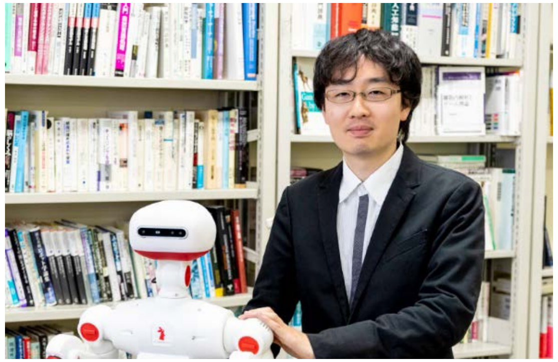
計算知能 (COMPUTATIONAL INTELLIGENCE)

第１回　計算知能の基礎

教員：　谷口彰

自己紹介



- 氏名： 谷口　彰　　　　　　　　　職位：講師
- 所属： 立命館大学 情報理工学部 創発システム研究室

担当講義

- 人工知能（大連理工大学）
- 計算知能（大連理工大学）

略歴

- 2018年 3月　立命館大学大学院 情報理工学研究科 博士後期課程　修了　博士（工学）
- 2015年 5月～2017年 3月　立命館大学　総合科学技術研究機構　リサーチアシスタント
- 2017年 4月～2019年 3月　日本学術振興会 特別研究員 (DC2, PD)
- 2019年 4月～2022年 3月　立命館大学 情報理工学部 特任助教
- 2022年 4月～　　　　　　　立命館大学 情報理工学部 講師

- 主な研究テーマ
  「確率的生成モデルに基づくロボットによる場所概念および語彙の獲得」
  - 位置情報・言語情報・画像情報といったマルチモーダル情報から，教師なし機械学習手法により場所概念（場所のカテゴリ）や場所に関する語彙の獲得を目指す研究.

講義内容(1/2)

- 第1回　計算知能の基礎
- 第2回　最適化の事例
- 第3回　進化型計算
- 第4回　ニューロコンピューティングの基礎
- 第5回　ニューラルネットワークの学習アルゴリズム（1/2）
- 第6回　ニューラルネットワークの学習アルゴリズム（2/2）
- 第7回　多層ニューラルネットワーク
- 第8回　前半復習問題と解説

講義内容(2/2)

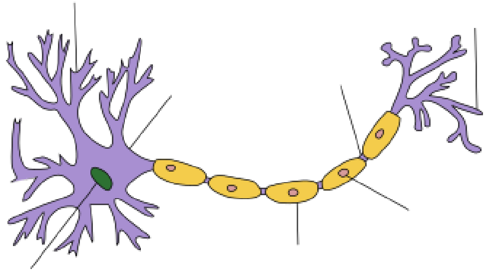
- 第9回　　プログラミング演習（1/2）
- 第10回　プログラミング演習（2/2）
- 第11回　畳み込みニューラルネットワーク
- 第12回　ニューラルネットワークまとめ
- 第13回　決定木
- 第14回　強化学習
- 第15回　まとめ

成績評価について

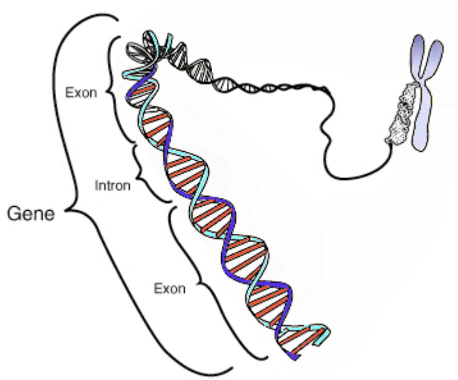
- 平常点　４０点
  - プログラミング演習のレポート課題
  - 出席
- 定期試験 ６０点
  - 講義の全範囲の内容について出題

今回の内容

- 計算知能の代表的なアルゴリズム概説
  - 遺伝的アルゴリズム
  - ニューラルネットワーク
  - 決定木、強化学習

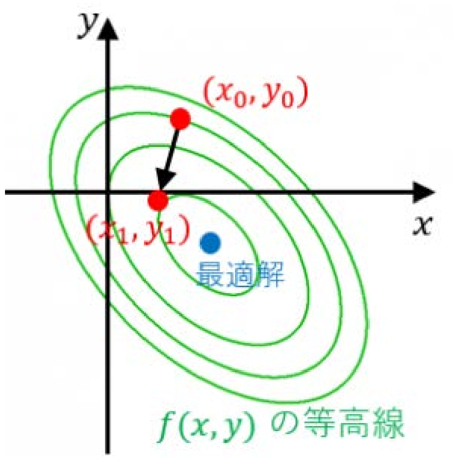


ニューロン
https://ja.wikipedia.org/wiki/%E7%A5%9E%E7%B5%8C%E7%B4%B0%E8%83%9E

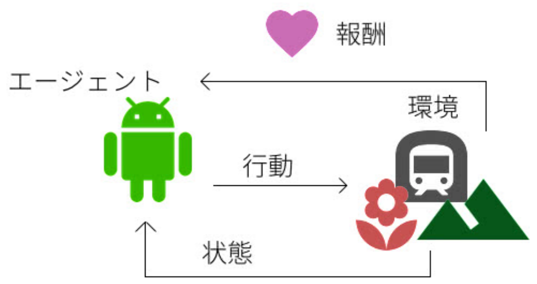


遺伝子
https://ja.wikipedia.org/wiki/%E9%81%BA%E4%BC%9D%E5%AD%90

- 最適化（探索）問題
  - 最急降下法
  - 焼きなまし法



最急降下法
https://mathwords.net/saikyukouka



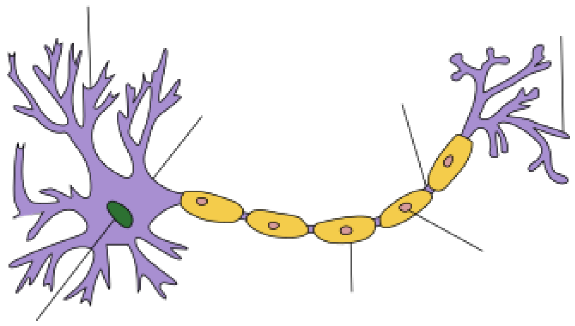
強化学習の概要
https://deepage.net/machine_learning/2017/08/10/reinforcement-learning.html

