

プライシング・アルゴリズムが 市場に与える影響*

加 藤 浩

1. イントロダクション

巨大なオンライン・プラットフォームの興隆に歩を合わせるかのように、プライシング・アルゴリズムの利用が進んでいる。プラットフォームを初めとするいくつかの市場では、マシンが価格決定のエージェントとなり、人間に代わって商品の価格付けをしている。マシンには価格決定の手続きが記述されたプライシング・アルゴリズムがプログラムされており、状況に適した価格を自動的に提示する。巨大プラットフォームでは、多数の売り手と買い手が参加し、膨大な数の商品が取引されている。膨大な数の商品それぞれについて、売り手と買い手をマッチングさせるように価格を定めるには、マシンが持つ高速な演算能力に頼らざるを得ない。e コマースの Amazon、民泊市場における Airbnb、個人タクシー市場の Uber は巨大プラットフォームの代表的な企業であり、それぞれ独自のプライシング・アルゴリズムを採用している。売り手と買い手のマッチングを実現する価格付けをすることは、プラットフォームが提供する価値あるサービスの1つである¹⁾。

プライシング・アルゴリズムの利用に関して一日の長にあるのが、ホテル、飛行機、球場、劇場の市場である。これの市場の特徴は、キャパシティ（座席

* 本稿は、九州経済法研究会第114回例会（公正取引委員会事務総局九州事務所、下関市立大学佐藤隆准教授との共同報告）で報告した『アルゴリズムとデジタルカルテル』をもとに、論文として書き下ろしたものである。なお、当会の研究報告では、加藤が理論分析、佐藤准教授が事例研究を担当した。

数や部屋数)が固定されており、期日(宿泊日, フライト時間, 上演日)が過ぎたらもうサービスを得ることができないという意味で、すぐに腐るサービス(perishable services)が販売されている点である。これらの分野では、コンピュータを用いたイールド・マネジメント(あるいはレベニュー・マネジメント)と呼ばれる手法で、収入が最大になるように価格が決定されている²⁾。アルゴリズムは、顧客のタイプに合わせて異なる価格(価格差別)を提示したり、時間によって異なる価格(ダイナミック・プライシング)を提示したりする³⁾。

最近、プライシング・アルゴリズムに人工知能を搭載し、膨大かつ多様なデータ(すなわちビッグ・データ)をアルゴリズムに投入することで、より緻密な価格付けを実現している。人工知能の具体的なタイプは自己学習であり、学習アルゴリズムをプライシング・アルゴリズムと組み合わせている。経験や実験から獲得した学習の成果を、予測や最適化に反映させることで、需要をより正確に予測し、利潤を最大にする価格を高速に計算できるようになった。人工知能を搭載したプライシング・アルゴリズムを使用することは、売り手に競争優位を与えるから、プラットフォーム以外の従来の市場でも、その利用が促進されることが期待される。

本稿では、人工知能が搭載されたプライシング・アルゴリズムとはどのようなものかを解説した後、このようなアルゴリズムが産業規模で用いられると、

- 1) Amazon マーケット・プレイスでは、価格をアルゴリズムによって決めるか、あるいは手作業で決めるかは、出品者に委ねている。Amazon が提供するアルゴリズムは、出品されている商品の最低価格に追随するプライス・マッチングの性質を持つ。Uber のアルゴリズムは、配車の需給を地域ごとに考慮して料金を決める。現在の需要量が供給量を超えるとき、サージ料金と呼ばれる料金の大幅引き上げを行う。Airbnb ではホストが料金を指定するが、物件の特性を考慮した、ホストとゲストの両者が受け入れる料金をアルゴリズムが算出し、ホストに料金のアドバイスする。
- 2) 短期的にはキャパシティを変えることができず、費用のほとんどがサunkコストであり、キャパシティ未満の供給量で運営しているときは、限界費用はゼロである。したがって、収入を最大化することは利潤を最大化することに等しい。低い価格を提示すると満席になり、それよりも高い価格を付けたなら得られたであろう収入を失う。高い価格だと空席が出てきて、空席だけ収入が失われる。詳細については、Phillips (2005)、Shy (2008)を参照せよ。
- 3) 飛行機の座席をエコノミー・クラス、ビジネス・クラス、ファースト・クラスで分けたり、コンサートの座席をS席、A席、B席で分けたりして、同じサービスに対して異なる料金を提示することは、価格差別である。予約した時期や、閑散期・繁忙期、昼夜によって異なる価格を提示することは、ダイナミック・プライシングである。

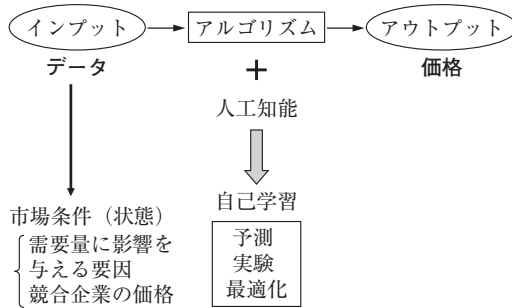


図1 プライシング・アルゴリズムの仕組み

市場構造がどのように変化するか、その結果、競争は促進されるか、制限されるかという問題を、産業組織論の視点から議論していく。

2. プライシング・アルゴリズムの仕組み

プライシング・アルゴリズムは、データをインプットとして入力し、価格をアウトプットとして出力する。より詳細に述べると、リアルタイムの市場データを収集して、分析をすることで、需要量に影響を与える諸要因や競合企業の価格といった市場条件（状態）に関する情報を得て、それをを用いて最適な価格を自動的に計算した上で、顧客にオンラインで価格を提示する（図1）。市場条件の変化はデータに反映され、アルゴリズムに新しいデータが入力されると、価格は自動的に調整される。

アルゴリズムに入力されるデータ量は膨大であり、種類も多様である。市場で実現した取引はデジタル化され、企業のマシンはインターネットを經由して、データとして収集できるようになった。例えば、自社や他社の価格・販売量のデータ、需要に影響を与える変数（季節、曜日、時間、気候等）のデータ、生産・販売・在庫の費用のデータがあり、利潤の計算に用いられる。また、個々の商品に関するデータ、具体的には、製品の在庫、品質、販売日時、販売場所、ユーザー・レビュー、サプライ・チェーンの位置といったデータも、情報技術（IT）の活用により容易に管理できるようになり、商品の需給を均衡させるた

めに活用される。さらには、顧客情報、すなわち、性別、年齢、職業、住所、所得、買い物時間、サイト訪問の頻度、サイトの閲覧履歴、購入履歴といったデータも、eコマースのサイトやIoT（モノのインターネット化）から入手できるようになり、パーソナライズ・プライシング（完全価格差別）を実現するために用いられる。マシンは、リアルタイムでこれらのデータを自動的に収集・処理・分析する⁴⁾。リアルタイムのデータに加えて、過去のデータ、さらには第三者が保有するデータなど、多様なデータを収集し、これらを融合（data fusion）させることで、ノイズを除去し、データが持つ情報としての本質的な部分を抽出する。これらの情報を用いて、ある事象の起こりやすさをより正確に予想することで、効果的に価格を決定する。

プライシング・アルゴリズムは、次の3つのタイプに分類される（Oxera (2017)）⁵⁾。

- heuristic タイプ：入手したデータから市場条件（状態）に関する情報を得て、単純なルールに従って価格を提示する⁶⁾。
- analytical タイプ：過去のデータを収集して統計分析を行うことで需要量との因果関係を見出し、そこから導かれたルールにより価格を提示する⁷⁾。プライシング・アルゴリズムは固定的で更新されることがないので、静学的なアルゴリズムである。
- autonomous タイプ：最初は過去のデータに基づいて価格を決めるが、継続的にパフォーマンスが評価され、観察された需要量や利潤をもとにアルゴリ

4) データが連続的に収集できるので、高頻度データ（high-frequency data）となる。

5) 実際のプライシング・アルゴリズムは、この3つのタイプがミックスされているため、はっきりと分類することは困難である。

6) 最も単純なルールには、競合企業の提示価格のうち最低水準に合わせるというプライス・マッチングや、ターゲットとなる競合企業の価格を k 倍した価格を提示するといったものがある。

7) 例えば、現在の販売量が過去の平均販売量よりも少ないというデータを得たとき、競合企業がより低い価格を付けていると推定し、アルゴリズムは自動的に価格を下げる。

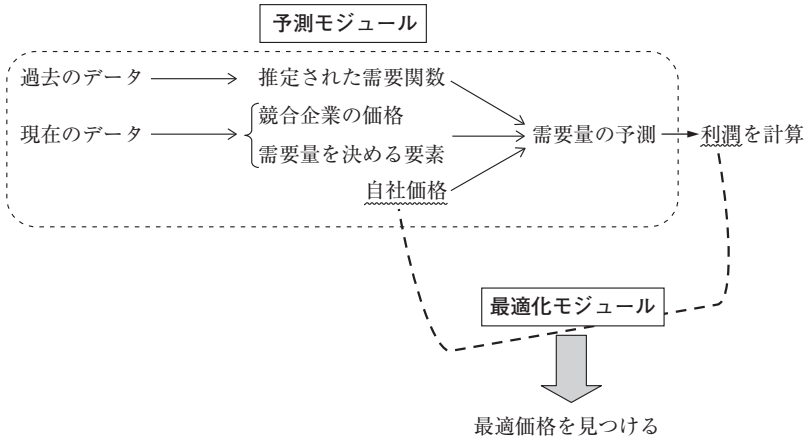


図2 予測モジュールと最適化モジュール

ズムが更新される。マシンが自らプライシング・アルゴリズムを更新するという点で、動的なアルゴリズムである。

学習アルゴリズムを備えたプライシング・アルゴリズムは、autonomousタイプになる⁸⁾。マシンには学習方法を教えるだけで、詳細な価格付けのルールはプログラムに記述しない。マシンは、実験とデータ収集を繰り返し行うことで市場環境を学習して、プライシング・アルゴリズムの更新に反映させる。学習アルゴリズムは、予測モジュールと最適化モジュールで構成される（図2）。

• 予測モジュール

過去に発生した市場条件と、そのときに提示した価格、またそのときに実現した需要量や利潤（＝パフォーマンス）をデータとして収集して、データ処理を通じて需要関数（＝市場環境）を推定する。これを踏まえて、市場条件に関するリアルタイムのデータを収集し、推定された需要関数から、提示価格に対してどのような需要量が実現するかを予測する。

8) プライシング・アルゴリズムは、自律エージェント、知的エージェントとなる。

- 最適化モジュール

候補となるいくつかの価格について⁹⁾、予測モジュールから需要量を導き出し、利潤の計算に用いる。また、競合企業が現在提示している価格をデータとして収集する。これらの需要量の予測と競合企業の価格の情報をを用いて、候補となる価格から得られる利潤の期待値を計算する。そして、算出された期待利潤のうち、最大水準を達成する価格を見つけて出力する。

