), (x_{b2}, y_{b2}), \dots, (x_{bD}, y_{bD})\}$

目的関数  $G_B(\Theta) = \frac{1}{D} \sum_{k=1}^K \text{loss}(y_{bk} - f_{\Theta}(x_{bk}))$

## 確率的勾配法に基づく最小化

Step 1 (初期点設定)  $\Theta := \Theta_0$

Step 2 (ミニバッチ取得)  $B$  をランダムに生成

Step 3 (勾配ベクトルの計算)  $g := \nabla G_B(\Theta)$

Step 4 (探索点更新)  $\Theta := \Theta - \alpha g$

Step 5 (反復) Step 2 に戻る

## バリエーション

Momentum

AdaDelta

RMSprop

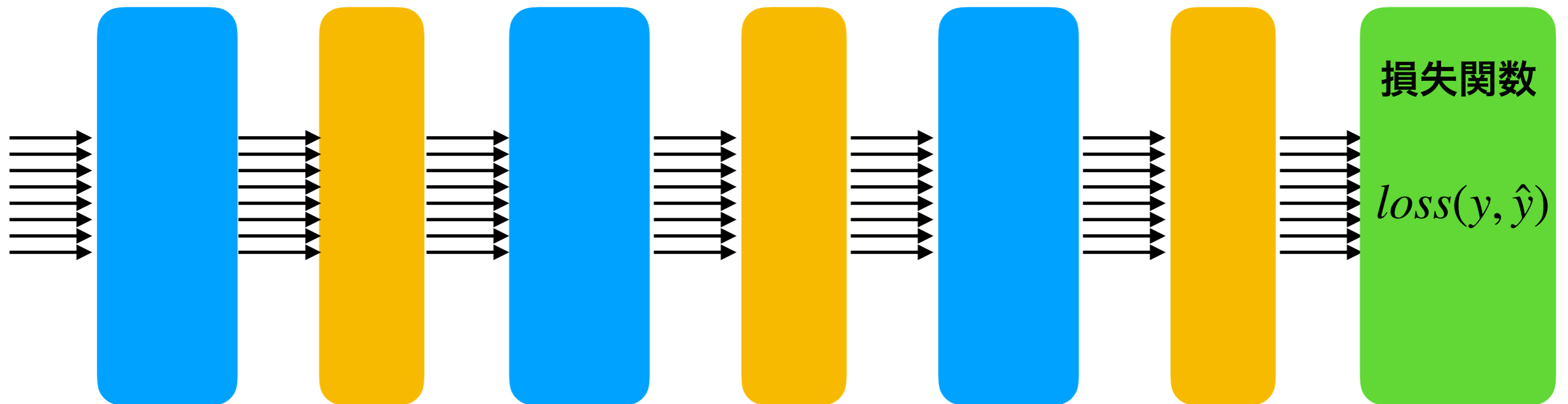
Adam

# 誤差逆伝播法(backprop)

- パラメータの勾配ベクトルを効率良く求めることが目的
- 微分の連鎖律の利用(BCJRアルゴリズムにととても良く似ている!)

前向き計算フェーズ (単なる値の評価)

注：ただし途中の計算結果を残す

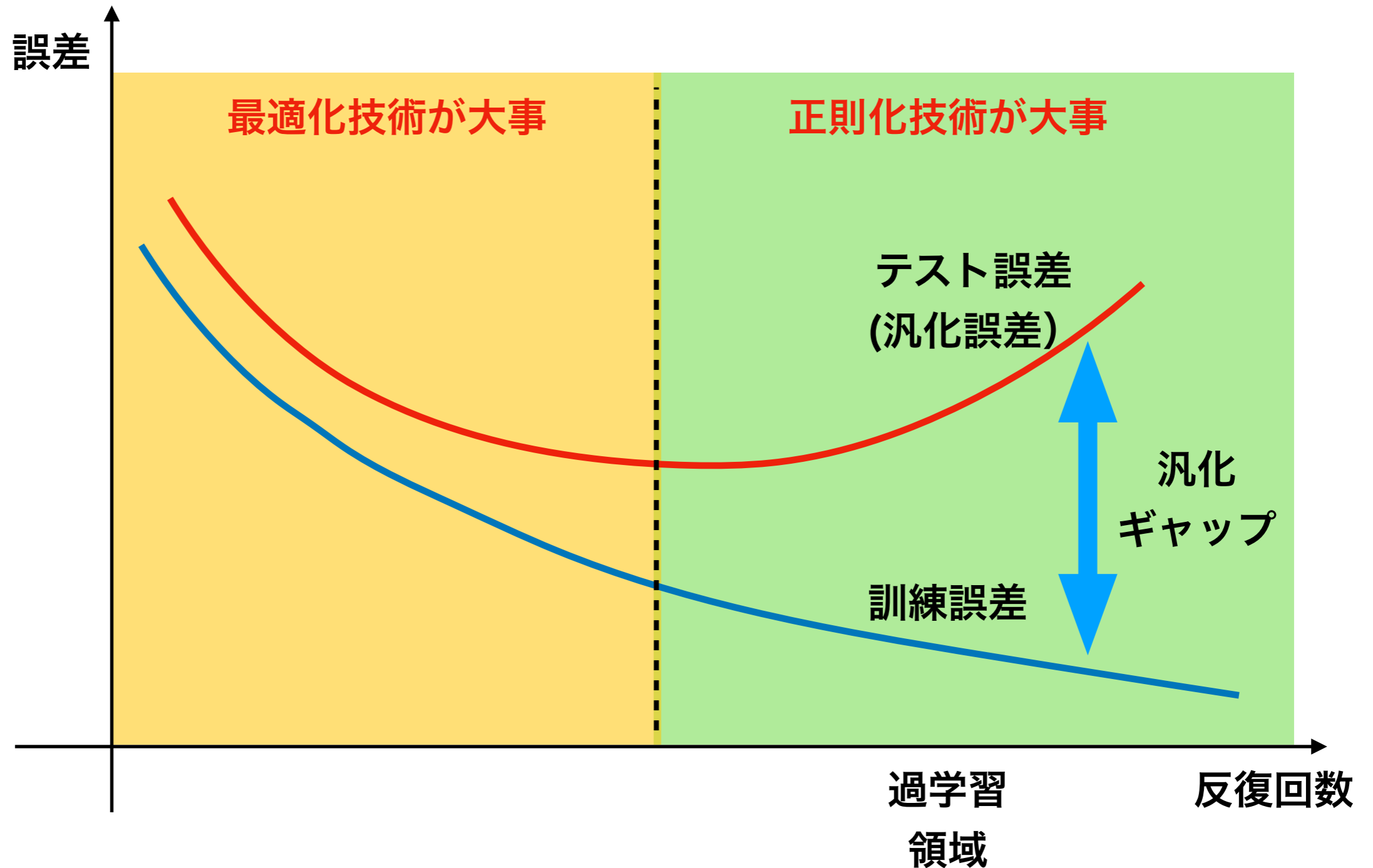


後ろ向き計算フェーズ

ヤコビ行列とベクトルの積を順次計算

# 汎化誤差

訓練データに含まれない**初見のデータ**に対して適切な出力を与えることが望ましい



# 深層学習フレームワーク

- PyTorchによるプログラム例

## イチオシは

```
class Net(nn.Module):  
    def __init__(self):  
        super(Net, self).__init__()  
        self.fc1 = nn.Linear(2, 2)  
        self.fc2 = nn.Linear(2, 2)  
    def forward(self, x):  
        x = F.sigmoid(self.fc1(x))  
        x = F.sigmoid(self.fc2(x))  
        return x
```



### ネットワークの定義部

順方向計算のみを記述すればよい。

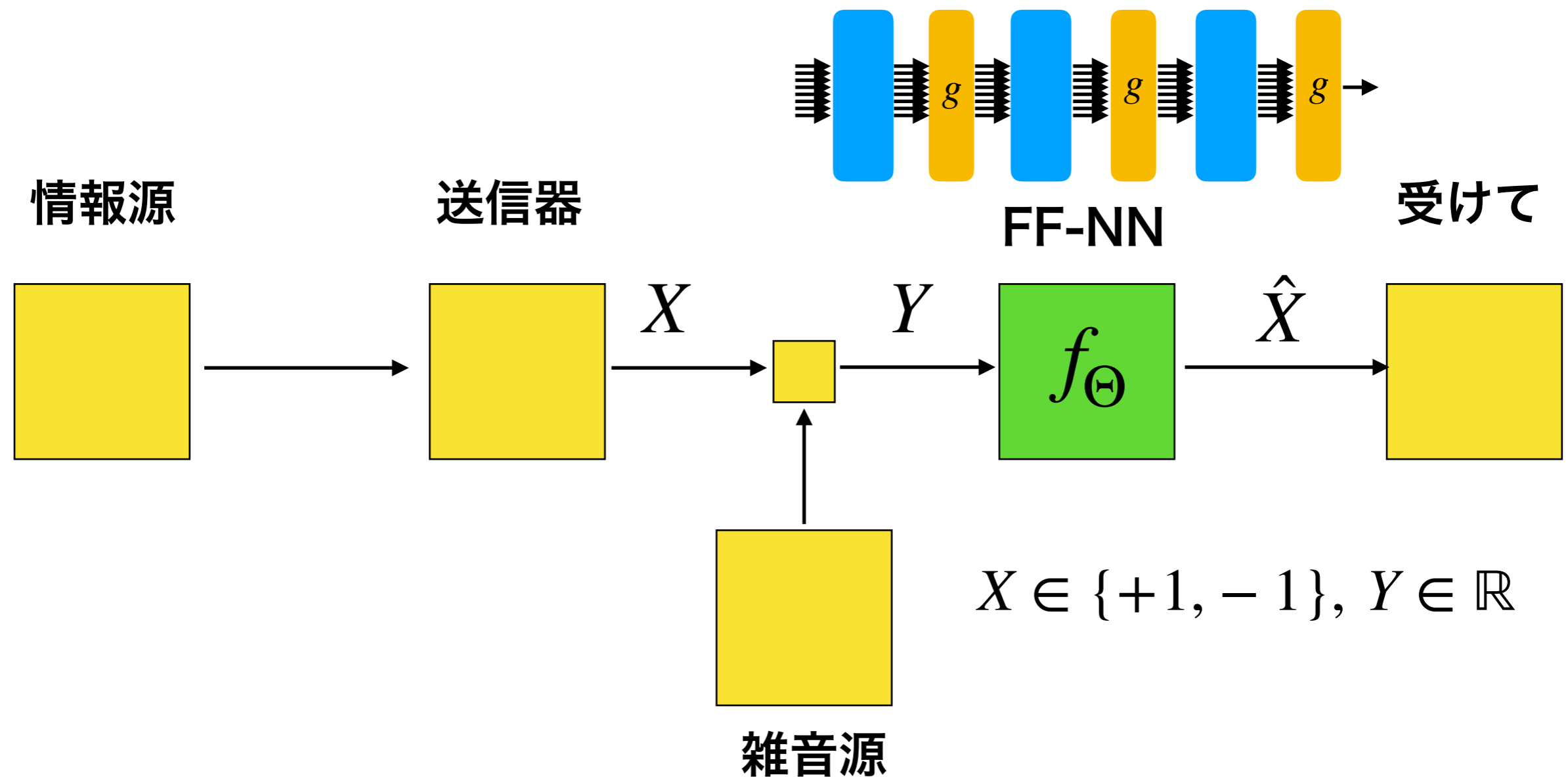
誤差逆伝播法の後向き計算フェーズは  
明示的にユーザが書く必要ない

```
model = Net()           ネットワークのインスタンス化  
loss_func = nn.MSELoss() 損失関数の指定  
optimizer = optim.Adam(model.parameters(), lr=0.1) オプティマイザの指定
```

# 機械学習を利用した無線物理層技術の分類

タイプ	アイデア
ブラックボックスモデル	通信路全体・検出アルゴリズム・復号アルゴリズムを深層ニューラルネットワークを利用して表現。データ駆動型設計。事前知識・過去の知見必要なし。
深層展開	既知の反復型アルゴリズムに学習可能パラメータを埋め込み深層学習技術を利用してパラメータを学習。
最適化模擬	凸最適化・非凸最適化アルゴリズムのニューラルネットワークによる模擬。
分散学習	各端末における学習とその学習結果の統合
強化学習	報酬に基づく最適戦略の決定、マルコフ決定プロセス。

# 信号検出器の学習



この形の検出器の論文はいっぱい出ています。その一例：

N.Samuel et al. "Learning to Detect," IEEE Trans. Signal Processing, May, 2019



# ブラックボックスモデルに基づくスパース信号再現

Ito and Wadayama, SITA2016

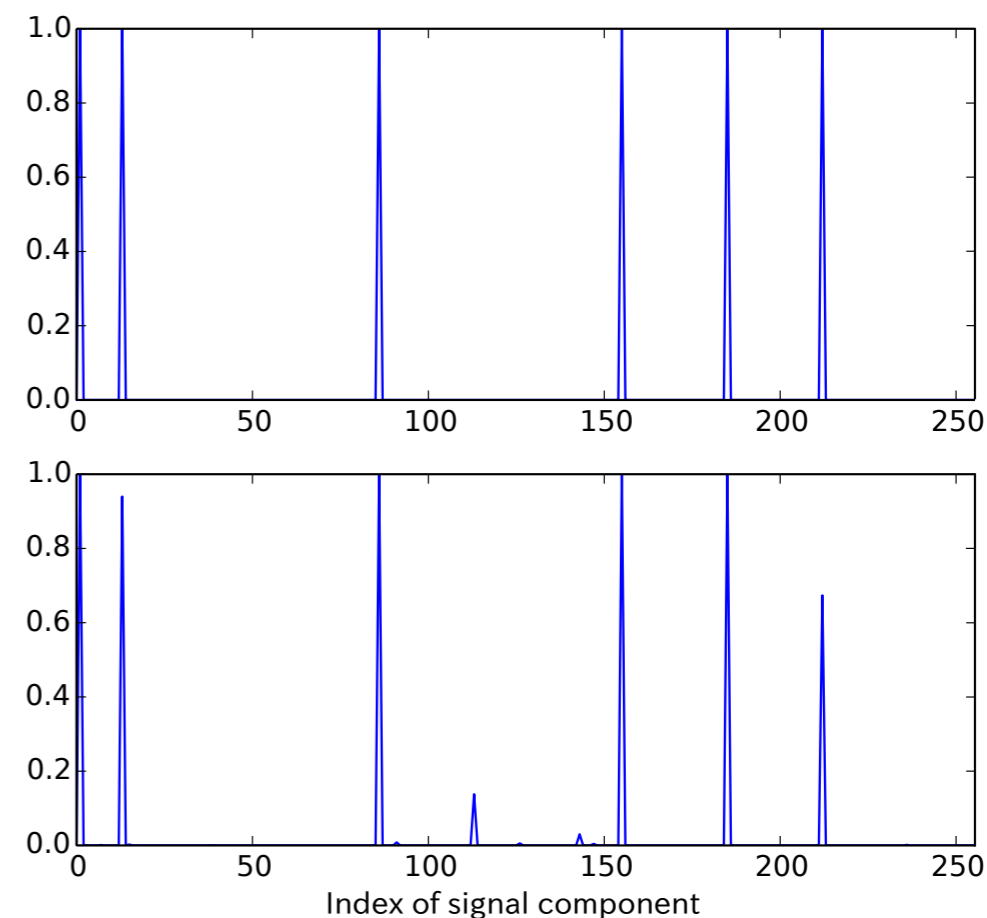
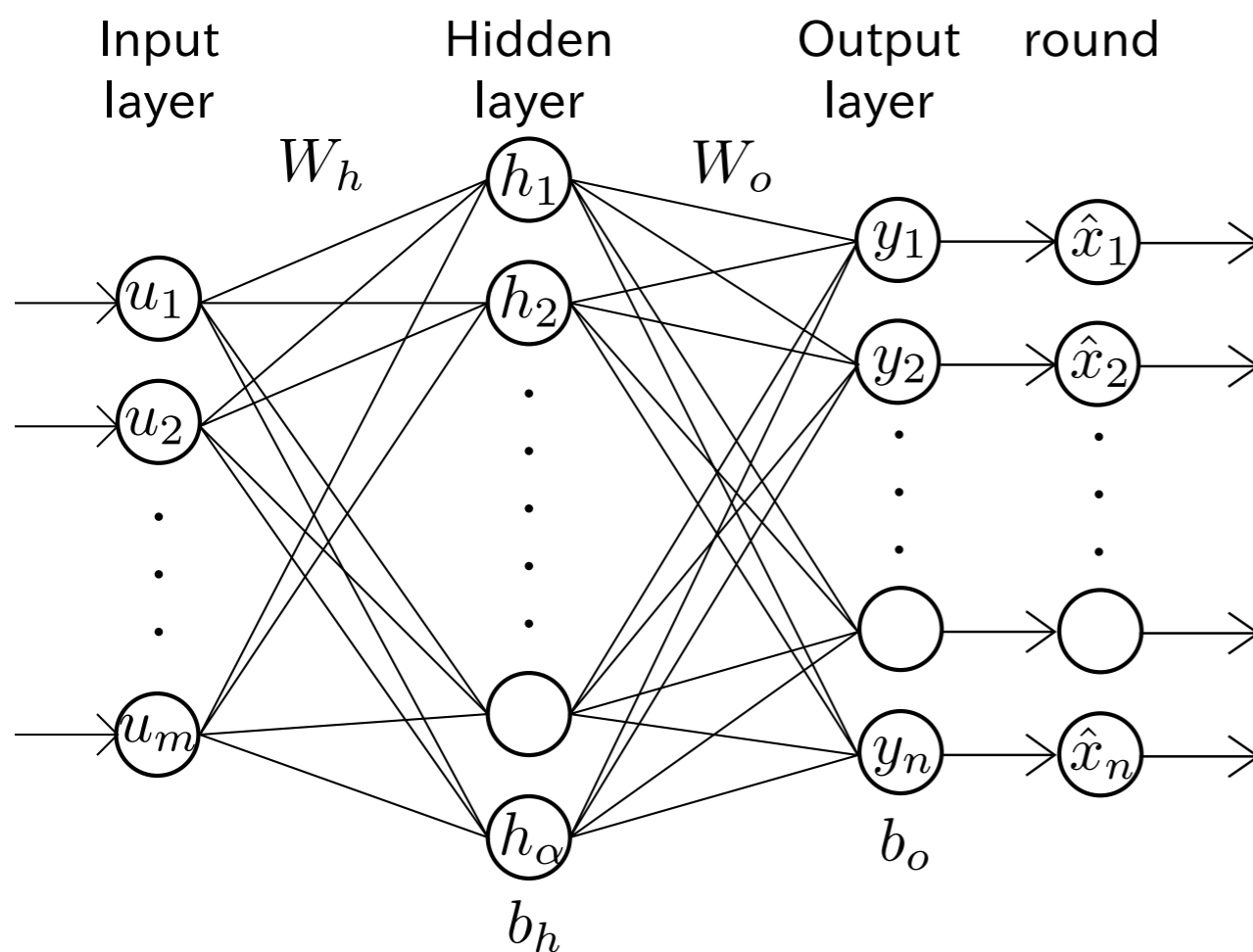


Fig. 2. Sparse signal recovery for a 6-sparse vector. (top: the original sparse signal  $x$ , bottom: the output  $y = \Phi_{\theta^*}(x)$  from the trained neural network.  $n = 256, m = 120$ )

ブラックボックスモデル

どうも大きなシステムにうまくスケールしない→深層展開へ方向転換

arXiv:1610.09463

# 機械学習を利用した無線物理層技術の分類

タイプ	アイデア
ブラックボックスモデル	通信路全体・検出アルゴリズム・復号アルゴリズムを深層ニューラルネットワークを利用して表現。データ駆動型設計。事前知識・過去の知見必要なし。
深層展開	既知の反復型アルゴリズムに学習可能パラメータを埋め込み深層学習技術を利用してパラメータを学習。
最適化模擬	凸最適化・非凸最適化アルゴリズムのニューラルネットワークによる模擬。
分散学習	各端末における学習とその学習結果の統合
強化学習	報酬に基づく最適戦略の決定、マルコフ決定プロセス。

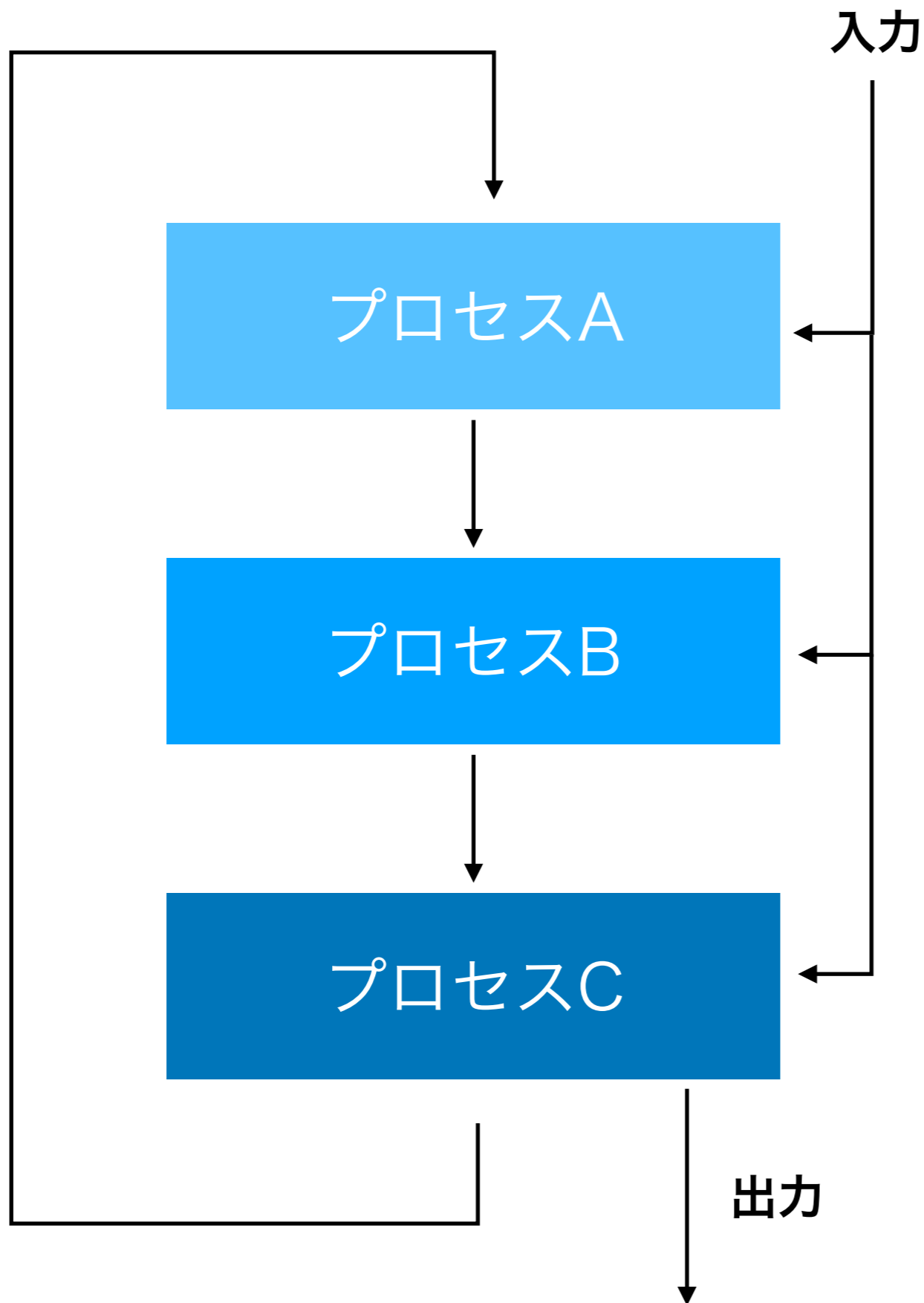
# 可微分プログラミング

- ✓ LeCunは、入出力を伴う処理(NNとは限らない)に対して、深層学習技術が適用可能であることを示唆している

… important point is that people are now building a new kind of software by **assembling networks of parameterized functional blocks** and by training them from examples using some form of gradient-based optimization…

