Decentralization vs. Blockchain Neutrality: The Unequal Burden of Ethereum's Market Mechanism on dApps*

Hanna Halaburda† Daniel Obermeier‡

January 26, 2024

Abstract

Blockchain platforms have been hailed for ushering our digital economy into Web3.0, a more transparent, inclusive, and equal era of the Internet. To do so, blockchain platforms aim to disintermediate digital platforms by substituting a centralized authority with a network of peers who collectively validate and record transactions based on rules predefined in a public protocol. As this decentralization necessitates limiting the transaction supply, most blockchain platforms rely on a market mechanism to allocate the transaction recording service. We study how this market mechanism influences what type of applications can be sustainably offered on such platforms. Based on a sample of 1,560 applications running on Ethereum, the most popular blockchain platform, we show that allocating transactions by a market mechanism favors some types of applications over others and reduces the heterogeneity of platform complements. This finding highlights a trade-off between decentralization and *blockchain neutrality*—a new notion we introduce as the principle that all actors on a blockchain platforms are treated equally. This trade-off is especially problematic as blockchain platform providers have no governance tools to mitigate this market discrimination and allow all types of applications to be offered on the platform.

Keywords: blockchain, decentralization, blockchain neutrality, platform orchestration, decentralized applications

*We thank Yannis Bakos, Chris Conlon, Guillaume Haeringer, Joachim Henkel, Aija Leiponen, Natalia Levina, Raveesh Mayya Foster Provost, and participants of the NYU Blockchain research group for their valuable discussions and comments. Further we thank all participants of the TUM Blockchain Workshop 2021, CIST 2022, WISE 2022, NBER Digitization Workshop 2023, and the EU-DPRN 2023 for their valuable feedback.

†NYU Stern School of Business; hhalaburda@gmail.com

‡NYU Stern School of Business; do2169@nyu.edu

1 Introduction

Blockchain technology aims to disintermediate digital platforms by substituting a centralized authority with a network of peers who collectively validate, enforce, and record transactions based on rules predefined in a public protocol (Nakamoto 2008). According to its proponents, this disintermediation limits platform providers' excessive power and allows platform architects to design platforms where the created value is distributed more evenly among all participating parties (Catalini and Tucker 2018, Vergne 2020). Based on these promises, Gavin Wood, one of Ethereum's founding fathers, envisioned that blockchain technology will usher us into Web3.0, a new, more transparent, inclusive, and democratic version of the Internet where nobody has and advantage over anyone else (Wood 2014). With this vision, he spurred a whole new industry that aims to disrupt prevailing digital platforms across industries such as finance, gaming, insurance, and health and foster the Internet's democratization.

However tempting this vision of a fully decentralized version of the Internet might be, it is also important to consider that disintermediation is no panacea free from limitations. For example, it is commonly known that blockchain platforms bear higher coordination costs as protocol changes require a consensus by the community, and higher storage costs as the same data is replicated across different nodes (Pereira et al. 2019). Further, while limiting the platform providers' power through decentralization can prevent them from exploiting their users (Wood 2014), it also removes their governance tools often necessary to orchestrate a healthy ecosystem of complements (Cennamo and Santaló 2019, Tiwana et al. 2010, Staub et al. 2022). In this study, we introduce another important trade-off: the trade-off between decentralization and what we refer to as *blockchain neutrality*—the principle that no complementor is prioritized over other complementors.

To replace a central platform authority with a peer-to-peer network that collectively validates, enforces, and records transactions, blockchain protocols typically limit the supply of transactions. Limiting the supply is necessary to allow as many validators as possible to join and help maintain the network. Not limiting the supply of transactions would favor validators with the most powerful machines as they could increase the transaction throughput up to a point where less powerful machines fail to stay synchronous to the longest chain preventing them from contributing new blocks and fostering the network's re-centralization. To allocate the limited transaction supply, most blockchain platforms like Bitcoin and Ethereum rely on a market mechanism to determine the price for transacting on the platform (Buterin 2014, Nakamoto 2008). The limited supply of transactions in combination with a market mechanism has led to skyrocketing transaction fees in the past. As a result, some dApp providers saw a decline in their dApp's usage and decided to leave the platform. Most prominently, Dapper Labs (dapperlabs.com), the developer of the CryptoKitties collectibles game, left and developed their own blockchain platform called "Flow". At the time of writing, Flow hosts 427 dApps.[1] To understand if the market exit of dApp providers like Dapper Labs underlies a systematic pattern, we investigate the consequences of decentralizing a transaction platform by replacing a centralized platform intermediary with a network of peers that allocates the limited supply of transactions through a market mechanism. Specifically, we are interested in whether such platforms are able to host a variety of different applications and become the general-purpose infrastructure necessary to deliver the promises of Web 3.0.

Although the platform literature emphasizes the importance of a healthy and diverse ecosystem of complements for platforms to prosper (Rietveld et al. 2020), available research on blockchain platforms mainly focuses on validators, users, and the overall stability of the transaction fee market

[1] Cf. https://www.flowverse.co/. Most of these dApps are gaming and NFTs collectibles dApps and are exclusive to the Flow blockchain. The most famous example is NBA Top Shots (nbatopshot.com), which has already attracted more than USD1.1b in sales until January 2023 (https://www.flowverse.co/applications/nba-top-shot).

(e.g., Basu et al. 2019, Easley et al. 2019, Ilk et al. 2021). How the usage of a market mechanism to allocate transactions affects complement heterogeneity, and thus the platform ecosystems health and appeal, remains understudied. To address this gap, we ask the following research questions: how does a market mechanism for the decentralized validation of transactions affect the usage of platform complements? What complements will be offered on blockchain platforms in the long run?

To answer these research questions, we draw parallels to the literature on net neutrality (Choi et al. 2018, Guo et al. 2012, Krämer and Wiewiorra 2012, Reggiani and Valletti 2016), introduce the notion of *blockchain neutrality*, and argue that current approach to reach decentralization by limiting transaction supply and allocating it through a market mechanism precludes blockchain neutrality as it favors some complements over others. Our core argument is that a transaction fee market mechanism only prioritizes complements based on the transaction fee sensitivity of their users. Whereas this leads to an efficient allocation for homogeneous transactions (e.g., like transactions on the Bitcoin network), it can lead to long-run inefficiencies in the case of heterogeneous complements. These inefficiencies occur because the market mechanism favors some types of complements over others based on their current user's transaction fee sensitivity but not on the dApp's overall quality and the value the complement might provide in the future. While platform complements generally face competition within the same category, the market mechanism for fees imposes an additional, cross-category externality in the form of congestion costs. With that, if one complement attracts more users and thus increases the demand for transactions, the transaction fees for all other complements—irrespective of their category—rise as well, as they all compete for the same supply of transactions.