計算知能(Computational Intelligence)とは

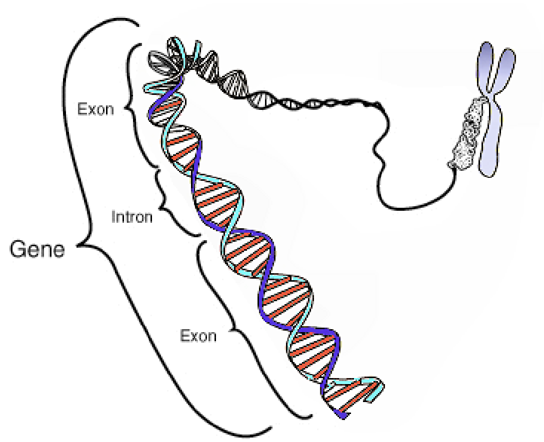
- 計算知能とは
  - 人工知能研究の一分野
  - ヒューリスティック（発見的）なアルゴリズムを扱う
  - 機械学習と密接に関係している
- 主要なテーマ
  - ニューラルネットワーク
  - ファジィシステム
  - 進化的計算：遺伝的アルゴリズム、遺伝的プログラミング
  - 群知能：粒子群最適化
- 発見的手法
  - 人間や生物にヒントを得た発見的なアルゴリズム
  - 必ず正しい答えを導けるわけではないが、ある程度のレベルで正解に近い解を得ることができる方法

本講義で扱う計算知能

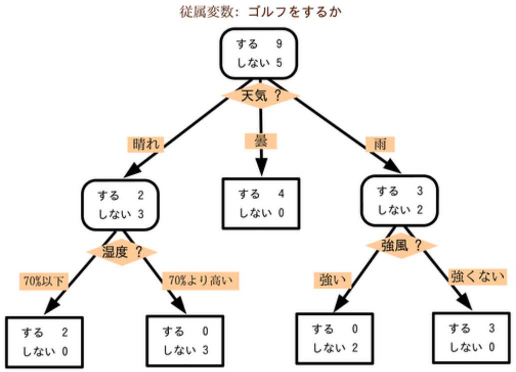
- 本講義で扱うアプローチ
  - 遺伝的アルゴリズム
  - ニューラルネットワーク
  - 決定木
  - 強化学習



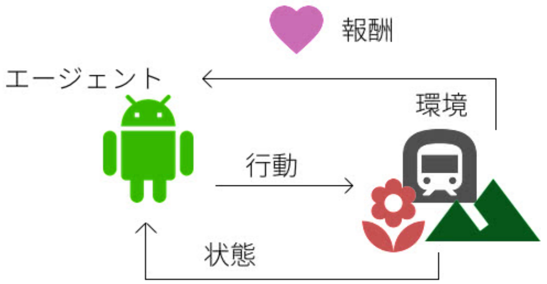
ニューロン
https://ja.wikipedia.org/wiki/%E7%A5%9E%E7%B5%8C%E7%B4%B0%E8%83%9E



遺伝子
https://ja.wikipedia.org/wiki/%E9%81%BA%E4%BC%9D%E5%AD%90

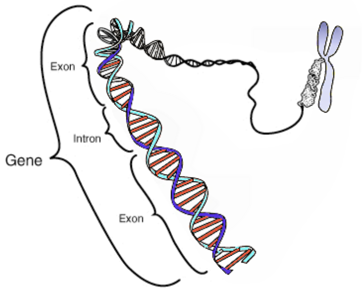


決定木の例
https://ja.wikipedia.org/wiki/%E6%B1%BA%E5%AE%9A%E6%9C%A8



強化学習の概要
https://deepage.net/machine_learning/2017/08/10/reinforcement-learning.html

遺伝的アルゴリズム



- 遺伝的アルゴリズムは、生物の進化（遺伝）の機構を模倣した確率的探索手法

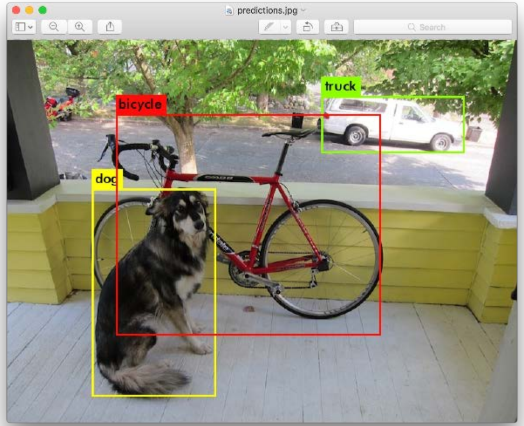
選択	交叉（こうさ）	突然変異

- 大規模組合せ最適化問題、多峰性の探索問題に有効

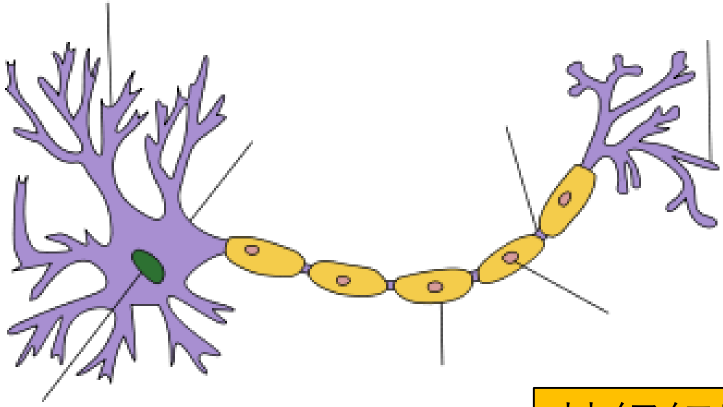
- 遺伝的アルゴリズム、遺伝的プログラミング、
進化的プログラミング、進化戦略を総称して進化的計算と呼ぶ