市場条件として、競合企業の提示価格に焦点を当てるアルゴリズムの場合、データを通して競合企業の提示価格を観察する。あるいは、過去に実現した全企業の提示価格の組み合わせ（価格プロファイル）がデータとして蓄積されているので、これまでの学習成果と合わせて、競合企業が提示する価格を予測する。競合企業の価格に対して、どのような価格で反応すると最大の利潤がもたらされるか、すなわち最適反応価格を計算する。この計算は、これまでの学習成果から自動的に実行されるので、起こりうるあらゆる競合企業の提示価格に対する最適反応を、事前にプログラムする必要はない。

具体例として、過去のデータから価格設定を学習するという、単純なアルゴリズムを考える。ここで、需要量を決定する要因のうち、需要関数のような定常的な要因を市場環境（market environment）と呼び、つねに変化する要因（所得、気候、曜日、競合企業の価格など）を市場条件（market conditions）あるいは状態（states）と呼ぶことにする。

- 予測モジュール

過去のデータから需要関数（=市場環境）を推定する。企業 i は、次のようなパラメータ化された需要関数をデザインする。

$$q_i = \alpha - \beta p_i + \gamma p_j + \kappa_1 x_1 + \kappa_2 x_2 + \dots + \kappa_N x_N. \quad (1)$$

q_i は需要量、 p_i は自社の提示価格、 p_j は競合企業の提示価格である。また、 x_1 ,

9) 価格集合の探索は見込みのある領域に集中し、モデル設計者の判断によらないため、効率的に最適解が導かれる。

```

input  $p[1], \dots, p[M], q[1], \dots, q[M]$  /* 候補となる価格と, その需要量の予測値を入力*/
input  $c, F$  /* 限界費用と固定費用を入力*/
 $\pi^* \leftarrow 0$  /* 初期化*/

/* 候補となる価格についてのメイン・ループ*/
for  $l = 1, \dots, M$ 
  if  $(p[l] - c)q[l] - F \geq \pi^*$  then /* より高い利潤をもたらす価格を発見した*/
     $\pi^* \leftarrow (p[l] - c)q[l] - F$ 
     $p^* \leftarrow p[l]$ 
  end if
end for

if  $\pi^* \geq 0$  then
  output  $p^*$  /* 利潤を最大化する価格を出力*/
else
  print “利潤は負です。この市場からの撤退を考慮してください!”
end if

```

表3 競合企業の提示価格を考慮に入れないときの最適価格の探索アルゴリズム

x_2, \dots, x_N は, 需要に影響を与える非価格要素 (選好, 所得, 気候, 曜日など) である。過去のデータ $(q_i, p_i, p_j, x_1, x_2, \dots, x_N)$ を収集し, 需要関数に当てはめ, 回帰分析等によりフィットするパラメータ $\alpha, \beta, \gamma, \kappa_1, \kappa_2, \dots, \kappa_N$ を求めて需要関数を推定する¹⁰⁾。 $s = (p_j, x_1, x_2, \dots, x_N)$ が市場条件 (状態) であり, リアルタイムのデータを収集することで, 現時点で実現している市場条件 s の値が判明する。そして, 候補となる価格 p_i を推定された需要関数に代入すると, 需要量 q_i の予測が導き出される。

• 最適化モジュール

需要量の予測や競合企業の提示価格に基づいて最適な価格を計算する。市場構造や販売戦略の違いにより, 価格決定のアルゴリズムは異なってくるが, どのような状況であれ, 利潤を最大化にする価格が探索される。Shy (2008) には, 様々な販売戦略に対応した価格決定アルゴリズムが掲載されている。その中でも素朴なアルゴリズムを2つ紹介する。表3は, 競合企業の価格は考慮せ

10) 機械学習のカテゴリーのうち, 「教師あり学習」に相当し, $(q_i, p_i, p_j, x_1, x_2, \dots, x_N)$ が教師データとなる。

```

input   $Q[1]$  /* 現在の実現している需要量のデータを入力 */
input   $P[2], \dots, P[N], Q[2], \dots, Q[N]$  /* 競合企業の価格と需要量のデータを入力 */
input   $\sigma$  /* スイッチング・コストを入力 */
input   $c, F$  /* 限界費用と固定費用を入力 */

if  $P[2] > P[N] + \sigma$  then
  print “データは整合的ではありません。停止します!”
end if

for  $j = 1, \dots, N - 1$ 
  if  $P[j] < P[j + 1]$  then
    print “停止します。企業を正しく並び替えてください。”
  end if
end for

 $p \leftarrow P[N] + \sigma$  /* 需要量  $Q[1]$  を維持できる最高の価格から始める */
 $\pi \leftarrow (p - c)Q[1] - F$ 

/* 初期化 */
 $j \leftarrow 1$ 
 $q \leftarrow Q[1]$ 

/* 価格を徐々に引き下げるためのメイン・ループ */
while  $(P[j] - \sigma > c$  and  $j < N)$  /* 価格を引き下げても利益が出る */
   $j \leftarrow j + 1$ 
   $q \leftarrow q + Q[j]$  /* 企業  $j$  からスイッチした顧客を加える */

  while  $(P[j] = P[j + 1]$  and  $j < N)$  /* 競合企業 2 社の価格が同じときの扱い */
     $j \leftarrow j + 1$ 
     $q \leftarrow q + Q[j]$  /* 企業  $j + 1$  からスイッチした顧客をさらに加える */
  end while

  if  $\pi < (P[j] - \sigma - c)q - F$  then /* 価格引き下げが利潤を増加させる */
     $p \leftarrow P[j] - \sigma$ 
     $\pi \leftarrow (P[j] - \sigma - c)q - F$ 
  end if
end while

output   $p$  /* 価格を出力 */

```

表 4 競合企業の提示価格を考慮に入れたときの最適価格の探索アルゴリズム

ず、候補となる価格と、そのときの需要量の予測の組み合わせをデータとして入力し、利潤が最大となる価格を出力するアルゴリズムである。

表 4 は、 N 企業が活動している市場で、競合企業から顧客を奪い取るように価格を決めるアルゴリズムである。したがって、競合企業の提示価格をデー

タとして入力する。さらに、スイッチング・コストも考慮してデータに入力する¹¹⁾。自社を企業1とする。企業 j ($j=2, \dots, N$) の提示価格は $P[j]$ であり、需要量は $Q[j]$ である。 $P[2] \geq \dots \geq P[N]$ となるように、企業のインデックスを並び替える。スイッチング・コストを σ とする¹²⁾。企業 j で購入し続けるときの消費者のコストは $P[j]$ 、企業1へスイッチするときのコストは $P[1] + \sigma$ である。ゆえに、 $P[1] < P[j] - \sigma$ と価格を提示すると、企業 j のすべての顧客が企業1へスイッチする。これを $j=2, \dots, N$ について計算し、利潤が減少しない限り次々と価格を引き下げる。

競合企業の価格改定といった市場条件の変化は、データに反映される。新しいデータを入手すると、需要パラメータが再度推定され、提示価格の再最適化がなされ、価格は自動的に調整される。このような価格の調整が高頻度かつ高速に実行される¹³⁾。

学習は継続的に行われ、時間を経るたびにアルゴリズムは進化し、予測や最適化の精度が向上する。投入されるデータが膨大かつ多様かつ高頻度なもの（＝ビッグ・データ）であるほど、アルゴリズムに豊富な学習機会を与え、緻密かつ効果的な価格付けが実現する。

3. プライシング・アルゴリズムの定式化

この節では、複占企業がベルトラン競争を繰り返しプレイするゲームを考える。複占企業のうち1企業が、学習アルゴリズムを搭載したプライシング・アルゴリズムを用いて価格を決定する。ここでは、代表的な3つの学習アルゴリズムを紹介する。

-
- 11) 商品の購入先を、現在の企業から別の企業へスイッチしたときに、消費者が負担するコストである。製品差別化が大きいほど、ブランド・ロイヤリティに逆らう購入行動をとるので、スイッチング・コストが大きくなる。
 - 12) すべての消費者について、また、どの企業へスイッチしても、スイッチング・コストはすべて同じ水準であるとする。
 - 13) 需要の変化に反応して速やかに価格を調整させるアルゴリズムは、ダイナミック・プライシング・アルゴリズムと呼ばれる。

3.1 有限オートマトン

(1) 有限オートマトン

heuristic タイプのプライシング・アルゴリズムは、有限オートマトンで表現することができる。有限オートマトン $m = (S, s_0, f, \tau)$ は、以下の4つの要素から構成される。

- i) 状態集合 S
- ii) 初期状態 $s_0 \in S$
- iii) 出力関数 $f: S \rightarrow P$
- iv) 状態推移関数 $\tau: S \times P \times P \rightarrow S$

第 t 期の状態とは、初期から第 t 期までに生じた事象に関して、有限オートマトンが持つ記憶のことであり、プレイの履歴を要約する変数である。 S は有限集合であるとする。 P は提示可能な価格の集合である¹⁴⁾。競合企業の価格集合も P とする。 f に価格付けのルールが記述されており、状態 $s \in S$ が実現すると、有限オートマトンは自動的に価格 $f(s) \in P$ を出力する。今期の状態と今期の価格プロファイルによって、次期の状態が決まる。

繰り返しゲームの戦略は、プレイの履歴 (= 過去の提示価格) に対して、提示すべき価格を対応させる写像である。繰り返しゲームの任意の戦略は、有限オートマトンで表現することができる (Mailath and Samuelson (2006))。また、プライシング・アルゴリズムは有限オートマトンで表現できるので、プライシング・アルゴリズムを繰り返しゲームの戦略と同一視することができる。

表5の利得表で表される囚人のジレンマ型のベルトラン競争を、無限回繰り返すゲームを考える。価格集合は $P = \{p_H, p_L\}$ である。具体的な有限オートマトンを以下に示す。

- トリガー戦略を表現する有限オートマトン $m_1 = (S, s_0, f, \tau)$
 - i) 状態集合 $S = \{(p_H, p_H), (p_L, p_L)\}$
 - ii) 初期状態 $s_0 = (p_H, p_H)$

14) 具体的には、商品が売れそうな価格帯 (price point) である。

		企業 <i>j</i>	
		p_H	p_L
企業 <i>i</i>	p_H	π^m, π^m	π^s, π^d
	p_L	π^d, π^s	π^c, π^c

(企業 *i* の利潤, 企業 *j* の利潤)

p_H = 共謀価格
 p_L = 競争価格
 $\pi^s < \pi^c < \pi^m < \pi^d$

表5 囚人のジレンマ型のベルトラン競争

iii) 出力関数 $\begin{cases} f((p_H, p_H)) = p_H \\ f((p_L, p_L)) = p_L \end{cases}$

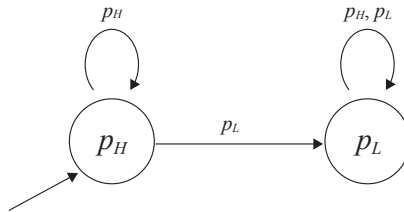
iv) 状態推移関数 $\tau(s, p_i, p_j) = \begin{cases} (p_H, p_H) & (s = (p_H, p_H) \text{ かつ } (p_i, p_j) = (p_H, p_H) \text{ のとき}) \\ (p_L, p_L) & (\text{それ以外}) \end{cases}$