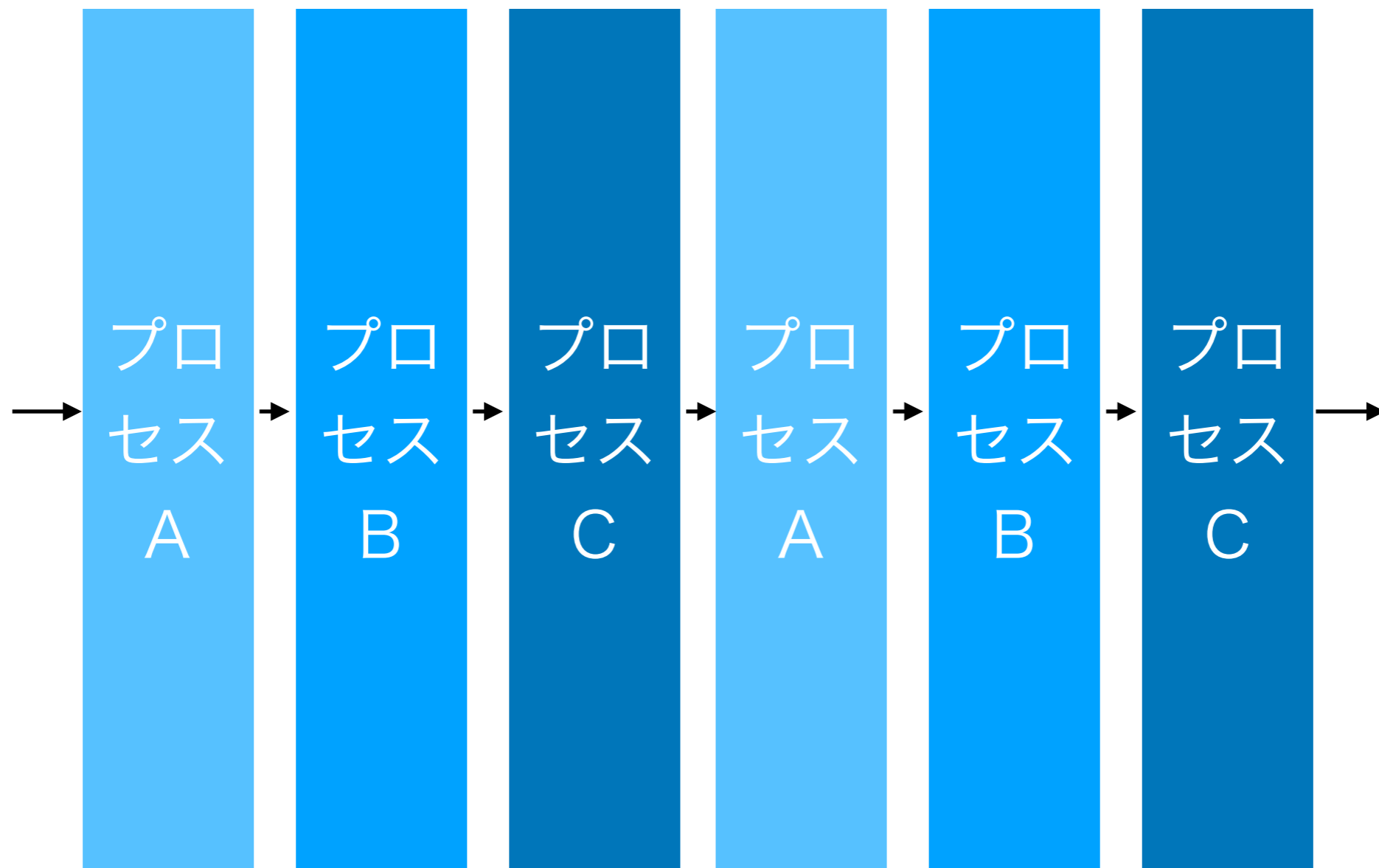
- ✓ 処理全体が微分可能であれば、内部に含まれるパラメータをbackprop+SGDで最適化できる  
→ **可微分プログラミング**  
(differentiable programming)

# 深層展開による反復アルゴリズムの改善



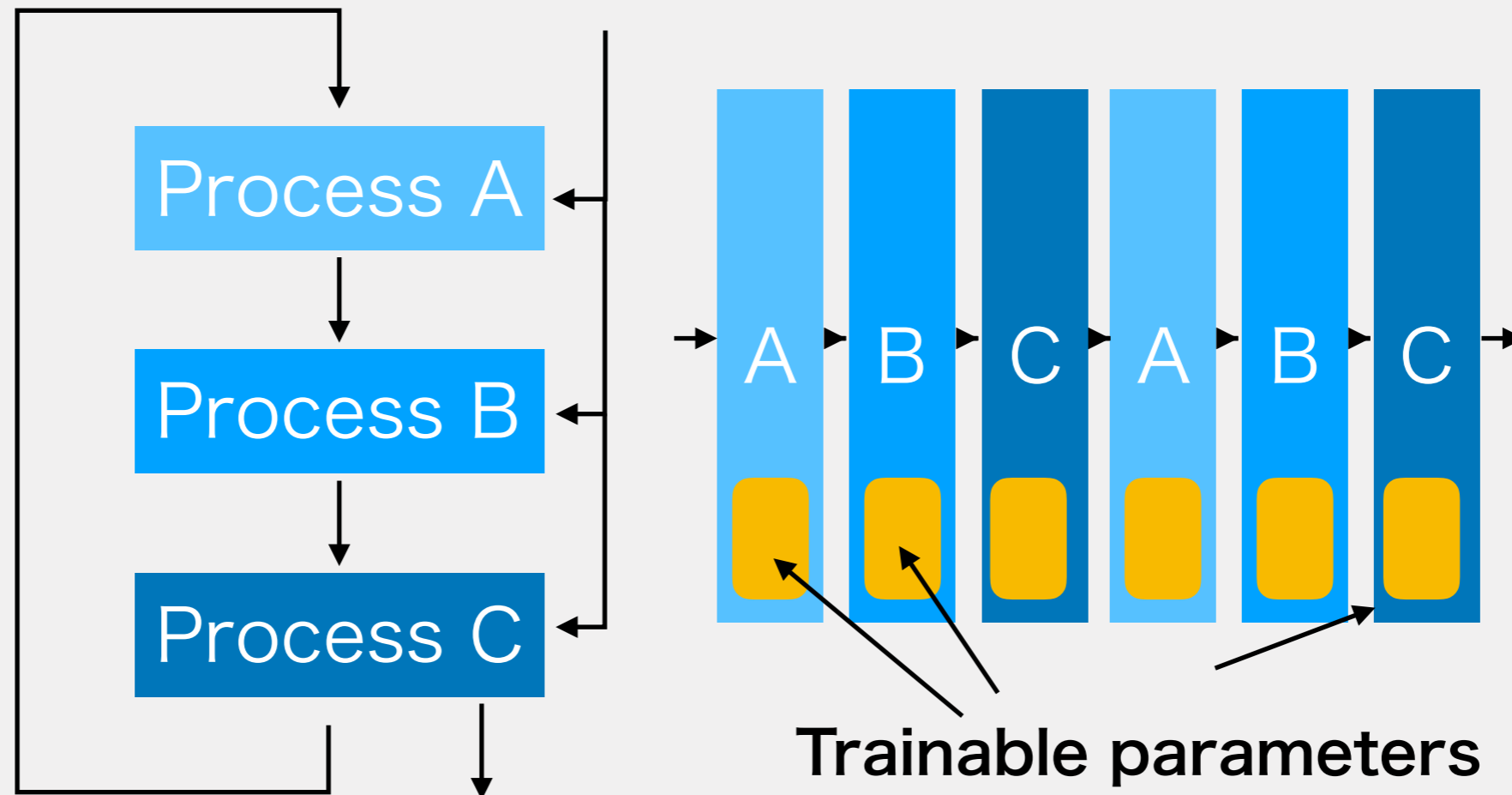
- ビリーフプロパゲーション
- ガウス・ザイデル法
- 共役勾配法
- 反復収縮法 (ISTA)
- AMP
- ADMM
- ...

# 信号フローを時間方向に展開



各プロセスが微分可能であり，かつ，その導関数が  
ほぼ至るところでゼロでなければ，**backprop可能**

# 深層展開 (Deep unfolding)



既知の反復型型アルゴリズムに学習可能パラメータを埋め込み深層学習技術を利用してパラメータを学習

A recent survey: A. Balatsoukas-Stimming and C. Studer, arXiv: 1906.05774, 2019.

# 射影勾配法に基づくLDPC復号法

- ✓ LP復号法(最適化ベース復号法) をもとにして
- ✓ 射影勾配法＋深層展開
- ✓ ペナルティ関数法も併用・符号制約ペナルティ
- ✓ 非線形目的関数
- ✓ BP性能を越すことが目標

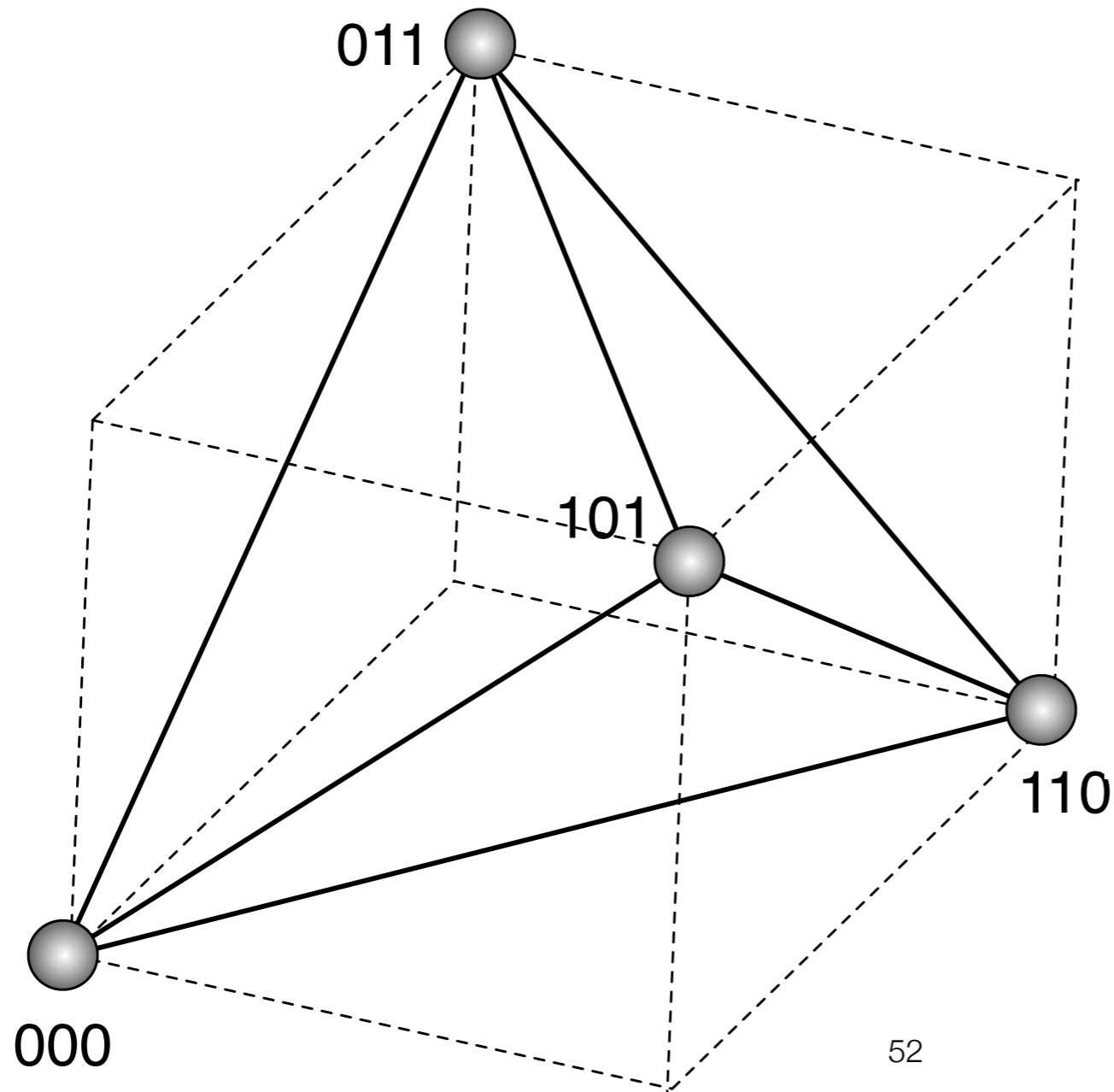
Wadayama, Takabe,

Deep Learning-Aided Trainable Projected Gradient Decoding for LDPC Codes

# 単一パリティ検査符号の凸包

単一パリティ検査符号 (n=3)

$C = \{000, 110, 101, 011\}$



$$x + (1 - y) + (1 - z) \leq 2$$

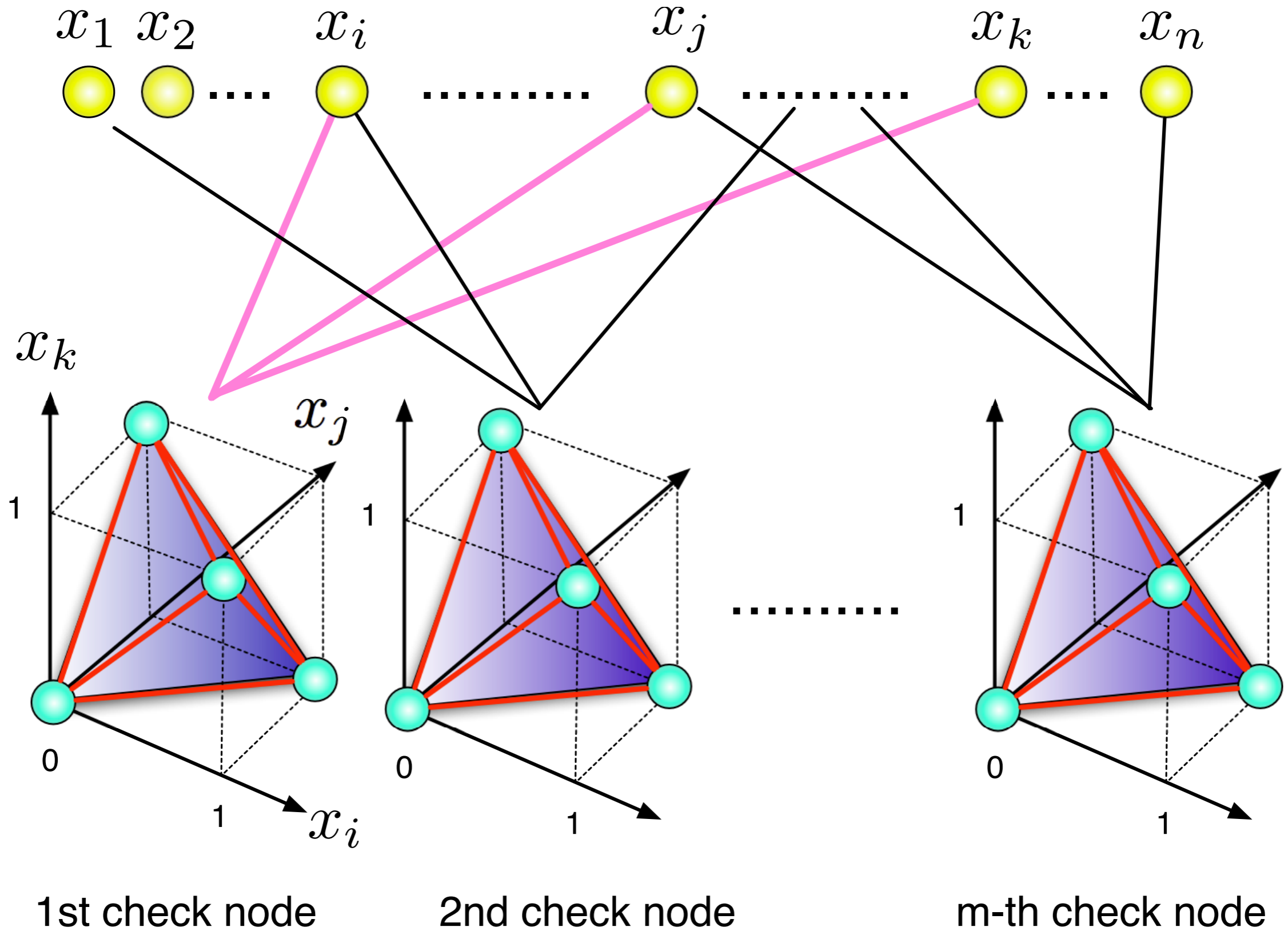
$$y + (1 - x) + (1 - z) \leq 2$$

$$z + (1 - x) + (1 - y) \leq 2$$

$$x + y + z \leq 2$$



# 基本多面体



# 線形計画 (LP) 復号法

## Feldman 2003

$$H \in \mathbb{F}_2^{m \times n}$$

… 検査行列

$$C(H) := \{x \in \mathbb{F}_2^n \mid Hx^T = \mathbf{0}\}$$

… 2元LDPC符号

$$A_i := \{j \in [n] \mid h_{i,j} = 1\}$$

… i-th インデックス集合

$$O_i := \{S \subset A_i \mid |S| \text{ is odd}\}$$

… 奇数サイズ部分集合

## 基本多面体

$$\mathcal{Q}(H) := \{x \in \mathbb{R}^n \mid x \text{ satisfies the parity constraint}\}$$

$$\forall i \in [m], \forall S \in O_i, \quad 1 + \sum_{t \in S} (x_t - 1) - \sum_{t \in A_i \setminus S} x_t \leq 0$$

## 線形計画復号法

$$\text{minimize}_{x \in \mathbb{R}^n} \lambda x^T \text{ subject to } x \in \mathcal{Q}(H)$$

# ペナルティ関数

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

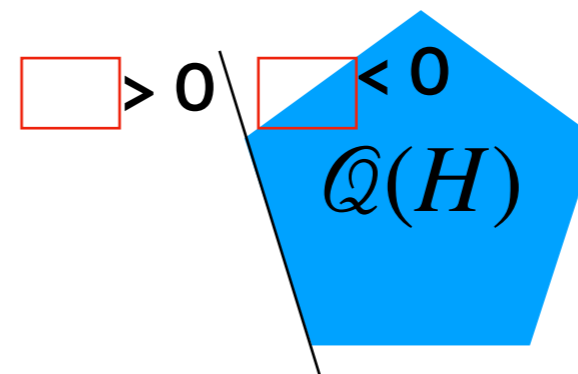
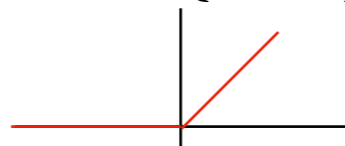
$$P(x) = 0 \text{ if } x \in Q(H)$$
$$P(x) > 0 \text{ otherwise}$$

## 基本多面体に対応するペナルティ関数

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

## ReLU 関数

$$\nu(x) := \max\{0, x\}$$



# ペナルティ関数の勾配ベクトル

## ペナルティ関数の偏微分

$$\frac{\partial}{\partial x_k} P(x) = \sum_{i \in [m]} \sum_{S \in O_i} \nu \left( 1 + \sum_{t \in S} (x_t - 1) - \sum_{t \in A_i \setminus S} x_t \right) (\mathbb{1}[k \in S] - \mathbb{1}[k \in A_i \setminus S])$$

## ペナルティ項の勾配ベクトル

$$\nabla P(x) = \nu(1 + (x - 1)Q - xR)D^T$$

$$Q, R, D \in \{0, 1\}^{n \times L}$$

$$L := \sum_{i=1}^m 2^{|A_i|} - 1$$

# 学習可能-射影勾配法

## 勾配ステップ

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

ステップ

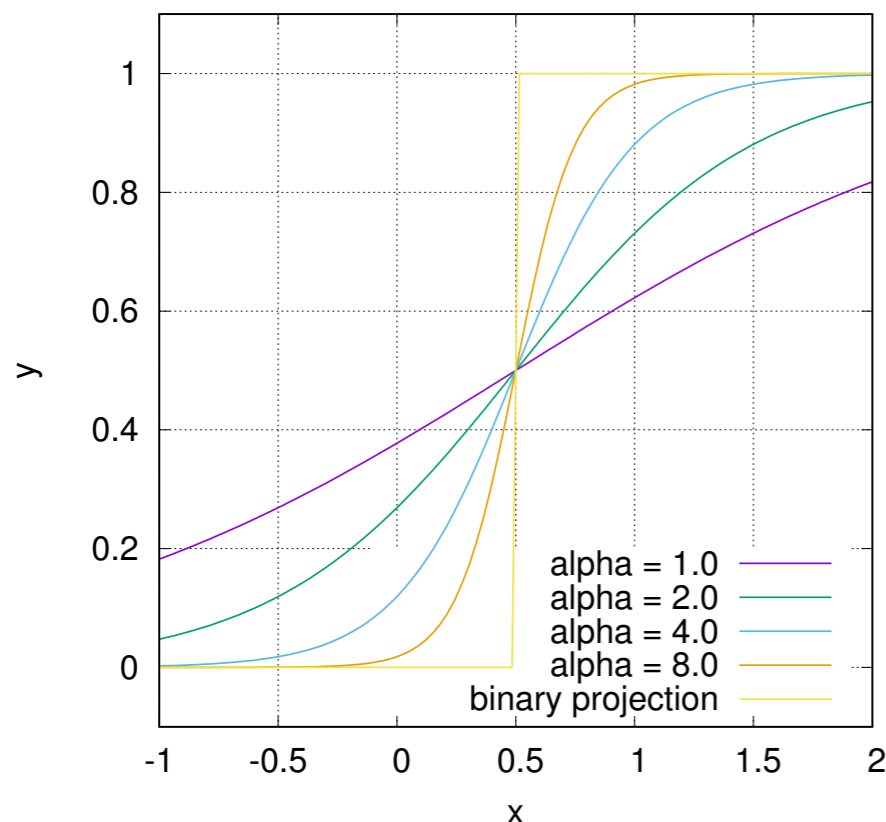
ペナルティ係数

## ソフト射影ステップ

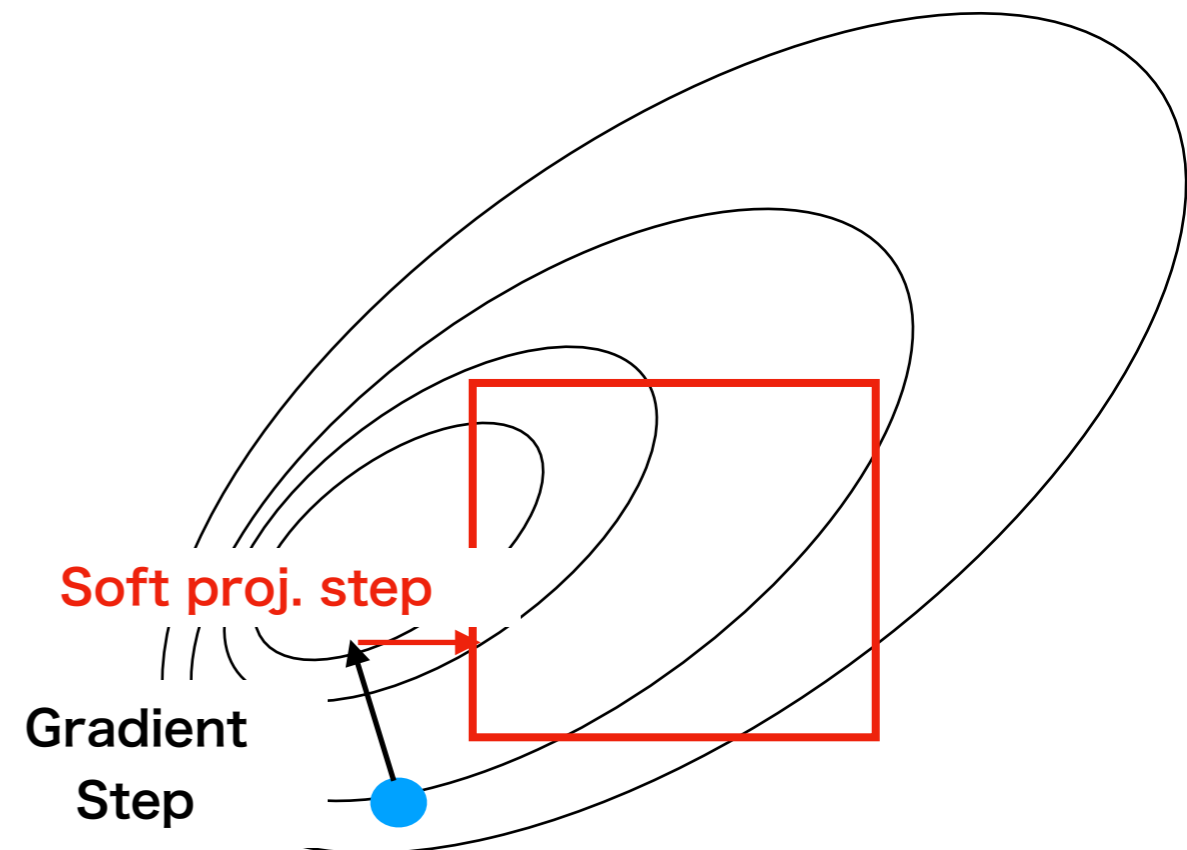
$$\mathbf{s}_{t+1} := \xi (\alpha (\mathbf{r}_t - 0.5))$$