This externality is problematic because, as we show in Section 7, there are several characteristics other than the quality of a complement that determine its users' sensitivity toward transaction fees and thus if a dApp will be used. And as the platform provider does not have any governance tools to protect complements from this externality, if necessary, even some high quality dApps that would benefit the platform in the long run might struggle to attract users in the short run and have to leave the platform. As we know from the literature on platform competition, users do not only care about complements' quantity and quality (Jin-Hyuk Kim et al. 2014) but also about their diversity (Rietveld et al. 2019). Thus, an unregulated reduction of complement heterogeneity and exits due to other reasons than complement's heterogeneity can hamper a platform's potential to leverage same-side and cross-side network effects (Rietveld and Schilling 2020). Further, as the net neutrality literature emphasizes, losing innovation on the edges of the platform can hamper the platform's overall innovation capabilities (Guo et al. 2012). Finally, it raises the concern whether blockchain platforms that rely on a market mechanism to enforce the execution of transactions can become a general purpose technology that hosts all types of applications and serves as the backbone infrastructure for Web 3.0.

We provide empirical evidence for our arguments by relying on the context of the Ethereum blockchain. Ethereum offers a unique opportunity to study our research questions for three reasons. First, Ethereum was the first decentralized blockchain platform to enable smart contracts, computer scripts that enable complementors to offer web applications to the platform users (Buterin 2014). As these applications run on top of a blockchain, they are called decentralized applications (dApps) (Wu et al. 2021). Accordingly, Ethereum qualifies as a multi-sided platform where complementors can offer arbitrary services to platform users. Second, Ethereum uses a market mechanism to allocate the limited supply of transactions among transaction senders (i.e., platform users). This market mechanism resembles a first-price auction where users must bid on how much they are willing to pay for the computational effort required by their transaction (Roughgarden 2020). Third, Ethereum served as the blueprint for many other blockchain platforms like Avalanche, Cosmos, or Polygon that now use a similar mechanism to allocate transactions and thus enhances the generalizability of our results.

For our empirical strategy, we use daily transaction data from a sample of 1,590 dApps on Ethereum and estimate different demand curves for different groups of dApps. To address the endogeneity issues arising from the simultaneous determination of transaction fees by demand and supply, we introduce Ethereum's difficulty bomb—which we decribe in detail in Section 6—as a novel supply-side instrument that has led to exogenous variation in the supply of transactions.

Our analysis yields several findings. First, by finding a downward-sloping demand curve, we can confirm that the law of demand also applies to transactions on Ethereum. While this finding seems theoretically trivial, the ongoing debate on the prevalence of speculation activity, extreme volatility, and illicit transaction activity questions whether blockchain platforms are subject to standard supply and demand dynamics comparable to other financial markets (Foley et al. 2019, Li et al. 2018). Our findings provide empirical evidence that helps to settle this debate and move on with further economic analysis of blockchain platforms. Second, we find that different groups of dApps significantly vary regarding their sensitivity towards transaction fees and that, in times of congestion, finance applications crowd out transactions to other applications by increasing the market price for transacting on the network. Third, our results suggest that building dApp-specific network effects and bundling transactions more efficiently are the only options a dApp has to influence its sensitivity towards transaction fees.

With our research, we contribute threefold. First, we contribute to the literature on net neutrality by extending its debate to the realm of blockchain technology, which according to its proponents, may become an important part of the Internet's IT infrastructure. Second, we contribute to the platform literature by showing that decentralizing platforms with blockchain technology and a market mechanism to allocate transactions can lead to the undesired consequence of losing innovation on the platform's edges. Our findings highlight that blockchain platforms may fall short in orchestrating an appealing ecosystem of complements and thus have difficulties competing with their centralized counterparts if they do not establish a working social layer or develop some governance tools. Further, our findings help to understand that blockchain platforms are less prone to "winner-take-all" dynamics and that it is more likely that we will see an ecosystem of different blockchain platforms than one dominate platform hosting all types of dApps. Finally, we also add to the burgeoning literature on transaction fees on blockchain platforms by introducing a novel instrument that helps to mitigate endogeneity concerns and help future scholars to investigate the economic dynamics on blockchain platforms that allow third-party providers to offer additional services in the form of dApps.

2 Related literature

Our research draws from three streams of prior research.

The first stream is the literature on net neutrality. As blockchain platforms prioritize transactions based on how much the transaction sender is willing to pay to the validation service providers for the execution of their transaction, we argue that many arguments of the net neutrality debate can be transferred to the blockchain realm. Specifically, we see similarities to a sub-stream of this literature focusing on the implications of net neutrality on content heterogeneity and innovation by small content providers at the edges of the Internet (e.g., Reggiani and Valletti 2016, Guo et al. 2012, Krämer and Wiewiorra 2012). To draw parallels to this debate, we define *blockchain neutrality* as the principle that all transactions on a blockchain are treated equally, and transaction validators cannot charge dApp providers for prioritizing their transactions. While superficially blockchain neutrality seems to be warranted as there is no payment stream between validators and dApp providers, the fact that dApps might differ regarding their transaction fee and congestion sensitivity might jeopardize blockchain neutrality as the market mechanism favors some dApps over others. By introducing the notion of blockchain neutrality, we seek to draw attention to the potential implications of violating this principle as it could lead, as observed in the context of net neutrality, to a loss of innovation on the edges of blockchain and hamper its overall innovation capability.

The second stream of literature is the literature on platform ecosystem governance. For a review see Rietveld and Schilling (2020). This stream investigates how the rules set and control exerted by the platform provider influence the emergence of a healthy ecosystem of complements, the platform's innovation capability, and the overall value created on the platform (e.g., Cennamo and Santaló 2019, Cennamo 2018, Tiwana et al. 2010, Staub et al. 2022, Wareham et al. 2014). It questions the often taken-for-granted tenet of network effects that suggests that a greater breadth and depth of complements is typically beneficial by emphasizing that unfiltered growth of the complementors side can pose the risk of platform congestion and decrease consumers' value (e.g., Casadesus-Masanell and Hałaburda 2014). Therefore, this literature makes a strong case that platform providers must carefully orchestrate a healthy ecosystem of complementors to maintain the platforms innovation capability and ensure its long-run success. Our research adds to this stream as blockchain platforms challenge the core assumption of a central orchestrator governing the platform and thus allow us to understand the implications of substituting a strong "visible" hand with an "invisible" hand of a decentralized market that prioritizes transactions based on the users' willingness to pay on the ecosystem of complements.

The third stream is the nascent literature that studies transaction fee mechanisms on blockchain platforms. Within this literature, scholars have already started to characterize blockchains as marketplaces where miners offer their services to transaction senders and study the dynamics of these marketplaces (e.g., Basu et al. 2019, Easley et al. 2019). There are only few empirical studies estimating impact of transaction fees on the usage of blockchain platforms, and most focus on the Bitcoin blockchain (e.g., Easley et al. 2019, Ilk et al. 2021). For Ethereum, this evidence is still lacking. Although few accounts investigate the relationship between network congestion and gas prices (Donmez and Karaivanov 2021) or gas prices and throughput (Azevedo Sousa et al. 2021, Spain et al.), or how high gas fees antagonize Ethereum's goal of inclusion and democratization by excluding users who cannot afford the increasing gas fees (Cong et al. 2022), there is a paucity of research that analyzes supply and demand dynamics on Ethereum and in particular how these impact dApp usage across different dApp groups and how this influences the variety of dApps offered on Ethereum in the long run.