前期に企業 *i* が提示した価格 p_i , および企業 *j* が提示した価格 p_j の価格プロフィール (p_i, p_j) が今期の状態である。 (p_H, p_H) は共謀（協調）が起きている状態、 (p_L, p_L) は競争が起きている状態である（図6）。

• しっぺ返し戦略を表現する有限オートマトン $m_2 = (S, s_0, f, \tau)$

i) 状態集合 $S = \{(p_H, p_H), (p_H, p_L), (p_L, p_H), (p_L, p_L)\}$

ii) 初期状態 $s_0 = (p_H, p_H)$



○ は状態を表す。○ 中の価格は、その状態から出力される価格である。
 → は状態の推移を表す。→ 上の価格は、競合企業の価格である。

図6 トリガー戦略と同値の有限オートマトン

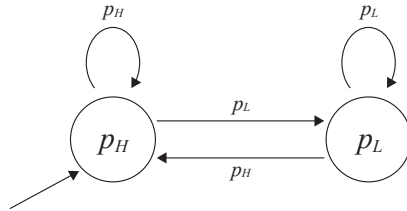


図7 しつぺ返し戦略と同値の有限オートマトン

- iii) 出力関数 $f((p_i, p_j)) = p_j$
- iv) 状態の推移関数 $\tau(s, p_i, p_j) = (p_j, p_i)$

マシンが学習をするときは次のように考える。マシンは複数のオートマトンから、1つをある確率で選び価格付けに用いる。以前と異なるオートマトンが選ばれたとき、プライシング・アルゴリズムが更新されたと解釈する。マシンは採用されたオートマトンのパフォーマンスを評価し、パフォーマンスの良いオートマトンほど採用確率が高くなるように調整される。パフォーマンスの良いオートマトンを見つけても、わずかな確率であるが、実験的に他のオートマトンを採用して、そのパフォーマンスを計測する。

学習モデルとして attraction-based 学習を考える (Camerer and Ho (1999), Hanaki et al.(2005))。実行可能なオートマトンの集合を M とすると、 M に含まれるすべてのオートマトンに誘引値 (attraction value) が割り振られる。オートマトン $m \in M$ の第 t 期の誘引値を $A_m(t)$ とする。第 t_0 期に m が採用され、第 $t_1 - 1$ 期まで使用されたとする。第 t 期に利潤 $\pi(t)$ 、 $t \in [t_0, t_1 - 1]$ が得られたとき、第 t_1 期のオートマトン m の誘引値は、

$$A_m(t_1) = (1 - \omega)A_m(t_0) + \omega \frac{\pi(t_0) + \dots + \pi(t_1 - 1)}{(t_1 - 1) - t_0} \tag{2}$$

と計算される。右辺第2項は、 $[t_0, t_1 - 1]$ 期間における m のパフォーマンスが評価されており、1期間当たりの平均利潤で計算される。良いパフォーマンスを残したオートマトンの誘引値は高くなる。 $\omega \in (0, 1)$ は、誘引値を更新する

とき、観察されたパフォーマンスにどれぐらい重きを置くかを表すパラメータである。

第 t 期にオートマトン m_0 が採用される確率は、次式で表される。

$$\Pr(m_0) = \frac{e^{\lambda A_{m_0}(t)}}{\sum_{k \in M} e^{\lambda A_k(t)}} \quad (3)$$

λ は、誘引値がオートマトンの採用確率に与える影響を表すパラメータである。

3.2 ニューラル・ネットワーク（深層学習）

深層学習は、過去のデータから自動的に需要関数の特徴抽出（feature extraction）をするというものである。つまり、市場条件のどの要因が需要量の決定に大きな影響を与えるかを見出すことで、需要関数の形を推定する。ニューラル・ネットワークは深層学習の代表的なモデルで、人の脳を模倣したデータ処理アルゴリズムである。膨大な数の疑似神経細胞（ニューロン）と、さらに膨大な数の疑似結合の集合から構成される、複数の隠れ層を持つネットワークである（図8）。データが入力されると、脳に似せた学習プロセスを辿り、需要に影響を与える本質的な特性だけが抽出され、それに基づいて価格が計算される。

企業 i の需要関数を、

$$q_i = \alpha - \beta p_i + \gamma p_j \quad (4)$$

と設定する¹⁵⁾¹⁶⁾。 q_i は企業 i の需要量、 p_i は企業 i の価格、 p_j は企業 j の価格、 α, β, γ は需要パラメータである。生産費用をゼロとすると、企業 i の利潤は次のようになる。

$$\pi_i = (\alpha - \beta p_i + \gamma p_j) p_i \quad (5)$$

15) 以下は、組織学習について分析した Barra and Saraceno (2005) のモデルに依拠して議論する。

16) (1) 式の $\kappa_{1x_1} + \kappa_{2x_2} + \dots + \kappa_{Mx_M}$ を α に収める。

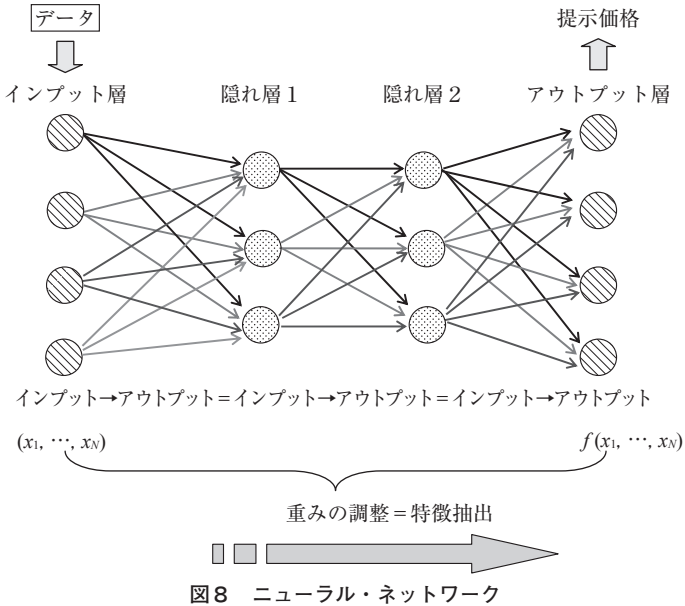


図8 ニューラル・ネットワーク

需要パラメータ α, β, γ の値が判明しているとき、企業 j の提示価格 \hat{p}_j に対する企業 i の最適反応関数は、以下の式で表される。

$$p_i = \frac{\alpha + \gamma \hat{p}_j}{2\beta}. \tag{6}$$

環境変数とは、需要関数 (= 市場環境) を決定する特性がデジタル化されたもので、マシンはデータとして受け取ることができる。ここでは、環境変数を需要パラメータ α, β, γ を決定する変数とする。環境を X 、環境変数を $\mathbf{x} \in X$ とする。毎期間、環境変数 \mathbf{x} が X からランダムに抽出され、 α, β, γ の値を決める。各期に実現した環境変数 \mathbf{x} を、その期の市場条件と呼ぶ。ここで、

$$\mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} \tag{7}$$

とする。下付き文字 $n=1, \dots, N$ は、需要パラメータに影響を与える各要因（環境特性）を表しており、 x_1, \dots, x_N はそれぞれ2進数である¹⁷⁾。 $x_n=0$ のときは、実現した市場条件に特性 n は存在しないことを意味し、 $x_n=1$ のときは存在することを意味する。市場条件 x と需要パラメータの（真の）関係を次のように設定する。

$$\alpha(x) = d_\alpha \left(\sum_{n=1}^N n a_n x_n \right)^2, \quad (8)$$

$$\beta(x) = d_\beta \sum_{n=1}^N n b_n x_n, \quad (9)$$

$$\gamma(x) = d_\gamma \sum_{n=1}^N n c_n x_n. \quad (10)$$

ただし、 $a_n, b_n, c_n \in (0, 1)$ であり、 $d_\alpha, d_\beta, d_\gamma$ は $\alpha(x), \beta(x), \gamma(x) \in [0, 1]$ と正規化する定数である。データ x の第 n ビット目 x_n が、需要パラメータ $\beta(x)$ に与える限界的貢献は $nb_n \Delta x_n = nb_n$ である。したがって、 $n=1, \dots, N$ は、需要パラメータに影響を与える重要度の順に特性を並べたものであり、 x_1 が最も重要度が低い特性であり、 x_N が最も重要度が高い特性となる。

マシンは(8)、(9)、(10)式で表わされる関係が分からないため、市場条件が需要パラメータにどう影響するかを繰り返し学習する。ただし、競合企業の提示価格 p_j のデータを学習に使わないものとする¹⁸⁾。

インプット層、隠れ層 h 、アウトプット層 o の3層から構成される、3層ニューラル・ネットワーク・モデルを考える（図9）。各層のニューロンはデータ処理単位であり、同じ層では並行してデータ処理が行われる。神経結合を經由して、各ニューロンから次の層のニューロンにデータが送られる。複数のニューロンから送られたデータは、重みを付けて足し合わされた上で、次の層の各ニューロンに送られる。重みは神経結合の強度を表しており、重要な

17) 1ビットの情報量を持つ変数である。 x は0と1を並べたもので、 N ビットの情報量を持つ。

18) 競合企業の価格はデータ x から予測し、利潤を計算するときのみに利用される。したがって、重みを調整するときに競合企業の価格が影響する。

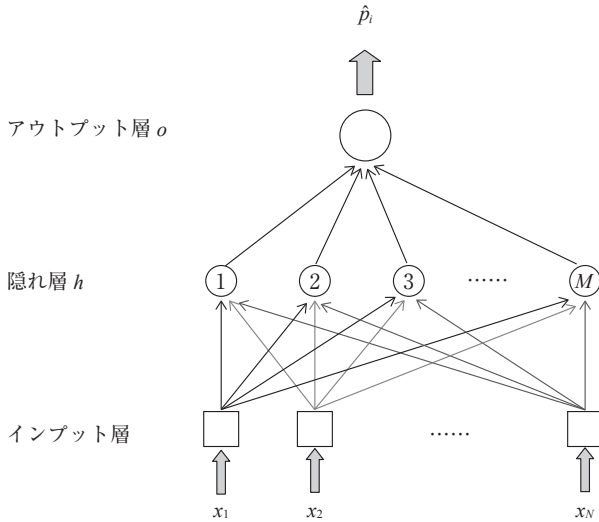


図9 3層ニューラル・ネットワーク・モデル

データほど重みが大きくなる。

入力層にデータ \mathbf{x} が入力された後、入力層の N 個のニューロンから、隠れ層 h の M 個にニューロンにデータが送られる。隠れ層 h の l 番目 ($l=1, \dots, M$) のニューロンは、重み

$$\mathbf{w}_l^h = (w_{l1}^h, \dots, w_{lN}^h) \tag{11}$$

の加重平均

$$y_l^h = \mathbf{w}_l^h \cdot \mathbf{x} = w_{l1}^h x_1 + \dots + w_{lN}^h x_N \tag{12}$$