ニューラルネットワーク(1/2)

- ニューラルネットワークとは神経回路網を意味する言葉
- 脳の神経回路の基本機能の一部をモデル化し、工学的に役立てることが狙い
- パターン認識などの分野で利用されることが多い



[パターン認識の例](https://pjreddie.com/darknet/yolo/)
https://pjreddie.com/darknet/yolo/

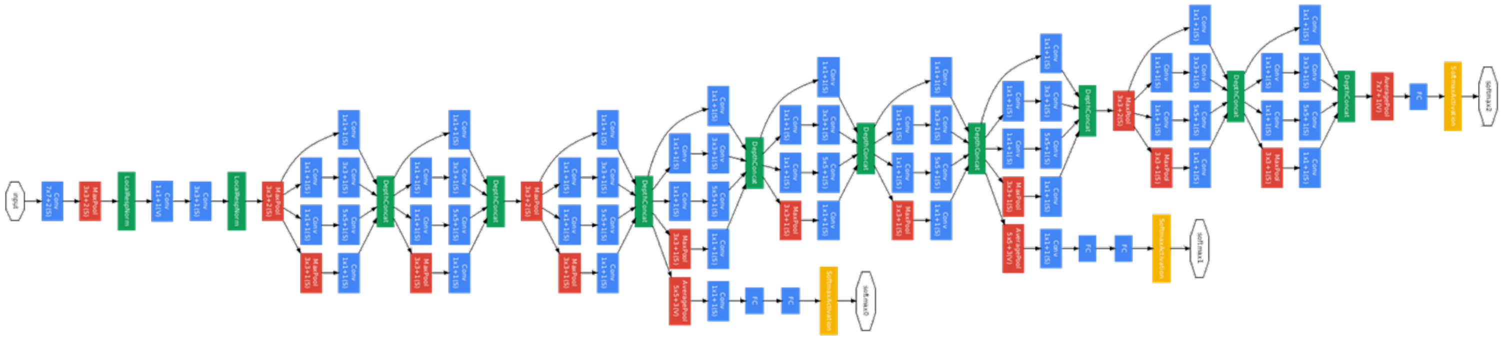


神経細胞（ニューロン）

ニューラルネットワーク(2/2)

- 近年、あらゆる分野で活用されているDeep Learningもニューラルネットワークの発展形

- 特に画像認識、自然言語処理で非常に良い性能を示す



GoogLeNet(ILSVRC-2014)

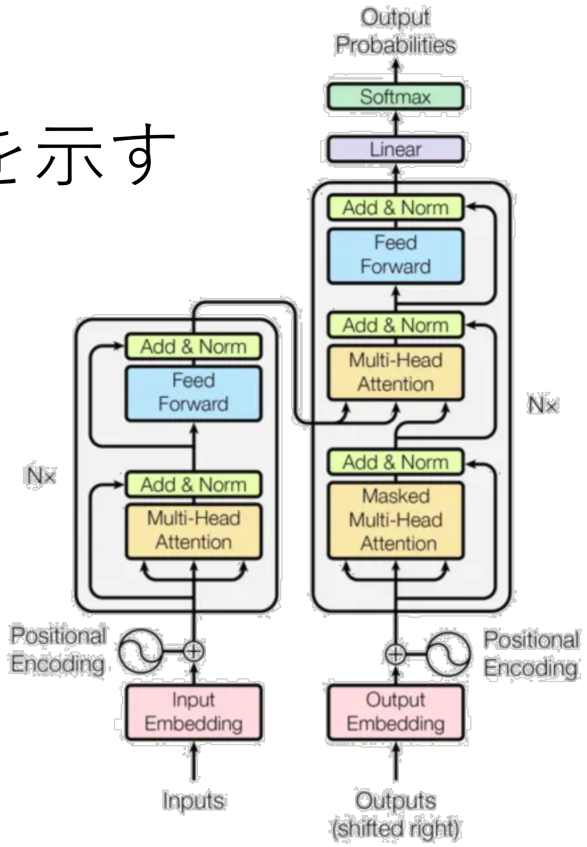


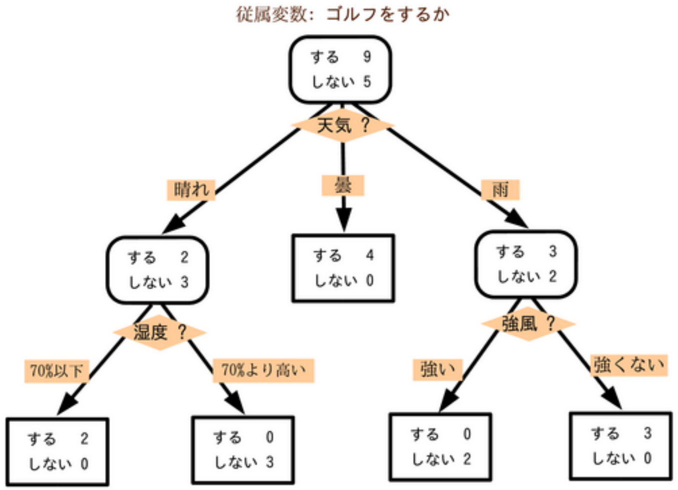
Figure 1: The Transformer - model architecture.