3 Ethereum's market for transactions

To validate, enforce, and record transactions users send to dApps, Ethereum uses a decentralized transaction mechanism. Prior scholars have already characterized Bitcoin mining, which uses a similar mechanism, as a two-sided market (e.g., Basu et al. 2019) and a market for data space more specifically Ilk et al. (2021). We also characterize Ethereum's transaction validation and execution process as a market but highlight some important differences due to Ethereum's capability to run smart contracts and offer dApps.

In contrast to Bitcoin and to facilitate dApps and arbitrary transactions, Ethereum does not charge a fee per transaction but a fee for the computational effort a transaction requires. A transaction's computational effort is measured in *units of gas* according to a list that indicates a fixed gas requirement for every atomic computation. To maintain decentralization by ensuring that miners with less powerful machines can also participate in mining transactions and to prevent that the network gets trapped in an infinite loop of computation, the maximum gas of a block is limited (*block gas limit*). In addition to limiting the total gas a block can use, the Ethereum protocol also tries to keep the average time it takes to find a new block (*average block time*) within a 12 to 14 seconds interval. These two limitations imply that the total amount of available gas has an upper limit. To allocate the limited gas supply, Ethereum uses a market mechanism that we conceptualize as a market for transactions or, more specifically, a market for the validation and enforcement service of transactions.

The commodity sold on this market is the gas required to validate a transaction.[2] Accordingly, users (transaction initiators) are the buyers, whereas miners are the sellers of this commodity. On the supply side, the supply of gas on each day is fixed due to the block gas limit and the limited average block time. Although miners can decide to what extent they use this limit, they cannot change it individually. Changing this limit requires successful voting by all miners and a protocol update. Also, suppose more miners join the network and participate in the race to solve the mining puzzle. In that case, the network will increase the mining difficulty (i.e., the number of hashes it takes on average to find a new block) to keep the average block time within the target window of 12 to 14 seconds and keep the supply of gas fixed.[3]

On the demand side, users cast transactions to other externally owned accounts or smart contracts. To initiate a transaction, users must indicate a *transaction gas limit* (i.e., the maximum amount of gas a miner is allowed to use to compute the transaction) and a *gas price*(e.g., the price the user is willing to pay for each unit of gas). If the gas limit is reached before the transaction is fully computed, the transaction will be aborted and not included in the block. Users only pay for the used gas if the computation is finished before reaching the limit. Also, only the actually used gas is considered for the block gas limit. Accordingly, the fees a user has to pay is the product of gas used and the gas price the users is willing to pay for every unit of gas.

As the supply of gas is limited, transaction senders compete with other senders by choosing a gas price that is high enough that miners pick their transactions from the pool of pending transactions. Typically, miners engage in profit maximization (Basu et al. 2019). Hence, they sort transactions by the indicated gas price and requirement and fill up the block until its gas limit is reached. Especially in times of congestion, offering too low a gas price means that a transaction will not be picked up by any miner and ultimately be deleted from the pool of pending transactions. This gas price mechanism has led to considerable fluctuations in the amount of gas used, and the price users have paid for a unit of gas. For illustration, Figure 1 depicts the daily gas usage on the left and the daily average gas price on the right.

——— insert Figure 1 about here ———

4 Conceptual framework

The driving force behind our framework is that the usage of a dApp—hence its success—on Ethereum depends on the usage of the platform, which in turn again depends on the usage of other dApps.[4] However, due to two countervailing forces, it is unclear if increasing the user base

[2] It is important to note that the transaction initiator only has to pay the gas fees for the computation of the transaction but not for the computational effort the miner has to invest solving the PoW puzzle that is required to find a new block.

[3] See Appendix A for the formula used to compute the mining difficulty.

[4] It is important to note that although our empirical analysis is—due to the selection of our instrumental variable—limited to a period when Ethereum relied on PoW as a consensus mechanism, our theoretical arguments also apply

and dApp base benefits all dApp providers. On the one hand, entering dApps attract new users to the platform, which fosters the platform's adoption, and enlarges the number of possible users of the focal dApp. On the other hand, the limited supply of transactions in combination with the first-price auction that allocates this limited supply aggravates the direct competition among dApps by introducing a negative externality: new dApps and users increase demand and intensify the competition for the limited supply of gas. The increasing demand and competition lead to increasing congestion costs and higher gas prices. Because transaction initiators need to pay transaction fees to interact with every dApp, increasing gas prices lessen the overall utility and, thus, the usage of dApps. Accordingly, the relative magnitude of these countervailing effects will determine the effect of Ethereum's market for transactions on the success of the platform complements.

Although the net impact of increasing gas prices as a response to more platform usage is theoretically undetermined—due to the countervailing forces described above—we can analyze which characteristics of a dApp expose it more to changes in the gas price. Understanding this is not only useful for the complementors' decision to enter such a market but also for the platform provider, as it might have important implications for the heterogeneity of complements offered on the platforms. We hypothesize that depending on four characteristics, dApps are more or less sensitive to changes in the gas price and, therefore, better or worse equipped to compete in a market for transactions.

First, we expect that the type of service a dApp offers influences its sensitivity towards changes in the gas price. This intuition becomes clear when considering that some dApps provide social and entertainment services while others provide financial or security-related services. Although finance dApps do not necessarily provide more utility to users than leisure-related dApps, it is easier to compute the expected utility of a finance transaction. Therefore, it should be easier for users to evaluate if they still want to send a transaction whereas for other dApps the uncertainty and cognitive effort to gauge the expected utility will deter them from sending a transaction. Further, finance-related transactions are often more time-sensitive, and as Donmez and Karaivanov (2021) show, users on Ethereum are more willing to pay higher gas fees for timely transactions.

Second, even within the same type of service, dApps can substantially differ regarding the requirements of the transaction. For example, dApps can differ in the complexity of the underlying transaction and hence the gas required for the computation of it. On the one hand, the gas requirement correlates with the complexity of the underlying functionality. On the other hand, it is also driven by the efficiency of the code itself. Particularly within the same type of service, where the functionality and complexity of transactions with dApps is similar, the code's efficiency should be the main determining factor for the gas requirement. Especially in times of high gas prices, we expect users to be more sensitive to such differences and use dApps that require less gas for the same functionality. Another factor determining a dApp's gas price sensitivity should be the value transferred in a transaction with a dApp. For example, considering that some NFTs are sold for well above $100,000, it becomes evident that even gas fees of a few dollars are negligible. Therefore, we expect that depending on the average transaction value that a dApp usually carries, the dApp should be more or less sensitive to changes in the gas price.

Third, dApps also differ in the overall quality of their services or their usability and hence in the value they create for their users. Accordingly, some dApps are more appealing to users than others. These dApps should not only perform better at baseline but are also more likely to benefit from the entry of other dApps. Consider, for example, that numerous new dApps enter Ethereum. This should attract additional users since users appreciate product variety. But once the users join, they will disproportionately choose the dApp offering more utility. This effect can be exacerbated if

to the period when Ethereum updated to PoS as PoS only removed the computationally expensive mining puzzle but still requires users to pay gas fees the dApp itself benefits from network effects, which should be the case for dApps such as currency exchanges, marketplaces, or social messengers. For such dApps, the increasing utility due to the larger network could counterbalance the additional fees resulting from the intensified competition for gas among dApp users.