サイズ

ソフトネス

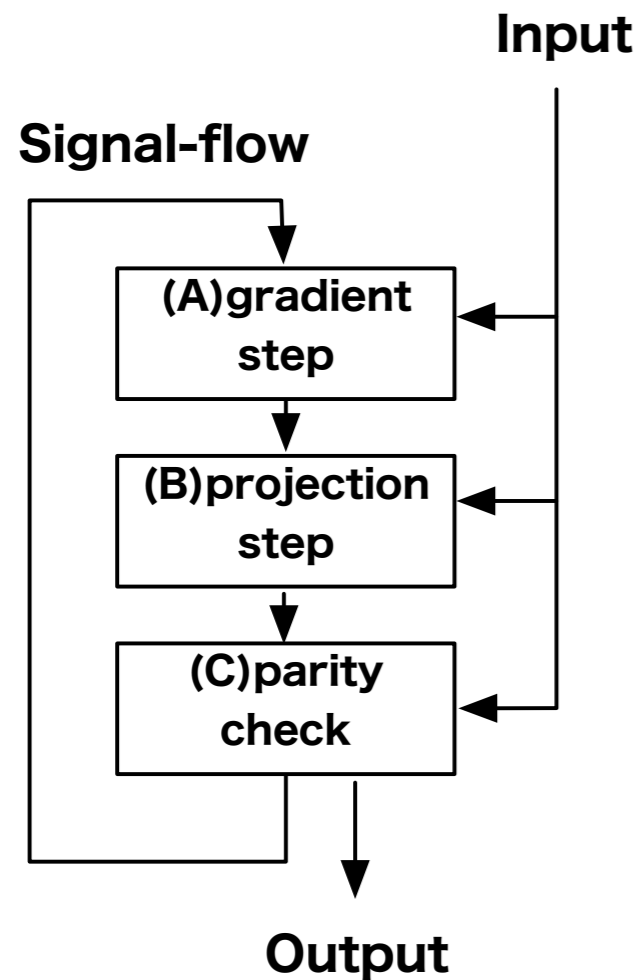


57

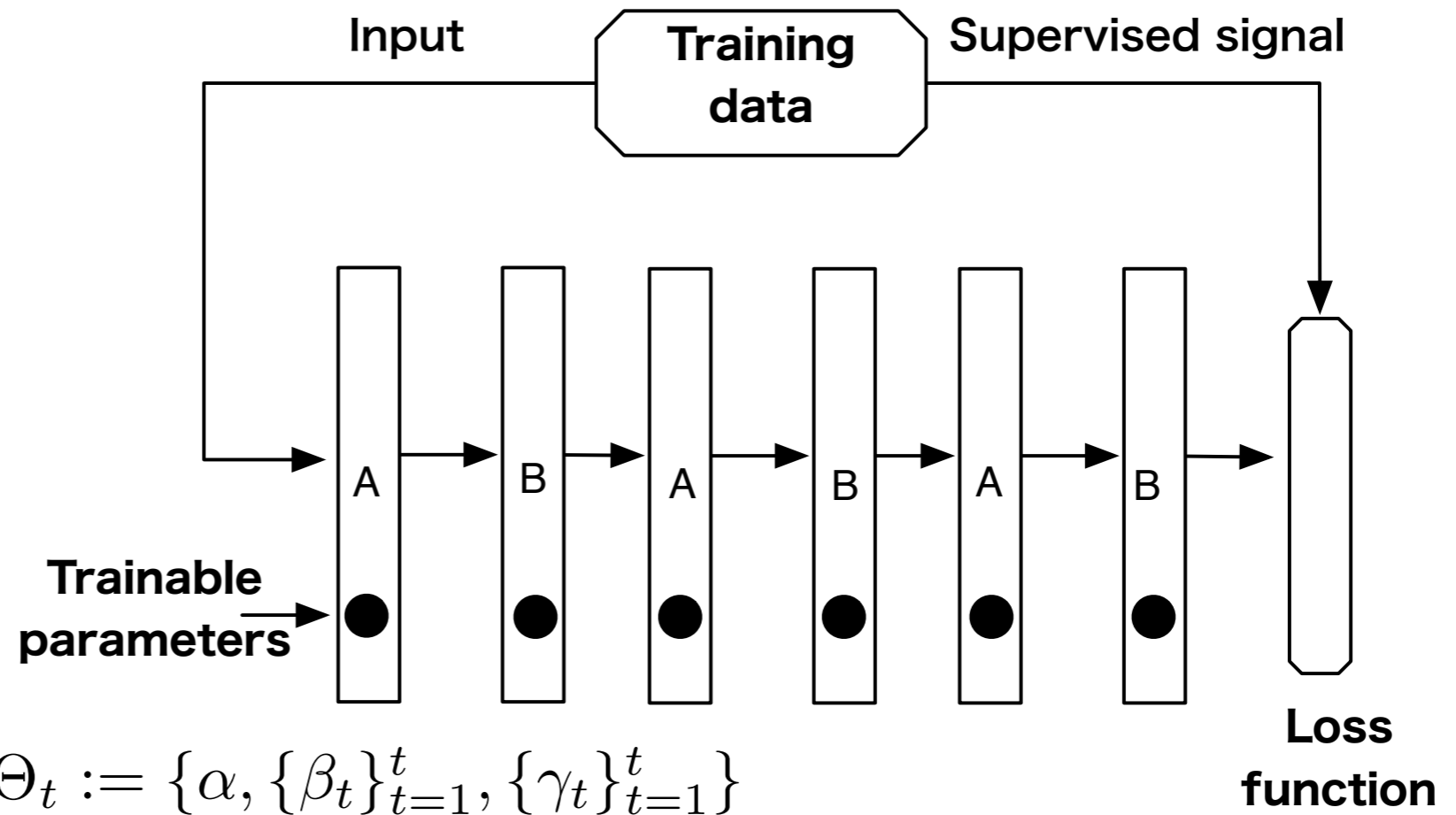


# 学習 (訓練) プロセス

$$B := \{(c_1, y_1, s_{1,1}), (c_2, y_2, s_{1,2}), \dots, (c_K, y_K, s_{1,K})\}$$



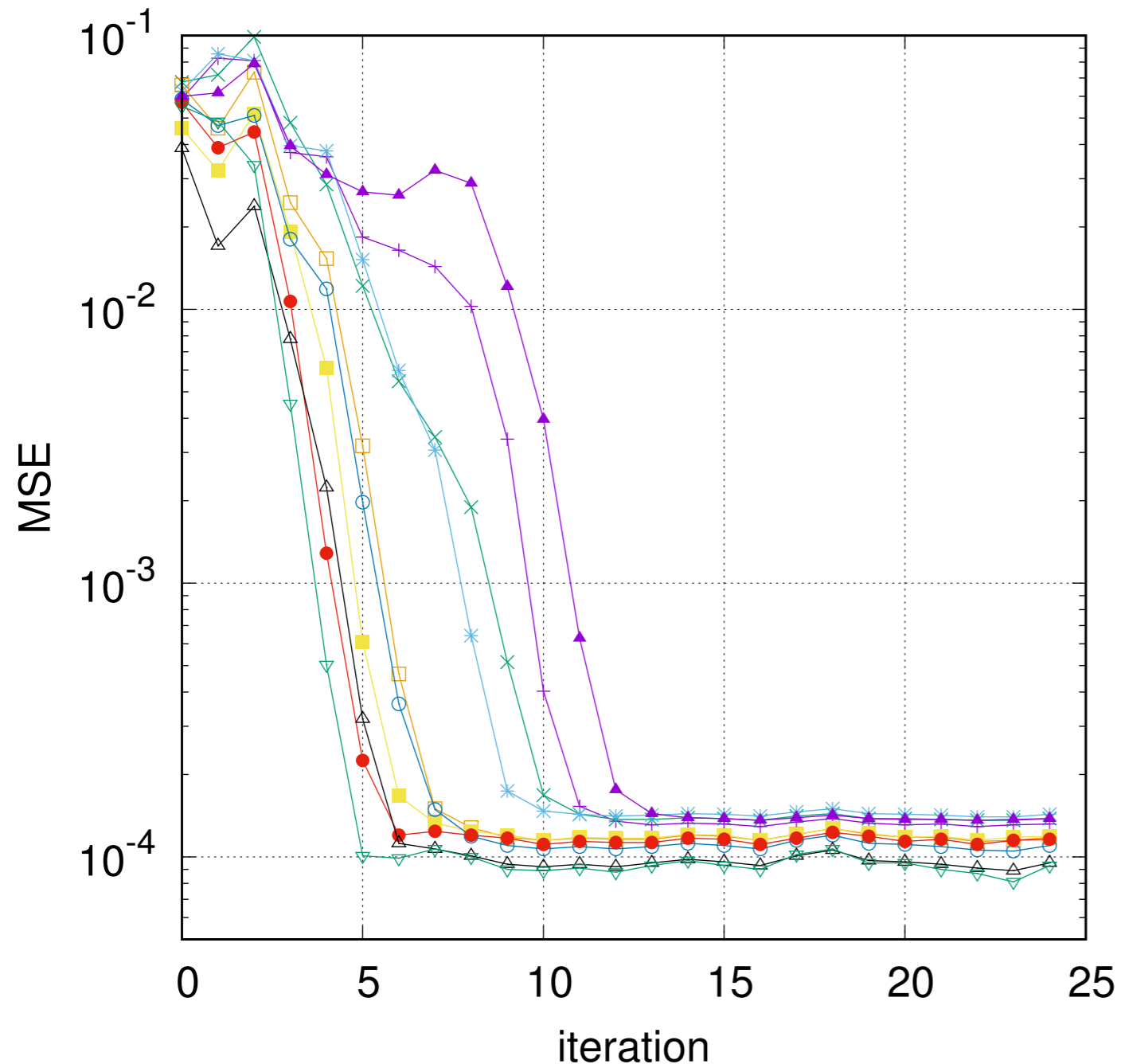
(a) Signal-flow diagram of TPG decoding



(b) Training process of a TPG decoder

# ランダムリスタート

## MSEデータ (10回の復号試行)

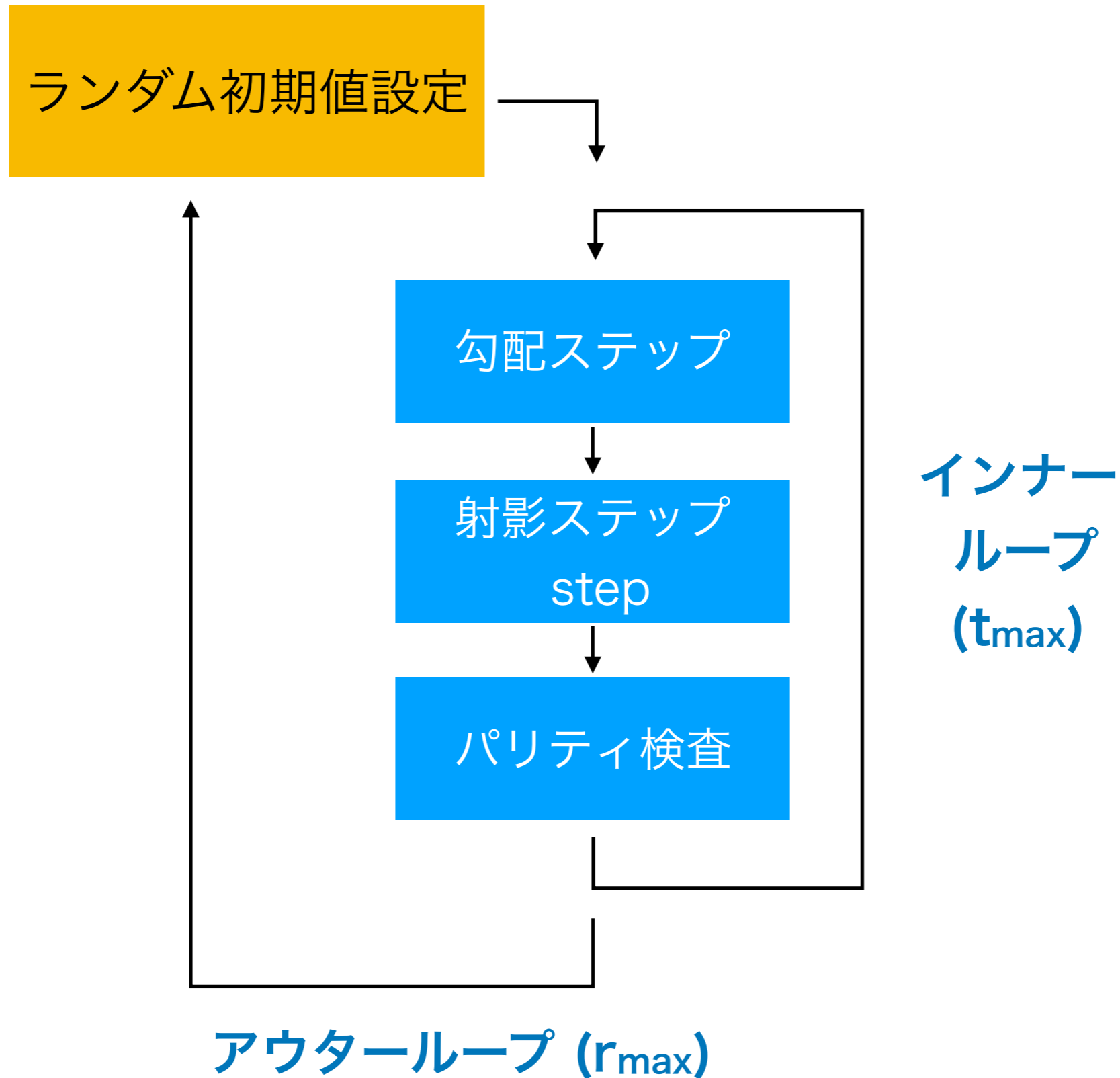


- 初期点はランダムに設定される
- 探索点の軌跡は初期値依存
- ランダムリスタートが復号特性の改善に役立つ可能性がある

固定された受信に対するMSEカーブ (ランダム初期値)

( $n = 204, m = 102, \text{SNR} = 4.0 \text{ (dB)}$ )

# 提案復号法全体のブロック図



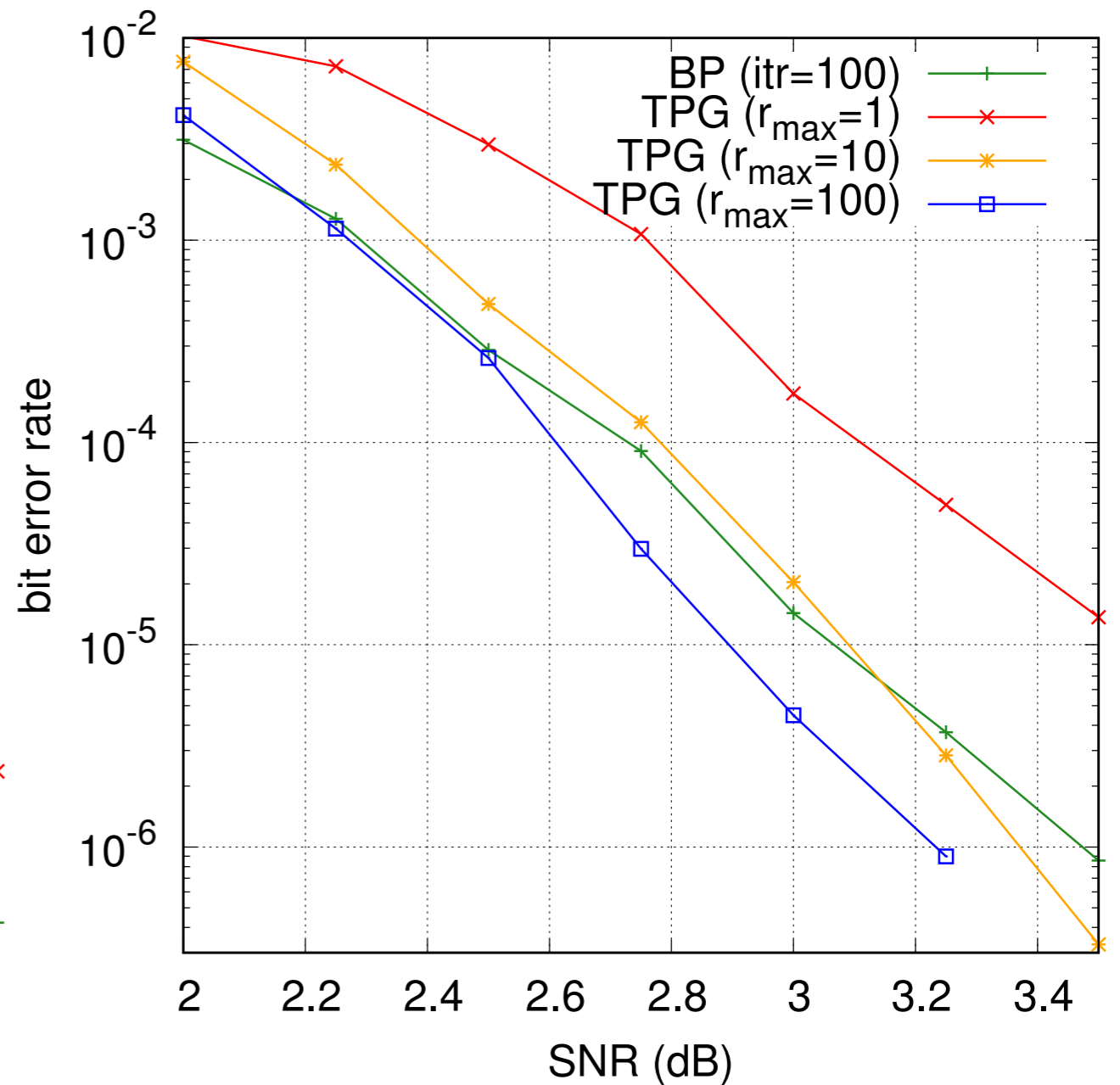
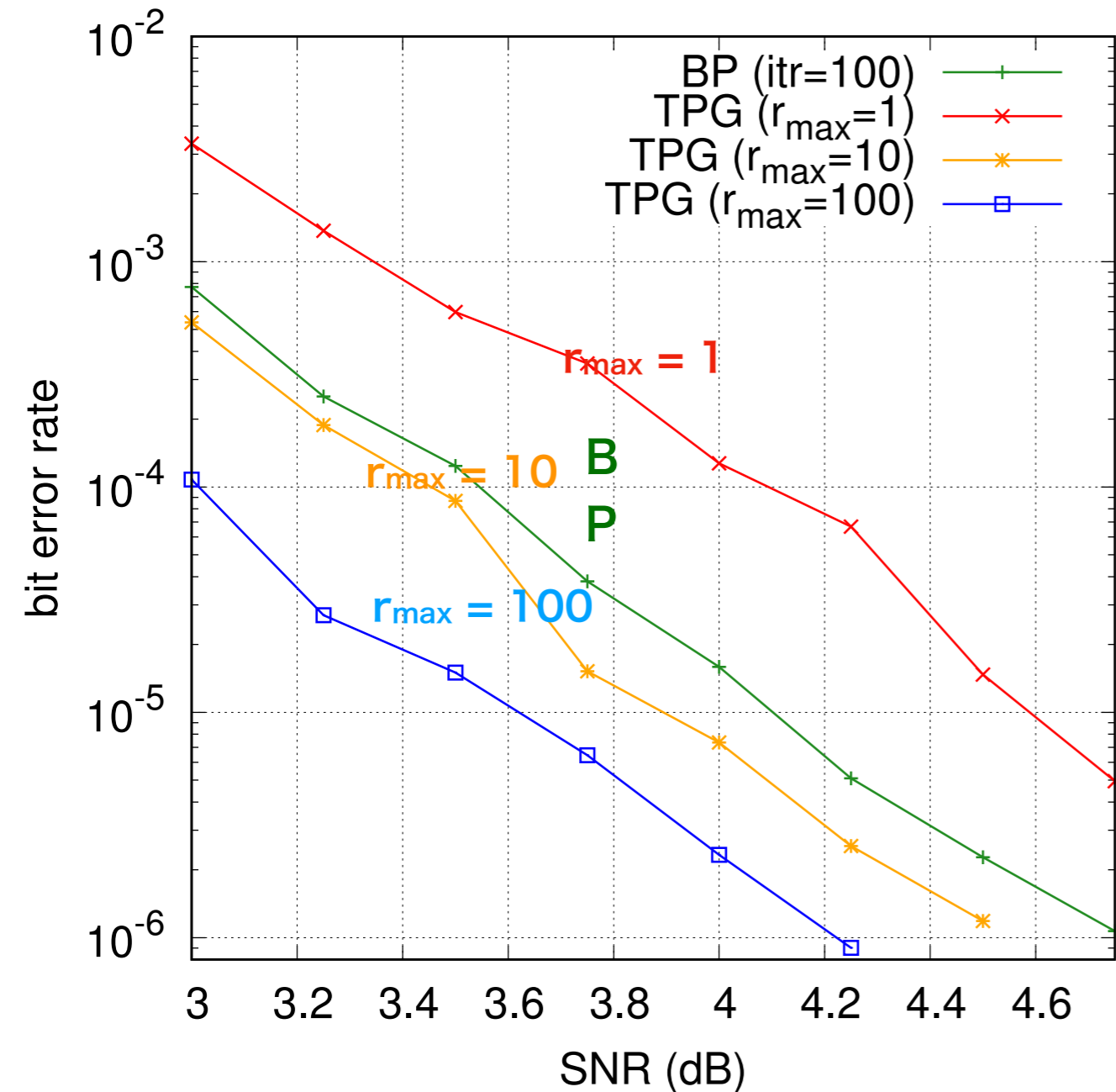