をインプットとして受け取る。 y_l^h をシグモイド関数

$$g(y) = \frac{1}{1 + e^{-y}} \tag{13}$$

に代入することで、隠れ層の l 番目のニューロンから z_l^h が出力される¹⁹⁾。シグモイド関数は $0 \sim 1$ の値をとり²⁰⁾、値の大きさはニューロンの活性度を表わし

ており、受け取ったデータ y_i^h がどれくらい重要な情報であることを示す。

$$z_i^h = g(y_i^h) \tag{14}$$

であるから、隠れ層から

$$\mathbf{z}^h = \begin{pmatrix} z_1^h \\ \vdots \\ z_M^h \end{pmatrix} = \begin{pmatrix} g(y_1^h) \\ \vdots \\ g(y_M^h) \end{pmatrix} \tag{15}$$

が出力される。神経結合を經由すると、重み

$$\mathbf{w}^o = (w_1^o, \dots, w_M^o) \tag{16}$$

が加わり、加重平均

$$y^o = \mathbf{w}^o \cdot \mathbf{z}^h = w_1^o z_1^h + \dots + w_M^o z_M^h \tag{17}$$

をアウトプット層はインプットとして受け取る。これをシグモイド関数に代入した値

$$\hat{p}_i = g(y^o) = g\left(\sum_{l=1}^M w_l^o g(\mathbf{w}_l^h \cdot \mathbf{x})\right) \equiv f(\mathbf{x}) \tag{18}$$

が、アルゴリズムの出力する価格である。

いま、企業 j は価格 \hat{p}_j を提示している。需要パラメータの値が判明しているとき、最適反応関数(6)式に従って価格を決めると、企業 i の価格 (= 正解値) は、

- 19) シグモイド関数は活性化関数の1つである。活性化関数は、加重平均 y_i^h を0, 1の値に対応させる関数である。値が0のときはニューロンはデータを無視し、情報として取るに足らないものとする。値が1のときはニューロンが反応(発火)をし、データを重要な情報として扱う。シグモイド関数は活性化関数の性質を保ちつつ、値を0から1まで連続的に出力するため、計算が容易になる。
- 20) 入力値が閾値を超えると急激に大きな値を出力するところが、シグモイド関数の特徴である。

$$p_i^* = \frac{\alpha + \hat{p}_i}{2\beta}, \quad (19)$$

利潤は,

$$\pi_i^* = \beta p_i^{*2} \quad (20)$$

となる。この値を理想的な利潤とする。一方、アルゴリズムが出力する価格から得られる利潤 (= パフォーマンス) は,

$$\hat{\pi}_i = \beta(2p_i^* \hat{p}_i - \hat{p}_i^2) \quad (21)$$

となる。アルゴリズムの出力値 \hat{p}_i と正解値 p_i^* の誤差により失われた利潤は,

$$L = \hat{\pi}_i - \pi_i^* = \beta(p_i^* - \hat{p}_i)^2 \quad (22)$$

であり、 L を損失関数と呼ぶ。出力値 \hat{p}_i と正解値 p_i^* の誤差 (2乗誤差) を,

$$\varepsilon = (p_i^* - \hat{p}_i)^2 \quad (23)$$

と置くと,

$$L = \beta\varepsilon, \quad (24)$$

$$\hat{\pi}_i = \pi_i^* - \beta\varepsilon \quad (25)$$

となるから、 $\varepsilon = 0$ のときアルゴリズムが生み出す利潤は理想的な水準となる。

学習アルゴリズムは、誤差逆伝播法 (backward propagation) に従い、プライシング・アルゴリズムを修正する。これは、損失関数 L が最小になるよう重みを調整し、入力・出力とは逆方向に、すなわちアウトプット層から隠れ層に伝播させることで、誤差を減らす方法である。重みを次式に従って調整することで、次期の価格決定におけるパフォーマンスを向上させる。

$$w_i^o(t+1) = w_i^o(t) - \eta \frac{\partial \varepsilon}{\partial w_i^o}, \quad (26)$$

$$w_{in}^h(t+1) = w_{in}^h(t) - \eta \frac{\partial \varepsilon}{\partial w_{in}^h}. \quad (27)$$

t は反復回数, η は学習率である²¹⁾。ここで,

$$\frac{\partial \varepsilon}{\partial w_i^o} = -2(p_i^* - \hat{p}_i) \hat{p}_i (1 - \hat{p}_i) z_i^h, \quad (28)$$

$$\frac{\partial \varepsilon}{\partial w_{in}^h} = -2z_i^h(1 - z_i^h)(p_i^* - \hat{p}_i) \hat{p}_i (1 - \hat{p}_i) w_i^o x_n \quad (29)$$

であるから²²⁾,

$$\phi_i^o = 2(p_i^* - \hat{p}_i) \hat{p}_i (1 - \hat{p}_i), \quad (30)$$

$$\phi_i^h = z_i^h(1 - z_i^h) \phi_i^o w_i^o \quad (31)$$

と置いてやると, 重みの反復式は,

$$w_i^o(t+1) = w_i^o(t) + \eta \phi_i^o z_i^h, \quad (32)$$

$$w_{in}^h(t+1) = w_{in}^h(t) + \eta \phi_i^h x_n \quad (33)$$

となる。重みが収束すると学習は完了し, アルゴリズムはこれ以上更新されない。

3.3 Q 学習 (強化学習)

強化学習は, 様々な価格を実験的に提示し, 需要量や利潤といった市場環境

21) 最急降下法による損失関数の最小化を行っている。関数 $F(\mathbf{x})$ の極小化解を求めるとき, 反復式

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \eta \nabla F(\mathbf{x}^{(k)})$$

に従って計算するものである。ただし, ∇F は F の勾配ベクトルである。

22) (28), (29)式で示されるように, ε の w_i^o に関する勾配を計算した後に, ε の w_{in}^h に関する勾配が計算される。したがって, 重みは入力・出力とは逆方向に調整される。これが誤差逆伝播法である。

からのフィードバックをデータで受け取り、学習するというものである。このような試行錯誤とフィードバックを繰り返して、各状態に対する最高のパフォーマンスを生み出す価格を学習し、アルゴリズムの更新に反映させる。実際に価格を提示してみて、得られた利潤を観察するという具合に、市場環境に直接働きかける学習方法なので、需要関数（＝市場環境）のモデル化と推定は必要としない。

強化学習のうち、よく用いられているモデルがQ学習である。戦略のパフォーマンスを数値で表したものをQ値と呼ぶ。戦略は実現した状態に対して提示価格を対応させるものであるから、戦略を状態と価格の組み合わせで表し、その価値を計算したものがQ値である。様々な状態と価格の組み合わせのQ値をマシンは記憶しており、再度同じ状態と価格が実現したとき、観察された利潤をもとにQ値を更新する。

状態 $s \in S$ 、価格 $p \in P$ に対する第 t 期のQ値を $Q_t(s, p)$ とする。第 t 期の状態 s_t と第 t 期に両企業が提示した価格により、第 $t+1$ 期の状態 s_{t+1} が決まる。第 t 期に状態 s が実現し、アルゴリズムが p を提示し、利潤利 π_t を得たならば、Q値は以下の式に従って更新される。

$$Q_{t+1}(s, p) = (1 - \alpha)Q_t(s, p) + \alpha \left\{ \pi_t + \delta \operatorname{Max}_{\tilde{p} \in P} Q_t(s_{t+1}, \tilde{p}) \right\}. \quad (34)$$

ただし $\alpha \in [0, 1]$ は学習率、 δ は割引率である。右辺第1項は、これまでの経験で得られたQ値である。第2項は、現在の経験が反映されており、次期は実現した状態に対してQ値が最大になるよう価格が選ばれるとする²³⁾。第 t 期に実現しなかった状態 s' 、価格 p' について、Q値は更新されない。つまり、

$$Q_{t+1}(s', p') = Q_t(s', p'). \quad (35)$$

exploitation と exploration という2つの方針のうち、いずれかに従って価格は提示される。exploitation は、現在までの学習を踏まえて最善の価格を提示することである。つまり、状態 s が実現したとき、Q値 $Q_t(s, p)$ を最大化する価

23) $\alpha = 1$ のときは、Q値を評価関数としたベルマン方程式になる。

格 $p^* \in \arg \text{Max } Q(s, p)$ を提示する。これは、現在得る利潤を重視した方針である。一方、exploration（探索）は試行錯誤であり、 p^* よりも良いパフォーマンスを実現する価格を探索するために、様々な価格をある確率で提示するものである。つまり、Q 値を最大にする価格とは異なる価格を実験的に提示して、実現した利潤のデータを収集する。こうして、市場環境に関してより正確な情報を得て、アルゴリズムの更新に反映する。これは、将来同じ状態が生じたとき、より高い利潤を得ることを考慮した方針である。

exploitation と exploration のどちらの方針が取られるかは、確率的に決まるとする。すなわち、状態 s が実現したとき、価格 $p \in P$ が提示される確率を、

$$\Pr(p) = \frac{\exp\left\{\frac{Q(s, p)}{\beta}\right\}}{\sum_{\tilde{p} \in P} \exp\left\{\frac{Q(s, \tilde{p})}{\beta}\right\}} \quad (36)$$

とする²⁴⁾。 $\beta \in (0, 1]$ は実験性向であり、この値が大きいほど実験を行う確率が高くなる。 β が 0 に近づくと、ある状態に対して 1 つの価格が確実に提示される。Q 値が高い価格ほど提示される確率が高くなるものの、Q 値が低い価格もわずかな確率で実験的に提示される。すなわち、exploitation は高い確率で実行されるが、低い確率ではあるが exploration も取られる。 $\beta \rightarrow 0$ のとき exploration は起こらず、アルゴリズムの学習は完了する²⁵⁾。

囚人のジレンマ型のベルトラン競争を無限回繰り返す（表 5）。前期に提示された価格プロファイルを今期の状態とする。したがって、状態集合は、

$$S = \{(p_H, p_H), (p_H, p_L), (p_L, p_H), (p_L, p_L)\}, \quad (37)$$

Q 値の集合は、

$$\{Q((p_H, p_H), p_H), Q((p_H, p_H), p_L), Q((p_H, p_L), p_H), Q((p_H, p_L), p_L), \\ Q((p_L, p_H), p_H), Q((p_L, p_H), p_L), Q((p_L, p_L), p_H), Q((p_L, p_L), p_L)\}$$

24) ボルツマン探索戦略と呼ばれる。

25) $\beta = \beta_0^t$, $0 < \beta_0 < 1$ と置く。 $t \rightarrow \infty$ のとき、 $\beta \rightarrow 0$ となり学習は完了する。