Fourth, the current performance of a dApp should influence users' willingness to pay for a transaction with the dApp. Again, especially for dApps that rely on network effects, the number of other users of a dApp should increase the utility of transacting with this dApp.

To understand how Ethereum's market for transactions influences dApp usage, we next empirically investigate the drivers of dApps' transaction fee sensitivity as hypothesized above.

5 Data and sample construction

5.1 Research context and data

We combine daily block and transaction-level data publicly stored on the Ethereum blockchain with three different data sources that provide supplementary off-chain data, such as the category of the dApp or the exchange rate for one Ether or other tokens.

5.2 Data collection procedure and sample

We obtained our data from four different sources. First, We use the Ethereum ETL to download all block-level and transaction-level data. Second, we use two websites that provide a curated list of dApps (stateofthedapps.com and defillama.com) to identify dApps that are running on Ethereum, the addresses of their associated smart contracts, and the category of the application. This step allows us to map the pseudonymous smart contract addresses on the blockchain to their respective dApp and is necessary because a dApp can consist of multiple smart contracts. Overall, we identified 1,590 dApps with 4,680 associated smart contracts active in our study period. Thrid, we use the Etherscan API to collect further daily network-level data, such as the network utilization, which measures the extent to which the block gas limit has been used. Finally, we retrieve the daily prices for one Ether and other tokens associated with the dApps in our sample from the CoinGecko API.[5] To ensure that all variables are on the same level and to mitigate high-frequency variation in the data, we first merge the block-level and transaction-level data by using the block hash reported for every transaction and then aggregate the resulting data at the daily level. Our consolidated dataset covers 1,279 days. Table 1 provides an overview of the number of dApps per group of categories.[6]

——— insert Table 1 about here ———

5.3 Data sets, variables, and measurement

Besides the daily aggregation, we further aggregate transactions on the level of a dApp.

Our main variable of interest is the quantity of gas used ($gasUsed_t$). It refers to the daily amount of computational validation effort demanded by all transactions with a dApp. It is measured in Giga gas units. This variable operationalizes the goods supplied by the miners and demanded by the transaction senders.

[5] https://www.coingecko.com/en/api/

[6] To mitigate multicollinearity issues arising from similar transaction patterns across similar categories, we aggregated the 17 categories into 5 groups that resemble in the type of service they offer. We obtained the groups by applying a cluster analysis to variables like daily transaction count and transaction value.

The gas price is the price (in GWei) transaction initiators must pay for each gas unit. As the gas price an initiator pays varies according to the outcome of a first-price auction, we define the gas price in times of the $marketGasPrice_t$ a sender would have had to pay for their transaction to just make it into one of the blocks on a given day. We proxy this market price with the daily average of the bottom fifth percentile gas price recorded on each block on that day in GWei. We use this proxy because there are blocks in whose validation miners circumvent the first-price auction mechanism by adding their own transactions with a gas price close to zero or even zero. Accordingly, using the marginal gas price (i.e., the lowest gas price on a day at which a transaction is just included in a block) would not correctly reflect the market mechanism. We also run several robustness checks with alternative gas price variables (e.g., different percentiles of the gas price in USD).

We define the variable difficulty bomb ($difficultyBomb_t$) as the average additional difficulty induced by Ethereum's difficulty bomb on a given day. Next to the automated adjustment of the mining difficulty, the difficulty bomb is the second mechanism encoded in Ethereum's protocol that influences the total network difficulty (i.e., the average number of hashes it takes to find a block). The goal of the difficulty bomb is to force miners to switch from PoW to PoS once the PoS update is available. To this end, the difficulty bomb exponentially increases the mining difficulty until it is almost impossible to find new blocks by solving the PoW puzzle. As Ethereum planned right from its start to switch to PoS at some point, the difficulty bomb was always part of the protocol. However, because the update to PoS was delayed several times, the difficulty bomb increased the difficulty too fast, resulting in a disproportionate increase that was not reflected by the network hash rate and the discovery of significantly fewer blocks per day. Because the resulting shortage in gas was not intentional (the plan was that PoS-blocks would grow at the same rate as the PoS-blocks would decline), the Ethereum community issued a protocol update that turned back the additional difficulty. Over our study period, this pattern occurred three times and is reflected in three protocol updates (EIP649, EIP1234, and EIP2384). As the difficulty induced by the difficulty bomb is not reported in any database, we leverage the fact that Ethereum's protocol continuously tried to keep the block time within the target window of 12-14 seconds and constructed the variable as follows. The difficulty induced by the difficulty bomb on a day d is the difference between the total observed difficulty and the theoretical difficulty required to reach the target block time, given the current hash rate in the network. Accordingly, the difficulty bomb on a day d is:

$$difficultybomb_d = (networkhashrate_d \times targetblocktime) - difficulty_{observed,d}$$

The unit of this variable is the number of Tera hashes it requires on average to find a new block. Due to the exponential growth and the fluctuation of the network difficulty within the target window, especially at the beginning of the activity of the difficulty bomb, the added difficulty is not always distinguishable from zero. To account for this fact, although the difficulty bomb is always active, we only assign a positive value to the difficulty bomb if the block time is noticeably above the target window (> 14s). According to this conservative approach, we only observe on 16% (182 days) of all days in our sample a difficulty bomb above zero. To establish robustness, we also use different cutoffs and approaches to measure the activity of the difficulty bomb. We will discuss our instrument's relevance and exogeneity later in the empirical strategy and results section. Figure 2 overlays the network hash rate with the observed total mining difficulty. Gaps between both curves indicate excessive difficulty added by the difficulty bomb.

——— insert Figure 2 about here ———

To account for the degree to which miners fill the blocks on a given day, we measure the network utilization ($networkUtilization$) as the fraction of total available gas (sum of the gas limit of all blocks) on a day that is used by all transactions on that day in percent. It captures the platform's usage level and has been used by prior researchers as a measurement for congestion (Donmez and Karaivanov 2021).

In addition to these variables, we compute several measures that allow us to study the transaction requirements of each dApp or their usage patterns. To reflect the complexity of an interaction with a dApp, we measure the average gas requirement (*avgGasRequirement*) of a transaction with a dApp. To reflect the requirements of a transaction with a dApp, we measure the average value of Ether (*avgValue*) or (*avgTokens*) a dApp receives as a proxy for how much value transactions with the dApp usually carry. In addition, we measure the following performance indicators for every dApp: average daily transaction activity (*avgDailyTxn*), average number of unique externally owned accounts (*avgDailyEOA*) that transactions with a dApp (i.e., our proxy for users),[7] the average gas price users pay for a transaction with a dApp (*avgGasPricePaid*), the average number of transactions per externally owned account on a given day (*avgTxnPerEOA*), and the surplus gas price the transaction senders paid beyond the market gas price on a given day *surplusGasPrice*.

We also control for the following network-level variables: Ether price (*EtherPrice*) measures the price of one Ether in USD on the day the transaction was executed; Ether volatility *EtherVolatility* measures the daily change in the exchange rate of one Ether; gas limit *gas Limit* measures the sum of all block gas limits on a day and accounts for the fact that over our sample period, the total units of gas that can be used in a block has been increased several times; and finally day of the week (*weekday*) and year (*year2017-2020*) dummy variables, and a trend (*trend*). Appendix 8 provides descriptive statistics and correlation scores for all variables in our data set.