# 実験結果: 提案法のBER性能

## Rate 1/2 (3,6)-regular LDPC codes

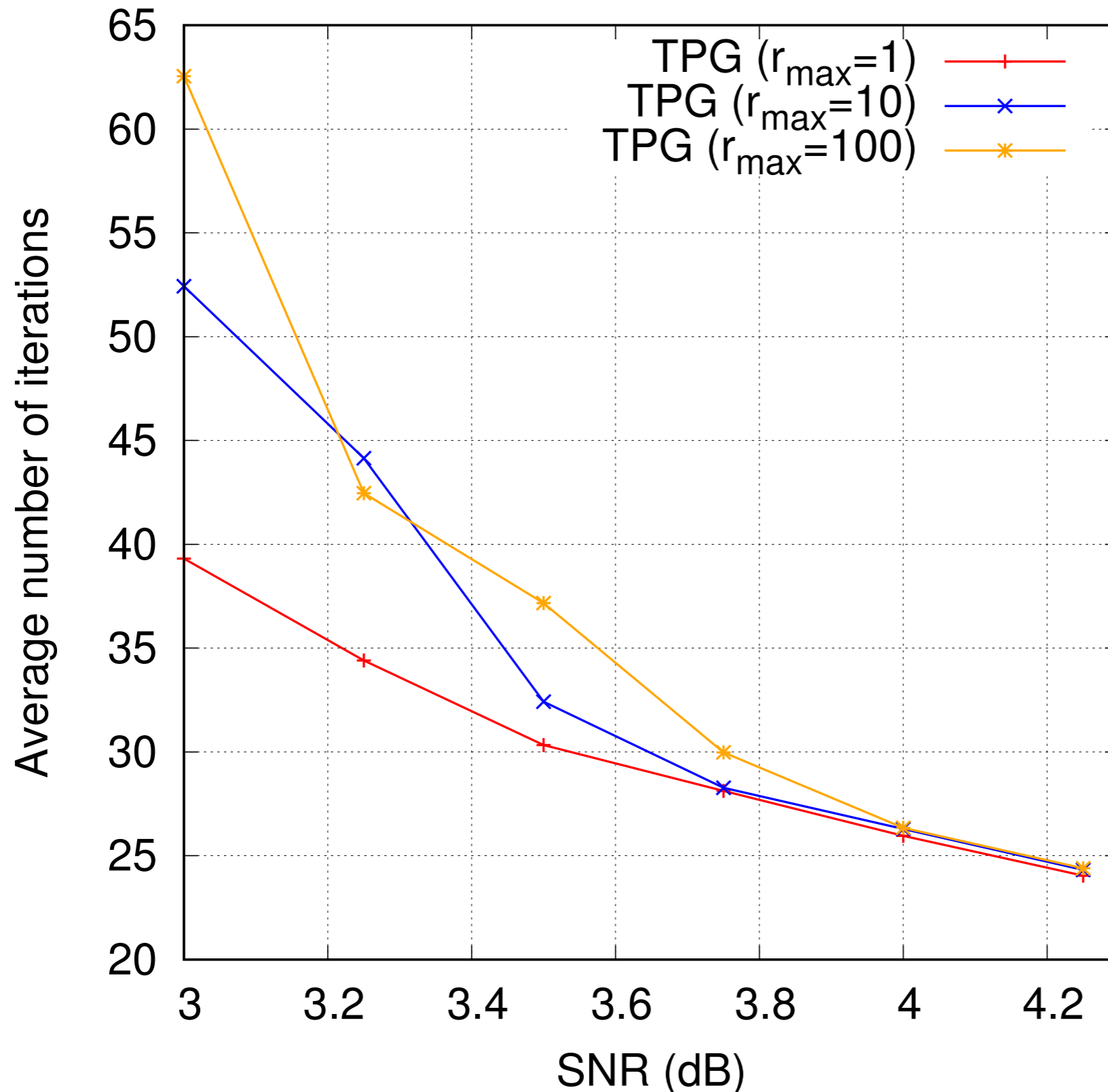
$n = 204, m = 102, t_{\max} = 100$

$n = 504, m = 252, t_{\max} = 100$



# 実験結果: 平均反復回数

$n = 204, m = 102, t_{\max} = 100$



提案法の時間計算量(1反復)

$$O(m2^r)$$

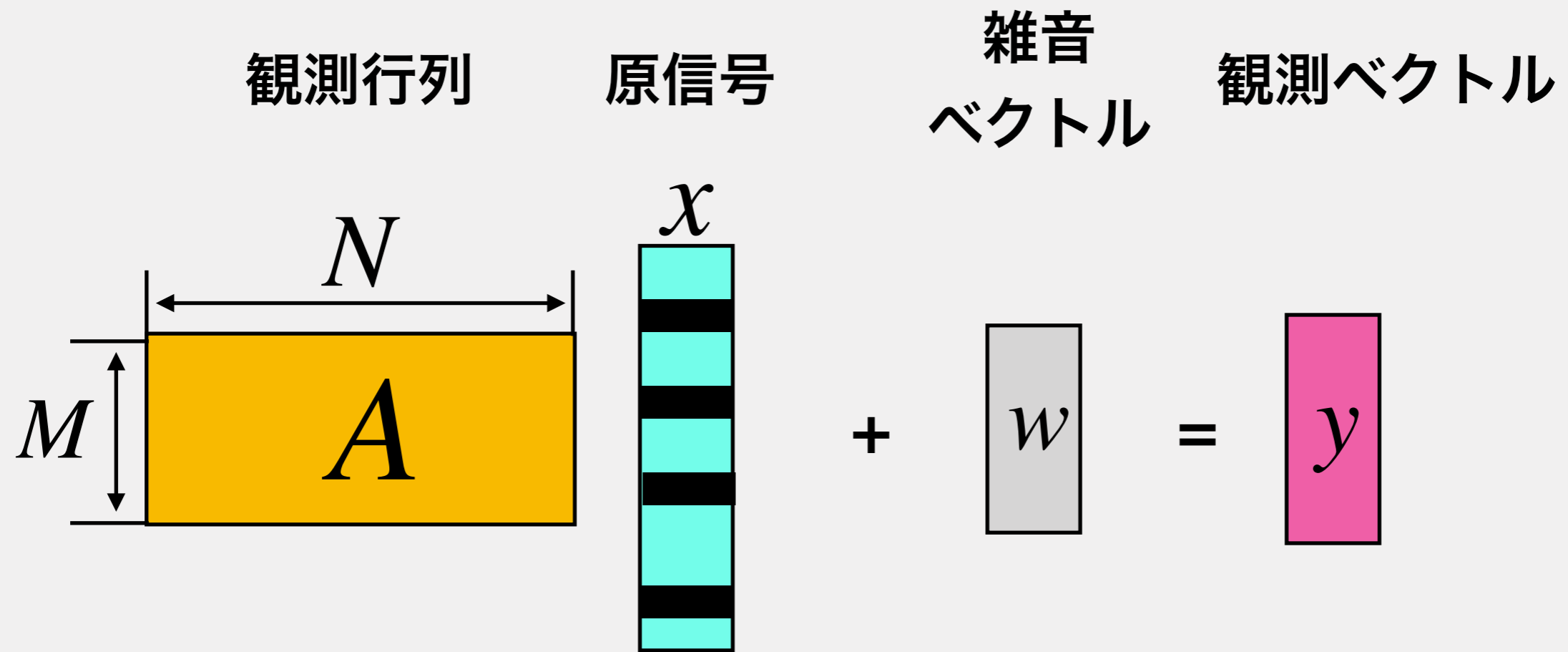
BPの時間計算量(1反復)

$$O(mr)$$

# 学習可能射影勾配復号法

- ✓ 深層展開がうまく動かすために必須の要素
- ✓ 基本多面体に基づくペナルティ関数の利用
- ✓ 射影勾配型復号法
  - ✓ ランダムリスタートの効用は大きい: 目的関数全体が非凸関数であるため、望ましくない停留点で探索点がトラップされる
  - ✓ BPよりは計算量的に少し重いがほとほとどの計算量
  - ✓ BPを越す復号性能を発揮
  - ✓ ニューラル型ハードウェアに好適

# 圧縮センシング



観測ベクトルを見て，疎ベクトルを可能な限り正確に推定したい

「未知変数の数  $>$  方程式の本数」 となる**劣決定系問題**

# 無線物理層における圧縮センシング

- ✓ 通信路推定（スパースなマルチパス）
- ✓ 到来波方向推定
- ✓ スペクトルセンシング
- ✓ NOMA

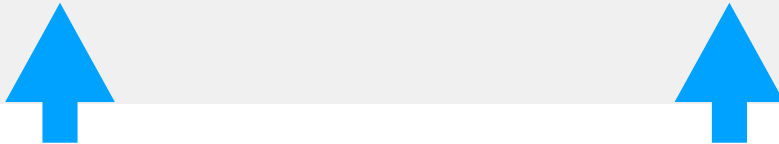
A User's Guide to Compressed Sensing for Communications Systems

K.HAYASHI, M. NAGAHARA and T. TANAKA

IEICE TRANS. COMMUN., VOL.E96-B, NO.3 MARCH 2013

# LASSO定式化

## LASSO

$$\hat{x} = \arg \min_{x \in \mathbb{R}^N} \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1$$


### 二次項

- (1)  $y$  と  $Ax$  をなるべく近く
- (2) 微分可能

### L1-正規化項

- (1) スパースベクトルを偏好
- (2) 微分不可能 (原点にて)

部分不能な正規化項が入っているため、単純な勾配法はうまく動かない

# 近接 (Proximal) 演算子

## 近接演算子

$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\text{Prox}_{\gamma f}(x) := \arg \min_{u \in \mathbb{R}^n} f(u) + \frac{1}{2\gamma} \|x - u\|_2^2$$

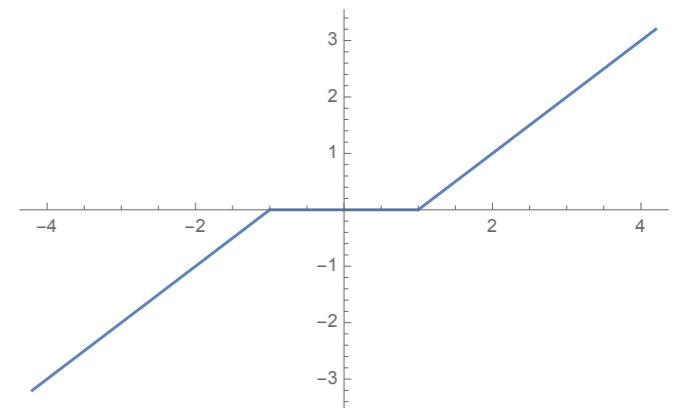
- (1) 射影演算子に似ている
- (2) 微分不能関数を滑らかに

## 例: L1 正規化項に対応する近接演算子

$$f(x) := |x| \quad (f: \mathbb{R} \rightarrow \mathbb{R})$$

$$\text{Prox}_{\gamma f}(x) := \text{sign}(x) \max\{|x| - \gamma, 0\} =: \eta(x; \gamma)$$

i.e., ソフトしきい値関数



# 近接勾配法

ゴール

$$\text{minimize } f(x) + g(x) \text{ s.t. } x \in \mathbb{R}^n$$

$f(x)$  : 微分可能

$g(x)$  : 簡潔な近接演算子を持つ

近接勾配法 (proximal gradient method)

$$x^{n+1} := \text{prox}_{\gamma g}(x^n - \gamma \nabla f(x^n))$$

いくつかの条件が満足されるならば(例えば目的関数の凸性)、  
近接勾配法において探索点は目的関数の最小点に収束する



# ISTA: LASSOに対する近接勾配法

## LASSO

$$\hat{x} = \arg \min_{x \in \mathbb{R}^N} \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1$$

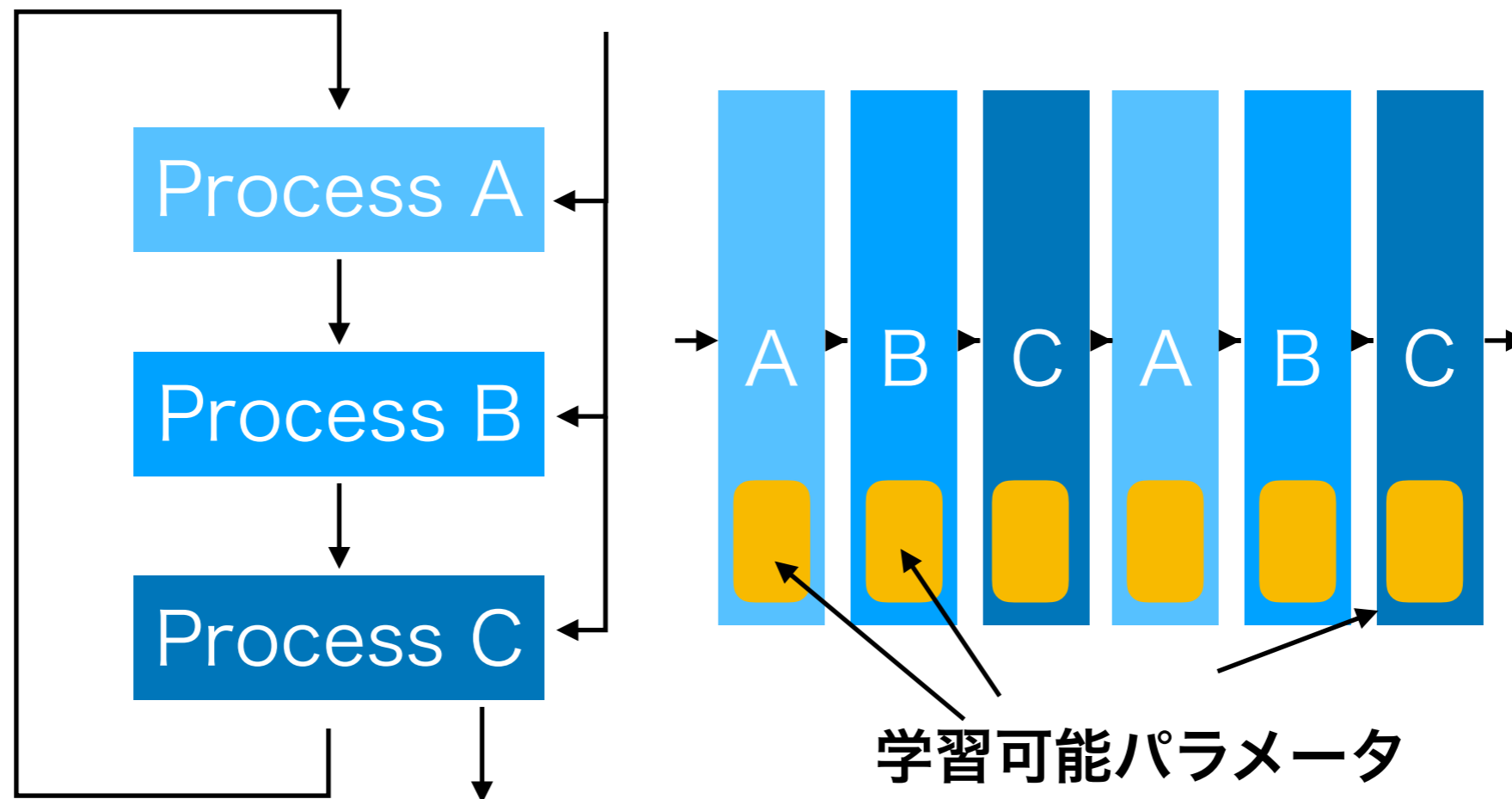
## ISTA (Daubechies et.al, 2004)

$$r_t = s_t + \beta A^T (y - As_t) \quad : \text{勾配ステップ}$$

$$s_{t+1} := \eta(r_t; \tau) \quad : \text{近接写像ステップ}$$

- (1) シンプルだが収束が遅い
- (2) ハイパーパラメータの調整がととても大変

# 「深層展開」の誕生



K. Gregor, and Y. LeCun,

“Learning fast approximations of sparse coding,”

Proc. 27th Int. Conf. Machine Learning, pp. 399--406, 2010.

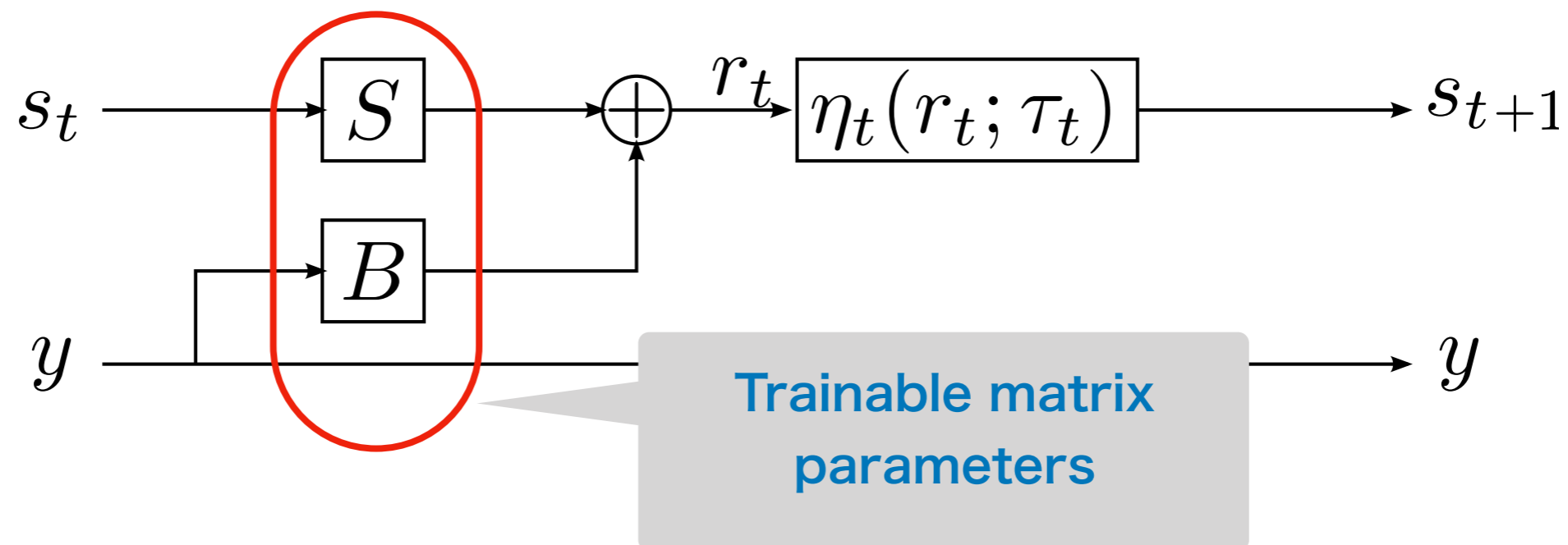
# LISTAの構成

Original ISTA:  $r_t = s_t + \beta A^T (y - As_t)$   
 $s_{t+1} := \eta(r_t; \tau)$

LISTA (Gregor and LeCun)

$$r_t = Bs_t + Sy$$

$$s_{t+1} = \eta(r_t; \tau_t)$$



# 深層展開における収束加速

- ✓ 単純な2変数2次関数を考える
- ✓ 初期値をランダム値とする勾配法

$$f(x_1, x_2) = x_1^2 + qx_2^2$$

## 勾配法

$$\mathbf{s}_{t+1} = \mathbf{s}_t - \gamma \nabla f(\mathbf{s}_t)$$

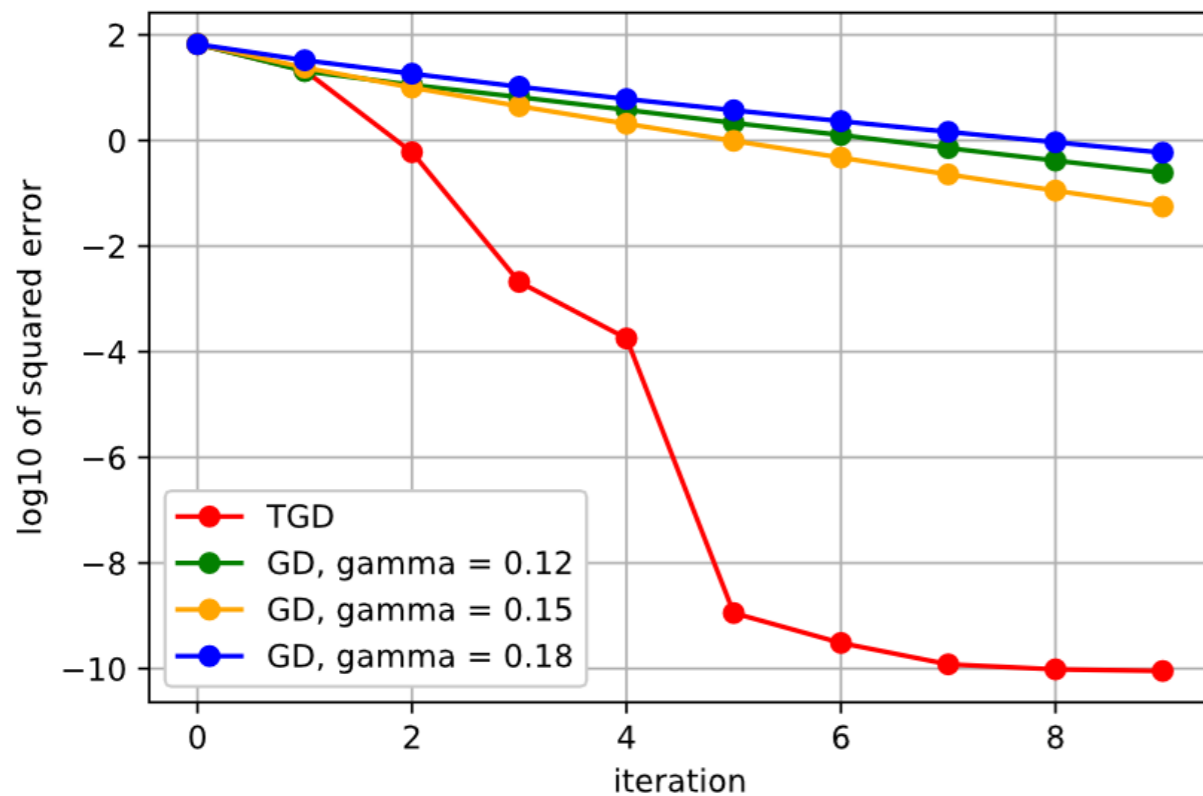
## 学習可能-勾配法

$$\mathbf{s}_{t+1} = \mathbf{s}_t - \underline{\gamma}_t \nabla f(\mathbf{s}_t).$$

