

## D. Ökonomische Erkenntnisse zur algorithmischen Kollusion

Die allgemeine ökonomische Literatur zu *tacit collusion* hat aufgezeigt, dass einige der Faktoren, welche Kollusion befördern können, durch algorithmische Preissetzung verstärkt auftreten. Zugleich wurden die generellen Hürden deutlich, aufgrund derer die Gefahr einer *tacit collusion* grundsätzlich als gering eingeschätzt wird. Im Folgenden werden die wichtigsten Erkenntnisse der Aufsätze, die sich explizit mit den Auswirkungen algorithmischer Preissetzung auf *tacit collusion* auseinandergesetzt haben, dargestellt und die Art der Untersuchungen zusammengefasst. Beginnend mit den theoretischen Auseinandersetzungen zur algorithmischen Preissetzung (I.), werden neben Computersimulationen (II.) auch auf Felddaten beruhende Analysen (III.) vorgestellt. Abschließend werden die Befunde in einem Zwischenergebnis zusammengefasst (IV.).

### I. Theoretische Ansätze

Verschiedene theoretische Modelle haben versucht, die Besonderheiten algorithmischer Preissetzung herauszustellen und ihre Auswirkungen auf den Wettbewerb zu analysieren. Auf Grundlage der bekannten Modelle werden hierbei unterschiedliche Eigenschaften betrachtet, die die algorithmische Preissetzung auszeichnen oder in Zukunft auszeichnen könnten.

#### 1. Informationsaustausch durch Preissetzungsalgorithmen

*Bruno Salcedo* betrachtet ein dynamisches *Bertrand*-Marktmodell, in dem beide Wettbewerber algorithmische Preissetzung einsetzen.<sup>651</sup> In seinem Modell kommt der Autor zu dem Ergebnis, dass die langfristigen Profite durch Algorithmen nahe dem Monopolgewinn liegen. *Salcedo* schreibt, dass *tacit collusion* in seinem Modell nicht nur ein mögliches Gleichgewicht sei, sondern sogar „*inevitable*.“<sup>652</sup> Allerdings trifft der Autor hierfür einige Annahmen, die in der Realität grundsätzlich nicht gegeben sein dürften.

---

651 *Salcedo* (2015).

652 *Salcedo* (2015), S. 3.

Demnach wird angenommen, dass die Algorithmen der Unternehmen in der Lage sind, die Strategie des Wettbewerbers nach einer gewissen Zeit abzuleiten oder den Algorithmus ganz zu entschlüsseln. Aus diesem Grund kennen die Unternehmen in seinem Modell nach einer zufällig bestimmten und vom Wettbewerber unabhängigen Zeit die Preissetzungsstrategie ihres Wettbewerbers und können ihre Strategie entsprechend anpassen.<sup>653</sup> Darüber hinaus setzt das Modell voraus, dass die Strategien der Algorithmen nur langfristig und nie zeitgleich verändert werden können.<sup>654</sup> Nur unter diesen Voraussetzungen kommt *Salcedo*'s Modell zu dem Ergebnis, dass die Algorithmen kollusive Marktergebnisse zur Folge haben.

## 2. Bessere Vorhersage der Zahlungsbereitschaft durch Preisalgorithmen

In Anlehnung an die Literatur zum Verhältnis von Markttransparenz zur Kollusion modellieren *Jeanine Miklós-Thal* und *Catherine Tucker* ebenfalls einen *Bertrand*-Markt und setzen sich mit dem Einfluss einer besseren Vorhersage der Nachfrage durch Algorithmen auseinander.<sup>655</sup> In ihrem Modell variiert die Zahlungsbereitschaft der Nachfrage zufällig: Mit je fünfzig Prozent Wahrscheinlichkeit haben die potenziellen Kundinnen eine hohe beziehungsweise eine niedrige Zahlungsbereitschaft für das angebotene Gut. Die Unternehmen können die Zahlungsbereitschaft der Nachfrage prognostizieren, bevor sie eine Preisentscheidung treffen und sich so der erwarteten Nachfrage anpassen.<sup>656</sup> Hierbei kommen die Autoren zu dem Ergebnis, dass eine genauere Vorhersage der Zahlungsbereitschaft auch positive Auswirkungen auf den Wettbewerb haben kann. Demnach erhöht sich der Anreiz, den Wettbewerber zu unterbieten, wenn sich eine hohe Nachfrage prognostizieren lässt.<sup>657</sup> *Jason O'Conner* und *Nathan E. Wilson* setzen sich ebenfalls mit der besseren Vorhersage der Zahlungsbereitschaft durch Algorithmen auseinander.<sup>658</sup> Im Gegensatz zu *Miklós-Thal* und *Tucker* bedienen sie sich eines anderen Grundmodells, dem Modell der Nachfrageungewissheit von

---

653 *Salcedo* (2015), S. 9 f.

654 *Salcedo* (2015), S. 3 f.

655 *Miklós-Thal/C. Tucker* (2019), *ManSci* 65 (4), 1552; Ihr Modell baut auf dem Kollusionsmodell von *Julio J. Rotemberg* und *Garth Saloner* auf (Theorie der Preiskriege in Zeiten von Hochkonjunktur), *Rotemberg/Saloner* (1986), *AER* 76 (3), 390.

656 *Miklós-Thal/C. Tucker* (2019), *ManSci* 65 (4), 1552 (1554).

657 *Miklós-Thal/C. Tucker* (2019), *ManSci* 65 (4), 1552 (1557).

658 *O'Connor/N. E. Wilson* (2021), *Information Economics and Policy* 54, 100882.

Green und Porter.<sup>659</sup> Konnten sich die Unternehmen im vorherigen Modell perfekt überwachen, haben die Unternehmen bei O'Connor und Wilsons Ansatz nur Informationen über das endgültige Marktergebnis.<sup>660</sup> In ihrer Analyse zeigt sich ein zweiseitiger Effekt der algorithmischen Vorhersage: Die bessere Vorhersage kann zum einen bei sich nicht verändernder Marktstruktur Kollusion wahrscheinlicher machen, zum anderen erhöht sie aber auch den Anreiz für eine Abweichung.<sup>661</sup> Die Ergebnisse entsprechen dem differenzierten Bild, welches die ökonomischen Modelle zur Markttransparenz bereits vorgezeichnet haben.<sup>662</sup>

### 3. Der frequenzielle Wettbewerb mit Algorithmen

Zach Brown und Alexander MacKay analysieren Daten von fünf Online-Händlern und nutzen diese im Anschluss, um den Erkenntnissen entsprechende Märkte zu modellieren.<sup>663</sup> Ihre Analyse der Online-Märkte zeigt auf, dass Verkäuferinnen ihre Preise in regelmäßigen Intervallen anpassen, wobei sich die Zeitpunkte der Anpassung zwischen den Unternehmen unterscheiden. Des Weiteren stellen sie das schnelle Anpassen einiger Händler an Preisanpassungen von Unternehmen mit einer geringeren Preisfrequenz fest, was darauf hindeutet, dass automatisierte Systeme zum Einsatz kommen.<sup>664</sup> Brown und MacKay modellieren einen Markt mit asymmetrischer Preissetzung und homogenen Gütern. Hierbei können die Unternehmen in einem ersten Modell in unterschiedlichen Frequenzen Preise festlegen.<sup>665</sup> Darüber hinaus bewerten sie algorithmische Preissetzung als ein Werkzeug, um sich als Unternehmen glaubhaft bezüglich einer Preisstrategie zu verpflichten. Deshalb erweitern sie ihr Modell in einem nächsten Schritt und gehen – ähnlich wie Salcedo – von einem Duopol aus, in dem die Unternehmen einen Preissetzungsalgorithmus wählen, welchen sie in regelmäßigen Intervallen anpassen können. Zwischen diesen Intervallen ist der Algorithmus jedoch auf eine Strategie festlegt, ungeachtet dessen, ob sie kurzfristig zum optimalen

---

659 Green/R. H. Porter (1984), *Econometrica* 52 (1), 87; siehe hierzu Kapitel C. I. 2. d) bb) (2).

660 Vgl. O'Connor/N. E. Wilson (2021), *Information Economics and Policy* 54, 100882 (3).

661 O'Connor/N. E. Wilson (2021), *Information Economics and Policy* 54, 100882 (18).

662 Siehe hierzu Kapitel C. II. 2. d) bb) (2).

663 Z. Brown/MacKay (2022), *AEJ: Micro* (im Erscheinen).

664 Z. Brown/MacKay (2022), *AEJ: Micro* (im Erscheinen) (11).

665 Z. Brown/MacKay (2022), *AEJ: Micro* (im Erscheinen) (15 ff.).

Ergebnis führt.<sup>666</sup> Ihre Ergebnisse zeigen auf, dass die Asymmetrien in der Preissetzung zu höheren Preisen als in einem synchronen *Bertrand*-Modell führen, unabhängig davon, ob sie aufgrund der differierenden Frequenzen oder der langfristigen Festlegung auf eine Strategie zu Stande kommen.<sup>667</sup> Hieraus schließen *Brown* und *MacKay*, dass „das *Bertrand*-Gleichgewicht auf Online-Märkten eher die Ausnahme als die Regel sein könnte.“<sup>668</sup>

## II. Simulationen

Neben rein theoretischen Arbeiten zu algorithmischer Preissetzung gibt es einige Arbeiten, in denen Wissenschaftlerinnen in Simulationen selbstlernende Algorithmen in einem Marktmodell zum Einsatz bringen. Im Kontrast zu Laborexperimenten treffen hierbei ausschließlich Algorithmen Entscheidungen, sodass sich die Interaktion mehrerer Algorithmen und ihre Fähigkeiten in einem Marktumfeld analysieren lassen. In einem Großteil der bisher durchgeführten Simulationen kommen selbstlernende Algorithmen zum Einsatz, welche sich des *reinforcement learnings* bedienen. Die Entscheidungen der hierbei verwendeten sogenannten *Q-learning* Algorithmen werden nach einer abgeschlossenen Lernphase im Wettbewerb untereinander analysiert. Im Folgenden werden zunächst die Eigenschaften der *Q-learning* Algorithmen dargestellt (1.). Anschließend wird ein Überblick über die wichtigsten Veröffentlichungen und Ergebnisse auf diesem Gebiet präsentiert (2.).

