数理・計算科学特論 C 第 5 回

河瀬 康志

2021 年 6 月 22 日

乱択アルゴリズムに対する競合比の定義

前回考えた競合比

$$\sup_{\text{入力列}} \frac{\mathbb{E}[\text{乱択オンラインアルゴリズムでのコスト}]}{\text{入力列を知っていた場合の最適コスト}}$$

- 乱数の出方によるアルゴリズムの挙動によらず入力列は固定している
- アルゴリズムの出力が将来の入力に影響を及ぼす状況では不適切
- どのようなモデルを考えれば良い？

今回の内容

- オンライン問題を数学的に定式化（リクエストアンサーゲーム）
- 乱択アルゴリズムに対する 3 種類のアドバーサリ（敵）を定義
  - オブリビアスアドバーサリ
  - アダプティブオンラインアドバーサリ
  - アダプティブオフラインアドバーサリ
- アドバーサリの違いによる競合比の関係を示す

アウトライン

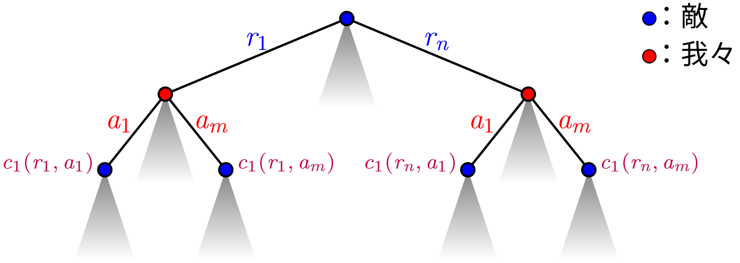
1. リクエストアンサーゲーム
2. アドバーサリモデル
3. 競合比の関係
4. 最悪ケースを越えて
5. まとめ

オンライン問題の形式的定義

オンライン問題を交互手番の展開型ゲームとして定式化する

リクエストアンサーゲーム $(\mathcal{R}, \mathcal{A}, \boldsymbol{c})$

- $\mathcal{R}$: リクエスト集合
- $\mathcal{A}$: 応答集合
- $\boldsymbol{c} = (c_1, c_2, \ldots)$: コスト関数 $c_t \colon R^t \times A^t \to \mathbb{R}_+ \cup \{\infty\}$



オンライン問題の形式的定義

オンライン問題を交互手番の展開型ゲームとして定式化する

リクエストアンサーゲーム $(\mathcal{R}, \mathcal{A}, \boldsymbol{c})$

- $\mathcal{R}$: リクエスト集合
- $\mathcal{A}$: 応答集合
- $\boldsymbol{c} = (c_1, c_2, \dots)$: コスト関数 $c_t \colon R^t \times A^t \to \mathbb{R}_+ \cup \{\infty\}$

アルゴリズムの流れ

- **For** $t = 1, 2, \dots$
  - アドバーサリが $r_t \in \mathcal{R}$ をリクエスト
  - アルゴリズムが $a_t \in \mathcal{A}$ で応答（a_t は過去に依存して決める）
  - アドバーサリは続ける (**continue**) か終わる (**break**) かを決める
- アルゴリズムのコストは $c_t(r_1, \dots, r_t; a_1, \dots, a_t)$

決定性アルゴリズムと乱択アルゴリズム

リクエストアンサーゲーム $(\mathcal{R}, \mathcal{A}, \boldsymbol{c})$

- $\mathcal{R}$: リクエスト集合
- $\mathcal{A}$: 応答集合
- $\boldsymbol{c} = (c_1, c_2, \ldots)$: コスト関数 $c_t \colon R^t \times A^t \to \mathbb{R}_+ \cup \{\infty\}$

決定性アルゴリズム ALG

- 関数列 $g_t \colon \mathcal{R}^t \to \mathcal{A}$ $(t = 1, 2, \ldots)$ により定義
- リクエスト列 $\sigma = (r_1, \ldots, r_T)$ に対して
  - アルゴリズムの応答: $\mathrm{ALG}[\sigma] = (a_1, \ldots, a_T)$　　ただし $a_t = g_t(r_1, \ldots, r_t)$
  - アルゴリズムのコスト: $\mathrm{ALG}(\sigma) = c_T(\sigma, \mathrm{ALG}[\sigma])$