Attention Is All You Need.(Ashish Vaswani, et al., 2017)

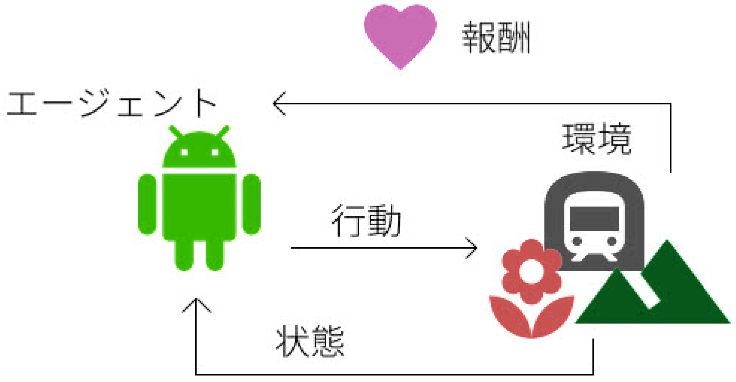
決定木、強化学習

■ 決定木
- データにおける規則を木の形で表現
- 教師あり学習の機能を実現

■ 強化学習
- 試行錯誤による経験の積み重ねの模倣
- 教師なし学習、探索の機能を実現



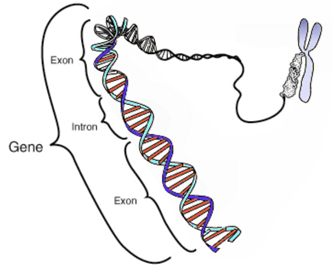
決定木の例

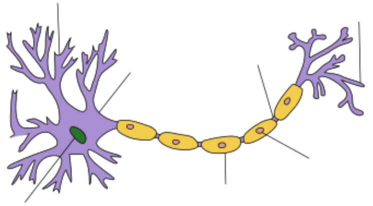


強化学習の例

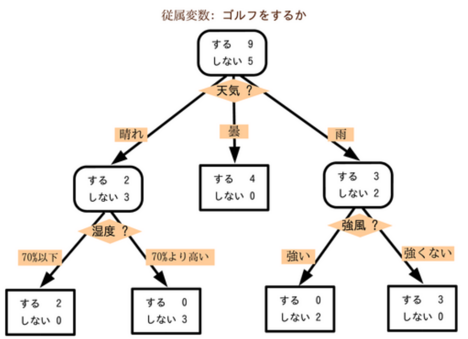
計算知能の特徴

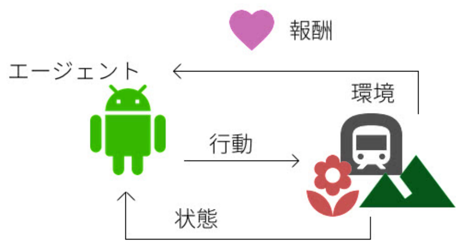
- 生物（特に人）が発想の原点にある技術
  - 遺伝的アルゴリズム：生物の進化をモデル化
  - ニューラルネットワーク：生体の神経回路網を模倣





- 生物（特に人）の考え方をモデル化
  - 決定木：人が考えるルールを木構造でモデル化
  - 強化学習：人の試行錯誤プロセスをモデル化





5 分休憩

最適化問題(1/2)

- 最適化問題とは、特定の集合上で定義された実数値関数について、その値が最小（もしくは最大）となる状態を解析する問題(数理計画問題も含む)
- 物理学やコンピュータビジョンにおける最適化問題は、考えている関数をモデル化された系のエネルギーを表すものと見なす(エネルギー最小化問題とも呼ばれる)

最適化問題(2/2)

$$\min_{x} \boxed{f(x)}$$ ←目的関数

$$\mathrm{s.t.} \boxed{x \in F}$$ ←制約条件

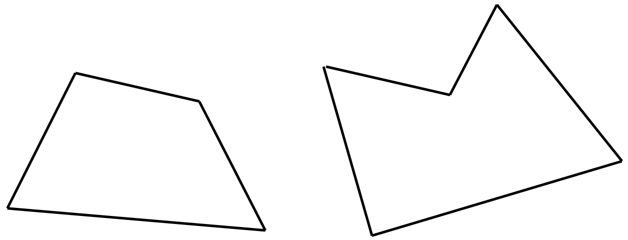
$$F \subseteq X$$

- x：決定変数（対象問題で決定すべき量）
  一般には複数あるので、ベクトルで表現
  （そのベクトル空間を X とする）
- $f(x)$：目的関数（ x の良さ/悪さを与える値）
- F：可能解領域（空間 X の中で解が許される範囲）

最適化 ： 目的関数の最大化/最小化

問題の分類、難しさのレベル(1/2)

- 解ける（数学的or解析的に最適解を求められる）場合
  ベクトル空間 X がn次元の実数値ベクトルで、
  F が綺麗な形（n次元凸集合）　←凸多角形の線形計画問題



凸多角形　　非凸多角形

- 厳密には解けないが、比較的単純な探索手続きがある
  （F がn次元凸多角形でない　etc. ）場合
  しかし、微分操作はできる
  →　ex.最急降下法で局所最適を何度も繰り返し求めて、
  　その中に本当の最適解が入っていることを期待する）

- 単純な手続きがなく、全解を列挙するしかない場合

問題の分類、難しさのレベル(2/2)