### 1. *Q-Learning* Algorithmen

*Q-learning* Algorithmen gehören zu den am häufigsten verwendeten und bekanntesten Algorithmen aus dem Bereich des *reinforcement learnings*.<sup>669</sup> Das auf *Christopher J. C. H. Watkins* zurückgehende *Q-learning* wurde ursprünglich zur Lösung von *Markow*-Entscheidungsproblemen entwickelt.<sup>670</sup> Hierbei handelt es sich um dynamische Entscheidungsprobleme mit mehre-

---

666 *Z. Brown/MacKay* (2022), AEJ: Micro (im Erscheinen) (23 ff.).

667 *Z. Brown/MacKay* (2022), AEJ: Micro (im Erscheinen) (20).

668 Aus dem Englischen übersetzt, siehe *Z. Brown/MacKay* (2022), AEJ: Micro (im Erscheinen) (41).

669 *Stone*, in: *Sammut/Webb* (Hrsg.), *Encyclopedia of Machine Learning and Data mining*, S. 1033.

670 *Watkins*, *Learning from Delayed Rewards*.

ren Perioden und Unsicherheit über die Folge einer Entscheidung.<sup>671</sup> Mittels Ausprobieren (*trial and error*) versucht der Algorithmus ohne Vorkenntnisse (*model-free learning*), den langfristigen Gewinn durch die hierfür bestmögliche Handlung zu maximieren.<sup>672</sup> Diese Form des Lernens wird als *temporal difference learning* bezeichnet.<sup>673</sup> Der zu Beginn unerfahrene Algorithmus trifft eine Entscheidung und erhält durch die Folgen seiner Handlung Informationen, aus denen er Schlüsse für die zukünftigen Entscheidungen ziehen kann.<sup>674</sup> Der Algorithmus beobachtet hierbei zunächst seine aktuelle Entscheidungssituation, trifft eine Entscheidung, bekommt eine Reaktion auf die Entscheidung und beobachtet die daraus folgende Entscheidungssituation.<sup>675</sup> Durch wiederholtes Ausprobieren erlernt der Algorithmus, welche Handlung am vielversprechendsten ist, um den diskontierten langfristigen Gewinn zu maximieren.<sup>676</sup> *Q-learning* muss dabei das Erkunden mit dem Ausnutzen kombinieren: Der Algorithmus steht dabei in dem Spannungsfeld, zum einen seine Handlungsoptionen umfassend zu erkunden (*exploration*) und zugleich das bereits erlernte Wissen zu nutzen (*exploitation*).<sup>677</sup>

Mathematisch lässt sich das Vorgehen wie folgt darstellen: Betrachtet wird die Periode  $t$ , in der der Algorithmus einen Zustand ( $s_t$ ) feststellt und eine Handlungsentscheidung trifft [ $a_t \in A(s_t)$ ].<sup>678</sup> Sowohl Zustands- als auch Handlungsraum sind hierbei endlich. Für jeden Zustand  $s_t$  und jede Handlung  $a_t$  erhält der Algorithmus eine Auszahlung  $\pi_t$  und gelangt zu einem neuen Zustand  $s_{t+1}$ . Ziel des Algorithmus ist es, eine Strategie zu wählen, welche den erwarteten Nutzen unter Berücksichtigung des Diskontierungsfaktors maximiert:  $E [\sum_{t=0}^{\infty} \pi_t * \delta^t]$ .<sup>679</sup> Hierbei ergibt sich aus einer Matrix der Zustände und der möglichen Handlungen je ein Wert für die *Qualität* einer Handlungsoption  $a_t$  in einem Zustand  $s_t$  (Q-Wert). Dieser Q-Wert gibt die aktuelle Approximation des Nutzens einer Entscheidung an. Durch Ausprobieren kann der Algorithmus die Werte der Tabelle anhand seiner gewonnenen Erfahrungen aktualisieren und so die Auswirkung

671 Hu/Yue, in: Hu/Yue (Hrsg.), Markov decision processes with their applications, S. 1.

672 Charniak, Introduction to Deep Learning, S. 119.

673 Stone, in: Sammut/Webb (Hrsg.), Encyclopedia of Machine Learning and Data mining, S. 1033.

674 Charniak, Introduction to Deep Learning, S. 117.

675 Watkins/Dayan (1992), Machine Learning 8 (3), 279 (281).

676 Watkins/Dayan (1992), Machine Learning 8 (3) (279).

677 Kaelbling et al. (1996), jair 4, 237 (243).

678 Nie/Haykin (1999), IEEE Trans. Veh. Technol. 48 (5) (1676).

679 Calvano et al. (2020), AER 110 (10), 3267 (3270).

seiner Entscheidungen besser einschätzen. Der Q-Wert berücksichtigt den unmittelbaren sowie den diskontierten zukünftigen Gewinn. Indem der Algorithmus in einem Lernprozess wiederholt Entscheidungen trifft, ihre Folgen beobachtet und die entsprechende Zelle der Q-Matrix aktualisiert, verbessert er die Prognose.

Trifft der Algorithmus eine Entscheidung in Periode  $t$  erhält er die Auszahlung  $\pi_t$  und gelangt in den Zustand  $s_{t+1}$ . Die entsprechende Zelle der Matrix  $Q(s_t, a_t)$  wird in der Folge aktualisiert  $[Q_{neu}(s_t, a_t)]$ , wobei  $\alpha \in [0,1]$  als Faktor für den Lernfortschritt berücksichtigt wird und angibt, welches Gewicht dem neuen Wert beziehungsweise dem alten Wert gegeben werden soll.<sup>680</sup> Dies ist notwendig, da die einzige Information über die Qualität eines Zuges in dem Wert der Zelle der Q-Matrix liegt und dieser nicht nur die neueste Erfahrung, sondern auch die zuvor gemachten Erfahrungen berücksichtigen sollte, um eine gute Prognose zu ermöglichen und um unwahrscheinliche Ergebnisse nicht überzubewerten.<sup>681</sup> So besteht  $Q_{neu}(s_t, a_t)$  aus einer Kombination des alten Q-Werts (multipliziert mit  $1 - \alpha$ ) sowie der neu gewonnenen Erfahrung (multipliziert mit  $\alpha$ ), welche jeweils den erwarteten Gewinn der folgenden Periode sowie den diskontierten Gewinn der zukünftigen Perioden angeben:

$$Q_{neu}(s_t, a_t) = (1 - \alpha) Q(s_t, a_t) + \alpha \left[ \pi_t + \delta * \max_{\alpha} Q(s_{t+1}, a) \right].$$

Alle übrigen Zellen (für  $s \neq s_t$  und  $a \neq a_t$ ) bleiben unverändert. Damit der Algorithmus eine möglichst genaue Prognose treffen kann, muss er alle Entscheidungen ausprobiert haben. Damit dies gelingt, trifft der Algorithmus in der Lernphase nicht immer die vermeintlich beste Entscheidung, sondern wählt mit der Wahrscheinlichkeit  $\varepsilon$  eine zufällige Handlungsoption aus (*epsilon greedy exploration*).<sup>682</sup> Um die tatsächliche Gewinn-Matrix bestmöglich abzubilden, muss der Algorithmus darüber hinaus mehrfach dieselbe Entscheidung in demselben Zustand treffen, weshalb der Lernprozess viel Zeit in Anspruch nimmt.<sup>683</sup>

680 T. Klein, RJE 52 (3) (2021), 538 (545).

681 Charniak, Introduction to Deep Learning, S. 119.

682 Rodrigues Gomes/Kowalczyk (2009), ICML '09, 369 (370).

683 Calvano et al. (2020), AER 110 (10), 3267 (3272).

## 2. Algorithmische Kollusion mittels *Q-learning* Algorithmen

### a) Das Scheitern zu Konkurrieren

*Ludo Waltman* und *Uzay Kaymak* gehören zu den ersten, die den Einsatz von *Q-learning* Algorithmen in einer Marktumgebung untersuchten.<sup>684</sup> Hierfür simulieren sie den Wettbewerb zwischen Algorithmen in *Cournot*-Oligopolyen und variieren die Marktgröße zwischen den Simulationen, von zwei bis hin zu sechs Wettbewerbern.<sup>685</sup> Für ihre Simulationen verwenden die Autoren sowohl Algorithmen mit der Fähigkeit vergangene Aktionen zu berücksichtigen (*memory*), als auch Algorithmen ohne ein solches „Gedächtnis“ (*no memory*).<sup>686</sup> Je umfangreicher das *memory* eines Algorithmus ist, desto mehr Zustände ( $s_t$ ) berücksichtigt dieser als eigene Zelle innerhalb seiner Matrix. In der Simulation von *Waltman* und *Kaymak* können die Algorithmen, die über ein Gedächtnis verfügten, die Kombination aus ihrem eigenen Produktionslevel, sowie dem Produktionslevel der Wettbewerber aus der vorherigen Periode als einen eigenen Zustand berücksichtigen (*memory one*).<sup>687</sup>

Innerhalb der *memory one* Algorithmen variieren die Autoren darüber hinaus die Weitsichtigkeit der Algorithmen der Wettbewerber. Während die weitsichtigen Algorithmen diskontierte zukünftige Gewinne in ihre Berechnung miteinbeziehen ( $\delta = 0,9$ ), berücksichtigen die kurzsichtigen Algorithmen die Auswirkungen ihres Handelns auf zukünftige Gewinne nicht, sodass der Diskontierungsfaktor null entspricht ( $\delta = 0$ ).<sup>688</sup> Alle Variationen werden über eine Millionen Perioden simuliert.