ゲーム木の各自分のノードにおいてどの手を出すか決めている

決定性アルゴリズムと乱択アルゴリズム

リクエストアンサーゲーム $(\mathcal{R}, \mathcal{A}, \boldsymbol{c})$

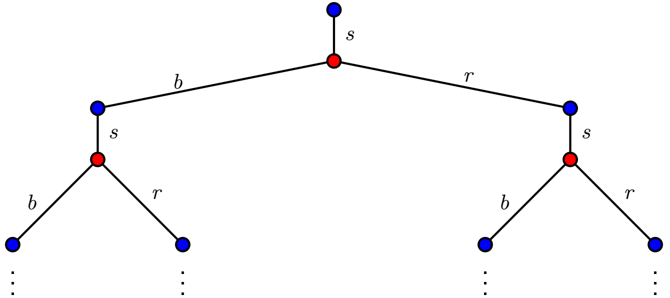
- $\mathcal{R}$: リクエスト集合
- $\mathcal{A}$: 応答集合
- $\boldsymbol{c} = (c_1, c_2, \ldots)$: コスト関数 $c_t \colon R^t \times A^t \to \mathbb{R}_+ \cup \{\infty\}$

乱択アルゴリズム RALG

- 決定性アルゴリズム上の確率分布により定義
- 確率変数 y を用いて ALG_y と表すことにする

例：スキーレンタル問題

$\mathcal{R} = \{\underbrace{s}_{\text{スキーに行く}}\}, \quad \mathcal{A} = \{\underbrace{b}_{\text{購入}}, \underbrace{r}_{\text{レンタル}}\}$



- $c_1(s; b) = 5, \; c_1(s; r) = 1$
- $c_2(ss; b*) = 5, \; c_2(ss; rb) = 6, \; c_2(ss; rr) = 2$
- $c_3(sss; b**) = 5, \; c_3(sss; rb*) = 6, \; c_3(sss; rrb) = 7, \; c_3(sss; rrr) = 3$
- $\vdots$

アウトライン

決定性アルゴリズムに対する競合比

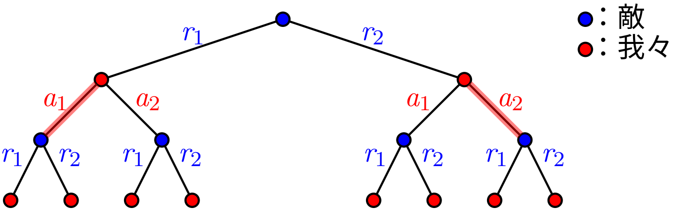
最適オフラインコスト

リクエスト列 $\sigma \in \mathcal{R}^T$ に対して，$\mathrm{OPT}(\sigma) = \min\{c_T(\sigma, \mathbf{a}) \mid \mathbf{a} \in \mathcal{A}^T\}$

決定性アルゴリズムの競合比

$$\rho_{\mathrm{DET}}(\mathrm{ALG}) = \sup_{\sigma} \frac{\mathrm{ALG}(\sigma)}{\mathrm{OPT}(\sigma)}$$