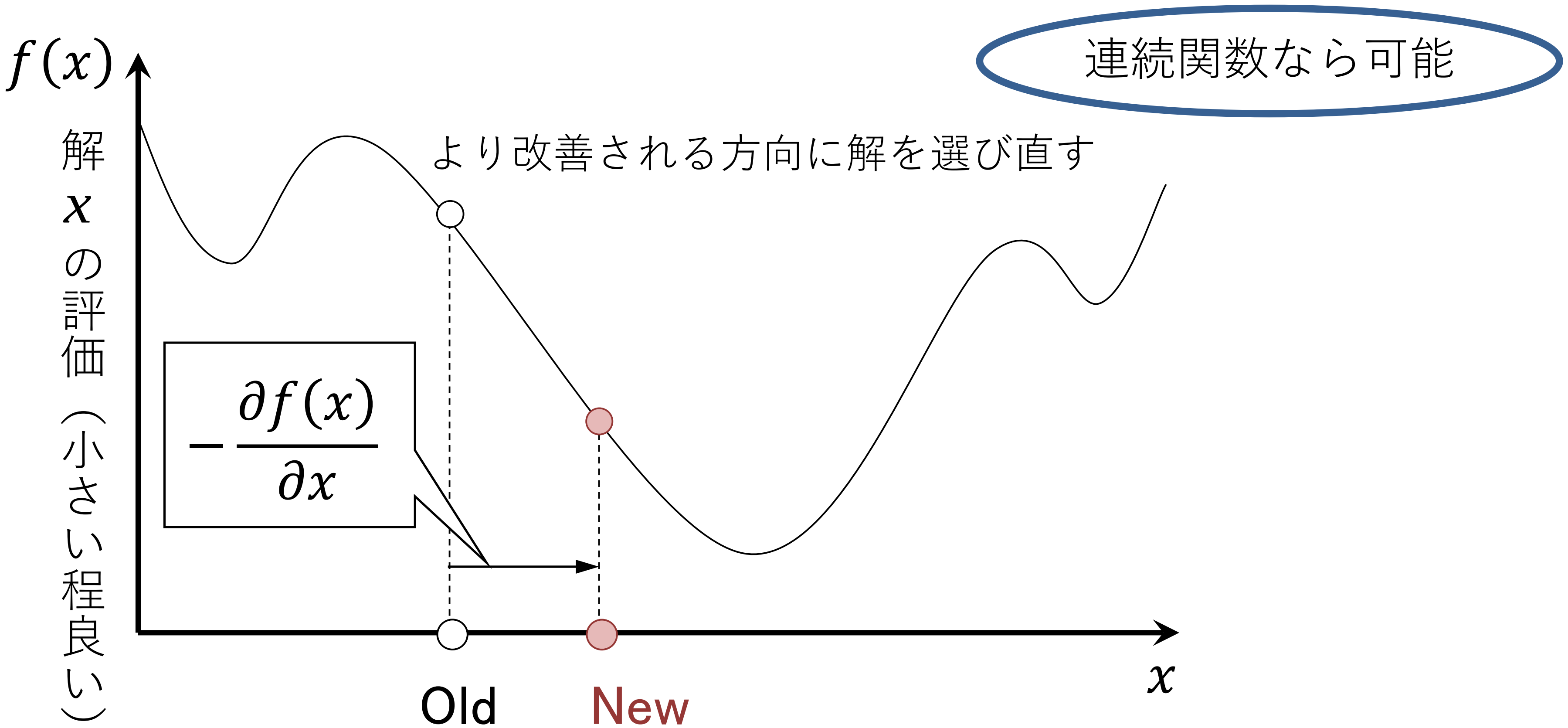
- 解ける（数学的or解析的に最適解を求められる）場合
  ベクトル空間 X がn次元の実数値ベクトルで、
  F が綺麗な形（n次元凸集合）　←凸多角形の線形計画問題

- 厳密には解けないが、比較的単純な探索手続きがある
  （F がn次元凸多角形でない　etc.）場合

- 単純な手続きがなく、全解を列挙するしかない場合
  - 変数xが離散的 → 頑張って探索
    ex.組合せ最適化問題
  - $f(x)$が不連続 → 不連続点では探索の手掛かりもない
    問題に応じて対応策を検討

最急降下法(1/4)

- 現在の x の値 x(Old) の微分($-{}^{\partial f(x)}/_{\partial x}$)をとって、$f(x)$ の値が減る方向へ新しい x(New) に更新する



最急降下法(2/4)

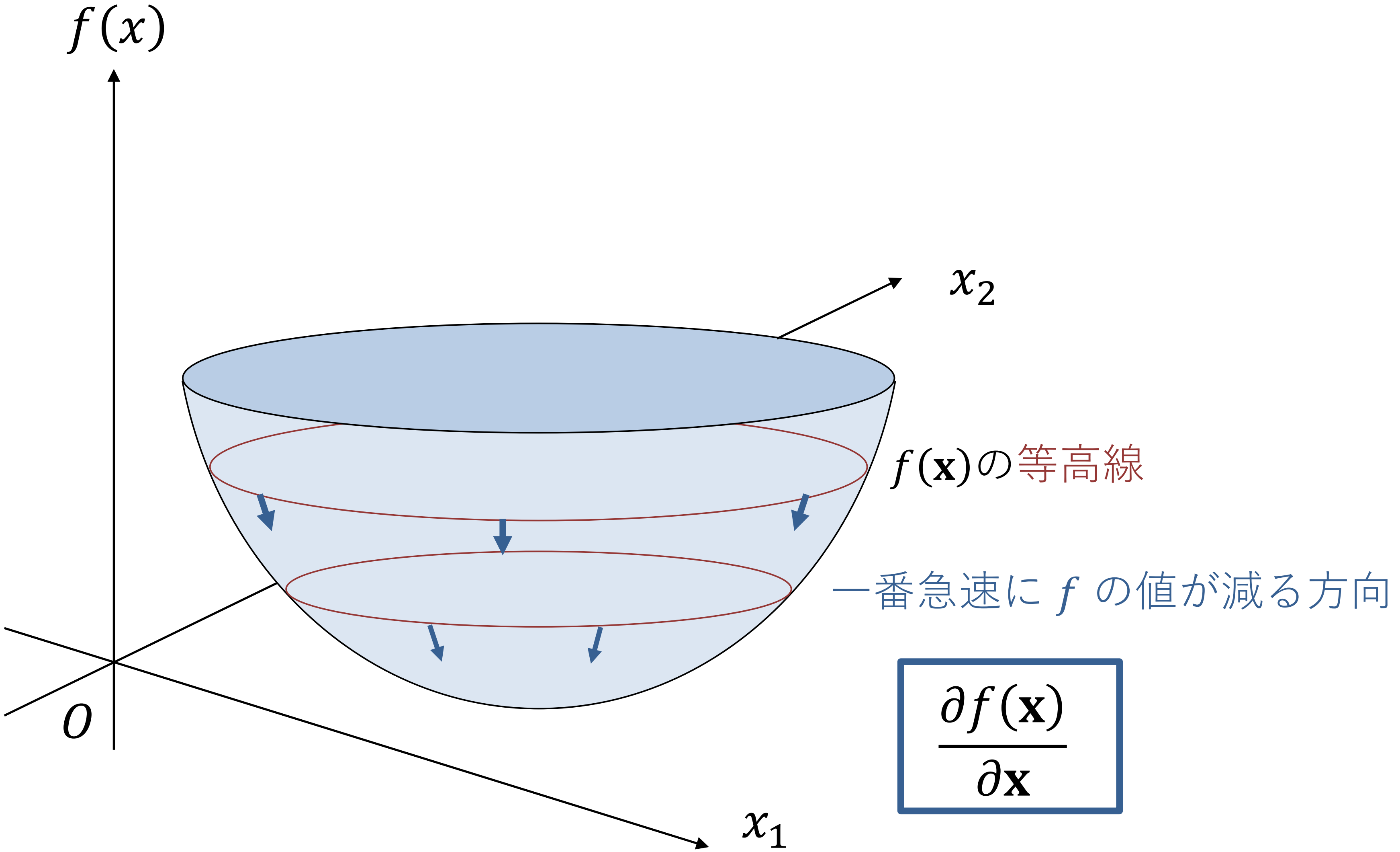
- n次のベクトル $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ を引数とする関数fを考え、この関数の極小値を求める