である。状態 $s \in S$ が実現したとき、共謀価格 p_H が提示される確率は、次のようになる。

$$\begin{aligned} \Pr(p_H) &= \frac{\exp\left\{\frac{Q(s, p_H)}{\beta}\right\}}{\exp\left\{\frac{Q(s, p_L)}{\beta}\right\} + \exp\left\{\frac{Q(s, p_H)}{\beta}\right\}} \\ &= \frac{1}{\exp\left\{\frac{Q(s, p_L) - Q(s, p_H)}{\beta}\right\} + 1}. \end{aligned} \tag{38}$$

$\beta \rightarrow 0$ とすると、 $Q(s, p_L) > Q(s, p_H)$ ならば $\Pr(p_H) = 0$ 、 $Q(s, p_H) \geq Q(s, p_L)$ ならば $\Pr(p_H) = 1$ となる。

4. プライシング・アルゴリズムが市場構造に与える影響

4.1 アルゴリズム軍拡競争

市場でプライシング・アルゴリズムを使用する企業が登場すると、アルゴリズム軍拡競争 (algorithm arms race) へと発展する。従来のまま人の手により価格を決めることは、次の3つの点から、価格競争を展開する上で不利となる。まず、価格決定に時間がかかる。市場条件がどのように変化したかを知るために、情報入手しないといけなく、入手した情報に基づいて、価格を改定するかどうかを判断し、改定するならばどれぐらい価格を変化させるかを決めないといけなく。第2に、価格決定に際してバイアスやエラーが発生して、緻密に価格を調整することができない²⁶⁾。最後に、価格改定により、値札を貼りかえたり、消費者や流通チャネルに新しい価格を通達したりすることで、メニュー・コストが発生する。

これに対して、アルゴリズムは、データの収集と処理を通じて迅速に情報を

26) 行動経済学的なバイアスとしては、損失回避、コンコルド効果 (sunk cost fallacy)、フレーミング効果がある。

入手でき、自動化された手続きによって価格を決定するので、市場状態の変化に対して即座に価格を調整することができる。また、アルゴリズムは、直感を排除した綿密な分析に基づいて価格を決定するので、バイアスやエラーがない。さらに、オンラインで価格を提示しているため、デジタル情報を書き換えるだけで、新しい価格を消費者や流通チャンネルに伝えることができる。このように、スピードの速さ、正確さ、コストの低さから、アルゴリズムを使用して価格付けをする企業に競争優位を与え、手作業で価格を決めている企業には、アルゴリズムの使用へシフトするように競争圧力がかかる。これは、オンライン小売業者のみならず、brick-and-mortar 小売業者にも同じ競争圧力がかかる。したがって、オンライン、オフラインに関係なく、産業規模でアルゴリズムが使用され、アルゴリズムの間で価格競争が起こる。

アルゴリズムの提示する価格は頻繁に変動するため、価格をそのまま提示すべきか、あるいは修正すべきかを判断する時間的余裕が人間にはない。それゆえ、人はアルゴリズムが計算した価格に唯々諾々と従うことになる。このことは、アルゴリズムが計算した価格が実際に提示される価格となり、したがって市場価格となることを意味する²⁷⁾。

産業規模でプライシング・アルゴリズムが使用されると、市場構造が次のように変化する。

(1) 市場の透明性 (market transparency) が向上する

市場の透明性は、市場で生じたあらゆる事象（価格、品質、製品特性、生産量、販売量、購入量、購入時点、購入先企業、商品情報、顧客情報など）が、どれくらい明確に観察できるかを表す概念である（図10）。アルゴリズムを用いる企業が増加すると、市場の透明性が向上する。アルゴリズムで決められた価格はデジタル化され、情報フローとして市場に流布する。市場で提示される多くの価格がアルゴリズムで決定されると、情報フローが増大し、企業はデータとしてリアルタイムにアクセスし、価格を観察することができる²⁸⁾。こうし

27) brick-and-mortar 小売で見られるような印刷された価格リストは、秘密の割引があるため実際の価格とかけ離れている可能性がある。

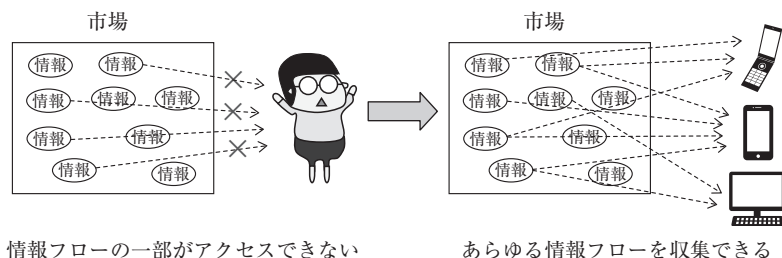


図10 市場の透明性の向上

て市場の透明性が向上する。

産業規模でアルゴリズムが用いられると、市場で提示されているあらゆる価格がデータとしてアクセスでき、どの企業もすべての価格を瞬時かつ正確に観察できる²⁹⁾。それゆえ、価格戦略にも影響を与える。企業は提示した価格を互いに監視することができ、さらに、競合企業に自社の価格をリアルタイムに知らしめ、競合企業の提示する価格を変えることができるからである³⁰⁾。

(2) 企業間の相互作用 (interaction) の頻度が高まる

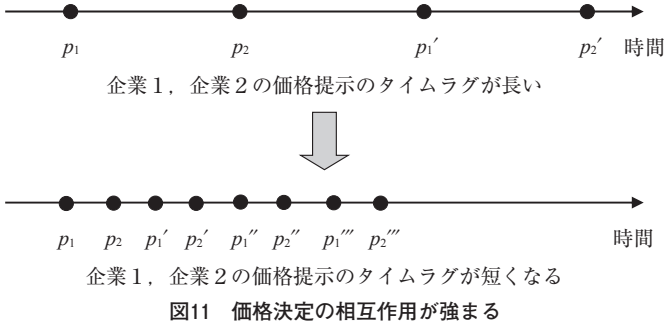
アルゴリズムは、競合企業の価格に対して最適反応となる価格を計算し、自動的に提示する。したがって、すべての企業がアルゴリズムを用いると、競合企業の価格改定に対してマシンは高速で反応し、さらに、この反応価格に対して、競合企業のマシンが高速で反応する。このようにして、企業間で生じる相互作用の頻度が高まる (図11)。それゆえ、価格が変動する要因は、コストの変化によるものではなく、競合企業の価格改定に対する最適反応によるものを中心となる。

手作業で価格を決めるとき、競合企業の価格改定とそれに対する反応との間にラグが発生する。例えば、競合企業の価格情報を知るために、紙に印刷され

28) さらに、価格以外の様々な市場取引もデジタル化され、データとして市場に流通するという「デジタル化された市場環境」が実現する。

29) すべての企業が完全モニタリングの状態にあり、企業の提示する価格は公の情報 (public monitoring) となる。

30) したがって、価格をシグナル、報酬、報復の手段として用いることができる (後述)。



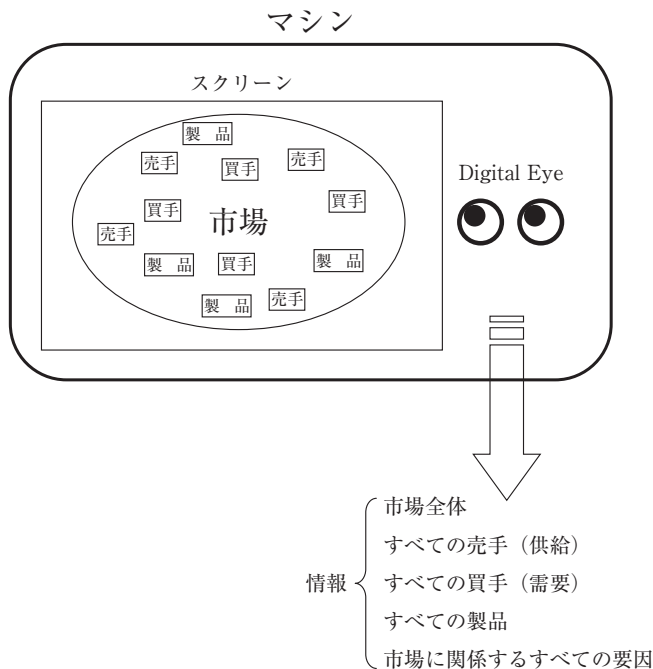
た価格リストを入手するには時間がかかる。その上で最適反応となる価格を計算して、価格リストに印刷して顧客に提示するので、さらに時間がかかる。

4.2 Digital Eye を持つ自律マシン

技術市場では絶えずアルゴリズムの開発競争が繰り広げられて、進化したよりスマートなアルゴリズムが生み出され、商用として普及する。しかし、企業間で同じようにスマートなアルゴリズムを使用しているにもかかわらず、投入されるデータ量やデータの多様性が異なると、アルゴリズムの学習量が異なるため、予測と最適化の能力に差が出てくる。つまり、蓄積しているデータが少ない企業は、学習不足によりアルゴリズムの性能が悪いため、競争上不利になり、最終的には市場から退出せざるを得なくなる³¹⁾。このことから、市場で生き残っている企業はすべて、スマートなアルゴリズムを採用し、かつビッグ・データをインプットとして投入しているため、高度に学習した高性能のプライシング・アルゴリズムで価格を決定していることになる。

このような高性能アルゴリズムを持つマシンは、「Digital Eye を持つ自律マシン」となる。マシンが Digital Eye を持つと、市場の透明性が極致に達して、任意の時点について市場で生じている事象をより明確に、より広範に、より詳細に見渡すことができる (図12)³²⁾。さらに、予測と最適化の精度が最高潮に

31) データを多く蓄積している企業には、ますます多くのデータが流入するというティッピングが生じる。



マシンに内蔵されているスクリーンに、市場の情報がすべて映し出されており、マシンの目（Digital Eye）がそれらを眺めているイメージである

図12 Digital Eyeを持つ自律マシン

達した自律マシンは、競合企業の価格改定を正確に予測し、それに対する最適な価格を瞬時に計算し、ミリ秒単位で反応する。

5. プライシング・アルゴリズムの競争促進効果および競争制限効果

産業規模でアルゴリズムが利用されるとき、競争が促進され、市場価格がよ

32) Ezrahi and Stucke (2016) は、マシンは God View で市場を見渡すと表現している。また、Uber は God View と呼ばれる情報管理システムで、顧客情報とドライバーの位置情報を管理している。

り低い水準になることも、逆に競争が制限され、競争水準以上の価格が均衡として実現することも、いずれの可能性も考えることができる。

5.1 競争促進効果

市場の透明性が向上すると、競合企業が提示する価格をより正確に観察ができ、それをベンチマークとして自社の価格を決定する。また、アルゴリズムは速やかに価格を調整できるので、競合企業が提示した価格にすぐ反応できる。このような要因により競争促進的な反応を引き起こす。具体的に述べると、次のような企業行動により市場は競争的になる。アルゴリズムは競合企業の価格に関するデータを入手し、それよりも価格を引き下げて顧客を奪うことで、利潤を増やす。また、競合企業よりも高い価格を提示していたら、速やかに価格を引き下げること、需要の減少を食い止める³³⁾。逆に、競合企業が価格を引き上げた後、それに追随すると、競合企業はすぐに価格を元の水準に戻してしまい、顧客が奪われ利潤は減少する。よって、競合企業が価格を引き下げると追随するが、価格を引き上げても追随しない³⁴⁾。