Die Analyse der Algorithmen konzentriert sich auf ihr langfristiges Verhalten, bei dem das zufällige Ausprobieren einer Handlung – das Training – bereits abgeschlossen ist ( $\epsilon = 0$ ). Ihre Ergebnisse zeigen, dass die Nutzung der *q-learning* Algorithmen sogar bei großen Märkten ( $n = 6$ ) zu kollusivem Verhalten führt und das sowohl bei den *memory one*, als auch bei den *no memory* Algorithmen.<sup>689</sup> Dieses Ergebnis ist insofern erstaunlich, als dass die Entscheidung der Algorithmen ohne Berücksichtigung der vorherigen Entscheidung einem *one-shot game* gleicht. So sind mögliche Bestrafungen einer

684 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275.

685 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275 (3280 ff.).

686 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275 (3282).

687 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275 (3282).

688 Hieraus ergibt sich für  $Q_{neu}$ :  $Q_{neu}(s_t, a_t) = (1 - \alpha) Q(s_t, a_t) + \alpha [\pi_t]$ .

689 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275 (3284).

Abweichung vom kollusiven Verhalten nicht möglich, sofern die Unternehmen dieses Verhalten nicht mit in die Entscheidung einfließen lassen können. Auch die kurzsichtigen Algorithmen sollten kein Interesse an einer Kollusion haben, da sie die langfristigen Vorteile einer Kollusion unberücksichtigt lassen. Dadurch ist ein kooperatives Verhalten der Algorithmen irrational. Die Ergebnisse von *Waltman* und *Kaymak* deutet somit weniger auf ein Erlernen von kollusivem Verhalten hin, als vielmehr auf ein gescheitertes Erlernen wettbewerblichen Verhaltens.

## b) Das Erlernen zu Kolludieren

*Emilio Calvano et al.* setzten sich ebenfalls mit der Frage selbstlernender Algorithmen und ihrem Einfluss auf den Wettbewerb auseinander.<sup>690</sup> Die Autoren lassen *memory one Q-learning* Algorithmen in einem *Bertrand*-Duopol miteinander in den Wettbewerb treten. Die Algorithmen haben dabei in der Ausgangssituation (*Baseline*) die Auswahl zwischen 15 möglichen Preisen. In 1.000 unabhängigen Simulationsdurchläufen (*Sessions*) pro *Treatment* lassen sie die Algorithmen mittels *trial and error* lernen, bis die optimale Strategie in einem Zustand für 100.000 Perioden konstant bleibt.<sup>691</sup> Abhängig von der Häufigkeit des Ausprobierens zufälliger Handlungen ( $\epsilon$ ), benötigten die Algorithmen zwischen 400.000 und mehreren Millionen Perioden, um den Prozess des Lernens abzuschließen und stabile Entscheidungen zu treffen.<sup>692</sup>

Im Ergebnis zeigen *Calvano et al.*, dass die Algorithmen kollusives Verhalten erlernen und im Duopol überwettbewerbliche Gewinne nahe dem Monopolgewinn erzielen.<sup>693</sup> Anders als bei *Waltman* und *Kaymak* erreichen die Algorithmen überwettbewerbliche Marktpreise nämlich nur dann, sofern es sich nicht um *no memory* Algorithmen handelt oder der Diskontierungs-

---

690 *Calvano et al.* (2020), AER 110 (10), 3267.

691 *Calvano et al.* (2020), AER 110 (10), 3267 (3273, 3276).

692 Die Autoren nutzen ein  $\epsilon$ -greedy Modell, bei dem die zwischen den Algorithmen variierende Zahl der zufälligen Entscheidungen mit der Zeit abnahm, *Calvano et al.* (2020), AER 110 (10), 3267 (3274, 3276).

693 Betrachtet man die Gewinne als prozentualen Wert über dem Bertrand-Gleichgewicht, wobei 0 % dem Gewinn im wettbewerblichen Gleichgewicht und 100 % dem Monopolgewinn entsprechen, so erzielen die Algorithmen mit nahezu homogenen Gütern zwischen 70-90 % ( $\bar{\epsilon} = 85\%$ ) zusätzlichen Gewinn im Duopol, *Calvano et al.* (2020), AER 110 (10), 3267 (3277 f.).



faktor  $\delta$  gering ist.<sup>694</sup> Ein weiteres Argument dafür, dass die Algorithmen tatsächlich kollusives Verhalten erlernen, ergibt sich bei Betrachtung der Strategien. So kommen die Autoren in ihrer Analyse zu dem Ergebnis, dass die Algorithmen das Abweichen des Wettbewerbers von einer kollusiven Strategie bestrafen und die Abweichungen so unrentabel machen.<sup>695</sup> Nach einigen Perioden der Bestrafung kehren die Algorithmen wieder zu ihrer kollusiven Strategie zurück und erzielen überwettbewerbliche Gewinne. Dies ist sogar bei beidseitiger Abweichung der Fall.<sup>696</sup> In ihren Simulationen betrachten *Calvano et al.* auch größere Oligopole. Zwar nimmt die Kollusion mit Zunahme der Marktteilnehmer ab, allerdings kommen die Algorithmen auch in einem Markt mit drei oder vier Unternehmen zu suprakompetitiven Preisen.<sup>697</sup>

Wenngleich es für eine solche Feststellung an einer Kontrollgruppe mit menschlichen Teilnehmern mangelt, vermuten die Autoren aufgrund ihrer Ergebnisse, dass die Kollusion mit Anstieg der Marktgröße langsamer abnimmt, als in vergleichbaren Märkten mit menschlicher Preissetzung. Die Autoren überprüfen ihre Ergebnisse im Duopol darüber hinaus unter Hinzunahme von Kosten- und Nachfrageasymmetrien, stochastischer Nachfrage, der Produkthomogenität und Marktzutritten. Alle Faktoren beeinflussen das Marktergebnis, wenngleich sie nie dazu führen, dass die Kollusion zusammenbricht und die Unternehmen zum vollständigen Wettbewerb zurückkehren.<sup>698</sup>

### c) Die Bestätigung der Befunde in weiteren Simulationen

Neben *Calvano et al.* gibt es weitere Veröffentlichungen, in denen Wissenschaftlerinnen kollusives Verhalten in Simulationen mit *Q-learning* Algorithmen feststellen. *Timo Klein* wählt einen ähnlichen Aufbau wie *Calvano et al.*, indem er Algorithmen in einem Duopol untersucht, die miteinander im Preiswettbewerb stehen.<sup>699</sup> Sein Ansatz unterscheidet sich dahingehend von

---

694 *Calvano et al.* (2020), AER 110 (10), 3267 (3280 f.).

695 *Calvano et al.* (2020), AER 110 (10), 3267 (3281 ff.).

696 *Calvano et al.* (2020), AER 110 (10), 3267 (3287).

697 So lagen die durchschnittlichen zusätzlichen Gewinne bei 64 % ( $n = 3$ ) bzw. 56 % ( $n = 4$ ), *Calvano et al.* (2020), AER 110 (10), 3267 (3288).

698 *Calvano et al.* (2020), AER 110 (10), 3267 (3289 ff.).

699 *T. Klein*, RJE 52 (3) (2021), 538.

Calvano et al., dass er – vergleichbar mit den theoretischen Ansätzen zur algorithmischen Kollusion von Salcedo sowie Brown und MacKay<sup>700</sup> – einen sequentiellen Preiswettbewerb betrachtet, bei dem die Preisentscheidungen asynchron erfolgen. Während der eine Algorithmus immer in den geraden Perioden eine Preisentscheidung trifft, passt der Wettbewerber in den ungeraden Perioden seine Preise an. Darüber hinaus variiert er die Anzahl der möglichen Preisoptionen ( $k \in \{6, 12, 14\}$ ).