- 最急降下法では反復法を用いて $\mathbf{x}$ を解に近づけていく。
  k回目の反復で解が $\mathbf{x}^{(k)}$ の位置にあるとき、
  最急降下法では右式のようにして
  値を更新する

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \eta \left({\partial f(\mathbf{x})} \big/ {\partial \mathbf{x}} \right)$$

$$= \mathbf{x}^{(k)} - \eta \begin{bmatrix} {\partial f(\mathbf{x})} \big/ {\partial x_1} \\ {\partial f(\mathbf{x})} \big/ {\partial x_2} \\ \vdots \\ {\partial f(\mathbf{x})} \big/ {\partial x_n} \end{bmatrix}$$

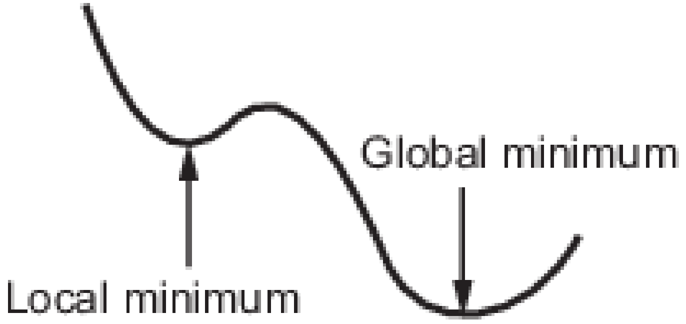
最急降下法(3/4)



$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$$

最急降下法(4/4)

- 利点
  - 実装が容易
  - 解空間が単峰性である場合、誤った方向への探索がない
- 欠点
  - パラメータに依存
  - 解空間が多峰性である場合、局所解に陥りやすい



https://jp.mathworks.com/help/optim/ug/local-vs-global-optima.html

練習問題1-1

- 下記の関数について、最急降下法を用いて最良解を探索せよ
  - 初期値は$(x, y) = (5,5)$、 $\eta = 0.1$とする
  - 探索回数は3回とする

- $f(x) = x^2$
- $f(x, y) = x^2 + y^2$
- $f(x) = 3x^2 + (x - 5)^2$
- $f(x) = (x - 5)^2(x + 5)^2$

練習問題1-1

- 下記の関数について、最急降下法を用いて最良解を探索せよ
  - 初期値は$\left(x^{(0)}, y^{(0)}\right) = (5,5)$、 $\eta = 0.1$とする
  - 探索回数は3回とする

$$f(x) = x^2$$

$$f'(x) = 2x$$

探索1回目：

$$x^{(1)} = x^{(0)} - \eta \left({}^{\partial f(x)}/_{\partial x} \right)$$
$$= 5 - 0.1 \cdot 10$$
$$= 4$$