# 二次関数の最小化

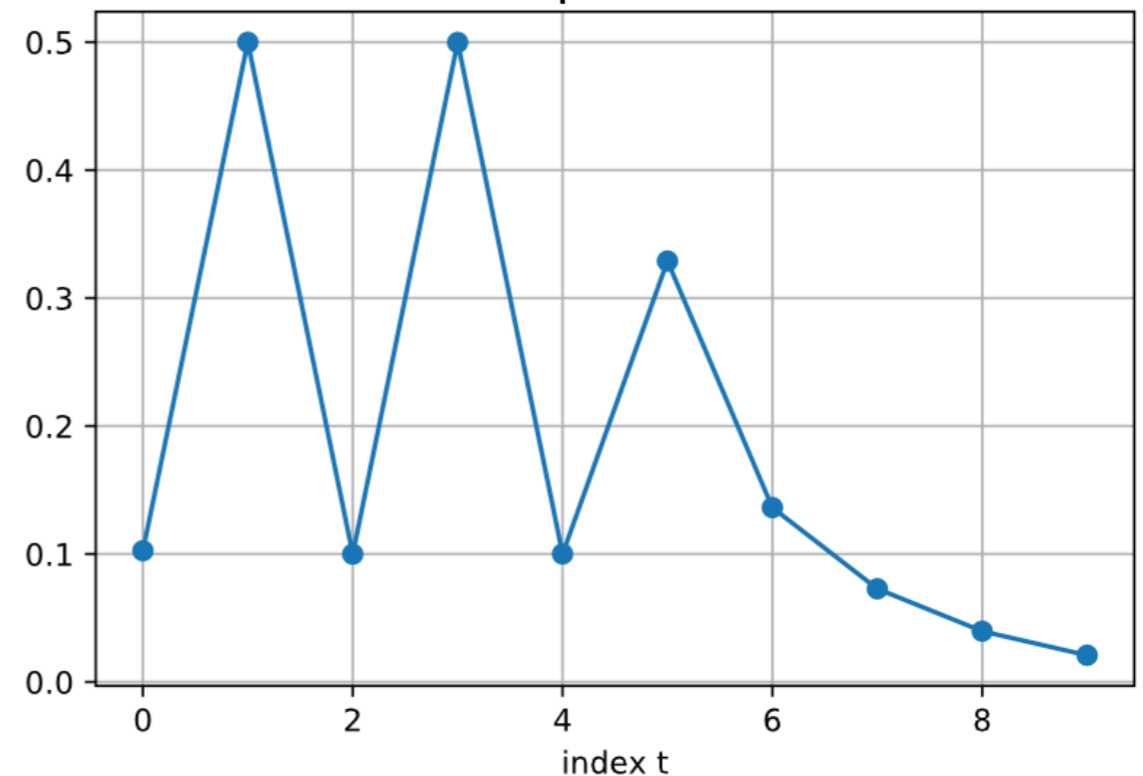
Squared errors v.s. iterations

$q = 5$



Values of  $\gamma_t$

$q = 5$

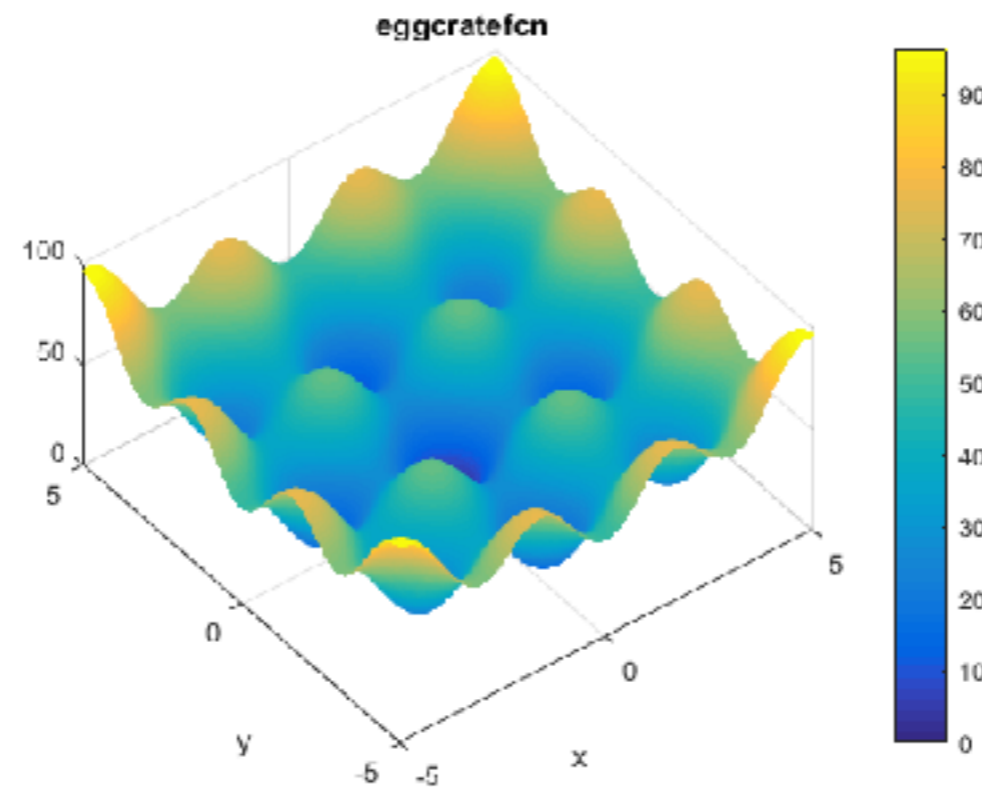


- ✓ 学習可能勾配法は、通常の勾配法に比べて圧倒的に収束が速い
- ✓ 反復インデックス依存のステップサイズパラメータによる恩恵は大きい

# 多峰性関数の場合

卵入れ関数 (egg crate function)

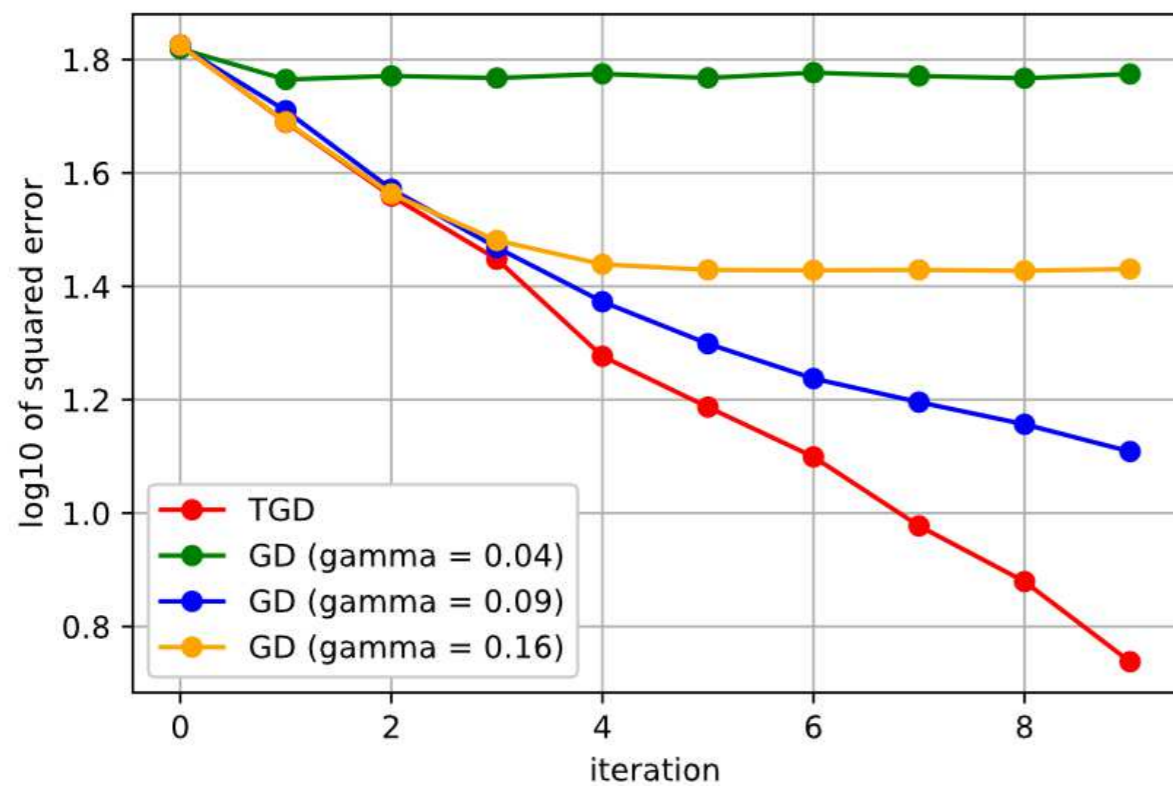
$$f(x_1, x_2) = x_1^2 + x_2^2 + 25(\sin^2(x_1) + \sin^2(x_2))$$



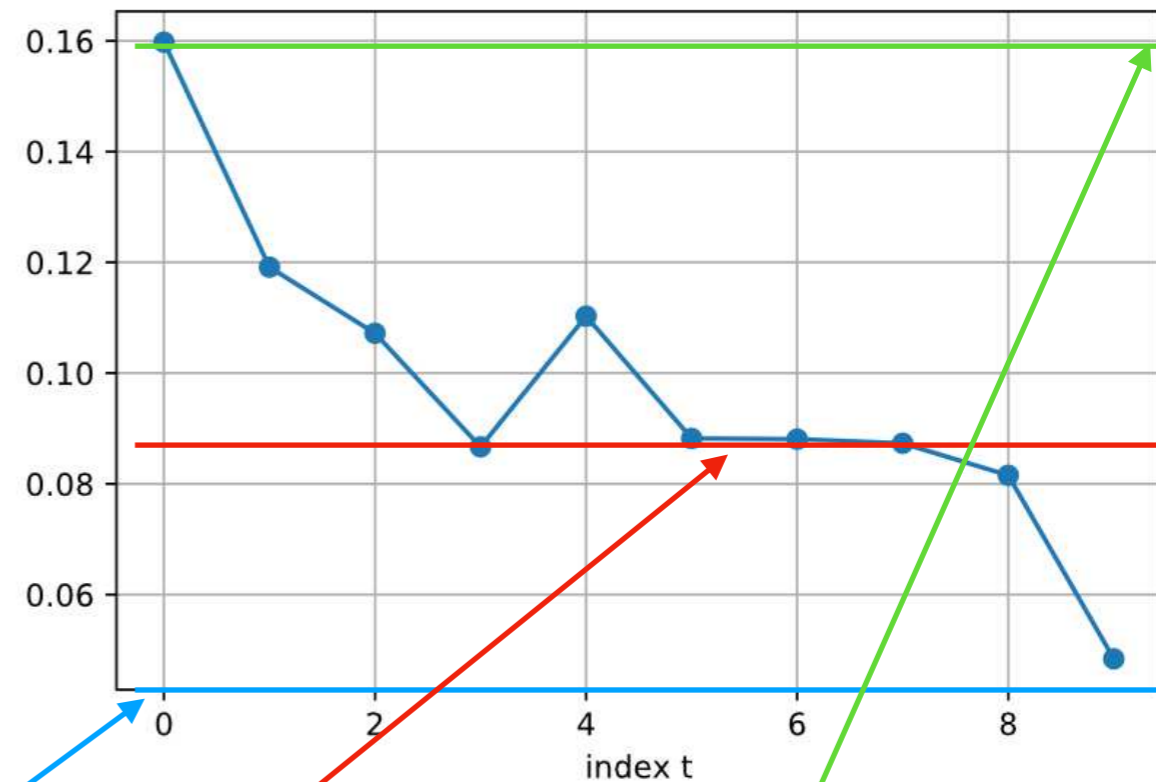
cited from: <http://benchmarkfcns.xyz/benchmarkfcns/eggcratefcn.html>

# 多峰性関数の場合

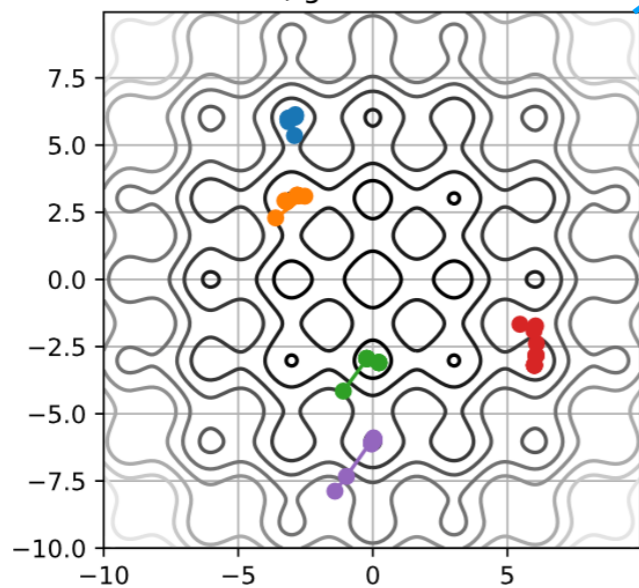
## 真値からの誤差



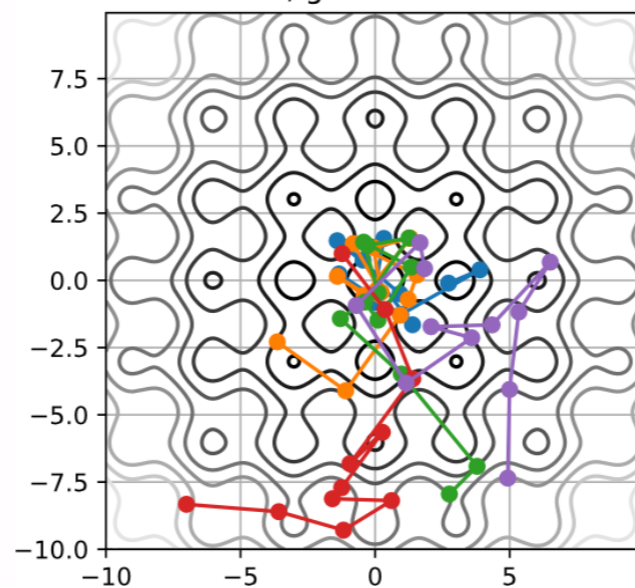
## 学習可能パラメータの値



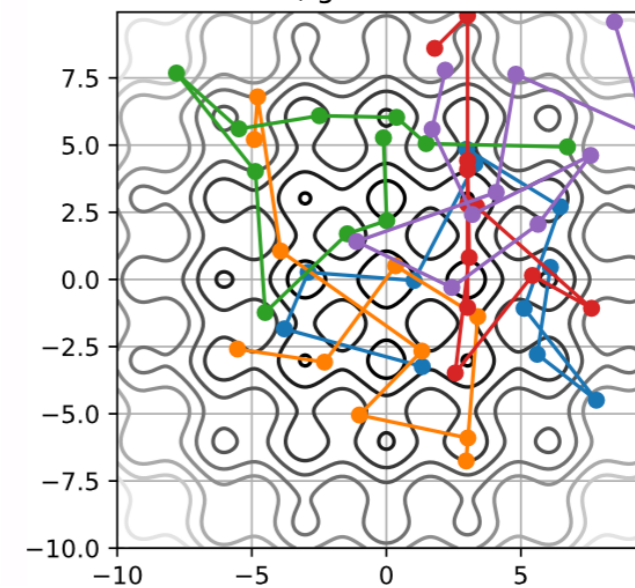
### GD, gamma=0.04



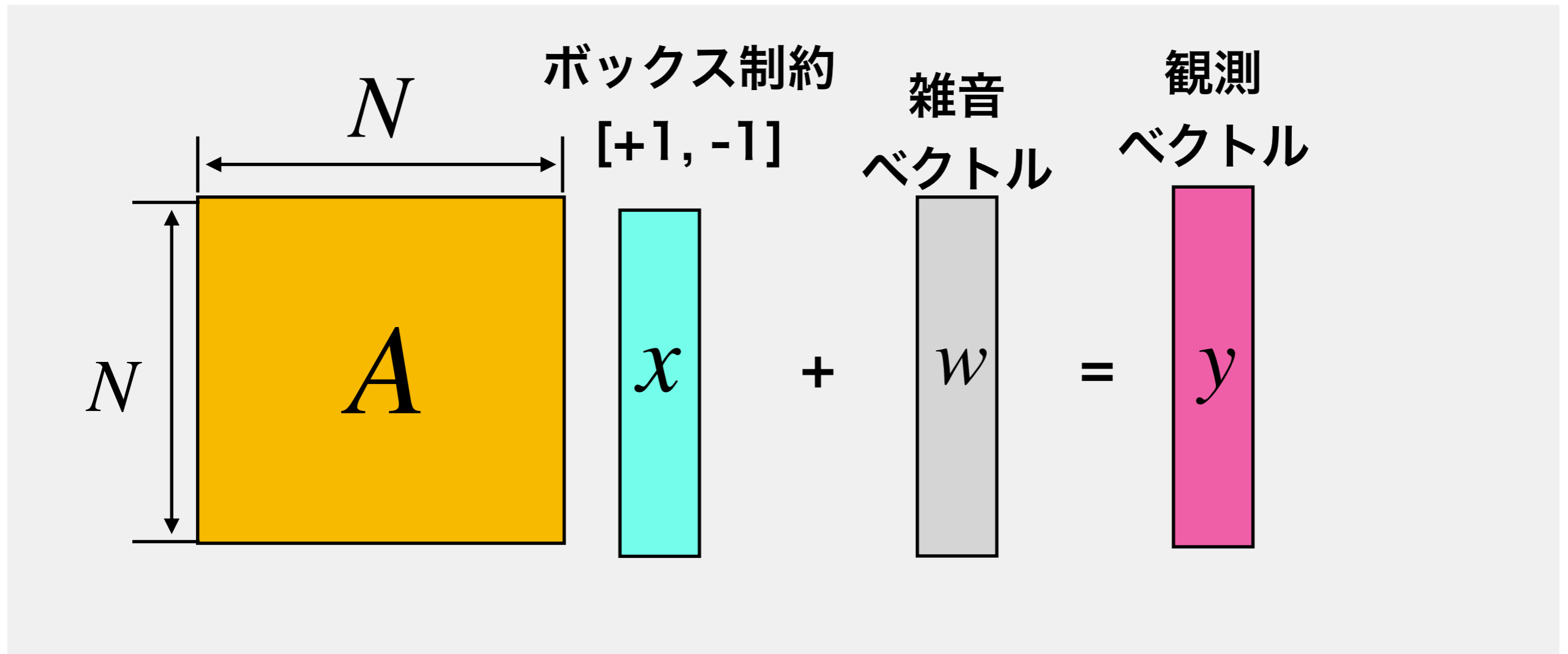
### GD, gamma=0.09



### GD, gamma=0.16



# 深層展開: 射影勾配法



$$\min_x \|y - Ax\|_2^2 \text{ subject to } x \in [-1, 1]^n$$



# 射影勾配法

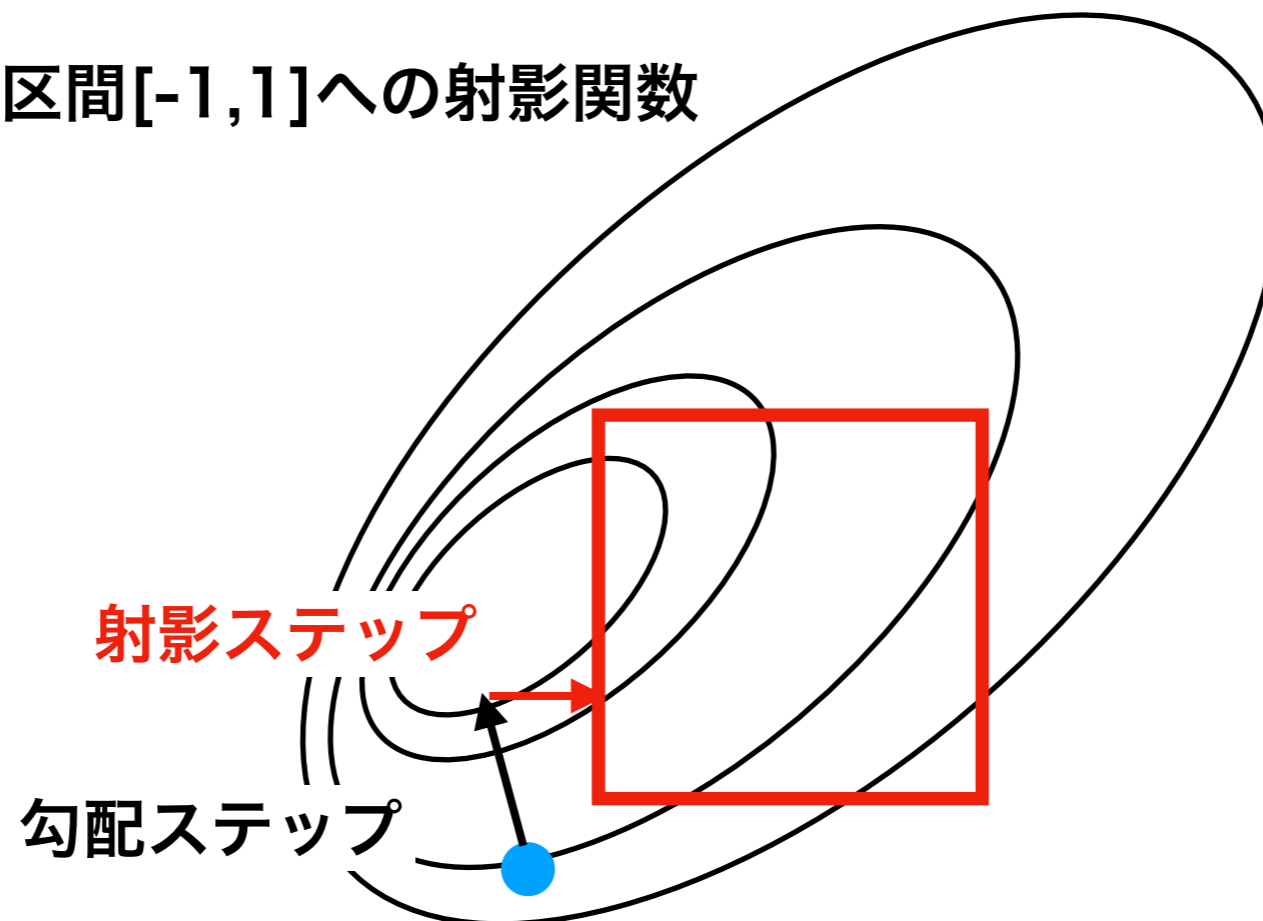
勾配ステップ

$$\mathbf{r}_t = \mathbf{s}_t + \gamma \mathbf{A}^T (\mathbf{y} - \mathbf{A} \mathbf{s}_t),$$

射影ステップ

$$\mathbf{s}_{t+1} = \varphi(\xi \mathbf{r}_t) \quad \square$$

$\varphi$  は閉区間 $[-1, 1]$ への射影関数



# 実験：射影勾配法

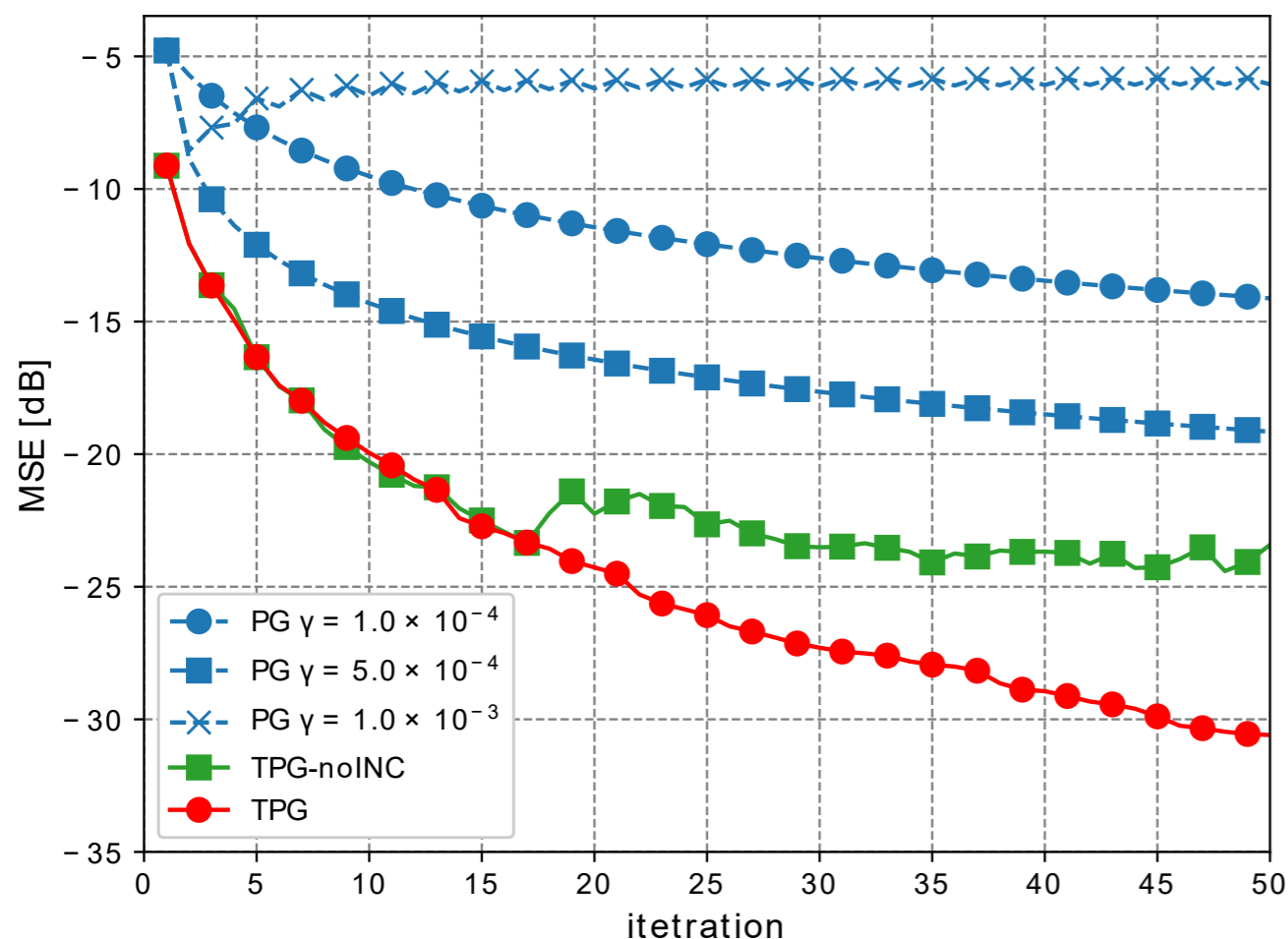
勾配ステップ

$$\mathbf{r}_t = \mathbf{s}_t + \underline{\gamma}_t \mathbf{A}^T (\mathbf{y} - \mathbf{A} \mathbf{s}_t),$$