Haben die Algorithmen die Auswahl zwischen sechs verschiedenen Preisoptionen, erlernen sie zunächst den kurzfristigen Gewinn zu erhöhen, indem sie den Wettbewerber immer wieder leicht unterbieten. Fällt der Preis jedoch zu tief, heben die Algorithmen das Preisniveau wieder an. Dieses Vorgehen entspricht den sogenannten *Edgeworth-Preiszyklen*. Der *Edgeworth-Preiszyklus* beschreibt ein wiederkehrendes asymmetrisches Preismuster, das von vielen kleinen Preissenkungen und wenigen großen Preissprüngen gekennzeichnet ist. Von einem hohen Preis beginnend unterbieten sich die Wettbewerber, bis sie den Preis wieder deutlich anheben und sich in der Folge erneut unterbieten.<sup>701</sup> Nachdem die Algorithmen diesen Kreislauf kennengelernt und seine Folgen verstanden zu haben scheinen, stabilisieren sie sich auf einem hohen kollusiven Niveau.<sup>702</sup> Somit führt der Versuchsaufbau ebenfalls dazu, dass die Algorithmen kollusives Verhalten selbstständig erlernen und ein suprakompetitives Gleichgewicht finden. Für den Versuchsaufbau mit einer größeren Auswahl von Preisoptionen ( $k = 12$  und  $k = 24$ ) entspricht der Preiswettbewerb dauerhaft einem *Edgeworth-Preiszyklus*.<sup>703</sup>

Auch die Arbeit von Ibrahim Abada und Xavier Lambin zeigt, dass *Q-learning* Algorithmen kollusive Marktergebnisse erzielen können.<sup>704</sup> Ihr Ansatz unterscheidet sich von den zuvor vorgestellten Simulationen dahingehend, dass sie ein Arbitrage Problem betrachten. In ihrer Simulation untersuchen sie – den Energiemarkt als Vorbild nehmend – den Wettbewerb von Algorithmen, welche homogene Güter auf einem Markt mit preiselastischer

---

700 Vgl. Kapitel D. I. 1. und 3.

701 Das von Eric Maskin und Jean Tirole entwickelte Modell bezeichneten diese auch als eine „Theorie von tacit collusion“, Maskin/Tirole (1988), *Econometrica* 56 (3), 571 (571, 592). Allerdings gibt es auch einige Stimmen, die *Edgeworth-Preiszyklen* als ein Indiz für einen stärkeren Wettbewerb verstanden wissen wollen, Noel, in: Jones (Hrsg.), *The new Palgrave dictionary of economics*, S. 3463.

702 T. Klein, RJE 52 (3) (2021), 538 (549).

703 T. Klein, RJE 52 (3) (2021), 538 (540).

704 Abada/Lambin (2020).

Nachfrage anbieten.<sup>705</sup> Auf dem Energiemarkt werden Speicherkapazitäten genutzt, um so die Marktschwankungen auszunutzen. So wird Energie zu niedrigen Preisen gekauft und gespeichert, um sie bei größerer Nachfrage zu hohen Preisen wieder zu veräußern.

Ihre Ergebnisse zeigen, dass die selbstlernenden Algorithmen in der Lage sind ihre Marktmacht auszuweiten und zu suprakompetitiven Ergebnissen zu gelangen.<sup>706</sup> Allerdings ziehen die Autoren in Zweifel, dass es sich hierbei tatsächlich um erfolgreiche *tacit collusion* handelt. So gehen sie nicht davon aus, dass ihre Algorithmen eine Bestrafungsstrategie erlernt haben und vermuten eher, dass die kollusiven Ergebnisse auf eine unzureichende Erfahrung mit alternativen Strategien zurückzuführen sind.<sup>707</sup>

*John Asker et al.* erweitern den Ansatz von *Calvano et al.* dahingehend, dass sie die Struktur der Algorithmen variieren.<sup>708</sup> Hierbei unterscheiden sie zwischen Algorithmen, die ihre Handlungen synchron beziehungsweise asynchron aktualisieren. Während die asynchron lernenden Algorithmen – entsprechend *Calvano et al.* – ihr Verhalten an der vorgenommenen Handlung und der resultierenden Auszahlung ausrichten, berücksichtigen die synchron lernenden Algorithmen darüber hinaus die Auszahlungen, welche auf alternative Handlungen gefolgt wäre.<sup>709</sup> Berücksichtigen die Algorithmen keine zukünftigen Gewinne ( $\delta = 0$ ), gelangen der synchron lernenden Algorithmen zum Bertrand-Gleichgewicht, wohingegen die asynchron lernenden Algorithmen überwettbewerbliche Preise erzielen.<sup>710</sup> Bei Berücksichtigung zukünftiger Gewinne erhöhen sich die Preise auch bei den synchron lernenden Algorithmen, bleiben aber deutlich hinter den asynchron lernenden Algorithmen zurück.<sup>711</sup> Mit Zunahme der Marktgröße nimmt der Preis ab.<sup>712</sup> Die Autoren kommen zu dem Ergebnis, dass das Verhalten der asynchron lernenden Algorithmen, welche selbst in Szenarien entsprechend eines *one shot games* zu suprakompetitiven Preisen gelangen, dazu führt, dass es für die Unternehmen im Duopol ein Gleichgewicht ist,

---

705 *Abada/Lambin* (2020), S. 8.

706 *Abada/Lambin* (2020), S. 23 ff.

707 *Abada/Lambin* (2020), S. 33 f.

708 *Asker et al.* (2021).

709 *Asker et al.* (2021), S. 13.

710 *Asker et al.* (2021), S. 26 f.

711 *Asker et al.* (2021), S. 32 ff.

712 *Asker et al.* (2021), S. 27.

entsprechend „naive“ asynchrone Algorithmen einzusetzen, um suprakompetitive Preise zu erzielen.<sup>713</sup>

Tobias Werners Arbeit verbindet Simulationen mit einem Laborexperiment und liefert mit einem sehr ähnlichen Aufbau wie *Calvano et al.* Evidenz dafür, dass *Q-learning* Algorithmen in der Lage sind Bestrafungsstrategien zu erlernen und tatsächlich zu mehr Kollusion gelangen als menschliche Entscheider in derselben Marktumgebung.<sup>714</sup> Der Autor untersucht das Verhalten von Algorithmen im Vergleich zu menschlichen Teilnehmern im Labor und kommt zu dem Ergebnis, dass zwei Algorithmen im Duopol signifikant höhere Marktpreise erzielen als zwei menschliche Entscheider.<sup>715</sup> Dabei erlernen die Algorithmen eine Strategie, die der *win-stay, lose-shift* Strategie ähnelt. Bei dieser Strategie behält der Algorithmus eine Entscheidung bei, wenn sie zu einer erfolgreichen Auszahlung geführt hat, wechselt die Strategie allerdings, sofern die Auszahlung nicht zufriedenstellend ist.<sup>716</sup> Weichen Wettbewerber ab, wählt der Algorithmus in den folgenden Perioden den Bestrafungspreis. Haben alle Marktteilnehmer diesen Bestrafungspreis gewählt kehrt der Algorithmus jedoch zu einen kollusiven Preis zurück.

Mit drei Wettbewerbern kommen die Algorithmen ebenfalls zu durchschnittlich leicht besseren Ergebnissen als menschliche Teilnehmer. Allerdings fällt der Preis in den Algorithmen-*Treatments* bei Vergrößerung des Marktes deutlich stärker, als in den menschlichen Gruppen, sodass der Effekt algorithmischer Preissetzung im Triopol nicht mehr signifikant ist.<sup>717</sup> Die von *Calvano et al.* geäußerte Vermutung, dass beim Einsatz von Algorithmen das kollusive Ergebnis mit der Zunahme der Marktgröße weniger stark abnimmt, als im Wettbewerb menschlicher Marktteilnehmer,<sup>718</sup> lässt sich nicht durch die Ergebnisse aus *Werners* Arbeit belegen.

In einer weiteren Simulation mit *Q-learning* Algorithmen betrachten die Autoren *Emilio Calvano et al.* das Verhalten der Algorithmen in Abhängigkeit zur Markttransparenz.<sup>719</sup> Entsprechend *Green* und *Porter* verwenden sie hierzu einen *Cournot*-Markt, bei dem sie variieren, welche Informationen

---

713 *Asker et al.* (2021), S. 40.

714 *Werner* (2021).

715 *Werner* (2021), S. 29.

716 *M. Nowak/Sigmund*, *Nature* 364 (6432) (1993), 56.

717 *Werner* (2021), S. 28.