各ノードでのアルゴリズムの選択は固定しているので，決定的な入力列だけを考える

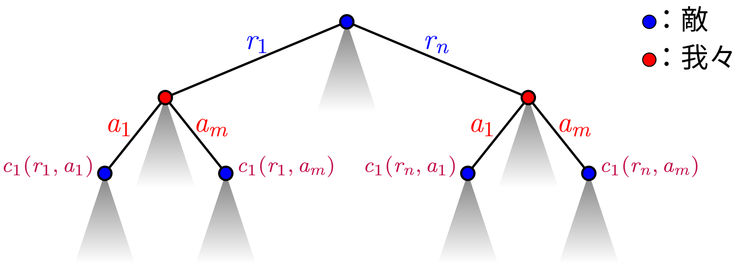


乱択アルゴリズムに対する競合比（オブリビアス）

- リクエストをアルゴリズムの挙動に依存せず決定（深さだけで決定）
- 前回解析した競合比

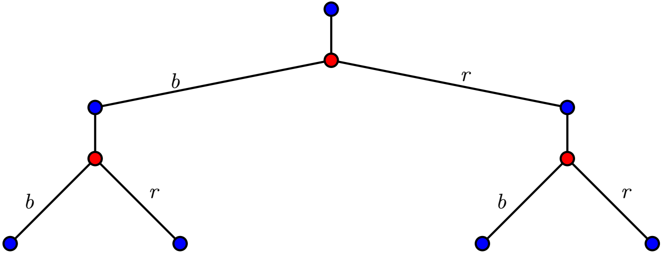
オブリビアスアドバーサリに対する競合比

$$\rho_{\mathrm{OBL}}(\mathrm{RALG}) = \sup_{\sigma} \frac{\mathbb{E}_y[\mathrm{ALG}_y(\sigma)]}{\mathrm{OPT}(\sigma)}$$



乱択アルゴリズムに対する競合比（アダプティブ）

アルゴリズムの選択に依存して，次のリクエストを決定 (乱数の直接観測はしない)



- リクエスト列は確率分布となる
- リクエスト列の確率分布に対する最適オンラインアルゴリズムは最適オフラインアルゴリズムとは異なる
  → 2 通りの比較相手を考えられる

乱択アルゴリズムに対する競合比（アダプティブ）

- リクエスト列をアダプティブに作成（現在のノードに依存して決定）
  - $\boldsymbol{s} = (s_t)_{t=1,2,\ldots}$，$s_t : \mathcal{A}^t \to \mathcal{R} \cup \{\mathrm{STOP}\}$
  - $\mathrm{ALG}_y = (g_t)_{t=1,2,\ldots}$ に対し，$\sigma(y, \boldsymbol{s}) = (r_1, \ldots, r_T)$ と定義

    $a_1 = g_1(r_1), r_2 = s_1(a_1), a_2 = g_2(r_1, r_2), r_2 = s_2(a_1, a_2), \ldots, s_T(a_1, \ldots, a_T) = \mathrm{STOP}$
- リクエスト列が y に依存 ⟶ 2 通りの OPT が考えられる

$\sigma(y, \boldsymbol{s})$ の結果を知った上での最適コストを採用

アダプティブオフラインアドバーサリに対する競合比

$$\rho_{\mathrm{AOFF}}(\mathrm{RALG}) = \sup_{\boldsymbol{s}} \frac{\mathbb{E}_y[\mathrm{ALG}_y(\sigma(y, \boldsymbol{s}))]}{\mathbb{E}_y[\mathrm{OPT}(\sigma(y, \boldsymbol{s}))]}$$

$\sigma(y, \boldsymbol{s})$ の分布のみで定まる最適オンラインアルゴリズムでのコストを採用

アダプティブオンラインアドバーサリに対する競合比

$$\rho_{\mathrm{AON}}(\mathrm{RALG}) = \sup_{\boldsymbol{s}} \frac{\mathbb{E}_y[\mathrm{ALG}_y(\sigma(y, \boldsymbol{s}))]}{\inf_{\mathrm{ALG}'} \mathbb{E}_y[\mathrm{ALG}'(\sigma(y, \boldsymbol{s}))]}$$

アダプティブオンラインとオフラインの違い

$k+1$ 種類のページしか存在しないページング問題

フォルトしたときに一様ランダムにページを消去するアルゴリズムを考える

- アダプティブアドバーサリは毎回フォルトを起こす要求ができる
  消去したページをリクエストすれば良い
- このアドバーサリに付随する最適オフラインアルゴリズムは
  将来の要求がもっとも遅いページを削除によって得られる
  ⟶ 前回の解析より $(k+1)H_{k+1}$ 回に 1 回フォルト
- このアドバーサリに付随する最適オンラインアルゴリズムは
  毎回一様ランダムにページ要求があるということだけ知っている
  ⟶ $k+1$ 回に 1 回フォルト

アドバーサリについて追加コメント

- オブリビアスアドバーサリに対して，「最適オフラインアルゴリズム」の代わりに「最適オンラインアルゴリズム」を考えても意味はない
- 各アドバーサリに乱数を使うことを許しても強さは変わらない