6 Estimation strategy

In this section, we discuss our baseline specification and the instrumental variable (IV) we use to address the endogeneity of the gas price.

6.1 Baseline specification

The specification for our dApp-level analysis is:

$$\begin{aligned}\log(gasUsed_{td}) = {} & \alpha_0 + \alpha_1 \log(marketGasPrice_t) + \alpha_2 networkUtilization_t + \\ & \alpha_3 networkUtilization_t^2 + \alpha_4 \log(EtherPrice_t) + \alpha_5 \log(EtherVolatility_t) + \\ & \alpha_6 \log(gasLimit_t) + \mu_{day\ of\ week} + \mu_{year} + \mu_d + trend + u_t\end{aligned}$$

where gas used is the equilibrium gas demand for each dApp d in the period t (day). We chose a log-log specification for gas used and market gas price to be able to interpret α_1 as the price elasticity of the demand. Due to the skewed distributions of Ether price, Ether volatility, and the gas limit, we use log-transformed versions of these variables in our specification. The network utilization allows us to control for the degree to which miners use the available block gas limit on a given day and has been used by prior scholars as a measure of network congestion (Donmez and Karaivanov 2021). We also add a quadratic term to account for the nonlinear relationship between gas price and network utilization.[8] In addition to these variables, we also control for the intrinsic

[7] Technically it is possible to differentiate between smart contract addresses and wallet addresses, but not if a wallet address is controlled by a bot. To account for this fact, we refrain from calling wallet addresses "users" and call them instead "externally owned accounts" to emphasize that they do not necessarily correspond to human users. Therefore, this variable is only a proxy.

[8] We also compute the same model with a threshold specification where we added only the linear term and dummy

growth of the dApp by adding age_{dt} as the number of days since the dApp entered the platform and specify μ_{d} as dApp fixed effects, $\mu_{\text{dayofweek}}$ as a day of week fixed effects, μ_{year} as a year fixed effects, and u_{t} as the error term.

6.2 Validity of the instrument

In this model, $\log(gasUsed_t)$ and $\log(marketGasPrice_t)$ are the endogenous variables, as both are jointly determined in equilibrium. To address this simultaneity issue, we use the *difficultyBomb* as an instrumental variable in a two-stage least squares approach (2SLS). In the first stage, we use the difficulty and all other control variables listed above to predict the $\log(marketGasPrice_t)$. In the second stage, we estimate the specification above by replacing the $\log(marketGasPrice_t)$ with its predicted value. The economic intuition underlying our approach is that we leverage the difficulty bomb as an exogenous supply shifter. Due to the consistent adjustment of the network difficulty and the resulting constant block time, the gas supply curve resembles a fixed vertical line. When the difficulty bomb is active, the added difficulty increases the block time and thus decreases the number of blocks on a given day. As the maximum gas a block can contain is limited, fewer blocks lead to a decrease in the gas supply and hence a horizontal shift of the supply curve to the left. We exploit this supply shift to identify the demand curve.

We argue that the difficulty bomb is exogenous and influences the gas demand only through the increased gas price for three reasons. First, it is programmed into the Ethereum protocol, and changing it requires a successful protocol update (called Ethereum Improvement Proposal or EIP) which is only possible after a majority vote and hence unlikely to be a response to a short-term market situation. Therefore, the difficulty bomb and its resets bomb can be seen as exogenous policy interventions. Second, as the difficulty level is not reported in wallet applications or by an API and has to be manually calculated (see Section 5.3), it is plausible to assume that ordinary Ethereum users were not aware of the existence of the difficulty bomb. Third, even if users were aware of the existence of the difficulty bomb, it is difficult for them to comprehend its exponential growth and differentiate its impact—at least in the initial phase—from normal fluctuations due to the exit and entry of miners. Further, it would also be difficult for users to predict the mining power and cost structure of every single miner and to evaluate when they cannot keep up with the difficulty level.

7 Results

7.1 Baseline dApp-level results

Following our baseline specification, Table 2 reports the results of our 2SLS demand curve estimation. Column 1 presents the first stage results, where we predict the gas price (log(Market gas price)) with our IV (difficulty bomb). Column 2 presents the second stage results, where we use the predicted gas price to estimate the price elasticity of the gas demand (log(Gas used)).

—– insert Table 2 about here —–

Consistent with our prediction, Columns 2 and 3 suggest a downwards-sloping demand curve for gas on Ethereum. The first stage reported in Column 1 shows that an increase in additional difficulty due to the difficulty bomb is significantly associated with increased gas prices. This is in line with

variable that takes on the value one if the utilization level exceeds 90%. They were qualitatively the same regarding the magnitude and significance of the coefficients we obtained.

our explanation that the added difficulty reduces the supplied gas—by reducing the number of blocks explored per day—and thus intensifies price competition among transaction senders. The coefficient of the difficulty bomb is highly significant even though we control for network utilization (i.e., the degree to which miners use the available block space), network utilization squared, the exchange rate of Ether to USD, the daily fluctuation of this exchange rate, the block gas limit, as well as day of the week and year dummies and a common trend.

Regarding the validity of our instrument, by comparing the first-stage with and without the instrument, we obtain an incremental F (121.39) that is well beyond the suggested cut-off of 10 (Stock and Yogo 2005) and thus suggests that our instrument strongly correlates with the endogenous gas price. Further, we compute the Stock-Yogo test for weak instruments, which shows that the Cragg-Donald-Wald F Statistic (2542.47) exceeds the predetermined critical value (16.38).

To interpret the magnitude of the effect of the gas price ($\log(marketGasPrice_t)$) on the demand of gas log(Gas used), the coefficient of -0.64 implies that a 1% increase in the market price of a unit of gas decreases the amount of gas demanded by 0.64%. Considering that the average transaction on Ethereum consumes 184,000 units of gas (which corresponds to a normal smart contract interaction), this equals a decrease of roughly 1,703 smart contract transactions per day or 14,923 Ether transfers which require 21,000 units of gas. Considering that the median dApp only receives eight transactions per day, the order of magnitude of this effect can have significant economic implications.

In sum, this analysis provides first empirical evidence that the well-established "law of demand" (Gale 1955) also applies to the validation service of transactions on Ethereum. It also provides evidence that Ethereum's gas price mechanism introduces a form of price competition among transaction senders that counteract the main prediction of the two-sided market literature (Rochet and Tirole 2006), i.e., that, due to the same-side network effect, an increase in the demand side draws even more consumers into the market and leads to subsequent increases in demand. On Ethereum, an increase in transaction senders increases not only the utility of transacting on Ethereum but also price competition. However, as the demand for gas is negatively associated with its price, the market mechanism underlying Ethereum's transaction validation process dampens the effectiveness of same-side network effects.

7.2 Differing demand curves per group

Column 3 in Table 2 reports the different demand curves for each group of dApps. We obtain these demand curves by interacting the instrumented market gas price with the group of a dApp.