718 Siehe *Calvano et al.* (2020), *AER* 110 (10), 3267 (3288).

719 *Calvano et al.* (2021), *IJIO* 79, 102712.

den Algorithmen zur Verfügung stehen.<sup>720</sup> Während das Verhalten der Algorithmen bei perfekter Überwachung dem festgestellten Verhalten aus den *Bertrand*-Simulationen<sup>721</sup> entspricht, nimmt die Kollusion bei Wegfall der Informationen ab, wenngleich sie mit 75% des perfekten Kollusionsgewinns noch immer deutlich über den nicht-kooperativen Gleichgewicht liegt.<sup>722</sup>

### 3. Algorithmische Kollusion mittel *deep learning* Algorithmen

#### a) Vorteile gegenüber Q-Learning

*Q-learning* Algorithmen sind relativ unkompliziert und können „mit wenigen Parametern, deren ökonomische Bedeutung klar ist, vollständig charakterisiert werden.“<sup>723</sup> Die einfache Anwendung hat zu einer weiten Verbreitung dieser Technologie geführt. Die sehr aufwendige und langwierige Trainingsphase bringt allerdings erhebliche Schwierigkeiten für ihre Anwendung auf realen Märkten mit sich. Jeder zusätzliche Wettbewerber, eine bessere Erkundung durch eine höhere Zufallsrate sowie die Zunahme an Preisoptionen, sorgen dafür, dass sich die Lernphasen der Algorithmen erheblich verlängern. So ist ein Lernprozess im laufenden Betrieb (*on-the-job*) schwer zu realisieren. Ginge man davon aus, dass ein Unternehmen täglich 30 Preisanpassungen in einer Stunde vornähme,<sup>724</sup> würde ein Algorithmus mit einer durchschnittlichen Lerndauer entsprechend der *Bertrand*-Simulation von *Calvano et al.* erst nach über drei Jahren zu zufriedenstellenden

---

720 So variieren sie zum einen, ob die Algorithmen die gewählte Menge der Wettbewerber in der vergangenen Periode überwachen können, zum anderen, ob sie die Nachfrage kennen. Der aus den Entscheidungen und der Nachfrage resultierende Preis ist stets ersichtlich, *Calvano et al.* (2021), *IJIO* 79, 102712 (7 f.).

721 Siehe *Calvano et al.* (2020), *AER* 110 (10), 3267.

722 *Calvano et al.* (2021), *IJIO* 79, 102712.

723 Aus dem Englischen übersetzt, siehe *Calvano et al.* (2020), *AER* 110 (10), 3267 (3270).

724 Bisher liegen die durchschnittlichen Preisanpassungen im Online-Handel erheblich niedriger. So ergab eine Stichprobe des Statistischen Bundesamtes von 2.680 Produkten aus 14 Internetgeschäften, dass es bei 88 % der beobachteten Produkte zu weniger als 15 Preisanpassungen in 3 Monaten gekommen ist. Beispielhaft für eine sehr häufige Preisanpassung wird ein Produkt mit einer Preisanpassung im Intervall von knapp über 2 Stunden benannt, *Blaudow/Burg* (2018), *WISTA* 2018 (2), 11 (15 ff.); *Chen et al.* analysieren die Preissetzung einiger Händler auf Amazon Marketplace. Hierbei kommen sie zu dem Befund, dass einige der Algorithmen die Preise „dutzende oder sogar hunderte Male am Tag anpassen.“, *Chen et al.*, *WWW '16*, 1339 (1348, 1343).

Ergebnissen gelangen.<sup>725</sup> Bei aufwendigeren Lernumgebungen würden die Algorithmen zum Teil sogar weit über fünf Jahre benötigen, bevor sie in der Lage wären, eine konstante Strategie zu verfolgen.<sup>726</sup> Eine Möglichkeit, dieses Problem zu umgehen, ist ein Training außerhalb des laufenden Betriebes (*off-the-job*), indem der Algorithmus mit historischen oder hypothetischen Daten trainiert und erst im Anschluss im realen Wettbewerb eingesetzt wird. Doch auch hierbei ergibt sich das Problem, der Anpassungsfähigkeit gegenüber neuartigen Marktphänomenen und der sich vom echten Wettbewerb unterscheidenden Trainingsbedingungen. Aus diesem Grund sind entsprechende Algorithmen „unpraktisch für komplexe Marktconstellationen, z.B. in großen Oligopolen.“<sup>727</sup> Dem überlegen können komplexere Algorithmen, wie *deep-Q-networks*, sein, welche in verschiedenen Marktsimulationen bereits zum Einsatz gekommen sind.

*Deep-Q-networks* wurden erstmalig von *Volodymyr Mnih et al.* entwickelt und kombinieren *reinforcement Q-learning* mit *artificial neural networks*.<sup>728</sup> Während beim *Q-learning* die Werte einer Tabelle fortlaufend aktualisiert wurden, werden Tabellen beim *deep-Q-network* durch ein neuronales Berechnungsmodell ersetzt.<sup>729</sup> Dieses dient dazu, sich der unbekanntem Q-Funktion so weit wie möglich anzunähern (Funktionsapproximation).<sup>730</sup> Einer der Vorteile ist, dass der Algorithmus nicht mehr auf die Optimierung einer Zelle pro Periode begrenzt ist und seinen Lernprozess erheblich beschleunigen kann. Außerdem ist der Algorithmus so in der Lage, deutlich größere Zustandsräume zu erlernen. *Mnih et al.* haben gezeigt, dass ihre Methode in 43 von 49 getesteten Atari Spielen besser abschneidet, als die besten Alternativen aus dem *reinforcement learning*.<sup>731</sup>

---

725 Hettich (2021), Fn. 3.

726 So benötigen die Algorithmen bei *Calvano et al.* in der Spitze über 2 Millionen Perioden zum Trainieren, was bei 30 Perioden pro Stunde 7,6 Jahren entspräche, *Calvano et al.* (2020), AER 110 (10), 3267 (3276).

727 *Kastius/Schlosser* (2022), J Rev Pricing Man 21, 50.

728 *Mnih et al.*, Nature 518 (7540) (2015), 529.

729 *Charniak*, Introduction to Deep Learning, S. 119.

730 *Mnih et al.*, Nature 518 (7540) (2015) (529).

731 *Mnih et al.*, Nature 518 (7540) (2015), 529 (530).

b) Der überlegene Algorithmus?

Matthias Hettich nutzt *deep-Q-networks* (DQN) für eine Simulation des Wettbewerbs von Unternehmen in einem *Bertrand*-Markt.<sup>732</sup> Der Autor zeigt, dass zwei DQNs ebenfalls in der Lage sind eigenständig kollusive Strategien zu erlernen. Hierbei wenden die Algorithmen Bestrafungsstrategien an, mit denen sie Abweichungen sanktionieren. Allerdings führt ein „gegenseitiges Verständnis“ dazu, dass die DQNs die Preise anheben, sofern sich diese nahe des *Bertrand*-Gleichgewichts bewegen.<sup>733</sup> In der Folge finden sie so zum kollusiven Gleichgewicht zurück und das selbst dann, wenn zuvor eine Abweichung durch beide Wettbewerber stattgefunden hat. Mit Zunahme der Marktgröße nimmt auch bei Hettichs Simulation der kollusive Gewinn der Unternehmen ab. So beträgt der durchschnittliche zusätzliche Gewinn im Duopol 67%, bei einem Markt mit drei Wettbewerbern liegt er noch bei 57%. Auch beim Wechsel von vier (36%) auf fünf Unternehmen (27%) nimmt der kollusive Gewinn ab, bevor er ab sechs Unternehmen im Markt unter die 20-Prozent-Marke fällt.<sup>734</sup>

Während die Algorithmen bei *Calvano et al.* im Schnitt 850.000 Perioden benötigen, um kollusive Strategien zu erlernen, erhöhen die *deep-Q-networks* bereits nach 20.000 Perioden ihre Preise und erzielen nach weniger als 100.000 Perioden suprakompetitive Gleichgewichte.<sup>735</sup> Hettich kommt somit zu dem Schluss, dass bei 30 Preisanpassungen pro Stunde immerhin nach einem Monat höhere Preise erzielt werden können. Bei einer Betrachtung heterogener Märkte, auf denen ein DQN und ein *Q-learning* Algorithmus für 100.000 Perioden im Wettbewerb stehen, zeigt er, dass der *deep-learning* Algorithmus dem *Q-learning* Algorithmus überlegen ist. Indem das DQN eine kompetitive Strategie anwendet, ist es in der Lage, hohe Gewinne zu erzielen und den *Q-learning* Algorithmus auszubeuten.<sup>736</sup> Hierbei profitiert er von der schnellen Anpassungsfähigkeit seines neuronalen Netzwerks. Stehen zwei DQNs mit einem kompetitiven statischen Algorithmus im Wettbewerb, gelingt es ihnen auch, suprakompetitive Marktergebnisse zu erzielen. Der statische Algorithmus wendet hierbei eine Penetrationsstrategie an, bei der er den Preis der Konkurrenten in der vorangegangenen Periode

---

732 Hettich (2021).

733 Hettich (2021), S. 11.

734 Hettich (2021), S. 13.

735 Hettich (2021), S. 10.

736 Hettich (2021), S. 14.

stets unterbietet. Die dynamischen Algorithmen stellen fest, dass sie das Verhalten des statischen Algorithmus nicht wirksam bestrafen können und lassen diesen in der Folge einen Gewinn nahe des Monopolgewinns ( $> 80\%$ ) realisieren.<sup>737</sup> Hierdurch sind die DQNs in der Lage, trotz des kompetitiven Wettbewerbers suprakompetitive Preise zu erzielen und einen Gewinn von durchschnittlich 61% zu realisieren. Dieser Wert liegt etwas unter dem Gewinn zweier DQN-Algorithmen im Duopol, aber über dem Gewinn im Tripol.