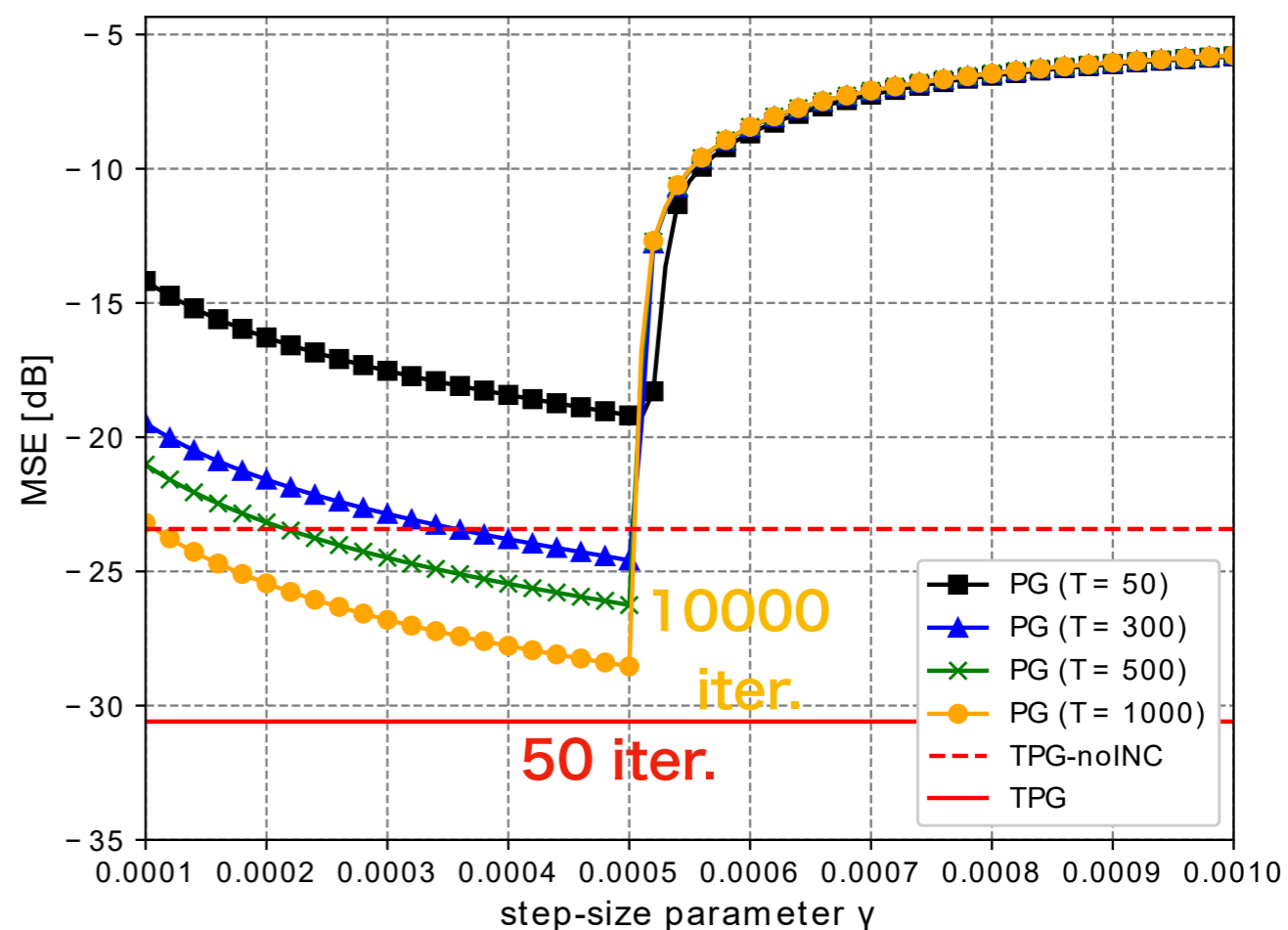
射影ステップ

$$\mathbf{s}_{t+1} = \varphi(\xi \mathbf{r}_t),$$

平均二乗誤差 v.s. 反復



平均二乗誤差 v.s.  $\gamma$  の値



$n = 1000$

# 観察

- ✓ 訓練プロセスにより適切なステップサイズパラメータが選択されている
- ✓ 反復ごとに独立したステップサイズパラメータが本質的に重要
- ✓ 訓練プロセスにより学習された結果は一種の「最小化のための戦略」と見ることができる
- ✓ 学習された戦略は必ずしも自明ではない

# TISTA (Ito, Takabe, W, 2019)

D. Ito, S. Takabe, and T. Wadayama, "Trainable ISTA for sparse signal recovery," IEEE Trans. Signal Processing, vol. 67, no. 12, pp. 3113-3125, Jun., 2019.

$$\begin{aligned}r_t &= s_t + \gamma_t W(y - As_t), \\s_{t+1} &= \eta_{MMSE}(r_t; \tau_t^2), \\v_t^2 &= \max \left\{ \frac{\|y - As_t\|_2^2 - M\sigma^2}{\text{trace}(A^T A)}, \epsilon \right\}, \\\tau_t^2 &= \frac{v_t^2}{N} (N + (\gamma_t^2 - 2\gamma_t)M) + \frac{\gamma_t^2 \sigma^2}{N} \text{trace}(WW^T)\end{aligned}$$

**$W$  は  $A$  のムーア・ペンローズ擬似逆行列**

# TISTA (Ito, Takabe, W, 2019)

## 訓練可能パラメータ

$$r_t = s_t + \gamma_t W(y - As_t),$$
$$s_{t+1} = \eta_{MMSE}(r_t; \tau_t^2),$$

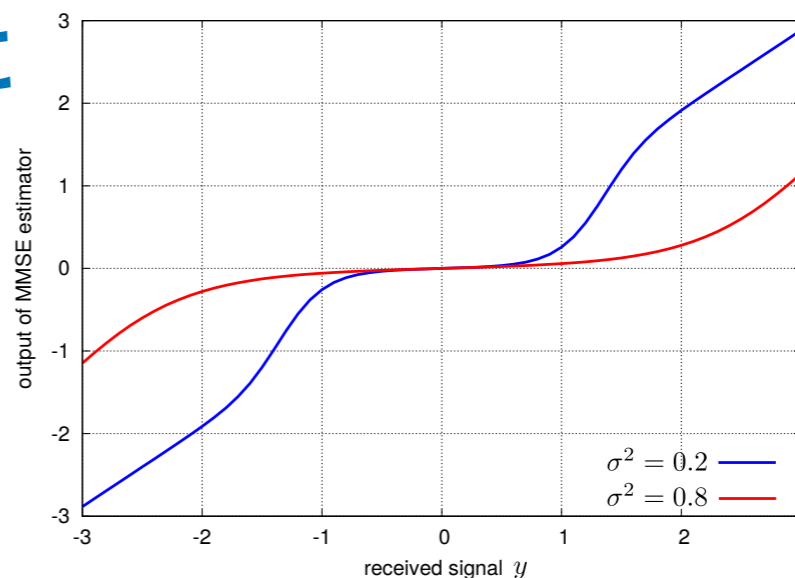
近接勾配法

$$v_t^2 = \max \left\{ \frac{\|y - As_t\|_2^2 - M\sigma^2}{\text{trace}(A^T A)}, \epsilon \right\},$$

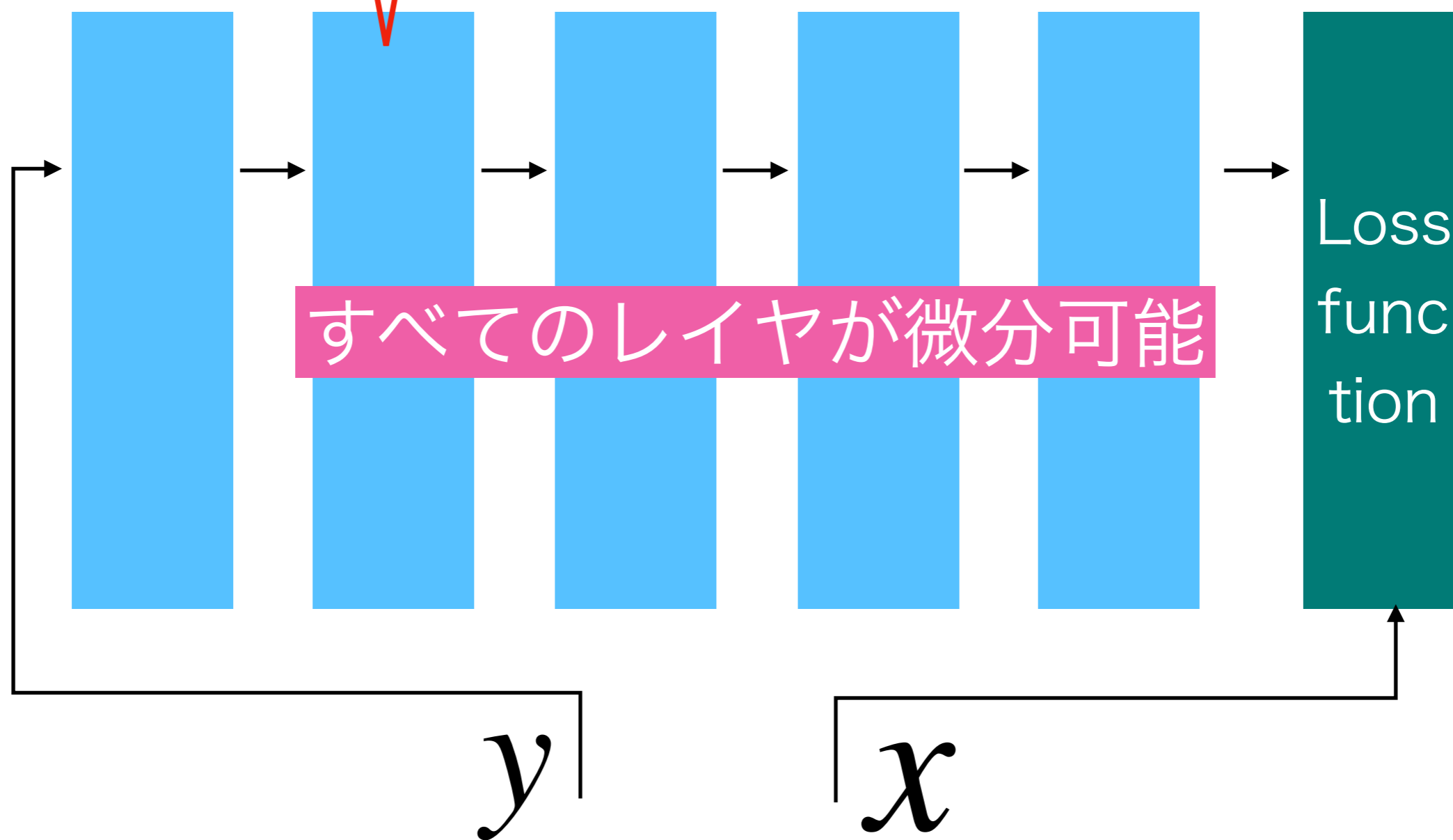
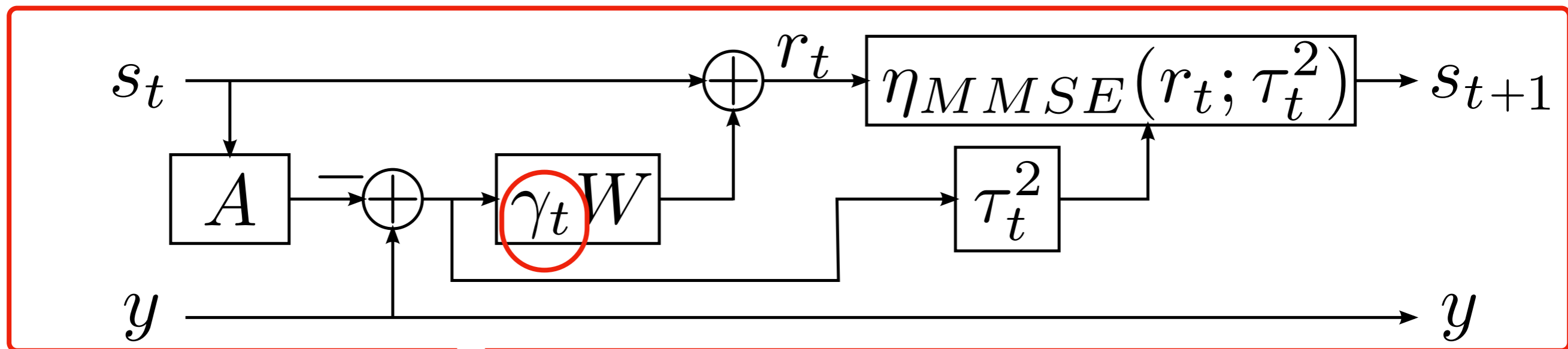
$$\tau_t^2 = \frac{v_t^2}{N} (N + (\gamma_t^2 - 2\gamma_t)M) + \frac{\gamma_t^2 \sigma^2}{N} \text{trace}(WW^T)$$

誤差分散推定

MMSE推定関数に  
基づく近接写像



# TISTAのブロックダイアグラム



# 比較：学習可能パラメータ数

	TISTA	LISTA	LAMP
# of params	$T$	$T(N^2 + MN + 1)$	$T(NM + 2)$

# TISTAの復元性能 (10%が非ゼロ元となる状況)

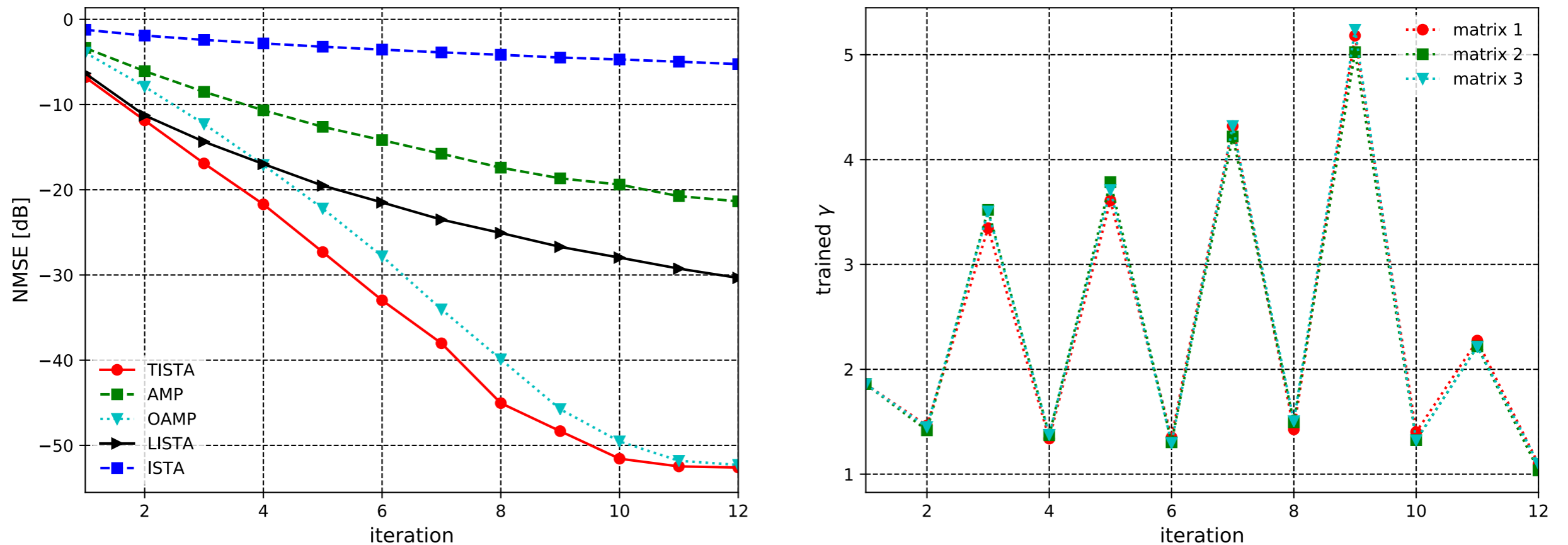


Figure 7: Sparse signal recovery for  $(n, m) = (300, 150)$ ,  $p = 0.1$  and  $\text{SNR} = 50$  dB. (Left) MSE performances as a function of iteration steps. (Right) Trained values of  $\gamma_t$  under three different measurement matrices. The initial value is set to 1.0.



# TISTA vs OAMP (1)

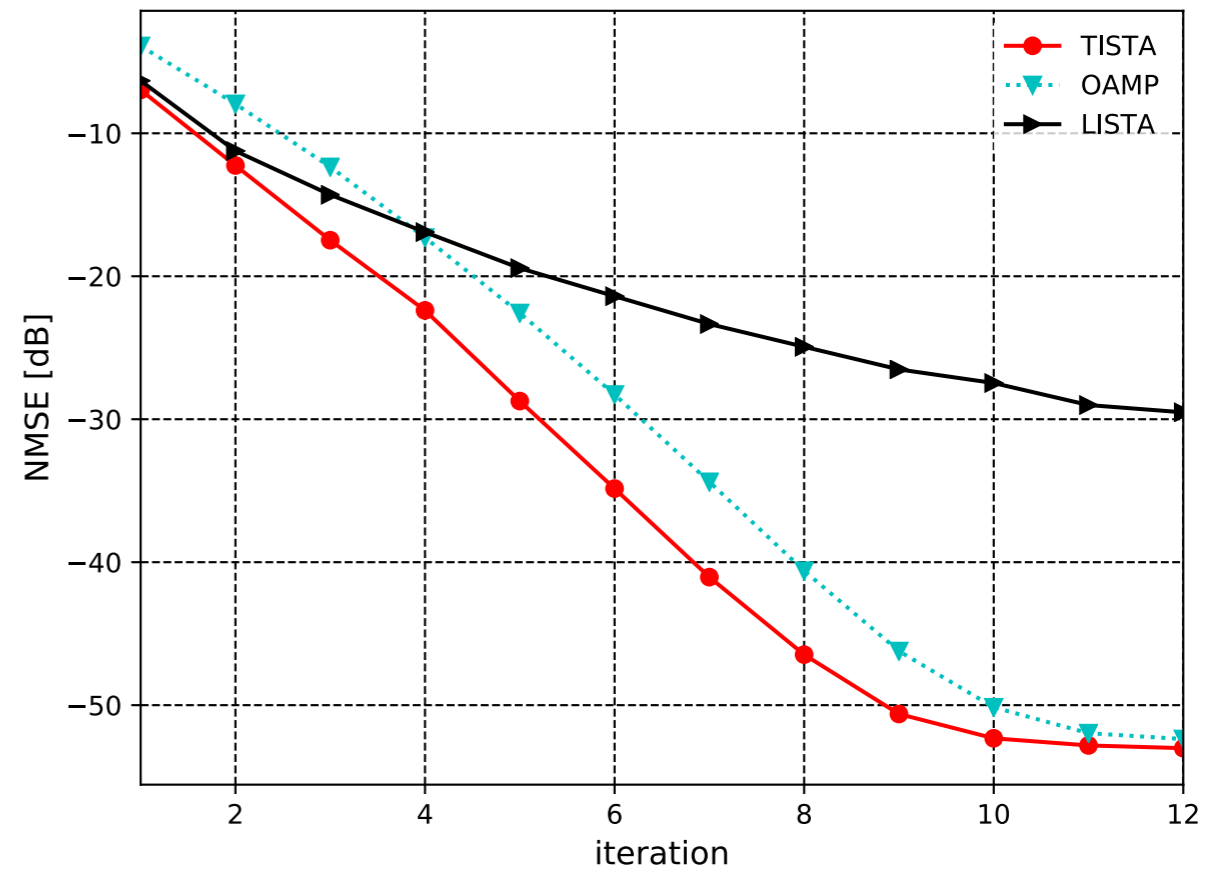
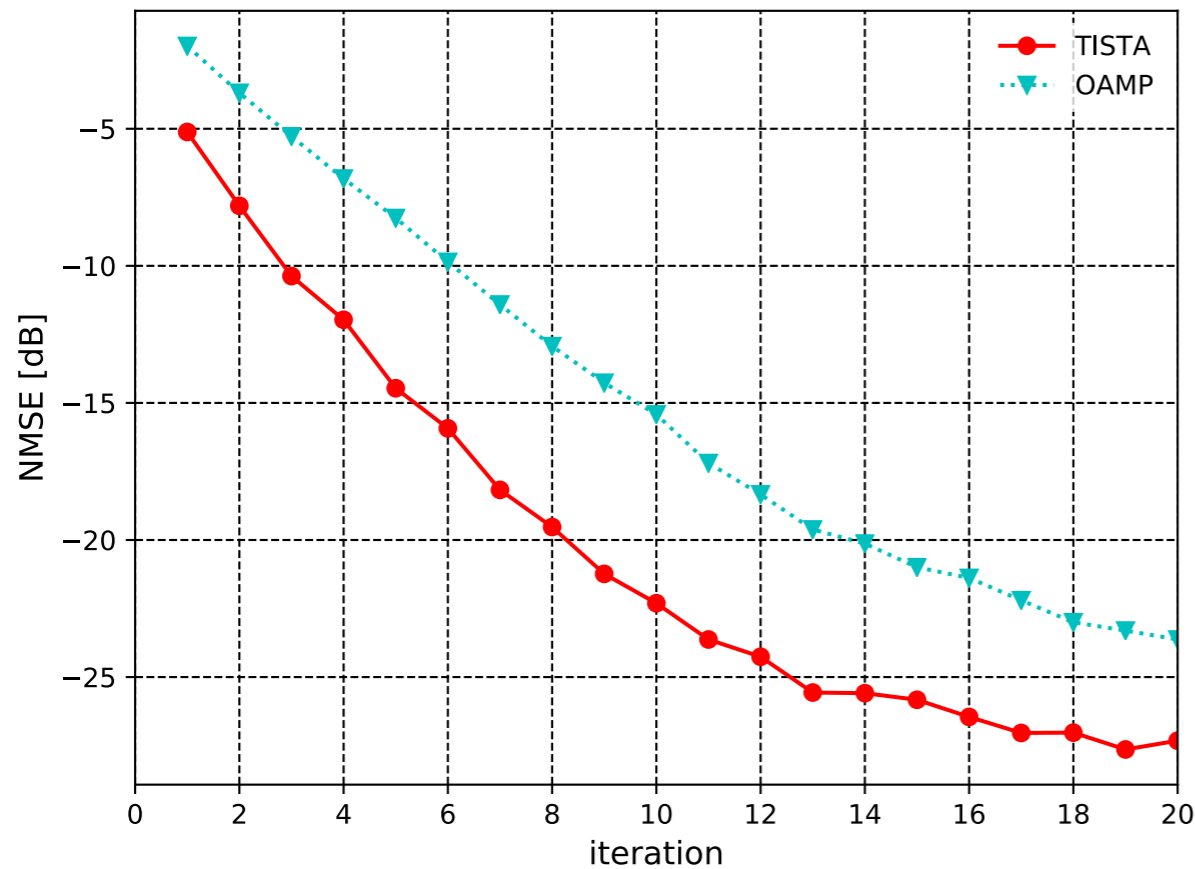


Figure 8: (Left) MSE performance of OAMP and TISTA for  $(n, m) = (300, 150)$ ,  $p = 0.18$  and SNR= 30 dB. (Right) MSE performances for  $(n, m) = (300, 150)$ ,  $p = 0.1$  and SNR= 30 dB with binary sensing matrix  $\mathbf{A} \in \{1, -1\}^{m \times n}$  as a function of iteration steps.