With a positive and significant coefficient (0.27) for our reference group (finance dApps), our results suggest that the demand curve for these dApps in upward-sloping. An explanation for this upward-sloping demand curve could be that the entry of additional finance-related dApps has caused an influx of high willingness-to-pay customers and that the network effects these finance-related dApps realize compensated for the higher transaction fees these transaction senders had to pay. This explanation is in line with prior research that describes networked goods (e.g., financial services) by irregularities such as an upward-sloping demand curve for low quantity levels (Economides and Himmelberg 1995). Particularly, if a service relies on strong network effects, no one will pay for the product if no one else uses it. Although the entry of high willingness-to-pay users is typically beneficial for a platform, the fact that we observe downward-sloping demand curves in the form of negative moderations of all other groups poses a danger that, particularly in times of high transaction fees, dApps from other groups are not used anymore and finally have to leave the platform. This reduction of complement heterogeneity can ultimately harm the long-term attractiveness of Ethereum, especially as a general-purpose platform.

7.3 Heterogeneous effect of Ethereum gas price mechanism

Beyond the category of a dApp, we use our rich data to further explore the characteristics of dApps that impact their sensitivity toward the gas price. We use these characteristics as additional moderating variables in our baseline analysis and add them as two-way interactions with the market gas price and three-way interactions with the market gas price and group of the dApp.

The first set of characteristics pertains to the formal requirements of a transaction with a dApp. These characteristics are the amount of gas a transaction with a dApp requires and the value of Ether and tokens a transaction with a dApp usually carries. We find that dApps with a higher average gas requirement suffer more from changes in the market gas price while dApp with higher average transaction value are less sensitive to changes in the market gas price. With some minor variation, this effect applies to all groups of dApps.

Next, we also compute average performance indicators for each dApp. For the average daily transactions and average daily EOA, we find a positive and significant two-way interaction with the gas price. This suggests that the demand for gas for transactions with dApps with a high average of daily transactions and users is less impacted by changes in the gas price. However, by adding the group dummies to these two-way interactions, we find that this interaction significantly differs between dApps in group one and all other groups. Whereas dApps in group 1 still seem to benefit from more transactions and EOAs—as indicated by the positive and significant two-way interactions between the gas price and the average number of transactions and the average number of daily EOA (Column 5, 0.39)—the three-way interactions with all other groups are highly significant and negative. This indicates that for dApps in these groups, the effect of receiving, on average, more transactions or having more unique EOAs transacting with them is less prevalent or even makes them more sensitive to changes in the gas price. Again, network effects could be a plausible explanation for this observation.

To further investigate network effects, we analyze the impact of dynamic usage indicators that vary for each dApp over time. Regarding the number of transactions per EOA, we find a positive interaction between the number of transactions per EOA and the gas price (*log(Market gas price)*). According to the three-way interactions, except for group 5, this moderation does not significantly differ between the different groups of dApps. Because for dApps in group five, the interaction is even stronger than for all other dApps, attracting heavy users might be a valid strategy for these dApps to survive the competition in a market for transactions. Considering that group 5 comprises dApps such as storage or energy services and given the strong lock-in effects these services typically exhibit, also these findings seem plausible.

Overall, the results of our heterogeneity analyses suggest that inherent features of dApp (e.g., its gas requirement or if it benefits from network effects) rather than its quality determines users sensitivity to changes in the market gas price.

7.4 Additional robustness checks

To assess the robustness of our analysis, we tested them against several alternative measures and samples. For example, we used the transaction count instead of gas used, applied different levels of winsorization to restrict the impact of possible outliers, used different percentile and levels of winsorization for the market gas price together with the average gas price, and also a different measurement of the difficulty bomb where we subtracted the observed number of blocks from the target number of blocks given the targeted block time. Further, we also conducted our analysis only for the periods where the difficulty bomb was active. Overall, we find the results to be consistent with the results of our baseline specification. Moreover, we further report two additional analyses that corroborate our results in Appendix 8 and 8.

8 Discussion and conclusions

Decentralized blockchain platforms like Ethereum have been hailed for challenging the dominance of centralized digital platforms that currently prevail in the digital economy (Murray et al. 2019, Vergne 2020). However, little is known about how the decentralized transaction validation mechanism, which distinguishes blockchain platforms from their centralized counterparts, impacts the platform by shaping its usage and complements. To investigate this question, we study Ethereum's transaction validation mechanism as a market for transactions and use a panel data set of 1,590 dApps together with a novel supply-side instrument to estimate different price elasticities of the demand for transactions with dApps. We find strong evidence that Ethereum's gas price mechanism leads to negative network effects (i.e., a growth of the transaction demand makes transacting more expensive) that counteract the positive network effects usually present on multi-sided platforms. Further, we find that the relative magnitude of these effects depends on characteristics of a dApp that are mostly predetermined. Particularly, the type and complexity of the service a dApp offers are decisive factors. For instance, across the board, the demand for transactions with finance or exchange dApps seems to be less impacted by changes in the gas price than dApps that offer games, gambling, social, or media-related services. This is especially problematic as the transaction validation mechanism adds a new externality to the existing competition on such platforms: all dApps—no matter what service they offer—must compete for the limited gas supply. Hence, it favors some dApps over others and finally forces disadvantaged dApps to leave the platform leading to a decrease in the heterogeneity of dApps offered on Ethereum and a reduced value for platform users who joined because of the variety of complements offered on the platform.

With these findings, we demonstrate that relying on a market mechanism to allocate the limited transaction supply, which is necessary to ensure decentralization, contradicts the principle of blockchain neutrality (i.e., no preferential treatment of complements). We have a reason to believe that these insights extend beyond Ethereum within the period that we study. For example, even though our empirical identification strategy crucially depends on mining present in PoW, even blockchains relying on PoS face the same problem. This is because the main factors driving the discrimination results are limited capacity of transactions and market mechanism to allocate those transactions. They do not depend on mining or staking procedures.

As decentralization strictly restricts the governance tools of blockchain platform to safeguard complementors from such discrimination, there are no simple solutions to this predicament. One potential resolution could involve abandoning the transaction supply limit and accepting a lower level of decentralization, as large blocks would require more powerful machines to validate and store. Another solution could be to develop different ways to allocate the limited transaction supply. Alternatively, we may have to acknowledge that blockchain platforms are not neutral, as they selectively favor some dApps over others, ultimately limiting the range of applications that can be offered on decentralized infrastructure. However, adopting this approach would jeopardize the vision of Web3.0 — a fully inclusive, transparent, and democratic version of the Internet designed to cater to the needs of all types of applications.

References

Azevedo Sousa J, Oliveira V, Valadares J, Dias Gonçalves G, Moraes Villela S, Soares Bernardino H, Borges Vieira A (2021) An analysis of the fees and pending time correlation in ethereum. *International Journal* *of Network Management* 31(3), ISSN 1055-7148, URL http://dx.doi.org/10.1002/nem.2113.

Basu S, Easley D, O'Hara M, Sirer E (2019) Towards a functional fee market for cryptocurrencies URL http://arxiv.org/pdf/1901.06830v1.

Buterin V (2014) Ethereum whitepaper. URL https://ethereum.org/en/whitepaper/.

Casadesus-Masanell R, Hałaburda H (2014) When does a platform create value by limiting choice? *Journal of Economics & Management Strategy* 23(2):259–293, ISSN 10586407, URL http://dx.doi.org/10.1111/jems.12052.