Alexander Kastius und Rainer Schlosser nutzen ebenfalls DQNs und darüber hinaus noch einen weiteren *deep reinforcement learning* Algorithmus, *Soft Actor Critic* (SAC),<sup>738</sup> und testen diese in verschiedenen Marktmodellen.<sup>739</sup> Zur Untersuchung kollusiven Verhalten ergänzen sie einen Wettbewerber, dessen Strategie eine Kollusion zu einem festen Betrag anbietet und die Algorithmen im Wettbewerb immer wieder unterbietet. Während der SAC Algorithmus bei einer kompetitiven Strategie bleibt und nicht zu einem kollusiven Gleichgewicht findet, lässt sich der DQN-Algorithmus auf den überwettbewerblichen Preis hochziehen.<sup>740</sup> Somit zeigt das Experiment, dass ein relativ einfache kollusive Strategie eines statischen Algorithmus dem DQN in dieser Marktumgebung eine Kollusion „aufzwingen“ kann. Die Autoren kommen zu dem Schluss, dass die Unternehmen „keine Informationen außer Preisreaktionen austauschen müssen, um ein Kartell zu bilden und den Wettbewerb auszuschalten.“<sup>741</sup> Bei der Untersuchung des Wettbewerbs zwischen den beiden *deep reinforcement learning* Algorithmen schafft es keiner der Algorithmen, eine überlegene Strategie zu entwickeln und den anderen Algorithmus zu „besiegen“. Beide Algorithmen finden jeweils Antworten auf die Strategie des Gegenübers, ohne dass ein Zyklus oder Gleichgewicht festzumachen ist.<sup>742</sup>

#### 4. *Overfitting* und mögliche Probleme in der Praxis

Die bisherigen Experimente haben gezeigt, dass (*deep*-)*reinforcement learning* Algorithmen kollusive Strategien erlernen können. Ein Problem

---

737 Hettich (2021), S. 15.

738 Siehe hierzu Haarnoja et al. (2018).

739 Kastius/Schlosser (2022), J Rev Pricing Man 21, 50.

740 Kastius/Schlosser (2022), J Rev Pricing Man 21, 50 (10).

741 Aus dem Englischen übersetzt, siehe Kastius/Schlosser (2022), J Rev Pricing Man 21, 50 (10).

742 Kastius/Schlosser (2022), J Rev Pricing Man 21, 50 (11).



bezüglich der Übertragbarkeit der Ergebnisse könnte jedoch darin begründet sein, dass sich die Algorithmen hierbei stets in ihrem Trainingsumfeld befunden haben.<sup>743</sup> Nachdem die Algorithmen das Training abgeschlossen hatten, setzten sie den Wettbewerb in derselben Umgebung fort.<sup>744</sup> Bei entsprechenden Simulationen lässt sich das Ergebnis somit nicht auf das Problem des *overfitting* überprüfen. *Overfitting* bezeichnet eine übermäßige Anpassung des von den Algorithmen entwickelten Modells an die Trainingsumgebung.<sup>745</sup> Sofern ein selbstlernender Algorithmus überangepasst auf seinen Trainingsdatensatz ist, liefert er sehr gute Ergebnisse innerhalb der Trainingsumgebung, allerdings schafft er es nicht, in einer anderen Umgebung, mit anderen Voraussetzungen und Einflüssen, entsprechende Erfolge zu übertragen.<sup>746</sup> *Nicolas Eschenbaum et al.* nehmen dies zum Anlass, kollusive *Q-learning* Modelle in einem Trainingsumfeld zu trainieren und in der Folge in einem leicht angepassten Testumfeld auszuprobieren.<sup>747</sup> Hierbei orientieren sich die Autoren an dem Versuchsaufbau von *Calvano et al.*<sup>748</sup> und nehmen einen *Bertrand*-Markt als Grundlage. Im Ergebnis schaffen es die Algorithmen aufgrund des *overfitting* nicht, ihr kollusives Verhalten in die leicht veränderte Umgebung zu übertragen. Unternehmen könnten dieses Problem jedoch verhindern, indem sie die Strategiemöglichkeiten der Algorithmen beschränken und eine Beobachtung der Preise des Wettbewerbers verhindern.<sup>749</sup> Die Autoren nehmen an, dass algorithmische Systeme „auf einfacheren Mustern beruhen und aufeinander abgestimmt sein“ müssen, um Kollusion in unterschiedlichen Umgebungen erfolgreich zu erzielen.<sup>750</sup>

---

743 Eine Ausnahme stellt *Werner* (2021) dar, dessen Ergebnisse allerdings ebenfalls auf eine schlechte Anpassung der Algorithmen auf neue Marktgegebenheiten hindeutet (s.u.).

744 *Eschenbaum et al.* (2022).

745 *Xue Ying* (2019), *Journal of Physics: Conference Series* 1168, 22022.

746 Verschiedene Untersuchungen haben sich bereits mit dem Problem des *overfitting*s bei selbstlernenden Algorithmen auseinandergesetzt. Sie statt vieler *C. Zhang et al.* (2018).

747 *Eschenbaum et al.* (2022).

748 *Calvano et al.* (2020), *AER* 110 (10), 3267.

749 Die Algorithmen können dann lediglich ihren eigenen Preis aus der vergangenen Periode beobachten, *Eschenbaum et al.* (2022), S. 20 f.

750 Aus dem Englischen übersetzt, siehe *Eschenbaum et al.* (2022), S. 24.

### III. Daten aus dem Feld

Die bisher dargestellte Literatur hat sich mit den (theoretischen) Möglichkeiten algorithmischer Preissetzung auseinandergesetzt und ihr kollusives Potenziale in Modellmärkten untersucht. Zur Beobachtung tatsächlicher Marktgegebenheiten werden darüber hinaus Daten im Feld erhoben und analysiert. Mit Hilfe dieser lässt sich die tatsächliche Verbreitung algorithmischer Preissetzung untersuchen und ihre vermeintliche Auswirkung auf den Wettbewerb analysieren. Hierbei gilt es jedoch zu beachten, dass auf realen Märkten – anders als in kontrollierten Laborumgebungen – unterschiedliche Einflussfaktoren gleichzeitig ihre Wirkung entfalten und sich kausale Zusammenhänge anhand von Felddaten nicht feststellen lassen. Auf dem Gebiet der algorithmischen Preissetzung gibt es bisher nur wenig empirische Feldforschung. Die wichtigsten Veröffentlichungen werden im Folgenden dargestellt.

#### 1. Verbreitung algorithmischer Preissetzung auf dem *Amazon Marketplace*

*Le Chen et al.* betrachten den Wettbewerb auf der digitalen Verkaufsplattform *Amazon Marketplace*.<sup>751</sup> Hierfür beobachten die Autoren im Jahr 2014 über einen Zeitraum von vier Monaten Händler der 1641 meistverkauften Produkte. Auf Grundlage zweier Kriterien analysieren sie, wie viele der Händler bei ihrer Preissetzung auf Algorithmen zurückgreifen. In einem ersten Schritt untersuchen *Chen et al.*, inwieweit die Preissetzung eines Händlers abhängig von den Preisen der Wettbewerber ist. In einem weiteren Schritt wird überprüft, wie häufig Preisanpassungen vorgenommen werden. Basiert die Preissetzung eines Händlers in einem gewissen Umfang auf den Preisen seiner Konkurrenz und werden seine Preise darüber hinaus hinreichend häufig angepasst, so gehen die Autoren davon aus, dass algorithmische Preissetzung zum Einsatz kommt.<sup>752</sup> Mit diesem Vorgehen identifizieren sie über 500 Händler (> 30%), die mit großer Wahrscheinlichkeit bei mindestens einem ihrer Produkte Algorithmen zur Festlegung der Preise verwenden.<sup>753</sup>

---

751 *Chen et al.*, WWW '16, 1339.

752 *Chen et al.*, WWW '16, 1339 (1344-1345).

753 Die Autoren der Studie gehen davon aus, dass Gründe hierfür das Verkaufsvolumen sowie die Bewertungen sind, *Chen et al.*, WWW '16, 1339 (1348).

## 2. Die Gefahr des Einfachen

*Marcel Wieting* und *Geza Sapi* nehmen in ihrer Untersuchung die größte niederländische Online-Verkaufsplattform *bol.com* in den Blick.<sup>754</sup> Zwischen 2018 und 2020 sammeln die Autoren zweimal über einen Zeitraum von einem Monat Daten von 2 846 Produkten auf der Plattform, um den Einsatz algorithmischer Preissetzung zu untersuchen. Entsprechend *Chen et al.* identifizieren die Autoren den Einsatz algorithmischer Preissetzung anhand einer Zunahme der Preisfrequenz sowie der Korrelation der Preise mit Preisen der Wettbewerber.<sup>755</sup> Bei der ganz überwiegenden Mehrheit der Produktmärkte (> 80%) können die Autoren keinen Algorithmus identifizieren, während es einige Märkte gibt, auf denen ein Händler algorithmische Preissetzung einzusetzen scheint (je nach Definition circa 5-15%) und wenige Märkte auf denen zwei oder drei Händler automatisiert Preise festzusetzen scheinen (< 2%).<sup>756</sup> Eine Analyse der Märkte mit algorithmischer Preissetzung ergibt, dass Algorithmen im Monopol deutlich niedrigere Preise verlangen, als Monopolisten ohne algorithmische Preissetzung auf vergleichbaren Märkten.<sup>757</sup> Allerdings scheinen in Oligopolen Händler mit Preissetzungssoftware häufiger die *Buy Box* zu gewinnen. Darüber hinaus scheinen Algorithmen von der wechselseitigen Präsenz weiterer Algorithmen zu profitieren, indem Märkte mit Algorithmen höhere Preise und Margen realisieren.<sup>758</sup>

Des Weiteren betrachten *Wieting* und *Sapi* die Preisstrategien der Algorithmen für 300 Produkte und können wiederkehrende Muster feststellen. Besonders verbreitet scheinen dabei sehr kurz andauernde Preissprünge zu sein.<sup>759</sup> Die Autoren erkennen bei circa der Hälfte der algorithmisch bepreisten Produkte Phasen, in denen die Preise kurzzeitig fallen oder ansteigen.<sup>760</sup> In 11% der Märkte mit algorithmischer Preissetzung stellen sie darüber hinaus einen hohen Anstieg des Preises, gefolgt von einer Abwärtsspirale entsprechend

---

754 *Wieting/Sapi* (2021).