# TISTA vs OAMP (2)

MNISTデータ  
セットに対する  
スパース信号再現

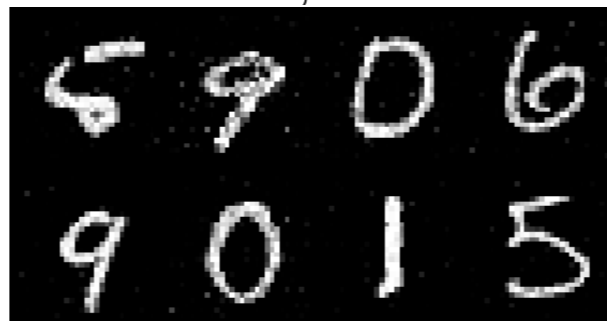
TISTA  $t = 5$ , MSE=0.0258



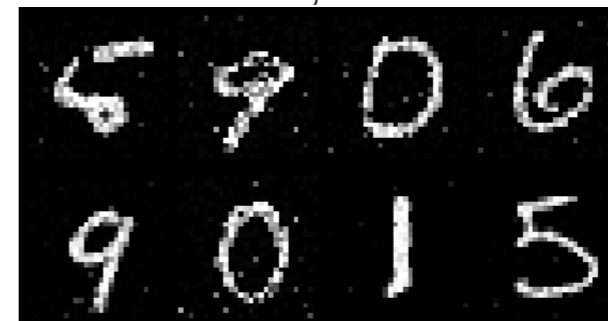
OAMP  $t = 5$ , MSE=0.0335



TISTA  $t = 10$ , MSE=0.0065

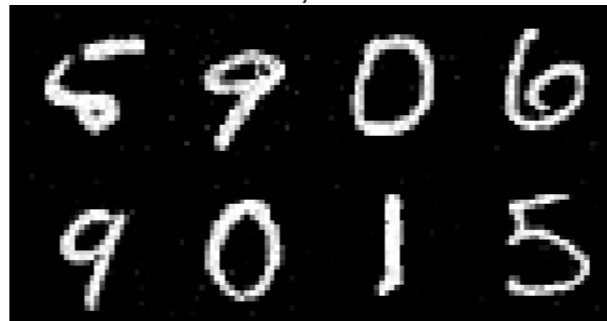


OAMP  $t = 10$ , MSE=0.0165

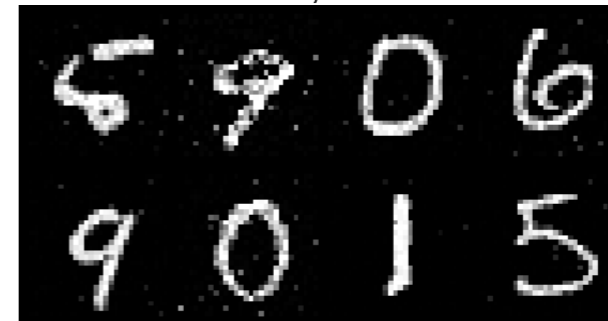


人工データだけではなく  
実データでも良好な特性

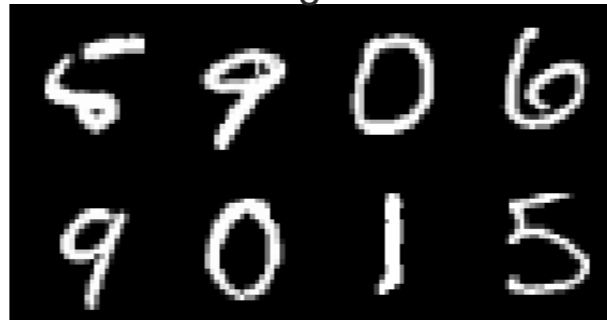
TISTA  $t = 20$ , MSE=0.0014



OAMP  $t = 20$ , MSE=0.0089



Original



$t=20$

TISTA



OAMP

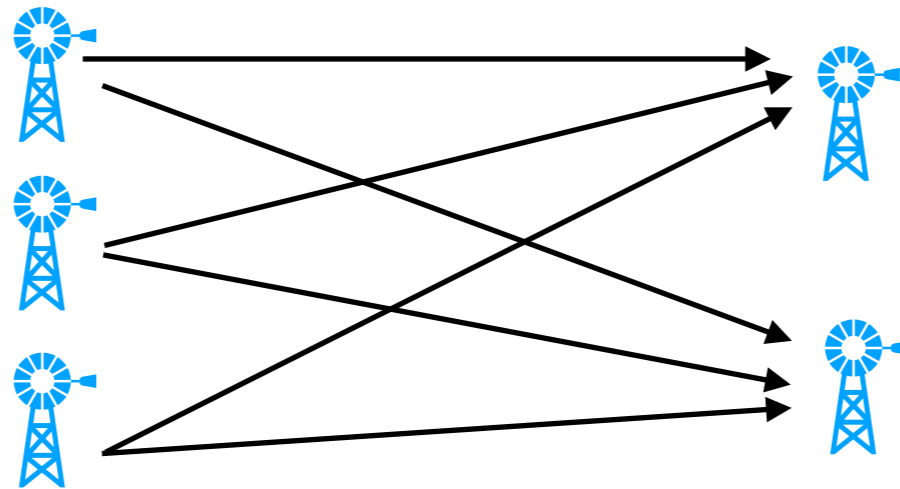


# TISTAのコード

<https://github.com/wadayama/TISTA>

# 過負荷MIMO検出問題

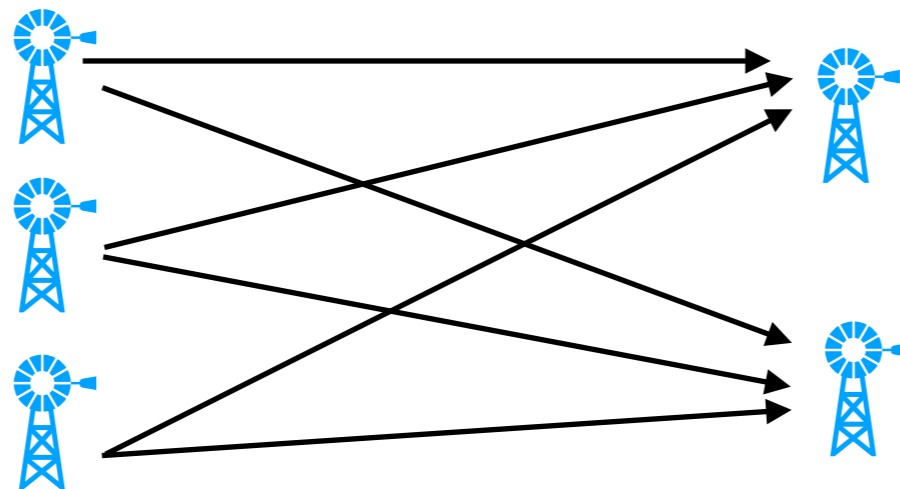
送信アンテナ数 > 受信アンテナ数



- ✓ 基地局側は多くのアンテナを持つ余裕があるが、  
端末側はコスト・消費電力などの点で、そんなに多くの  
アンテナを持ってない
- ✓ mmWave MIMOなどでは、アンテナの小型化・  
高集積化が進む

# 過負荷MIMO検出問題

送信アンテナ数 > 受信アンテナ数



観測行列 **2値ベクトル** ノイズ 観測ベクトル

$$\begin{matrix} & \xrightarrow{N} \\ M \updownarrow & \boxed{H} \\ & \downarrow \end{matrix} \quad \begin{matrix} \color{cyan} \boxed{x} \\ \color{cyan} \end{matrix} + \begin{matrix} \color{gray} \boxed{w} \\ \color{gray} \end{matrix} = \begin{matrix} \color{magenta} \boxed{y} \\ \color{magenta} \end{matrix}$$

劣決定系問題

# MIMO通信路モデル

複素モデル  $\tilde{y} = \tilde{H}\tilde{x} + \tilde{w},$

実数モデル  $y = Hx + w,$

$$y \triangleq \begin{bmatrix} \text{Re}(\tilde{y}) \\ \text{Im}(\tilde{y}) \end{bmatrix} \in \mathbb{R}^M, \quad H \triangleq \begin{bmatrix} \text{Re}(\tilde{H}) & -\text{Im}(\tilde{H}) \\ \text{Im}(\tilde{H}) & \text{Re}(\tilde{H}) \end{bmatrix},$$
$$x \triangleq \begin{bmatrix} \text{Re}(\tilde{x}) \\ \text{Im}(\tilde{x}) \end{bmatrix} \in \mathbb{S}^N, \quad w \triangleq \begin{bmatrix} \text{Re}(\tilde{w}) \\ \text{Im}(\tilde{w}) \end{bmatrix} \in \mathbb{R}^M,$$

# 提案手法：TPG-検出器

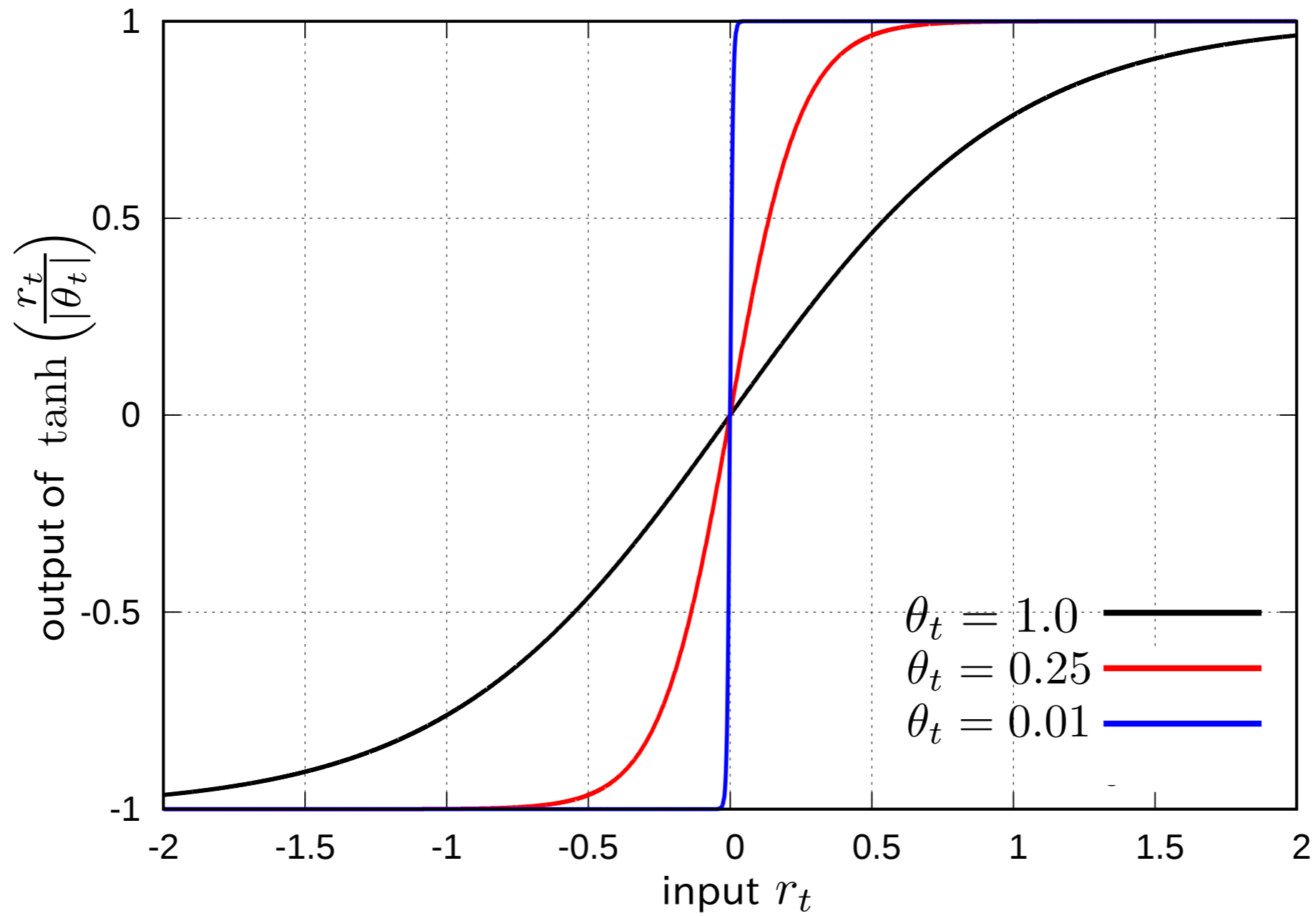
Satoshi Takabe, Masayuki Imanishi, Tadashi Wadayama, Ryo Hayakawa, Kazunori Hayashi, "Trainable Projected Gradient Detector for Massive Overloaded MIMO Channels: Data-driven Tuning Approach", IEEE Access, July 2019.

$$\mathbf{r}_t = \mathbf{s}_t + \gamma_t \mathbf{W} (\mathbf{y} - \mathbf{H} \mathbf{s}_t),$$

$$\mathbf{s}_{t+1} = \tanh \left( \frac{\mathbf{r}_t}{|\boldsymbol{\theta}_t|} \right),$$

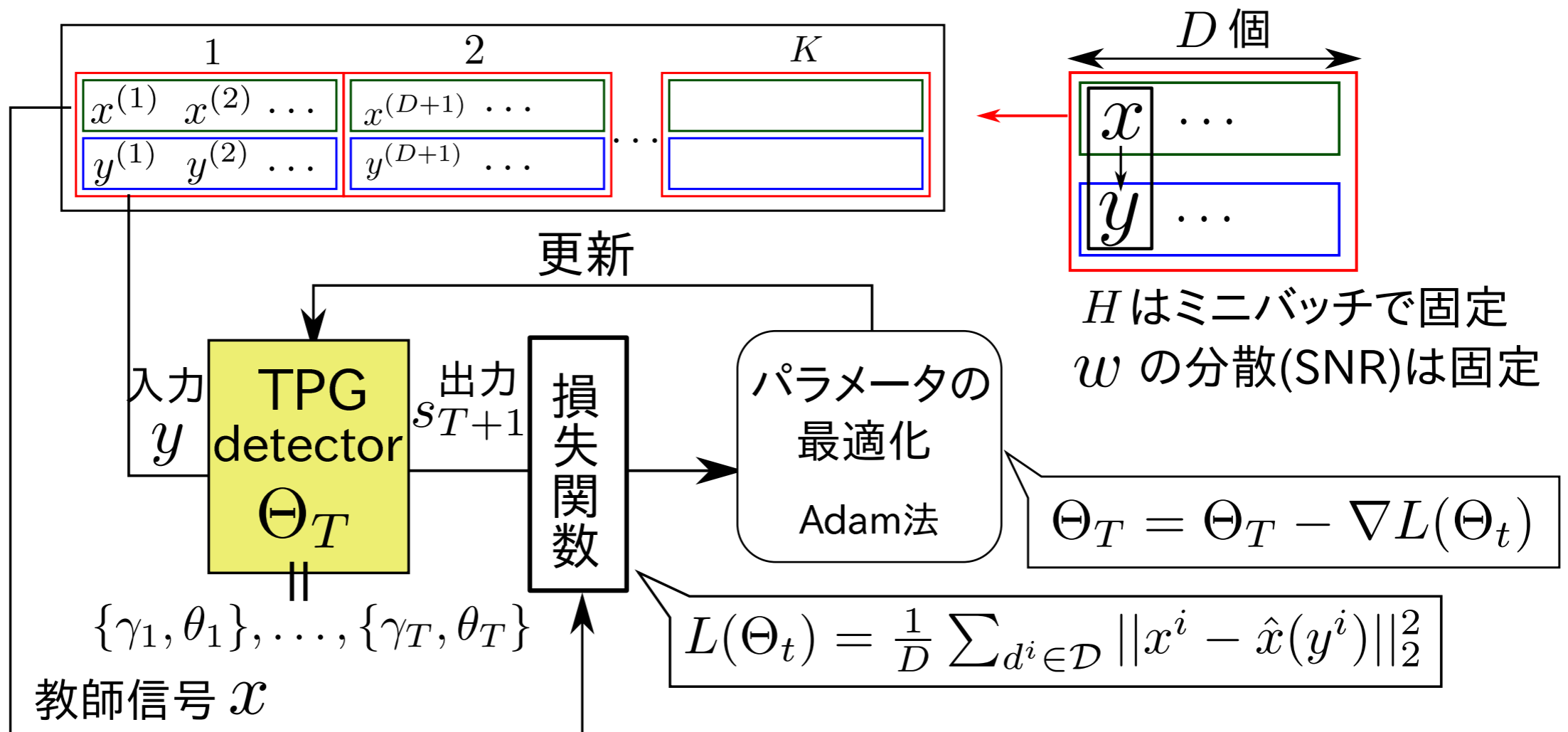
$$\mathbf{W} \triangleq \mathbf{H}^T (\mathbf{H} \mathbf{H}^T + \alpha \mathbf{I})^{-1}$$

# 軟射影関数

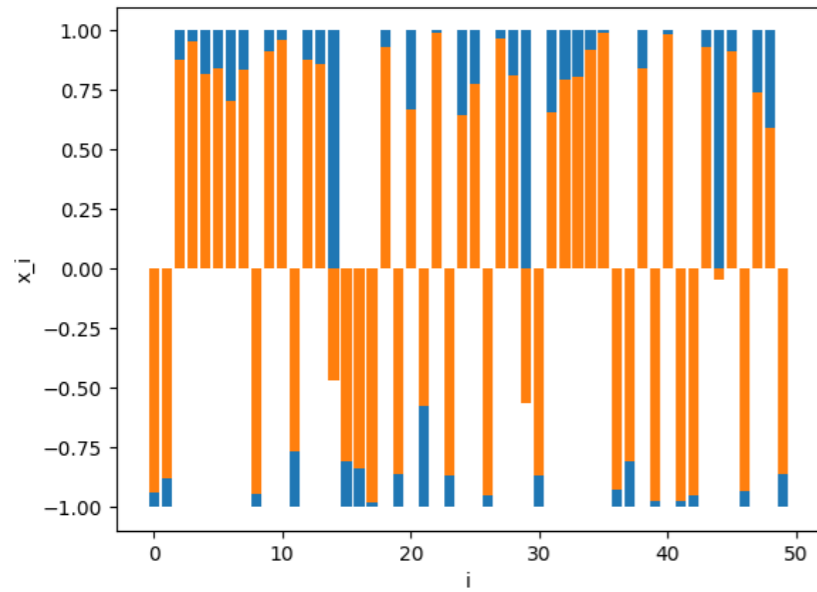




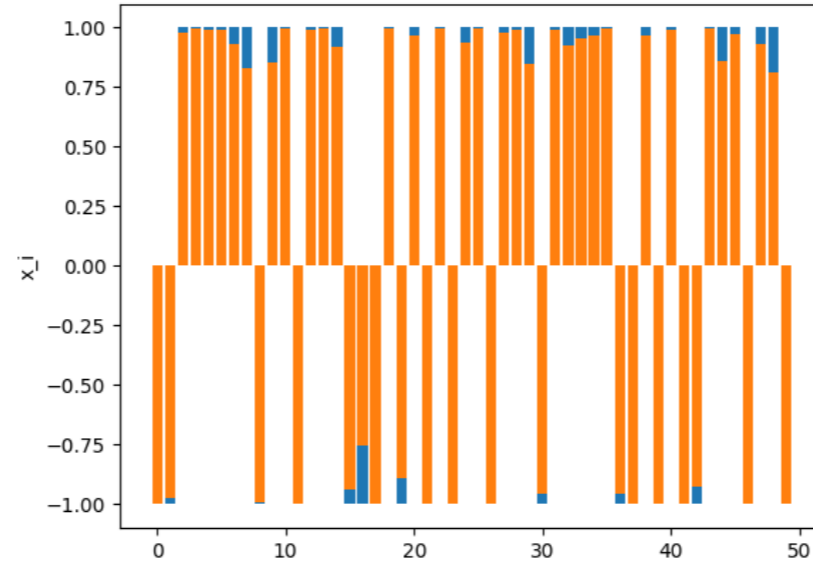
# 学習の過程



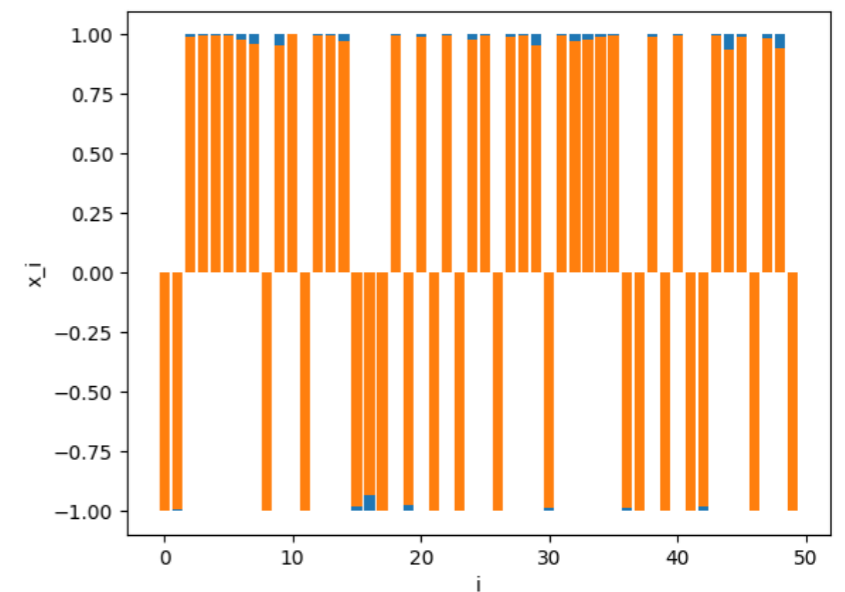
# 推定の様子



**t=1**

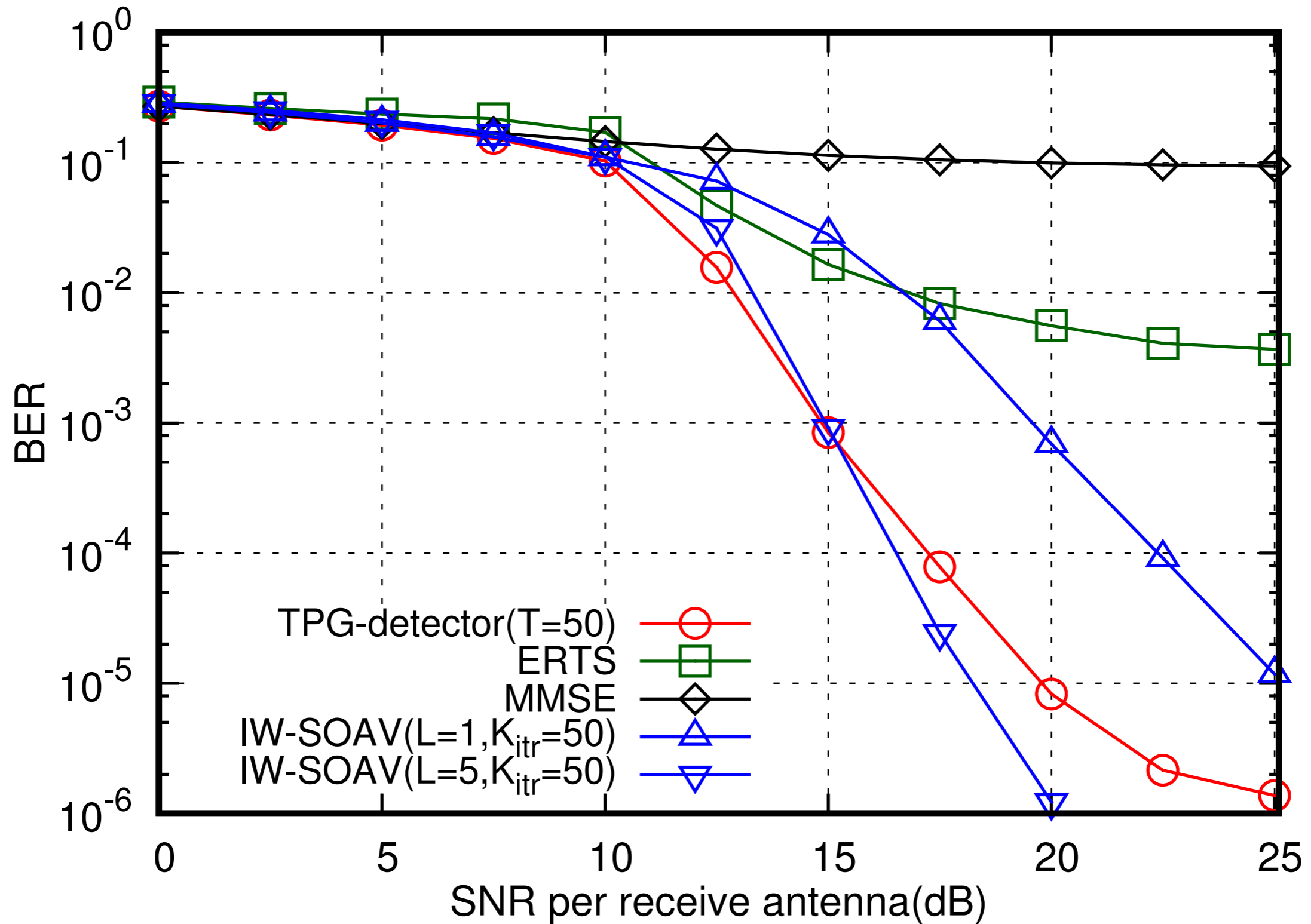


**t=5**



**t=10**

# BER比較：(n,m)=(150,96)



# 複素TISTA (C-TISTA)

Takabe and Wadayama, <https://arxiv.org/abs/1904.07409>, 2019

## 非線形観測問題

$$\mathbf{y} = f(\mathbf{A}\mathbf{x}) + \mathbf{w},$$

関数 $f$ はコンポーネントワイズ関数で、非線形関数でもよい

### C-TISTA

$$\begin{aligned} \mathbf{r}^{(t)} &:= \mathbf{s}^{(t)} + \beta_t h(\mathbf{s}^{(t)}), \\ \mathbf{s}^{(t+1)} &:= \eta(\mathbf{r}^{(t)}; \lambda^{(t)}), \end{aligned}$$