Catalini C, Tucker C (2018) Antitrust and costless verification: An optimistic and a pessimistic view of the implications of blockchain technology. *SSRN Electronic Journal* 1–14, URL http://dx.doi.org/10.2139/ssrn.3199453.

Cennamo C (2018) Building the value of next-generation platforms: The paradox of diminishing returns. *Journal of Management* 44(8):3038–3069, ISSN 0149-2063, URL http://dx.doi.org/10.1177/0149206316658350.

Cennamo C, Santaló J (2019) Generativity tension and value creation in platform ecosystems. *Organization Science* 30(3):617–641, ISSN 10477039, URL http://dx.doi.org/10.1287/orsc.2018.1270.

Choi JP, Jeon DS, Kim BC (2018) Net neutrality, network capacity, and innovation at the edges. *The Journal of Industrial Economics* 66(1):172–204, ISSN 00221821, URL http://dx.doi.org/10.1111/joie.12161.

Cong L, Tang K, Wang Y, Zhao X (2022) Inclusion and democratization through web3 and defi? initial evidence from the ethereum ecosystem. *SSRN Electronic Journal* URL http://dx.doi.org/10.2139/ssrn.4229308.

Donmez A, Karaivanov A (2021) Transaction fee economics in the ethereum blockchain. *Economic Inquiry* 60:265–292, URL https://onlinelibrary.wiley.com/doi/pdfdirect/10.1111/ecin.13025?download=true.

Easley D, O'Hara M, Basu S (2019) From mining to markets: The evolution of bitcoin transaction fees. *Journal of Financial Economics* 134(1):91–109, ISSN 0304405X, URL http://dx.doi.org/10.1016/j.jfineco.2019.03.004.

Economides N, Himmelberg C (1995) critical mass and network size with application to the us fax market. *Mimeo, Stern School of Business at New York University)* .

Foley S, Karlsen J, Putniņš T (2019) Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies? *The Review of Financial Studies* 32(5):1798–1853, ISSN 1465-7368, URL http://dx.doi.org/10.1093/rfs/hhz015.

Gale D (1955) The law of supply and demand, , vol. 3. *Mathematica Scandinavica* 3(1):155–169.

Guo H, Cheng HK, Bandyopadhyay S (2012) Net neutrality, broadband market coverage, and innovation at the edge*. *Decision Sciences* 43(1):141–172, ISSN 1540-5915, URL http://dx.doi.org/10.1111/j.1540-5915.2011.00338.x.

Ilk N, Shang G, Fan S, Zhao J (2021) Stability of transaction fees in bitcoin: A supply and demand perspective. *MIS Quarterly* 45(2):563–692, ISSN 02767783, URL http://dx.doi.org/10.25300/MISQ/2021/15718.

Jin-Hyuk Kim, Jeffrey Prince, Calvin Qiu (2014) Indirect network effects and the quality dimension: A look at the gaming industry. *International Journal of Industrial Organization* 37:99–108, ISSN 01677187, URL http://dx.doi.org/10.1016/j.ijindorg.2014.08.005.

Krämer J, Wiewiorra L (2012) Network neutrality and congestion sensitive content providers: Implications for content variety, broadband investment, and regulation. *Information Systems Research* 23(4):1303–1321, URL http://dx.doi.org/10.1287/isre.1120.0420.

Li T, Shin D, Wang B (2018) Cryptocurrency pump-and-dump schemes. *SSRN Electronic Journal* ISSN 1556-5068, URL http://dx.doi.org/10.2139/ssrn.3267041.

Murray A, Kuban S, Josefy M, Anderson J (2019) Contracting in the smart era: The implications of blockchain and decentralized autonomous organizations for contracting and corporate governance. *Academy of Management Perspectives* ISSN 1558-9080, URL http://dx.doi.org/10.5465/amp.2018.0066.

Nakamoto S (2008) Bitcoin: A peer-to-peer electronic cash system. URL https://bitcoin.org/bitcoin.pdf.

Pereira J, Tavalaei M, Ozalp H (2019) Blockchain-based platforms: Decentralized infrastructures and its boundary conditions. *Technological Forecasting and Social Change* 146:94–102, ISSN 00401625, URL http://dx.doi.org/10.1016/j.techfore.2019.04.030.

Reggiani C, Valletti T (2016) Net neutrality and innovation at the core and at the edge. *International Journal of Industrial Organization* 45:16–27, ISSN 01677187, URL http://dx.doi.org/10.1016/j.ijindorg.2015.12.005.

Rietveld J, Ploog JN, Nieborg DB (2020) The coevolution of platform dominance and governance strategies: Effects on complementor performance outcomes. *Academy of Management Discoveries* (6):1–45, URL http://dx.doi.org/10.5465/amd.2019.0064.

Rietveld J, Schilling M (2020) Platform competition: A systematic and interdisciplinary review of the literature. *Journal of Management* (in press).

Rietveld J, Schilling MA, Bellavitis C (2019) Platform strategy: Managing ecosystem value through selective promotion of complements. *Organization Science* 30(6):1232–1251, ISSN 1047-7039, URL http://dx.doi.org/10.1287/orsc.2019.1290.

Rochet JC, Tirole J (2006) Two-sided markets: a progress report. *The RAND journal of economics* 37(3):645–667.

Roughgarden T (2020) Transaction fee mechanism design for the ethereum blockchain: An economic analysis of eip-1559 URL http://arxiv.org/pdf/2012.00854v1.

Spain M, Foley S, Gramoli V (????) The impact of ethereum throughput and fees on transaction latency during icos. URL http://dx.doi.org/10.4230/OASICS.TOKENOMICS.2019.9.

Staub N, Haki K, Aier S, Winter R (2022) Governance mechanisms in digital platform ecosystems: Addressing the generativity-control tension. *Communications of the Association for Information Systems* 51:906–939, URL http://dx.doi.org/10.17705/1CAIS.05137.

Stock J, Yogo M (2005) Testing for weak instruments in linear iv regression. Stock JH, Andrews DWK, eds., *Identification and inference for econometric models*, 80–108 (New York: Cambridge university press), ISBN 9780521844413, URL http://dx.doi.org/10.1017/CBO9780511614491.006.

Tiwana A, Konsynski B, Bush AA (2010) Research commentary —platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research* 21(4):675–687, URL http://dx.doi.org/10.1287/isre.1100.0323.

Vergne JP (2020) Decentralized vs. distributed organization: Blockchain, machine learning and the future of the digital platform. *Organization Theory* 1(4):1–26, ISSN 2631-7877, URL http://dx.doi.org/10.1177/2631787720977052.

Wareham J, Fox PB, Cano Giner JL (2014) Technology ecosystem governance. *Organization Science* 25(4):1195–1215, ISSN 1047-7039, URL http://dx.doi.org/10.1287/orsc.2014.0895.

Wood G (2014) apps: What web 3.0 looks like. URL http://gavwood.com/dappsweb3.html.

Wu K, Ma Y, Huang G, Liu X (2021) A first look at blockchain–based decentralized applications. *Software: Practice and Experience* 51(10):2033–2050, ISSN 0038-0644, URL http://dx.doi.org/10.1002/spe.2751.