探索2回目：

$$x^{(2)} = 3.2$$

探索3回目：

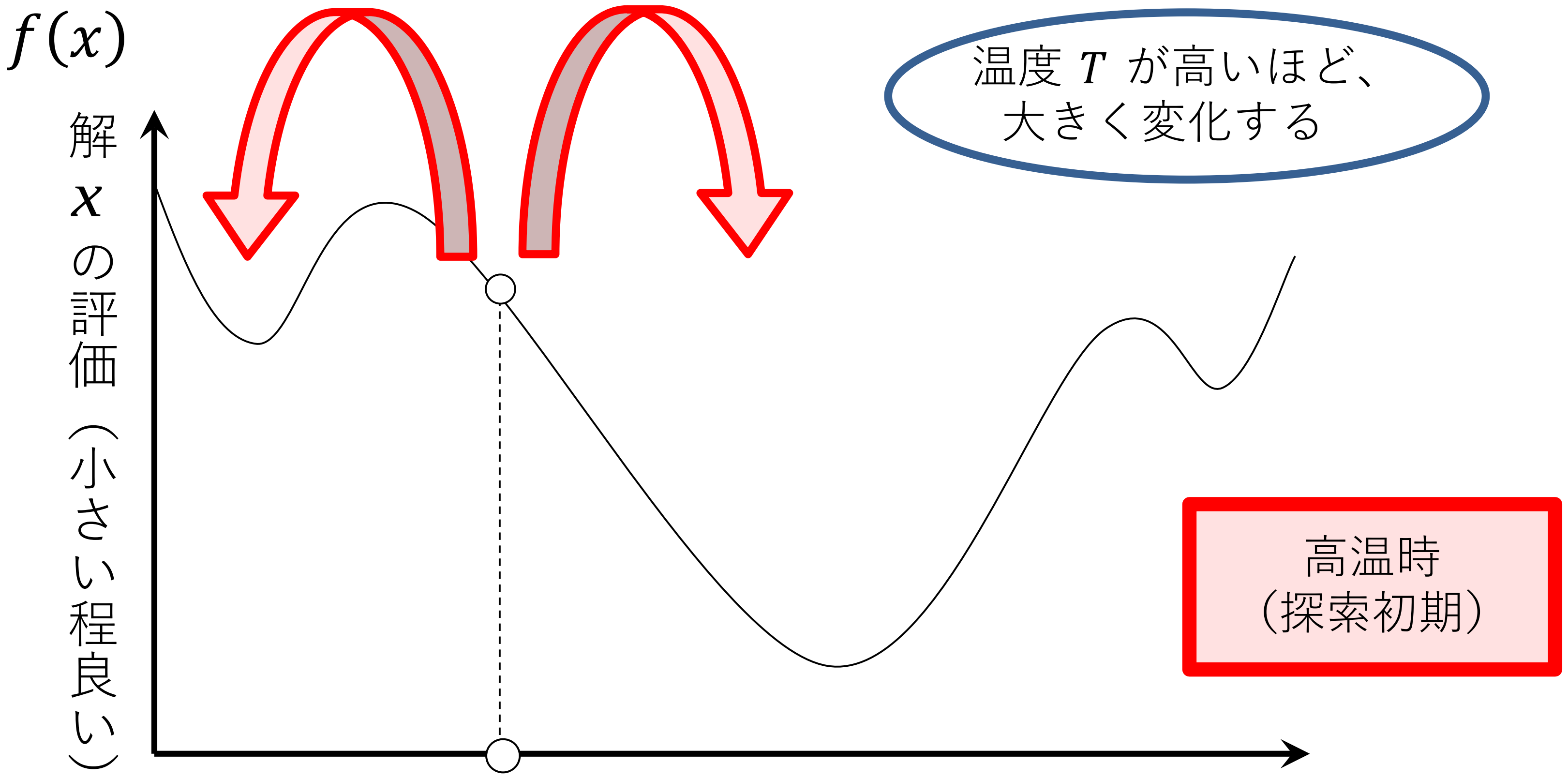
$$x^{(3)} = 2.56$$

焼きなまし法(1/5)

- 大域的最適化問題解決のための汎用的な乱択アルゴリズム
- 広大な探索空間内で、与えられた関数の大域的最適解に対して、よい近似を取得するための手法

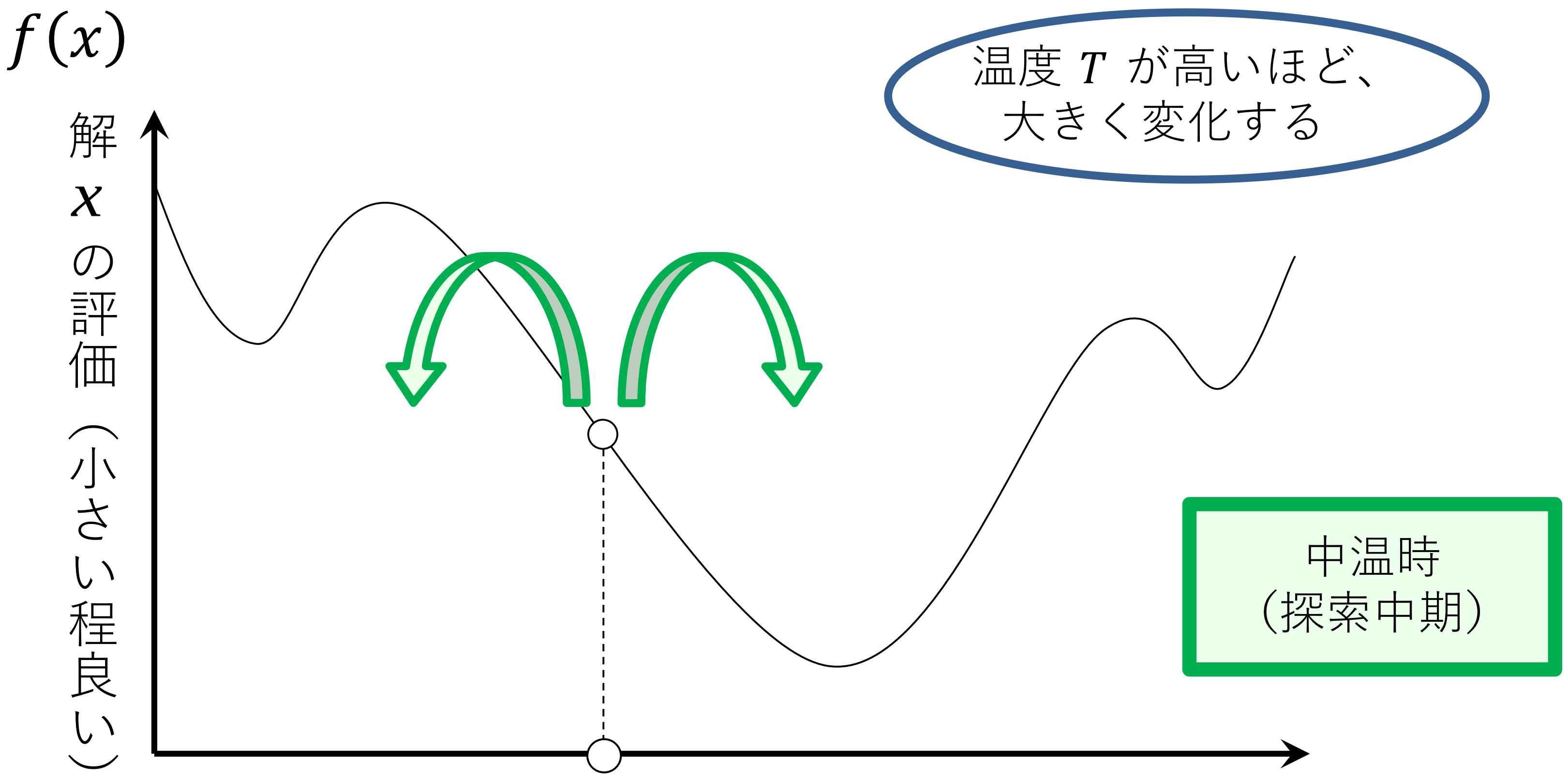
焼きなまし法(2/5)