(複素関数 $f$ のウィルティンガー微分のチェインルールに基づく)

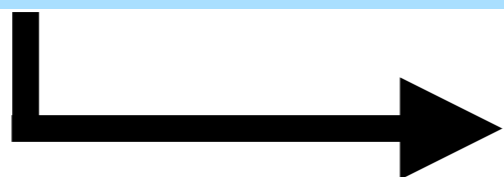
$$\lambda^{(t)} := a_t + b_t \frac{\|\mathbf{y} - f(\mathbf{A}\mathbf{s}^{(t)})\|_2^2}{\text{Tr}(\mathbf{A}^H \mathbf{A})},$$

$$\begin{aligned} h(\mathbf{s}) &:= \mathbf{W} \left[ \{\mathbf{y} - f(\mathbf{A}\mathbf{s})\}^* \odot \frac{\partial f}{\partial z^*}(\mathbf{A}\mathbf{s}) \right. \\ &\quad \left. + \{\mathbf{y} - f(\mathbf{A}\mathbf{s})\} \odot \frac{\partial f^*}{\partial z^*}(\mathbf{A}\mathbf{s}) \right] \end{aligned}$$

# C-TISTAの展開

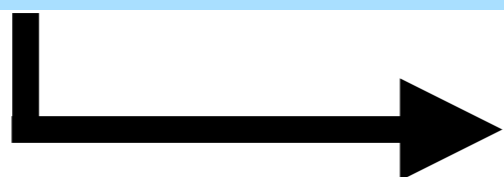
$$y = f(Ax) + w,$$

A:IDFT 行列、f:クリッピング関数



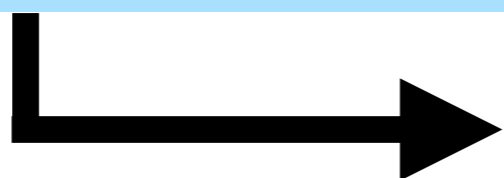
OFDMにおけるPAPR改善  
(クリッピング保証)

A:複素疎行列、f:恒等関数



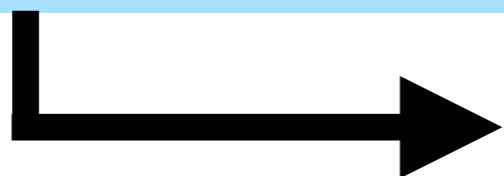
SCDMAの検出アルゴリズム

A:IDFT 部分行列、f:クリッピング関数



複素数体上の誤り訂正符号

A:通信路行列、f:量子化関数



量子化を伴うOverloaded  
MIMO検出

# C-TISTAによるクリッピング補償

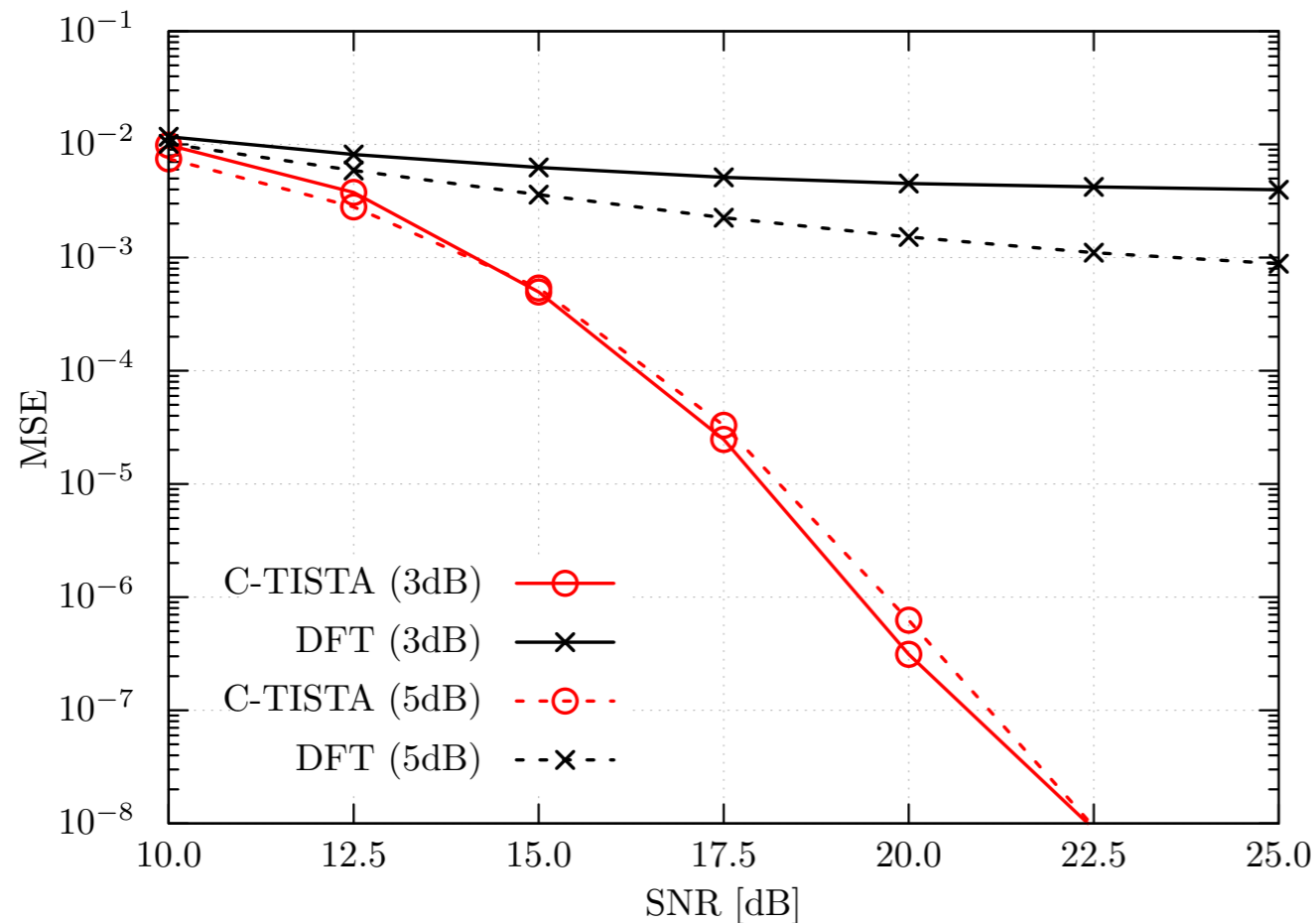


Fig. 5. MSE as a function of SNR in the clipped OFDM system ( $n = 128$ ). Symbols represent proposed C-TISTA ( $T = 10$ ; circles) and DFT (cross marks) when PAPR=3dB (solid) and 5dB (dashed).

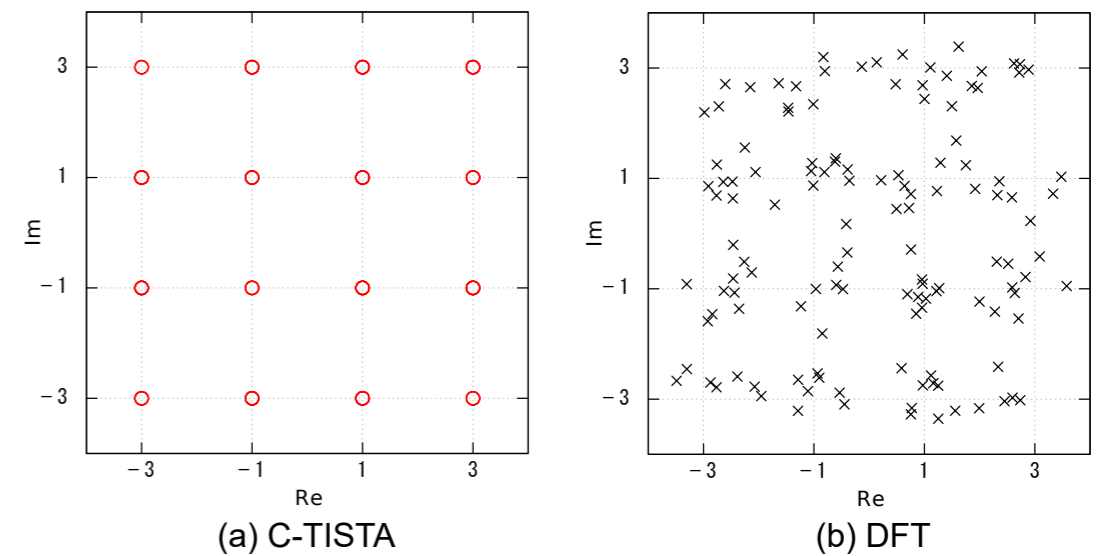
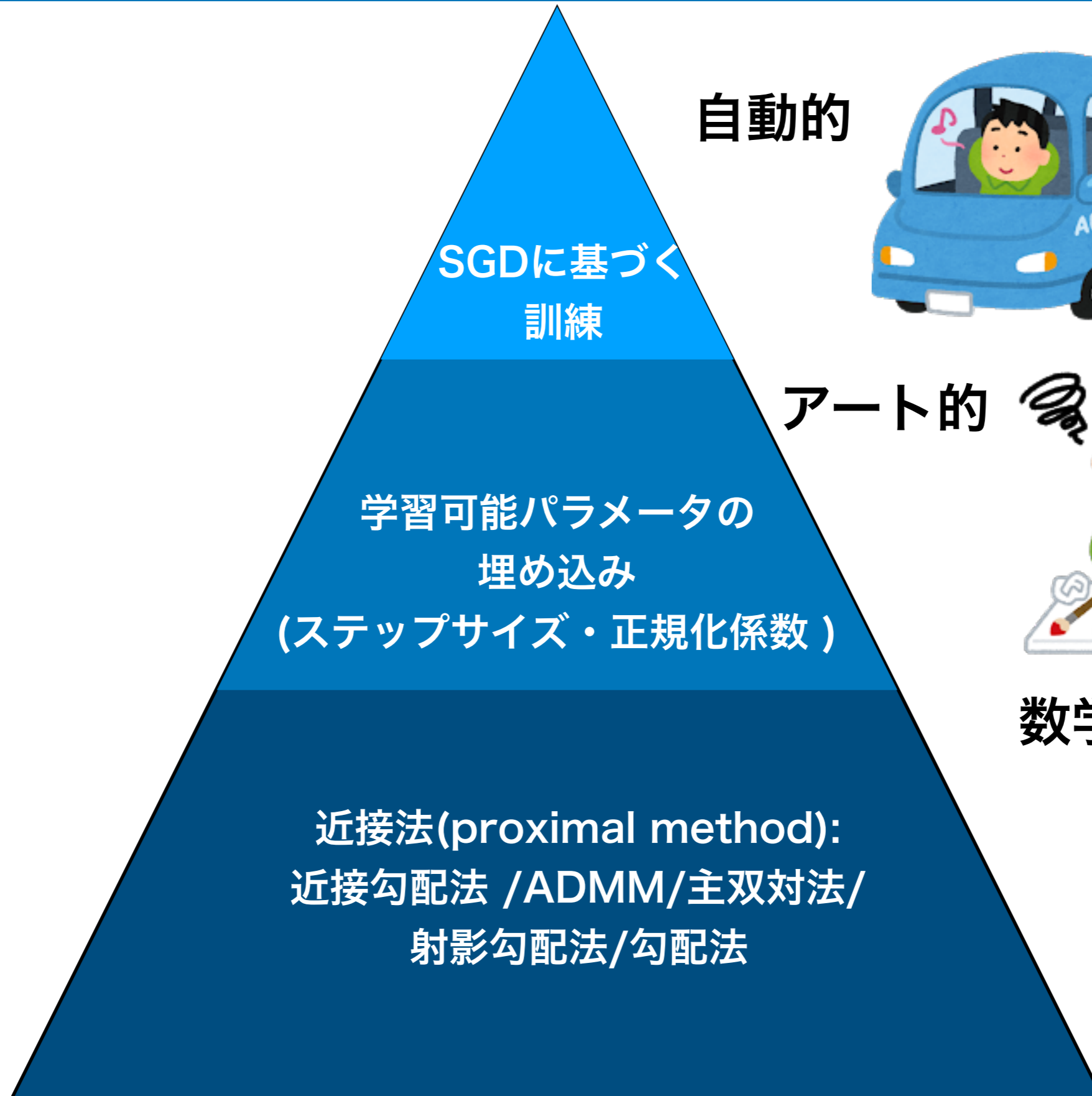
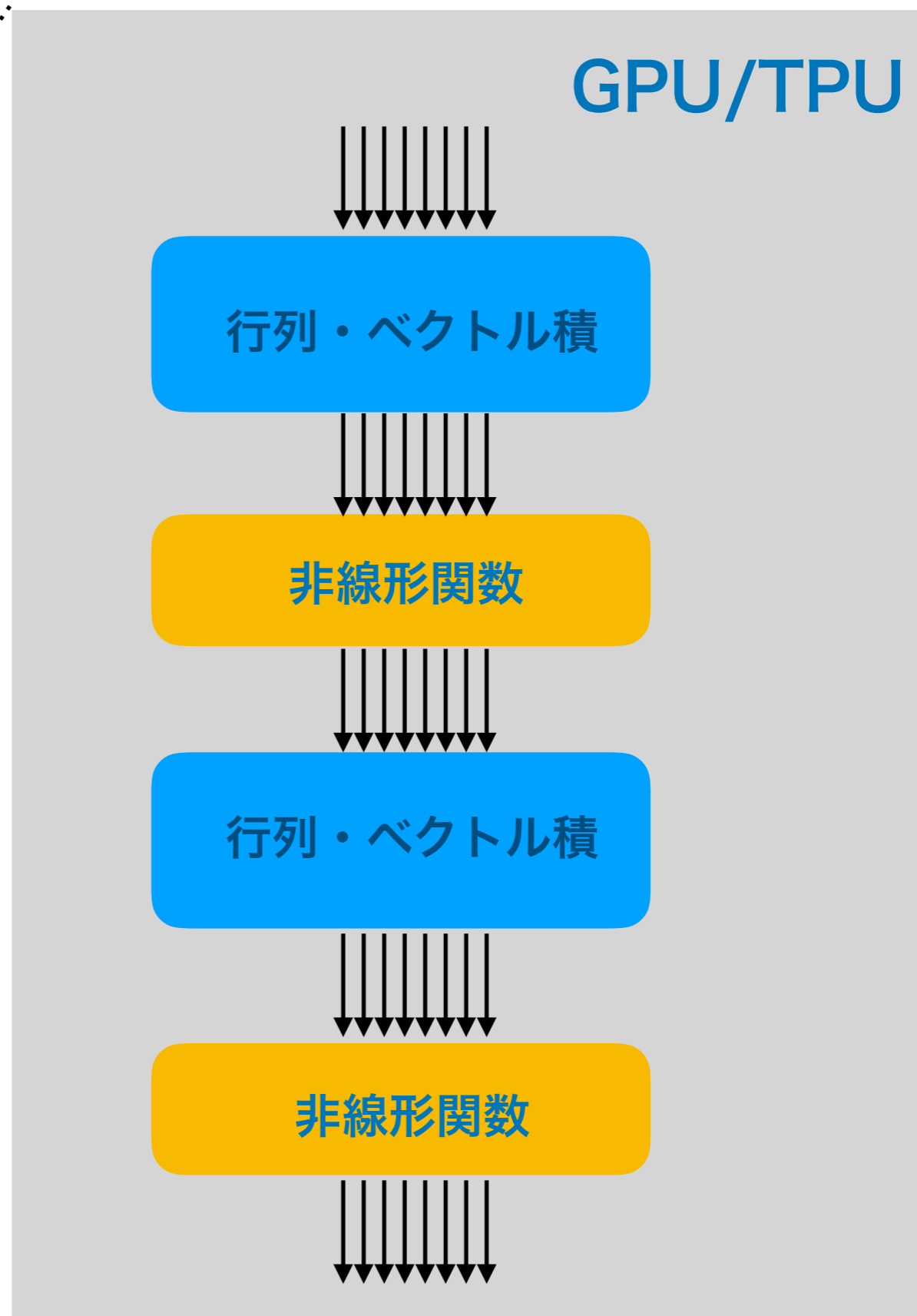
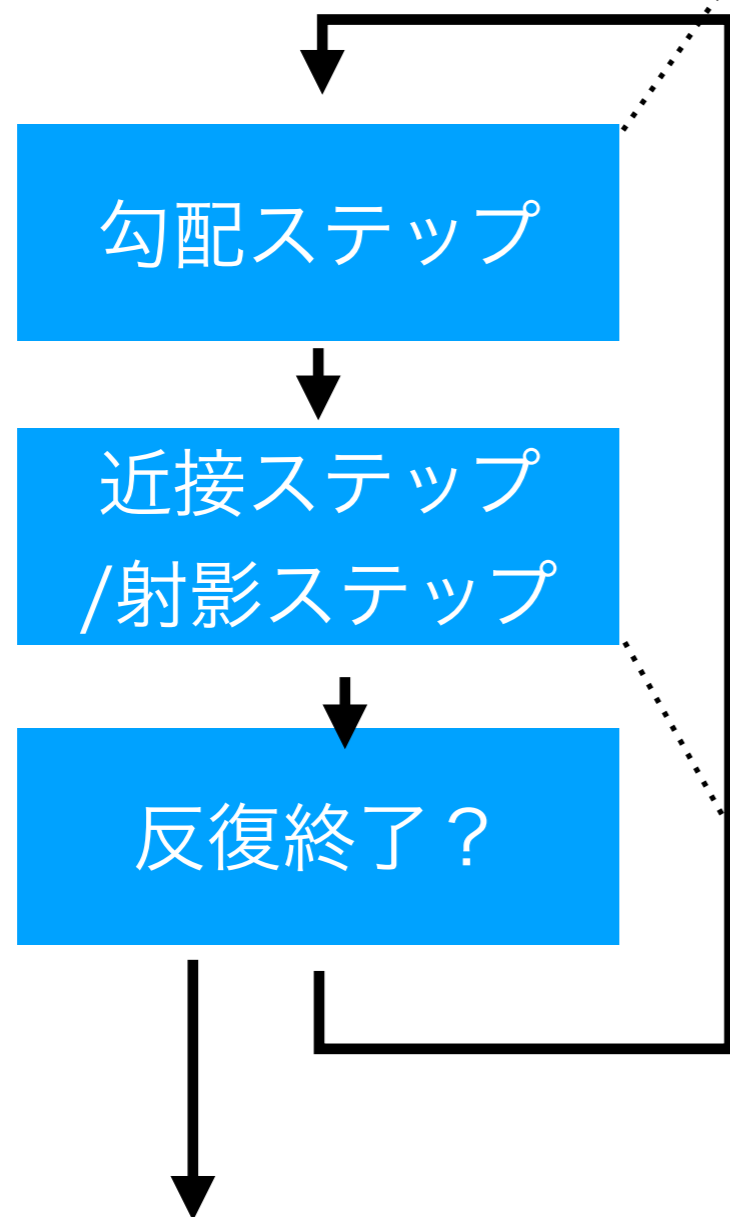


Fig. 6. Example of detected signals by (a) C-TISTA and (b) DFT when SNR=17.5dB and PAPR=3dB. Each symbol represents a detected signal  $\{\hat{x}_i\}_{i=1}^{n=128}$  before hard thresholding.

# 近接勾配アルゴリズムの深層展開



# 近接勾配法とニューラルネット用ハードウェア





# 本グループの関連発表

現在、popular Items 50にランクイン！！  
IEEE Trans.on Signal Processing

## 関連論文

Trainable ISTA (TISTA)  
スパース信号再現

IEEE ICC Workshop 2018  
IEEE Trans. on Signal Processing, 2019

BP復号器に適したニューラル量子化器

IEEE Globecom 2018

学習可能射影勾配法に基づく過負荷MIMO検出  
アルゴリズム

IEEE ICC 2019/ IEEE Access, 2019

学習可能射影勾配型復号法 (LDPC符号)

IEEE ISIT 2019

複素TISTA(C-TISTA)  
非線形観測再現

TBD

## 現在進行中プロジェクト

Sparse CDMA, Quantized CS/MIMO, Phase retrieval  
近接勾配法に基づく組み合わせ最適化, 分散最適化

# モデルベース v.s. ブラックボックスベース

## モデルベース アプローチ

## ブラックボックス アプローチ

### Pros:

**ドメイン知識の有効活用**  
原理からアルゴリズムの演繹的導出  
(例:事後確率計算→同時分布の  
因子分解→分配則の適用→BP)

**ドメイン知識不要**  
データ駆動型 (帰納的) アルゴリズム  
設計

### Cons:

既存のアルゴリズムをよりパワー  
アップ(性能・柔軟性)するにはどうす  
れば。。。?

**解釈可能性問題**  
**スケーラビリティ**

# モデルベース v.s. ブラックボックスベース

モデルベース  
アプローチ

深層展開

ブラックボックス  
アプローチ

Pros:

**ドメイン知識の有効活用**

原理からアルゴリズムの演繹的導出  
(例:事後確率計算→同時分布の  
因子分解→分配則の適用→BP)

**ドメイン知識不要**

データ駆動型 (帰納的) アルゴリズム  
設計

Cons:

既存のアルゴリズムをよりパワー  
アップ(性能・柔軟性)するにはどうす  
れば。。。?

**解釈可能性問題**

**スケーラビリティ**

# むすび

- ✓ 信号検出問題における**非凸最適化アプローチ**について、特にGDBF復号法を中心に説明を行った→非凸最適化アプローチは信号処理アルゴリズムデザインにおける強力な原理
- ✓ 非凸最適化も**恐れずにやってみる**ことが重要（非凸最適化の時代）
- ✓ 特に**深層展開**について、その方法論・長所について詳細をご紹介した
- ✓ 深層学習を利用する研究においては:
  - ✓ われわれの慣れている研究の方法論（原理→アルゴリズム: 演繹的）とは異なる方法論(データ→アルゴリズム: 帰納的)が必要であり、**頭(考え方)の切り替え**が必要
  - ✓ どうしてもブラックボックス的部分は残る(解釈可能性問題)→そのような部分が**最小となるようにアルゴリズムを構成**するのがよいのでは？
  - ✓ ブラックボックスモデルによりもたらされる柔軟性
  - ✓ 深層展開は**シンプルかつ強力**・非凸最適化問題のための頼れる武器