人の手で価格を決めるときは、メニュー・コストの存在により価格を頻繁に変えられない。これは、価格の調整にはラグが発生し、価格が硬直的となることを意味する。いったん価格を決定すると、 Δ 時間の間だけ価格は変えられない。すなわち、価格は $p(t), p(t+\Delta), p(t+2\Delta), \dots$ と変動するため、 Δ 時間だけ価格をコミットしていることになる。そこで、価格を引き上げて、競合企業の追随を誘ってみる。 Δ 時間だけ価格を変更できないので、 Δ が大きいほど、この誘いは魅力的になり、競合企業も値上げに追随する。こうして、競争水準よりも高い均衡価格が実現することになる (Maskin and Tirole (1988))。

33) 市場の透明性は消費者行動にも影響を与え、競争を促進させる。市場取引がデジタル化されることで、消費者は価格を容易に検索でき、複数企業の価格を比較する。こうして、消費者のサーチ・コストが低下するので、各企業の価格支配力は低下し、市場価格も低下する (Salop and Stiglitz (1977), Varian (1980), Bakos (1997))。さらに、購入先をクリック1つで変更することができるため、高い価格をつけている企業は、クリック1つでライバル企業に顧客を奪われる。

34) したがって、シグナリング・アルゴリズムが事前にプログラムされており、なおかつ共謀戦略を採用していなければ、価格を引き上げはシグナルとして機能しない (後述)。

5.2 競争制限効果 — アルゴリズムに共謀プログラムが入力されている場合

プライシング・アルゴリズムは、暗黙の共謀（tacit collusion）を容易に実現させるツールとなる。暗黙の共謀とは、企業間で合意（agreement）や直接的なコミュニケーションはないが、先導者が価格を引き上げたときは、それに追従して価格を引き上げることが、また、一方的に値下げをしないことが、利潤最大化を実現する最適な戦略であることを、どの企業も独立に認識している状態である。さらに、競争相手も同様の認識をしていると確信している状態である。これにより、どの企業も競争水準の価格を超える価格（＝共謀価格）を設定するので、明白な共謀（＝カルテル）と同じ結果が市場で観察される。暗黙の共謀が実現するためには、2つの要因が必要である。

(1) 共謀への参加を誘う³⁵⁾

先導者が価格を引き上げて、競合企業に対して共謀への参加（＝追従）を誘い、競合企業の反応を持つ。価格引き上げに追従しないならば、直ちに価格を元の水準に戻す。

以下の条件がすべて実現可能ならば、価格の引き上げが共謀の誘いとして有効な手段となる。i) 競合企業が価格の引き上げをすぐさま観察し、ii) 速やかに反応する。iii) 競合企業が追従しない場合は即座に価格に戻す。

(2) 共謀戦略を採用する

競合企業が追従し共謀価格を提示したならば、その後も共謀価格を維持させなければ暗黙の共謀は崩壊してしまう。暗黙の共謀を安定化させる観点から、競合企業が提示した価格を将来の報酬・報復に結び付け、値下げを断念させ、共謀価格を維持させるように誘導する戦略が採用される。このような、過去の提示価格と因果関係を持つ報酬・報復スキームが組み込まれた戦略を共謀戦略と呼ぶ³⁶⁾。具体的には次のような戦略である。競合企業が共謀価格を維持し続けるならば、自社も共謀価格を提示し続ける（報酬フェーズ）が、ある時点で

35) 有限オートマトンの初期状態を (p_H, p_H) に導くことを意味する。

競合企業が価格を引き下げ、顧客を奪った場合は、報復として自社も価格を引き下げる（報復フェーズ）³⁷⁾。

共謀戦略が、競合企業に逸脱（＝価格の引き下げ）を思いとどまらせて、共謀価格を維持させることができるかどうかは、次のプロセスが、速やかにかつ確実に実行されるかどうかにか依存する。すなわち、i) 競合企業の価格を監視し、ii) 逸脱を発見し、iii) 逸脱者に対して報復する。

産業規模でアルゴリズムが使用されることで、市場の透明性が高まり、企業は互いの提示価格を容易かつ正確に観察することができる。さらに、企業間の相互作用が強まることで、競合企業の提示価格に対する反応のスピードが上昇する。これらは、暗黙の共謀が実現する状況を拡大させる要因である（図13）。

(1) 共謀への参加を誘うシグナルを送る

先導者のアルゴリズムは試みに価格を引き上げて、共謀への参加を誘うシグナルを送る。競合企業のマシンは直ちにシグナルを観察し、すぐ反応するため、シグナルを送るのは数秒だけで済む。競合企業が価格引き上げに追随しないならば、アルゴリズムは価格を即座に戻して、シグナルを撤回することができるので、顧客が競合企業に奪われるリスクがない。また、競合企業の追従が得られなくても、再度シグナルを送る機会がつねにあり、何度でも共謀に誘うことができる。

これに対して、価格が人間の手で決められているときは、価格引き上げをアナウンスする期間を長くとり、競合企業に認識してもらい、さらに反応を待たないといけない。また、秘密の値下げが可能であるため、競合企業の価格リストに共謀価格が書かれていても、実際の販売価格が共謀価格であるかどうかを、

36) 過去の提示価格（＝プレイの履歴）がどのようなものであるかに関係なく価格を決めるとき、1回限りのゲーム（one-shot game）のナッシュ均衡が每期プレイされる。つまり、つねに競争価格を提示する戦略が唯一の均衡戦略となり、暗黙の共謀が実現することはない。

37) 暗黙の共謀を、すべての企業が共謀戦略を採用している状態であると定義することができる（Harrington (2018)）。すべての企業が競争水準以上の価格を提示している状態は、暗黙の共謀そのものではなく、暗黙の共謀の結果に過ぎない。

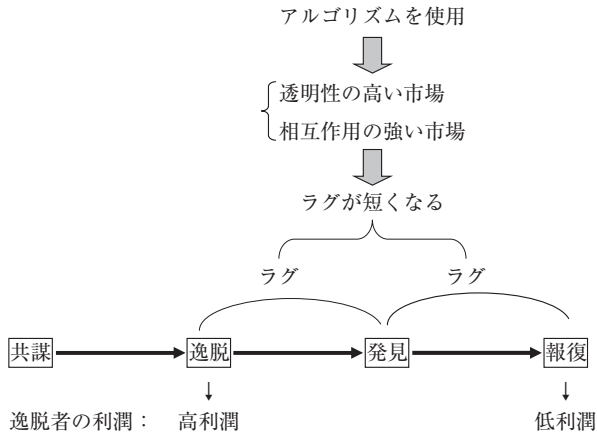


図13 暗黙の共謀の実現可能性

正確に把握することは困難である。

(2) 報復が徹底的かつ効果的となる

アルゴリズムはデータから逸脱（＝価格引き下げ）を瞬時に発見し、自動的かつ速やかに報復（＝価格引き下げ）する。また、逸脱者もアルゴリズムを使用しているので、逸脱者に報復をリアルタイムで知らしめることができる。このように、逸脱の発見と逸脱に対する報復のスピードが上昇するので、徹底的かつ効果的に報復できる。

人の手で価格が決められるとき、市場の透明性は低く、また反応速度も遅い。このとき、逸脱によって得られる利潤が増加する。市場の透明性が低いときは、競合企業の価格を不完全にしか観察できない³⁸⁾。そのため、逸脱がないにも関わらず、誤った判断で報復が実行される可能性がある。逆に、秘密の値下げをしても見逃され、報復されない可能性もある。また、反応にラグがある場合は、逸脱が発見されても即座に報復は起こらないので、報復が起こるまで、逸脱者

38) 競合企業の価格が正確に観察できないので、逸脱が生じたかどうかは、自社の販売量から推測して判断するしかない。販売量が減少した原因が、競合企業の価格引き下げによるものか、あるいは需要関数が変化したことによるものか、どちらの可能性が高いか推測した上で、報復を実行する (Green and Poter (1984))。

は高い利潤を享受することができる。このように、報復が不完全であるため、逸脱を阻止することが困難となる³⁹⁾。

市場の透明性が向上すると、逸脱が発生してからそれを発見するまでのラグが短くなる。また、企業間の相互作用の頻度が高まると、逸脱を発見してから報復を実行するまでのラグが短くなる。これにより、暗黙の共謀はより安定的になる。この知見が正しいことをモデルを用いて確認する⁴⁰⁾。企業間の相互作用は連続的に起こるので、複占企業が無限連続時間で価格を決定するモデルを構築して議論する。共謀（報酬フェーズ）から得る利潤を π^m 、競争状態（報復フェーズ）で得る利潤を π^c 、共謀から逸脱するとき得る利潤を π^d と定め、その水準は毎時点一定とする。また、 $\pi^d > \pi^m > \pi^c$ とする。共謀価格を維持するときの割引現在価値は、次のようになる。

$$V^m = \int_0^{\infty} e^{-rt} \pi^m dt. \quad (39)$$

ただし、 r は割引率である。逸脱の発見に要する時間を T 、報復の実行に要する時間を L とする。 T は市場の透明性を測る指標、 L は企業間の相互作用の頻度（反応の速度）を測る指標である。共謀から逸脱するときの割引現在価値は、次のようになる。

$$V^d = \int_0^{T+L} e^{-rt} \pi^d dt + \int_{T+L}^{\infty} e^{-rt} \pi^c dt. \quad (40)$$

共謀価格が維持される条件は、

$$V^m \geq V^d \quad (41)$$

である。(39)、(40)式を代入すると、

$$\pi^m \int_0^{\infty} e^{-rt} dt \geq \pi^d \int_0^{T+L} e^{-rt} dt + \pi^c \int_{T+L}^{\infty} e^{-rt} dt. \quad (42)$$

39) 価格を人の手によって決めている企業が共謀のメンバーに存在すると、暗黙の共謀は安定しない。価格決定の担当者が、双曲割引率を持っていたり、あるいは担当の任期が来るまでに実績を求めたりするので、すぐに利益を出すべく、値引きをして商品を売り切るインセンティブを持つからである。

40) OECD (2017) で展開されたモデルに依拠して議論する。

これより,

$$\pi^m \geq \pi^d - (\pi^d - \pi^c) e^{-r(T+L)} \quad (43)$$

となる。これが暗黙の共謀の安定条件である。右辺は T と L の増加関数であるから、市場の透明性が向上する (T が小さくなる) ほど、また企業の相互作用の頻度 (反応の速度) が高まる (L が小さくなる) ほど、右辺の値は小さくなり、暗黙の共謀はより安定的となる。