755 Im Gegensatz zu *Chen et al.* müssen die beiden Faktoren hierbei nicht zwingend kumulativ vorliegen, sodass z.B. eine sehr hohe Preissetzungsfrequenz ausreichen kann, damit der Einsatz von Algorithmen angenommen wird. Insgesamt liefern die Autoren drei unterschiedliche Definitionen, anhand derer sich drei verschiedene Verteilungen ergeben, *Wieting/Sapi* (2021), S. 15 ff.

756 *Wieting/Sapi* (2021), S. 17 ff.

757 *Wieting/Sapi* (2021), S. 2.

758 *Wieting/Sapi* (2021), S. 25 ff.

759 *Wieting/Sapi* (2021), S. 19 ff.

760 *Wieting/Sapi* (2021), S. 20 mit dem Hinweis, dass auch *Chen et al.* von entsprechenden Preissprüngen berichtet, *Chen et al.*, WWW '16, 1339 (1341).

einem *Edgeworth*-Preiszyklus fest.<sup>761</sup> In 6% der Fälle beobachten die Autoren einen umgekehrten Zyklus, indem der Preis niedrig startend ansteigt und ab einer gewissen Höhe auf den Ausgangspunkt zurückfällt.<sup>762</sup> Auch ein vorübergehendes, aber sehr häufiges Hin- und Herspringen zwischen zwei Preisen sowie häufige und scheinbar zufällige Preiswechsel werden als Preismuster beobachtet. Insgesamt schlussfolgern die Autoren, dass die beobachteten Händler „*relatively unsophisticated*“ Preissetzungssoftware verwenden, die aus einer endlichen Menge von „*if-then statements*“ zu bestehen scheint.<sup>763</sup>

### 3. Hohe Preise aufgrund zyklischer Strategien

*Leon A. Musolff* verwendet Daten eines *Repricing*-Unternehmens, das Drittanbieter auf der Plattform *Amazon Marketplace* im Preiswettbewerb unterstützt.<sup>764</sup> Er nutzt die Daten um Preisstrategien zu analysieren und auf dieser Grundlage die Entwicklung der Märkte in einem Modell zu prognostizieren.<sup>765</sup> Der Autor zeigt zwei häufig angebotene und verwendete algorithmische Preissetzungsstrategien auf: Eine zyklische Preisstrategie, sowie eine Strategie des Unterbietens.<sup>766</sup> Sowohl bei der unterbietenden Strategie, als auch der zyklischen Strategie werden die Preise für das Unternehmen stets entsprechend einem vorgegebenen Betrag unterhalb der Preise der Wettbewerber festgelegt. Den zyklischen Strategien ist darüber hinaus ein kritischer Wert oder ein bestimmtes Ereignis vorgegeben, ab dem der eigene Preis auf einen hohen Preis „zurückgesetzt“ wird.<sup>767</sup>

Da auch die Algorithmen mit einer unterbietenden Strategie ihre Preise in Abhängigkeit der Wettbewerber bestimmen, kommt es auch hier zu einer zyklischen Preissetzung.<sup>768</sup> *Musolffs* Daten zeigen, dass an Tagen des Zurücksetzens der Preise die Preise des Unternehmens im Schnitt 7,75%

---

761 *Wieting/Sapi* (2021), S. 20 f; vgl. hierzu auch die Ergebnisse bei *T. Klein*, RJE 52 (3) (2021), 538.

762 *Wieting/Sapi* (2021), S. 21.

763 *Wieting/Sapi* (2021), S. 40 f.

764 *Musolff* (2021), S. 12.

765 *Musolff* (2021); der Datensatz umfasst Preisänderungen von 319.514 Produkten zwischen dem 26.8.2018 und dem 30.10.2019.

766 *Musolff* (2021), S. 10 f.

767 Bezüglich der Ereignisse, die ein Zurücksetzen auslösen können, kommen beispielsweise gelegentliche Anpassungen zu Zeiten geringer Nachfrage in Betracht. *Musolff* (2021), S. 10 ff.

768 Vgl. hierzu auch *Wieting/Sapi* (2021); *T. Klein*, RJE 52 (3) (2021), 538.

höher sind, als an anderen Tagen.<sup>769</sup> Darüber hinaus nehmen auch die Preise der Wettbewerber an diesen Tagen um durchschnittlich 1,09% zu.<sup>770</sup> *Musolff* modelliert in der Folge den Wettbewerb der Unternehmen in Bezug auf die Übertragung der Preisentscheidung an einen Algorithmus. Sein Modell geht davon aus, dass der Anstieg der Preise nicht an die Grenzkosten gekoppelt und keine zwingende Folge ruinösen Wettbewerbs ist, sondern der Zyklus so festgelegt wird, dass er Durchschnittspreise nah am Monopolpreis zur Folge hat.<sup>771</sup> Der Autor kommt zu dem Ergebnis, dass es bei der Auswahl zwischen den beiden aufgezeigten Strategien sowie einer passiven Strategie eines festgelegten Preises langfristig zu einer Verbreitung der beiden algorithmischen Preissetzungstrategien und somit einer zyklischen Preissetzung kommen sollte.<sup>772</sup>

*Musolff* vermutet, dass der Wandel auf dem von ihm untersuchten Markt gerade erst begonnen hat.<sup>773</sup> Die Einführung dieser Strategien könnte zu anfänglich geringeren Preisen führen und die Täuschung hervorrufen, es gäbe keinen Grund zur Sorge.<sup>774</sup> Langfristig ergäben sich aus seinem Modell hingegen erhebliche Wohlfahrtsverluste. Sofern sich seine Befunde in weiteren Untersuchungen bestätigen würden, „sollte dies als ernsthafte Warnung vor den möglichen Auswirkungen einer automatischen Preisanpassungssoftware auf die gesamtgesellschaftliche Wohlfahrt“ verstanden werden.<sup>775</sup>

#### 4. Der Preisanstieg an der Tankstelle

Algorithmen werden ebenfalls in einigen Bereichen des stationären Handels zur Preissetzung eingesetzt. So bietet unter anderem der dänischen Softwareentwickler *a2i systems A/S* Preissetzungssoftware für Tankstellen an, sodass diese mit Hilfe künstlicher Intelligenz automatisiert Preise festlegen lassen können.<sup>776</sup> *Stephanie Assad et al.* untersuchen die Auswirkungen algorithmischer Preissetzung auf die deutschen Tankstellenmärkte.<sup>777</sup> Grundlage hierfür bilden die Daten der Markttransparenzstelle für Kraftstoffe (MTS-

---

769 *Musolff* (2021), S. 15.

770 *Musolff* (2021), S. 15.

771 *Musolff* (2021), S. 39.

772 *Musolff* (2021), S. 19 ff.

773 *Musolff* (2021), S. 2.

774 *Musolff* (2021), S. 27.

775 Aus dem Englischen übersetzt, siehe *Musolff* (2021), S. 38.

776 Siehe <https://www.a2isystems.com/products/>.

777 *Assad et al.* (2020).

Kraftstoffe),<sup>778</sup> wodurch den Autoren Preisinformationen über alle gängigen Kraftstoffe von 16 661 Tankstellen in Deutschland in minütlichen Intervallen zur Verfügung stehen. In einem ersten Schritt versuchen *Assad et al.* zu identifizieren, welche Tankstellen ab welchem Zeitpunkt automatisierte Preissetzungen eingesetzt haben und in einem weiteren Schritt, welche Auswirkungen die Einführung auf den Wettbewerb hat.<sup>779</sup> Zur Identifizierung der Einführung betrachten die Autoren das Preissetzungsverhalten der Tankstellen und nehmen eine Adaption algorithmischer Preissetzung an, sofern sich zwei der drei folgenden Faktoren innerhalb eines Zeitraums von acht Wochen zugleich umgestellt haben: Zunächst wird überprüft, ob die Frequenz der Preissetzung erhöht wurde, darüber hinaus wird untersucht ob sich die Sprünge innerhalb einer Preisanpassung reduziert haben und zuletzt, ob eine schnellere Reaktion auf Preisanpassungen der Wettbewerber erfolgte. Die Analyse der Autoren erlaubt die Vermutung, dass 4 441 Tankstellen ( $\approx 30\%$ ) ihre Preissetzung auf Algorithmen umgestellt haben, der Großteil von ihnen in der Mitte des Jahres 2017.<sup>780</sup>