- x、$f(x)$の他に温度 T を考える
- x の変化量は T に応じて 大きく変化する



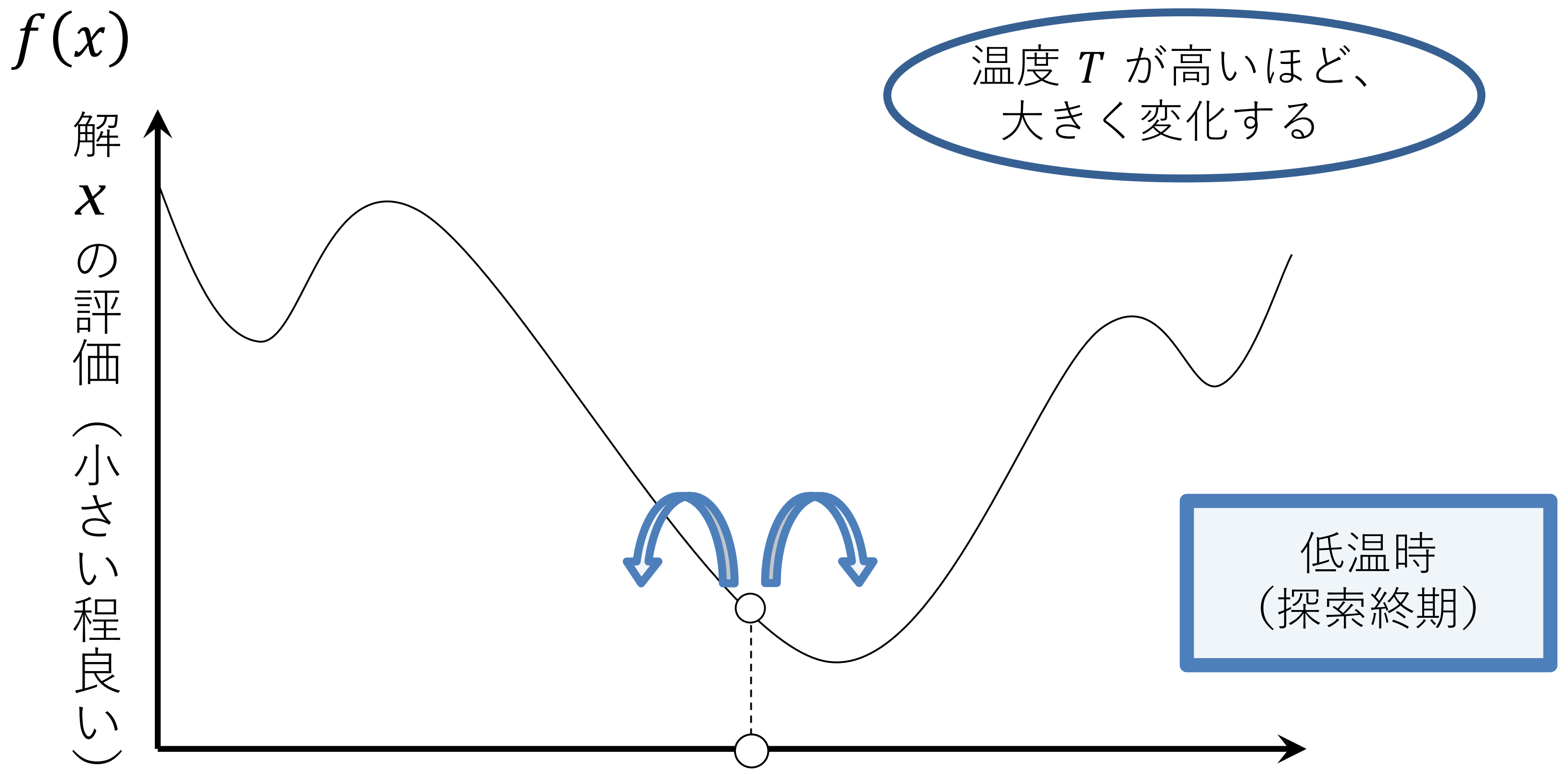
焼きなまし法(3/5)

- x、$f(x)$の他に温度 T を考える
- x の変化量は T に応じて 大きく変化する



焼きなまし法(4/5)

- x、$f(x)$の他に温度 T を考える
- x の変化量は T に応じて 大きく変化する



焼きなまし法(5/5)

- 温度 T は最初は高くして、例えば探索回数に応じて減少させる
(探索初期：高 → 探索終期：低)
- 解空間の形状があらかじめ判明している場合、
探索初期：最大の山を超えられるように（局所解に陥らないように）
探索終期：大きな山は越えず、局所的な最適解を取得できるように

- 探索初期：　大域的に（大雑把に）良い解を探索
- 探索中期：　ある程度近傍に限定して、解を探索
- 探索終期：　（ほぼ探索が完了しているものとして）
近傍の最適解を厳密に探索

それぞれに異なる目的を持ち、異なる探索を実施

非常に多くの手法に応用されている

まとめ

- 計算知能の概要を学んだ。
- 計算知能の主要な要素技術を概観した。
- 計算知能における主要な概念である最適化問題について学んだ。
- 最適化問題の代表的なアルゴリズムとして最急降下法と焼きなまし法を学んだ。

復習問題

1. 最適化問題の難しさを三つのレベルに分類せよ
2. 最適化問題における最適化は、具体的にはどのような計算によって達成されるか？
3. 目的関数の微分によって決定変数を更新するアルゴリズムを何というか？
4. 大域的最適化問題解決のための汎用的な乱択アルゴリズムを何というか？

復習問題

1. 最適化問題の難しさを三つのレベルに分類せよ
   ⇒解析的に解ける、厳密には解けないが探索手続きがある、単純な手続きが無い
2. 最適化は、具体的にはどのような計算によって達成されるか？
   ⇒目的関数の最小化あるいは最大化
3. 目的関数の微分によって決定変数を更新するアルゴリズムを何というか？
   ⇒最急降下法
4. 大域的最適化問題解決のための汎用的な乱択アルゴリズムを何というか？
   ⇒焼きなまし法

次回の講義

- 最適化の事例
  - 組み合わせ最適化問題の典型例
  - 欲張り法
  - 粒子群最適化法