特に、Digital Eye を持つ自律マシンは、市場の透明性が極致に達し、またミリ秒単位で反応できるので、 $T \rightarrow 0$, $L \rightarrow 0$ となる、(43)式の右辺は、

$$\lim_{\substack{T \rightarrow 0 \\ L \rightarrow 0}} \{\pi^d - (\pi^d - \pi^c) e^{-r(T+L)}\} = \pi^c \quad (44)$$

となるから、 $\pi^m > \pi^c$ である限り割引率や逸脱利潤の大きさに関係なく、安定条件はつねに成立する。

暗黙の共謀を実現させるアルゴリズムはどのようなものであろうか。OECD (2017) は、以下に挙げる3つのアルゴリズムがマシンにプログラムされているとき、暗黙の共謀が実現すると指摘している。

(1) シグナリング・アルゴリズム

共謀への参加を誘うシグナルを、一方的に競合企業に送るアルゴリズムである (図14)。シグナルを送り、それに対する競合企業の反応を待つ。シグナルには、具体的な共謀価格の水準が指定されていたり、価格引き上げの提案が含まれていたりする。競合企業のマシンは、即座にかつ確実にシグナルを読み取り、素早く反応してシグナルを送り返す。合意のシグナルが送られてこないときは、さらに異なるシグナルを送り、競合企業からの反応を待つ。競合企業から合意が得られるまで、シグナルを何度も送り続ける。シグナルのやり取りは素早く行われ⁴¹⁾、またシグナルは人間には読めない⁴²⁾ので、シグナルが送られた

41) マシン間で、シグナルの発信 (offer) と、シグナルによる反応 (accept) を繰り返すことは、マシン間でコミュニケーションをしていると解釈することができる。

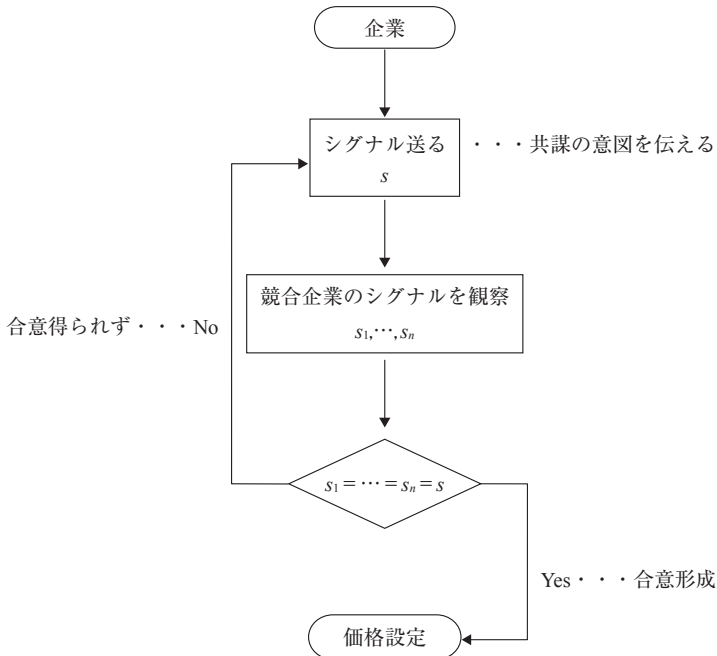


図14 シグナリング・アルゴリズム

ことを人間が気が付くことはない。

(2) 並行アルゴリズム

先導者のアルゴリズムは共謀価格を計算し、価格を引き上げる。競合企業の価格を観察し、追随するならば共謀価格を維持し続け、追随しないならば瞬時に競争価格に戻す。一方、追随者のアルゴリズムは、つねに先導者の価格に追随する (図15)。

(3) モニタリング・アルゴリズム

競合企業の提示価格をつねに監視するアルゴリズムである (図16)。共謀価格が提示されているならば、共謀価格を維持し、監視を続ける。共謀価格以外の価格が提示されたときは、永遠に競争価格を提示する⁴²⁾。

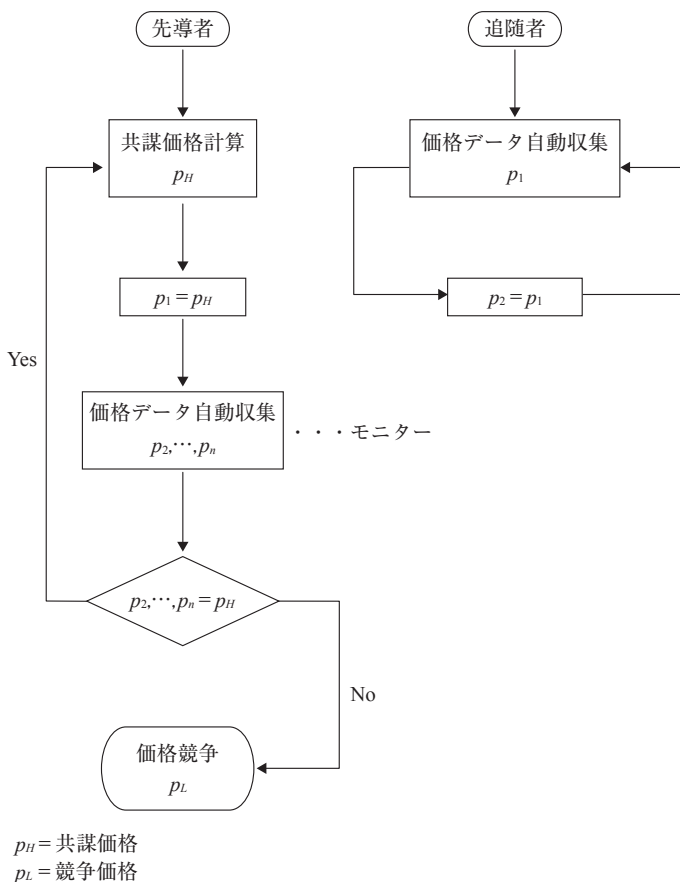


図15 並行アルゴリズム

5.3 競争制限効果 — 学習アルゴリズムが搭載されている場合

以上の議論で扱われた暗黙の共謀は、共謀を実現する諸要因がプログラムに記述されているという点で、人為的に起こされたものである。アルゴリズムは、暗黙の共謀を実現するための道具として使用されているに過ぎない。ところが、

42) モニタリング・アルゴリズムを採用する企業が多いほど、個々の価格が多くの企業に監視されるので、逸脱は容易に発見される。さらに、逸脱者は、多くの企業から集団的に報復されるため、集中産業における報復と同じ効果を持つ。

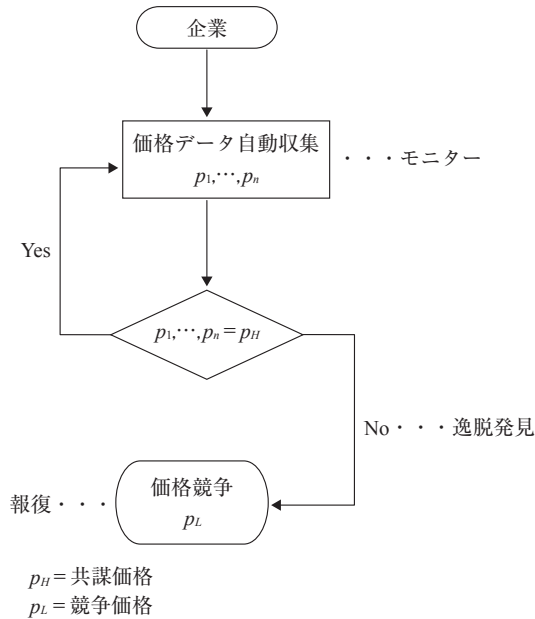


図16 モニタリング・アルゴリズム

学習アルゴリズムが搭載されたプライシング・アルゴリズムは、特別な命令を与えなくても、暗黙の共謀を引き起こす可能性がある。経験や実験から、共謀戦略を採用すること、あるいは、競合企業の値上げに追随し、一方的に値下げをしないことが、利潤最大化の目的に合うことを学習し、そのようにアルゴリズムが書き換えられる。経営者はアルゴリズムがどのように学習して、どのような価格付けルールを採用しているか分からない。したがって、経営者の与り知らないところで共謀が生じる。

(1) 有限オートマトン

トリガー戦略の性質を持つオートマトン（図6）や、しっぺ返し戦略の性質を持つオートマトン（図7）は、前期の価格プロファイルに基づいて価格を出力し、なおかつ報酬・報復スキームを備えている。 (p_H, p_H) が共謀状態（報酬フェーズ）、 (p_L, p_L) が競争状態（報復フェーズ）となる。学習過程で、様々

なオートマトンの中から、これらのオートマトンが採用されるようになると、暗黙の共謀が起こる⁴³⁾。

(2) ニューラル・ネットワーク

結合利潤が最大になる状態が理想であることをマシンに教え込むならば、共謀価格が出力されるかもしれない。しかしこれは、共謀を実現するようにアルゴリズムにプログラムしているだけで、マシンが自主的に実行しているわけではない。

そもそも需要関数を推定する過程で、暗黙の共謀が実現することはない。アルゴリズムが共謀戦略の性質を持つならば、予測・最適化モジュールは次のようになっているだろう。予測モジュールにおいては、過去に競合企業が提示した価格のデータを収集し、競合企業が提示する価格を予測する。最適化モジュールでは、競合企業の提示価格に与える影響を考慮に入れて、自社の価格を決める。ニューラル・ネットワークでは以下のような状態に相当するだろう。競合企業の価格のデータがインプット層に入力され、このデータに大きな重みが置かれた入力値が、各ニューロンに送られる。そうなる、共謀価格が出力される可能性がある⁴⁴⁾。

(3) Q 学習

学習が完了したとき、Q 値がトリガー戦略と類似した性質を持つとする。つまり、

$$Q((p_L, p_L), p_L) > Q((p_L, p_L), p_H), \quad (45)$$

$$Q((p_H, p_H), p_H) > Q((p_H, p_H), p_L), \quad (46)$$

$$Q((p_H, p_L), p_L) > Q((p_H, p_L), p_H), \quad (47)$$

43) ただし、まず初期状態として (p_H, p_H) を実現しないと、暗黙の共謀は起こらない。

44) ただし、各層で起きていることはブラック・ボックスであるので、人間の目では見ることができない。それゆえ、マシンがどのデータを使ってどのような特徴を抽出し、特定の価格を導き出したか人間には分からない。

$$Q((p_L, p_H), p_L) > Q((p_L, p_H), p_H) \quad (48)$$

となっている。このとき、プライシング・アルゴリズムは、状態が (p_H, p_H) のときだけ共謀価格 p_H を出力し、それ以外の状態では競争価格 p_L を出力する。両企業の Q 値がこのようになっているとき、状態が (p_H, p_H) のとき両企業は p_H を提示し、次期の状態も (p_H, p_H) となり、共謀状態が持続する。それ以外の状態では、両企業とも p_L を提示し、次期の状態は (p_L, p_L) となり、以降は競争状態となる。したがって、初期状態が (p_H, p_H) であるときは、(高い確率で) 暗黙の共謀が実現する⁴⁵⁾。また、それ以外の状態でも、両企業が同時に実験を行い共謀価格 p_H を提示したならば、次期の状態は共謀状態 (p_H, p_H) となり、やはり暗黙の共謀が実現する。