In einem nächsten Schritt betrachten *Assad et al.* die Veränderung der täglichen Margen. Auf monopolistischen Märkten lässt sich nach der Umstellung kein signifikanter Anstieg der Margen feststellen. Stellt eines von zwei Unternehmen in einem Duopol auf algorithmische Preissetzung um, lässt sich ebenfalls keine Veränderung der Margen feststellen.<sup>781</sup> In einem Duopol, in dem beide Unternehmen einen Algorithmus eingeführt haben, steigen die Margen allerdings durchschnittlich um 29% an. Aufgrund der Tatsache, dass sich die Margen im ersten Jahr nicht verändert haben und erst danach langsam ansteigen, gehen die Autoren davon aus, dass die Algorithmen nicht – wie bei *Waltman* und *Kaymak*, *Asker et al.* und anderen<sup>782</sup> – daran scheitern, effektiv zu konkurrieren, sondern vielmehr lernen zu

---

778 Die Markttransparenzstelle für Kraftstoffe ist eine Organisation des Bundeskartellamtes und dient der Förderung der Transparenz auf Seiten der Verbraucher. Tankstellenbetreiber übermitteln seit 2013 in Echtzeit Preisänderungen der Kraftstoffe an die Markttransparenzstelle, welche diese über Drittanbieter an die Verbraucher weiterleitet, siehe *Deutscher Bundestag*, Unterrichtung durch die Bundesregierung, 3.8.2018; in Kapitel E. I. 1. wird näher auf das Konzept eingegangen.

779 *Assad et al.* (2020), S. 12 f.

780 *Assad et al.* (2020), S. 22 f.

781 *Assad et al.* (2020), S. 29 ff.

782 *Waltman/Kaymak* (2008), J.E.D.C. 32 (10), 3275; *Asker et al.* (2021); auch *Karsten Hansen et al.* zeigen auf, dass selbstlernende Algorithmen zu suprakompetitiven Preisen gelangen können, indem sie aufgrund eines unzutreffenden Modells die Preissensitivität des Marktes überbewerten. *K. T. Hansen et al.* (2021), *MarkSci* 40 (1), 1.

kolludieren.<sup>783</sup> Somit scheint die Adaption algorithmischer Preissetzung *tacit collusion* im Tankstellenmarkt zu befördern, sofern beide Unternehmen eines Duopols auf Algorithmen zurückgreifen.<sup>784</sup>

#### IV. Zwischenergebnis

Die Potenziale algorithmischer Preissetzung wurden in unterschiedlichen wissenschaftlichen Studien untersucht. Sowohl theoretische Modelle als auch Marktsimulationen und empirische Untersuchungen realer Märkte haben sich mit der algorithmischen Preissetzung auseinandergesetzt.

In der theoretischen Literatur wurden unter anderem die Auswirkungen einer höheren Transparenz sowie einer besseren Vorhersage zukünftigen Nachfrageverhaltens durch Algorithmen analysiert. Hierbei zeigen sich gegenläufige Effekte: Die bessere Vorhersage durch Algorithmen kann wettbewerbsförderndes Potenzial entfalten, indem es den Anreiz für eine Abweichung erhöht. Zugleich können die in Aussicht gestellten zukünftigen Gewinne sowie eine bessere Überwachung der Wettbewerber eine Kollusion befördern und stabilisieren. Darüber hinaus deuten theoretische Modelle darauf hin, dass die höhere Frequenz algorithmischer Preissetzung sowie die zeitlich versetzte Preissetzung der aufeinander reagierenden Algorithmen suprakompetitive Marktergebnisse wahrscheinlicher macht.

In Simulationen wurde insbesondere die Fähigkeit selbstlernender Algorithmen untersucht, selbstständig kollusive Marktergebnisse zu erzielen. Eine Vielzahl entsprechender Untersuchungen zeigt, dass *Q-learning* Preissetzungsalgorithmen nach einem längeren Lernprozess eigenständig und ohne eine entsprechende Programmierung zu einem erhöhten Preisgleichgewicht gelangen können. Ebenso zeigt sich, dass selbstlernende Algorithmen eigenständig Bestrafungsstrategien entwickeln können, die darauf hindeuten, dass die hohen Preise nicht Folge eines Scheiterns am Wettbewerb, sondern eines aktiven Erlernens kollusiven Verhaltens sind. Insbesondere die Ergebnisse zu in heterogenen Märkten ausbeutenden und in homogenen Märkten kollusiv spielenden DQNs legen nahe, dass selbstlernende Algorithmen grundsätzlich dazu in der Lage sind, *tacit collusion* eigenständig zu erlernen.

---

783 Assad et al. (2020), S. 43.

784 Denkbar ist hierbei auch, dass es sich um Algorithmen desselben Anbieters handelt und so ein Fall entsprechend des *hub-and-spoke* Szenarios vorliegt.

Allerdings handelt es sich bei diesen Simulationen um stark vereinfachte Modellmärkte, deren Eigenschaften sich deutlich von realen Marktgegebenheiten unterscheiden.<sup>785</sup> Sowohl die Zunahme der Marktgröße, als auch die Erweiterung des Strategieraums durch zusätzliche Marktfaktoren schwächt die kollusiven Ergebnisse algorithmischer Systeme ab. Insbesondere in heterogenen Marktzusammensetzungen scheinen selbstlernende Algorithmen Schwierigkeiten zu haben, kollusive Gleichgewicht zu erreichen. Darüber hinaus ist unklar, inwiefern es in einigen der Simulationen zu einem *overfitting*-Effekt gekommen sein könnte, der die Übertragbarkeit der Ergebnisse auf Marktgegebenheiten außerhalb der Trainingsumgebung in Frage stellt. Desweiteren lässt die lange Lernphase der in den vorgestellten Simulationen verwendeten Algorithmen einen Einsatz auf realen Märkten unwahrscheinlich erscheinen. Die verwendeten Algorithmen waren erst nach langen *off-the-job* Trainingsphasen in der Lage, kollusive Ergebnisse zu erzielen. Ein *on-the-job* Lernen dürfte mit den hier dargestellten Algorithmen keine für Unternehmen realistische Option darstellen. Selbst den *Q-learning* Algorithmen überlegene DQNs benötigen mehrere Wochen mit einer Vielzahl täglicher Preisänderungen, ehe sie kollusive Gleichgewichte erzielen können.

Dennoch zeigen die Ergebnisse der Simulationen das kollusive Potenzial algorithmischer Systeme an. Insbesondere die Vermutung, dass es für Unternehmen sinnvoll erscheinen kann, in ihren Fähigkeiten beschränkte Algorithmen einzusetzen, welche aufgrund ihrer Unbedarftheit eine *tacit collusion* erreichen, dürfte die Sorge vor algorithmischer Kollusion zusätzlich verstärken. Vor allem einfache Strategien scheinen ein vielversprechendes Mittel zu sein, um *tacit collusion* in unterschiedlichen Marktgegebenheiten zu ermöglichen. Simulationen haben gezeigt, dass relativ einfache – *tacit collusion* belohnende – Strategien auch selbstlernende Algorithmen von einer Kollusion überzeugen können.

Empirische Untersuchungen haben die Verbreitung algorithmischer Preissetzung auf realen Märkten sowie ihre Auswirkungen auf den Wettbewerb untersucht. Es zeigt sich, dass eine Vielzahl von Unternehmen bereits seit längerem auf entsprechende Unterstützung zurückzugreifen scheint. Wenngleich die Vorteile algorithmischer Preissetzung darauf hindeuten, dass sich diese langfristig flächendeckend verbreiten könnte, scheint sie derzeit jedoch noch nicht alleinig die Preissetzung zu bestimmen. Die Daten deuten darauf hin, dass der Einsatz algorithmischer Preissetzung auf realen Märkten zu höheren Preisen führen kann. Insbesondere für Märkte, auf

---

785 Vgl. Schwalbe/Zimmer, Kartellrecht und Ökonomie, S. 541.



denen Algorithmen gegeneinander im Wettbewerb stehen, konnten für Online Plattformen leichte und für den deutschen Tankstellenmarkt starke Preisanstiege festgestellt werden.

Die bisherigen Erkenntnisse lassen vermuten, dass gerade einfache Strategien dazu beitragen können, hohe Gleichgewichtspreise in einem Markt durchzusetzen. Durch die Reduzierung möglicher Strategieoptionen, direkte Abhängigkeiten in der Preissetzung, die leichte Interpretierbarkeit der Strategien sowie die Selbstbindung der Unternehmen wird eine Koordination über den Markt erheblich erleichtert. Es ist deshalb zu erwarten, dass auf digitalen Märkten vermehrt Preiszyklen auftreten, welche im Ergebnis zu Lasten der Verbraucher sowie des Wettbewerbs gehen. Darüber hinaus scheint es für Unternehmen leichter eine kollusive Absicht zu bekunden und sich durch den Algorithmus an diese zu binden. Indem die Wettbewerber durch die eigene Preisüberwachung die Preissetzung analysieren und interpretieren können, liegt es an ihnen, ein solches „Angebot“ für überwettbewerbliche Preise anzunehmen. Es ist zu vermuten, dass sich die tatsächliche wettbewerbschädigende Wirkung erst mit der weiteren Verbreitung algorithmischer Preissetzung zeigen wird.<sup>786</sup>

---

786 Vgl. *Musolff* (2021), S. 2.