  AON の場合は少し面倒だが，競合比版 Yao の原理の証明と似たような議論により示すことができる
- 多くの論文ではオブリビアスアドバーサリを仮定している

  どのような状況をモデル化したかによって使い分けるべき

アウトライン

競合比の関係

アドバーサリの自由度の高さの関係により以下の関係が成立

定理

任意の乱択アルゴリズム RALG について

$$\rho_{\mathrm{OBL}}(\mathrm{RALG}) \leq \rho_{\mathrm{AON}}(\mathrm{RALG}) \leq \rho_{\mathrm{AOFF}}(\mathrm{RALG})$$

決定性アルゴリズムに対してはどのアドバーサリ同じ強さなので以下が成立

定理

$$\rho_{\mathrm{OBL}} \leq \rho_{\mathrm{AON}} \leq \rho_{\mathrm{AOFF}} \leq \rho_{\mathrm{DET}}$$

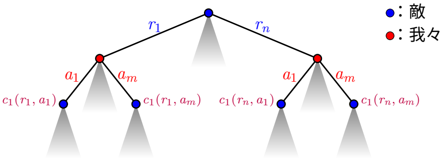
- $\rho_{\mathrm{OBL}}, \rho_{\mathrm{AON}}, \rho_{\mathrm{AOFF}}$: それぞれ対応する競合比の RALG についての下限
- ρ_{DET}: 対応する競合比の ALG についての下限

$\rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}}$

定理

AOFF に対して競合比が α 以下となる乱択アルゴリズムが存在
$\Longrightarrow$ 競合比 α 以下の決定性アルゴリズムが存在

証明:



- 敵の立場で競合比を悪くすることを考える
- その場で STOP すれば競合比 $> \alpha$ となる敵ノードを勝利ノードと呼ぶ
- RALG に対し勝利ノードに必ず辿り着ける
  ➡ 競合比は α より大きい
- 勝利ノードに辿り着けないことがある
  ➡ アルゴリズム側はうまく応答を決めることで勝利ノードを回避可能
  ➡ 競合比 α 以下の決定性アルゴリズム

$\rho_{\mathrm{OBL}}, \rho_{\mathrm{AON}}, \rho_{\mathrm{AOFF}}$ の関係

定理

$$\rho_{\mathrm{AON}} \le \alpha \text{ かつ } \rho_{\mathrm{OBL}} \le \beta \Longrightarrow \rho_{\mathrm{AOFF}} \le \alpha \cdot \beta$$

証明

- $\mathrm{RALG} = \{\mathrm{ALG}_y\}$ について $\rho_{\mathrm{AON}}(\mathrm{RALG}) \le \alpha$ と仮定
  $\Longrightarrow \mathbb{E}_y[\mathrm{ALG}_y(\sigma(y, \boldsymbol{q}))] \le \alpha \cdot \mathbb{E}_y[\mathrm{ALG}'(\sigma(y, \boldsymbol{q}))]\ (\forall \boldsymbol{q}, \mathrm{ALG}')\ (*)$
- $\mathrm{RALG}' = \{\mathrm{ALG}_z\}$ について $\rho_{\mathrm{OBL}}(\mathrm{RALG}') \le \beta$ と仮定
  $\Longrightarrow \mathbb{E}_z[\mathrm{ALG}_z(\sigma)] \le \beta \cdot \mathrm{OPT}(\sigma)\ (\forall \sigma)$
- $(*)$ の ALG' として RALG' を使用すると

$$\begin{aligned}
\mathbb{E}_y[\mathrm{ALG}_y(\sigma(y, \boldsymbol{q}))] &\le \alpha \cdot \mathbb{E}_z[\mathbb{E}_y[\mathrm{ALG}_z(\sigma(y, \boldsymbol{q}))]] \\
&= \alpha \cdot \mathbb{E}_y[\mathbb{E}_z[\mathrm{ALG}_z(\sigma(y, \boldsymbol{q}))]] \\
&\le \alpha\beta \cdot \mathbb{E}_y[\mathrm{OPT}(\sigma(y, \boldsymbol{q}))]
\end{aligned}$$