Q 値がしつぺ返し戦略と類似した性質を持つときは、(45)、(46)、(47)式に加えて、

$$Q((p_L, p_H), p_H) > Q((p_L, p_H), p_L) \quad (49)$$

となる。トリガー戦略と異なるのは、状態が (p_L, p_L) のときだけ両企業が p_L を提示し、以降は競争状態となる点である。

(4) Digital Eye を持つ自律マシン

市場で活動しているすべての企業のマシンが、Digital Eye を持つ自律マシンとなると、これまでにないタイプの暗黙の共謀が実現する。すべてのマシンが、類似したビッグ・データを蓄積し、学習レベルが同じ高性能アルゴリズムを用いて、同じ市場環境の下で、利潤最大化の目的を共有している。それゆえ、各マシンが採用する戦略は、共通の戦略集合に属する要素に絞られるので、実現した市場条件に対して競合企業がどのような戦略を取ろうとしているかを、正確に予測することができる。直感的な表現を用いると、このような状況である。自社のマシンと競合企業のマシンは全く同じものであり、同じように予測

45) どちらか一方の企業が実験して p_L を提示すると、次期は競争状態になる。

をし、同じ方法で最適化をする。マシンは競争相手のマシンが何を考えて（＝予測）、どう行動しようとしているか（＝最適化）、手に取るように分かるという、マシンの間で以心伝心がある状態である。

したがって、どのマシンも、競合企業による競争の主導権（competitive initiative）（ライバルから市場シェアを奪うことを目論んだ価格の引き下げ）が実行される前に、その予兆を察知することができる。このことから、Digital Eyeを持つ自律マシンに対して競争の主導権（competitive initiative）を仕掛けることが不可能であることを、Digital Eyeを持つ自律マシンは理解する。それに伴い、各々のマシンは競争を引き起こさないような価格（＝共謀価格）を見つけ出し、顧客を企業間で割り当てて、市場の安定化を図る⁴⁶⁾。こうして、市場で競争の自主規制が起こる。

6. ま と め

人工知能を搭載したプライシング・アルゴリズムは、自己学習を通じて予測と最適化の精度を高めることで、実現した市場条件に対する最適価格を正確に計算し、市場条件の変化に対しては即座に価格を調整する。アルゴリズムを用いる企業が増えるほど、市場の透明性が向上し、企業間の相互作用が強くなる。このような市場構造の変化により競争形態が変わり、暗黙の共謀が容易に実現する環境が整備され、結果、競争が起こらず、市場価格が高い水準に維持される可能性がある。

巨大プラットフォームに関して、このような議論が成立する可能性は高い。したがって、暗黙の共謀を防止する観点から、巨大プラットフォームによるビッグ・データのアクセスやアルゴリズムの使用について、独禁法に基づいた制限を課さないといけないだろう。しかし、一般の市場でアルゴリズムを使用することが、暗黙の共謀に実現に繋がるかどうかは疑問の余地が残る。巨大プ

46) 独禁法の用語で、意識的並行行為（conscious parallelism）という概念がある。価格競争をしないことが利潤を増加させるという企業間の相互理解（mutual understanding）が、意思の疎通を図らなくても生じることである。

ラットフォームでは見られない様々な制約が、暗黙の共謀の実現を阻害するからである。まず、市場で活動する企業の異質性が著しいとき、共謀価格をどの水準に定めるかという、焦点 (focal point) の問題が生じる。企業規模、製品差別化の程度、費用、キャパシティ、市場シェア、消費者のロイヤリティが企業間で異なるとき、焦点となる価格が多数存在し、どの企業も受け入れられる価格が1つに定まることはない。

結合利潤を最大にする価格を共謀価格とするにしても、暗黙の共謀は容易には実現しない。結合利潤は、

$$\Pi = \sum_{i=1}^n \left\{ (p_i - c_i) \left(\alpha_i - \beta_i p_i + \sum_{j \neq i} \gamma_{ij} p_j \right) - F_i \right\} \quad (50)$$

となり、これを最大にする価格プロファイル $\{p_1, \dots, p_n\}$ を共謀価格とする。報酬・報復スキームが組み込まれた戦略を構築するためには、競合企業の需要パラメータ $\alpha_i, \beta_i, \gamma_{ij}, \dots, \gamma_{in}$ と、費用パラメータ c_i, F_i を推定しないとイケない。それには、顧客の選好や所得、生産技術、要素価格といった競合企業の内部情報にアクセスできることが前提となるが、通常これらの情報は企業秘密となっているであろう。また、共謀戦略のように、前期の価格プロファイルをもとに最適価格を決めるとき、企業数が増加すると「次元の呪い」により計算量が爆発的に増加する。市場に n 企業が活動しているとき、状態集合は $S = P \times \dots \times P$ であり、各企業の戦略は $S \rightarrow P$ であるから、最適戦略を見つけるときに必要な計算量は $(\#P)^{n+1}$ となり、企業数が増加すると計算量が指数関数的に増加する。

プライシング・アルゴリズムは、ダイナミック・プライシングと価格差別を同時に実現するので、焦点の問題はさらに深刻になる。ダイナミック・プライシングが行われている場合、購入・予約する時間を区切って異なる価格が付けられる。また、価格差別が行われている場合、顧客をグループに分けて異なる価格が付けられる。したがって、探索すべき共謀価格の数は、「時間単位×顧客グループ数」だけ存在し、その数は膨大である。

アルゴリズムが持つ技術上の制約も、暗黙の共謀を容易には実現させない。

市場環境が頻繁に変化するとき、最適価格がつねに変動するため、実験ばかり行われ、アルゴリズムの学習過程は収束しない。また、競合企業のアルゴリズムにエラーがあり、異常な価格設定をすると、自社のアルゴリズムもそれに反応してしまう⁴⁷⁾。いずれの場合も、暗黙の共謀が一時的に成立したとしても、すぐに崩壊するであろう。

このような制約を考慮に入れると、「人工知能は競争なき市場を創り出す」という命題は、一般の市場では、限られた状況でしか成立しないと推測できる。それでは、アルゴリズムを使用すると、競争が制限されやすくなる市場はどのような性質を持つのであろうか。理論分析とシミュレーション分析を踏まえて、明らかにしていかなければならない。

参考文献

- [1] Agrawal, A., J. Gans., and A. Goldfarb. (eds), (2019), *The Economics of Artificial Intelligence*, University of Chicago Press.
- [2] Bakos, J. (1997), “Reducing Buyer Search Costs : Implications for Electronic Marketplaces.”, *Management Science*, Vol.43, pp.1676-1692.
- [3] Barr, J. and F. Saraceno. (2005), “Cournot Competition, Organization and Learning.”, *Journal of Economic Dynamics and Control*, Vol.29, pp.277-295.
- [4] Camerer, C. and T-H. Ho. (1999), “Experience-Weighted Attraction Learning in Normal Form Game.”, *Econometrica*, Vol.67, pp.827-874.
- [5] CMA (Competition and Markets Authority) (2018), “Pricing Algorithms : Economic Working Paper on the Use of Algorithms to Facilitate Collusion and Personalised Pricing.”, working paper.
- [6] Deng, A. (2017), “When Machines Learn to Collude : Lessons from a Recent Research Study on Artificial Intelligence”, Bates White Economic Consulting.
- [7] Ezrachi, A. and M. E. Stucke. (2016), *Virtual Competition : The Promise and Perils of the Algorithm-Driven Economy*, Harvard University Press.
- [8] Ezrachi, A. and M. Stucke. (2017), “Artificial Intelligence & Collusion : When Computers Inhibit Competition.”, *University of Illinois Law Review*, Vol.2017, pp.1775-1809.
- [9] Green, E and R. Porter. (1984), “Noncooperative Collusion under Imperfect Price Information.”, *Econometrica*, Vol.52, pp.87-100.
- [10] Hanaki, N., R. Sethi., I. Erev., and A. Peterhansl. (2005), “Learning Strategy.”, *Journal*

47) 2011年に Amazon で付けられた生物学のテキストの価格が、2,370万ドルまで跳ね上がったことがある。マーケットプレイスのある売り手 i のアルゴリズムは、他の売り手 j の付けた価格をベンチマークして価格を決めるというもので、 $p_i = 1.27059p_j$ に従い価格を決めていた。別の売り手も同様のアルゴリズムを採用しており、 $p_i = 0.9983p_j$ であった (Mehra (2015))。

- of Economic Dynamics and Control*, Vol.36, pp.523-542.
- [11] Harrington, Jr., J. E. (2018), “Developing Competition Law for Collusion by Autonomous Price-Setting Agents.”, working paper.
- [12] Ittoo, A. and N. Petit. (2017), “Algorithmic Pricing Agents and Tacit Collusion : A Technological Perspective.”, working paper.
- [13] Mailath, G. and L. Samuelson. (2006), *Repeated Games and Reputations : Long-Run Relationships*, Oxford University Press.
- [14] Maskin, E. and J. Tirole. (1988), “A Theory of Dynamic Oligopoly, II : Price Competition, Kinked Demand Curves, and Edgeworth Cycles.”, *Econometrica*, Vol.56, pp.571-599.
- [15] Mehra, S. (2015), “Antitrust and the Robo-Seller : Competition in the Time of Algorithms”, *Minnesota Law Review*, Vol.100, pp.1323-1375.
- [16] OECD (2017), “Algorithms and Collusion : Competition Policy in the Digital Age.”
- [17] Oxera (2017), “When Algorithms Set Prices : Winners and Losers”, working paper, 19.
- [18] Phillips, R. (2005), *Pricing and Revenue Optimization.*, Stanford Business Books.
- [19] Salcedo, B. (2015), “Pricing Algorithms and Tacit Collusion”, Pennsylvania State University.
- [20] Shy, O. (2008), *How to Price : A Guide to Pricing Techniques and Yield Management*, Cambridge University Press.
- [21] Salop, S. and J. Stiglitz. (1977), “Bargains and Ripoffs : A Model of Monopolistically Competitive Price Dispersion.”, *Review of Economic Studies*, Vol.44, pp.493-510.
- [22] Varian, H. (1980), “A Model of Sales.”, *American Economic Review*, Vol.70, pp.651-659.
- [23] Varian, H. (2017), “Artificial Intelligence, Economics, and Industrial Organization.”, NBER Working Paper No. 24839.
- [24] Vives, X. (1999), *Oligopoly Pricing : Old Ideas and New Tools*, MIT Press.
- [25] Waltman, L. and U. Kaymak. (2008), “Q-learning Agents in a Cournot Oligopoly Model.”, *Journal of Economic Dynamics and Control*, Vol.32, pp.3275-3293.