$y, \boldsymbol{q}$ を固定した時，$\sigma(y, \boldsymbol{q})$ は一意に決まる

$\rho_{\mathrm{OBL}}, \rho_{\mathrm{AON}}, \rho_{\mathrm{AOFF}}$ の関係

定理

$$\rho_{\mathrm{AON}} \leq \alpha \text{ かつ } \rho_{\mathrm{OBL}} \leq \beta \Longrightarrow \rho_{\mathrm{AOFF}} \leq \alpha \cdot \beta$$

系

$$\rho_{\mathrm{AON}} \leq \alpha \Longrightarrow \rho_{\mathrm{AOFF}} \leq \alpha^2$$

$\rho_{\mathrm{OBL}} \leq \rho_{\mathrm{AON}} \leq \alpha$ より成立

競合比の関係まとめ

- $\rho_{\mathrm{OBL}} \leq \rho_{\mathrm{AON}} \leq \rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}}$
- $\rho_{\mathrm{AOFF}} \leq \rho_{\mathrm{OBL}} \cdot \rho_{\mathrm{AON}}$
- $\rho_{\mathrm{AOFF}} \leq \rho_{\mathrm{AON}}^2$

競合比の関係の応用例

ページング問題について

- $\rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}} = k$
- $\rho_{\mathrm{OBL}} = \Theta(\log k)$

⟶ $\rho_{\mathrm{AON}} \geq \rho_{\mathrm{AOFF}} / \rho_{\mathrm{OBL}} = \Omega(k / \log k)$

競合比の関係まとめ

- $\rho_{\mathrm{OBL}} \le \rho_{\mathrm{AON}} \le \rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}}$
- $\rho_{\mathrm{AOFF}} \le \rho_{\mathrm{OBL}} \cdot \rho_{\mathrm{AON}}$
- $\rho_{\mathrm{AOFF}} \le \rho_{\mathrm{AON}}^2$

競合比の関係の応用例

ページング問題について

- $\rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}} = k$
- $\rho_{\mathrm{OBL}} = \Theta(\log k)$

$\longrightarrow$ $\rho_{\mathrm{AON}} \ge \rho_{\mathrm{AOFF}} / \rho_{\mathrm{OBL}} = \Omega(k/\log k)$

アウトライン

資源拡大モデル

- 比較相手として全てを知っているのは強すぎる
- アドバーサリにハンデをつけてアルゴリズムの評価指標を緩くする
- 資源拡大モデルの例
  - ページング問題におけるキープできるページの数
  - k サーバー問題におけるサーバー数
  - スケジューリング問題におけるマシンの処理速度や台数
  - ナップサック問題におけるナップサックの容量
  - ビンパッキング問題におけるビンの容量

アドバイス複雑度

- 指定された解の性能（競合比 c など）を達成するために
必要なアドバイスのサイズ（どれだけの情報が欠けているか）を解析
- b ビットのアドバイスがあれば，2^b 通りの決定性アルゴリズムの中で
ベストなものを選ぶことができる
- スキーレンタル問題の場合は 1 ビットあれば最適値を達成可能
初日に買うべきかどうかをアドバイスすれば良い

アウトライン

1 リクエストアンサーゲーム

2 アドバーサリモデル

3 競合比の関係

4 最悪ケースを越えて

5 まとめ

まとめ

- リクエストアンサーゲームによりオンライン問題を数学的に定式化
- 乱択アルゴリズムに対する 3 種類のアドバーサリ（敵）を定義
  - オブリビアスアドバーサリ (OBL)
  - アダプティブオンラインアドバーサリ (AON)
  - アダプティブオフラインアドバーサリ (AOFF)
- アドバーサリの間の関係を証明
  - $\rho_{\mathrm{OBL}} \leq \rho_{\mathrm{AON}} \leq \rho_{\mathrm{AOFF}} = \rho_{\mathrm{DET}}$
  - $\rho_{\mathrm{AOFF}} \leq \rho_{\mathrm{OBL}} \cdot \rho_{\mathrm{AON}} \leq \rho_{\mathrm{AON}}^2$