Tables and Figures

Figures

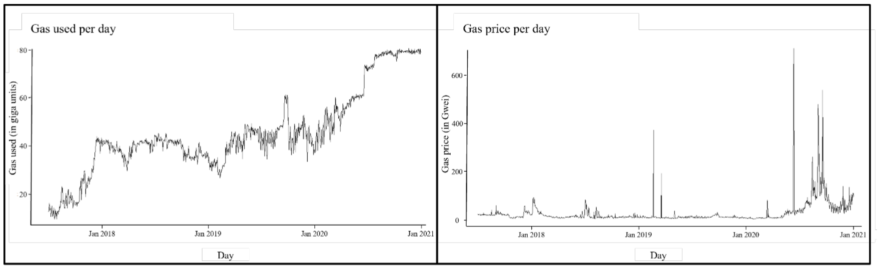

Figure 1: Daily gas used and gas price

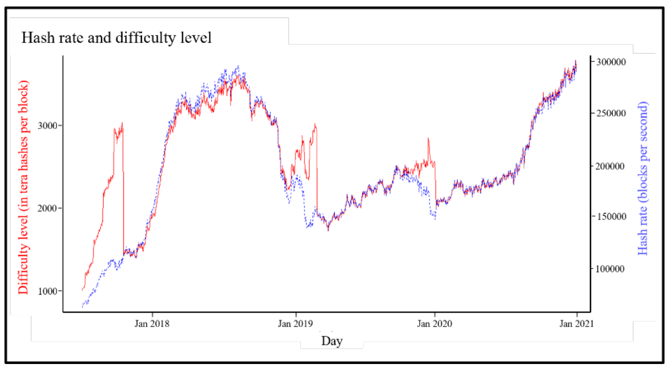

Figure 2: Hash rate and the impact of the difficulty bomb

Tables

Table 1: Groups of dApps

	dApp categories	examples	dApps
Group 1	finance, exchanges, wallets, insurance, security	Sushi swap, OmiseGo, Status, Nexus Mutual, Chainlink	507
Group 2	identity, property	ENS Manager, Decentraland	45
Group 3	games, marketplaces	Axie Infinity, Cryptokitties	464
Group 4	gambling, social, health	FunFair, Minds, BEAT	397
Group 5	energy, governance, media, storage	Dovui, Aaragon, CryptoTunes, XCloud	177

Table 2: Demand curve estimation – baseline model (dApp level)

	(1) log($marketGasPrice$)	(2) log($gasUsed$)	(3) log($gasUsed$)
difficultyBomb	0.20*** (0.0000)		
log($marketGasPrice$)		-0.64*** (0.21)	0.27*** (0.05)
log($EtherPrice$)	-0.0004 (0.01)	0.15*** (0.04)	0.18*** (0.04)
log($EtherVolatility$)	-0.01*** (0.0004)	0.01** (0.004)	0.02*** (0.003)
networkUtilization	-2.36*** (0.06)	-1.20** (0.47)	0.30*** (0.11)
networkUtilization2	16.30*** (0.37)	8.59*** (3.29)	-1.89*** (0.68)
log($gasLimit$)	2.40*** (0.03)	1.89*** (0.53)	0.13 (0.20)
Age	0.001*** (0.0000)	-0.002*** (0.0003)	-0.002*** (0.0002)
Year2018	-0.82*** (0.02)	-0.68*** (0.22)	-0.09 (0.15)
Year2019	-1.09*** (0.02)	-0.66*** (0.25)	0.07 (0.15)
Year2020	-0.95*** (0.02)	-0.28 (0.24)	0.36** (0.16)
weekdayThursday	-0.02*** (0.001)	-0.03*** (0.01)	-0.0001
weekdaysFriday	0.02*** (0.001)	-0.02** (0.01)	-0.03*** (0.01)
weekdaysWednesday	-0.005*** (0.001)	-0.001 (0.01)	0.002 (0.01)
weekdaysMonday	-0.02*** (0.001)	-0.03*** (0.01)	-0.02** (0.01)
weekdaysSaturday	0.01*** (0.002)	-0.07*** (0.01)	-0.08*** (0.01)
weekdaysSunday	0.01*** (0.002)	-0.08*** (0.01)	-0.09*** (0.01)
log($marketGasPrice$)$group2$			-0.43*** (0.15)
log($marketGasPrice$)$group3$			-0.64*** (0.12)
log($marketGasPrice$)$group4$			-0.49*** (0.10)
log($marketGasPrice$)$group5$			-0.28*** (0.09)
Observations	370,392	370,392	370,392
R2	0.78	0.11	
Incremental F	121.39		
C-D Wald F Stat.	2542.47	118.07	
Stock-Yogo Critical Value	16.38	26.87	
Kleibergen-Paap LM Stat.	70.04***	25.16***	

HAC standard-errors in parentheses.
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Appendix A – Additional formulas

Block time

Ethereum adjusts the mining difficulty for every new block according to the following function:

$$blockTime_b = \frac{miningDifficulty_b}{networkHashRate_{b-1}}$$

Where $miningdifficulty_b$ is the average number of hashes it requires to find a new block and $networkhashrate_{b-1}$ is the number of hashes computed per second by all miners while searching for the previous block.

Mining reward

To incentivize miners to provide their computation service, they are rewarded with a mining reward for every block they find. This reward consists of a static block reward (at the time of writing, 2 Ether) for finding a new block plus the sum of all gas fees (usually measured in $GWei$; 1 Ether = 10^9 GWei) paid by all transactions t which a miner includes in this block. Hence, the mining reward for every block b is:

$$miningReward_b = 2 + \sum_{\forall t \epsilon b} \frac{gasPrice_t \times gasUsed_t}{10^9}$$

Transaction fees

On Ethereum, users only pay for the used gas if the computation is finished before reaching the limit. Also, only the actually used gas is considered for the block gas limit. Accordingly, the fees a user has to pay for a transaction t are computed as follows:

$$transactionFees_t = \frac{gasPrice_t \times gasUsed_t}{10^9}$$

Appendix B – Descriptive Statistics

Table 3: Descriptive Statistics

	dapps*days (N = 379,748)	Mean	SD	Min	Max	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)	(17)	(18)	(19)
(1)	gasUsed	180,178	1,288,645	21	85,346,148	1																		
(2)	transactionActivity	893	9,213	1	518,357	0.89	1																	
(3)	EOA	288	3,143	1	168,900	0.82	0.97	1																
(4)	avgGasPricePaid	28	44	0	6,250	0.05	0.05	0.05	1															
(5)	marketGasPrice	14	1	54	0.06	0.06	0.06	0.71	1															
(6)	difficultyBomb	65	210	0	1,610	-0.01	-0.01	-0.01	-0.09	-0.13	1													
(7)	networkUtilization	301	195	84	1,385	0.02	0.02	0.02	0.25	0.23	-0.2	1												
(8)	networkUtilization2	0.24	20	-228	153	0.01	0.01	0.01	0.01	0.06	0.001	0.06	1											
(9)	log($EtherPrice$)	0.85	0.1	0.3	0.98	0.03	0.04	0.04	0.37	0.54	-0.12	0.34	0.04	1										
(10)	log($EtherVolatility$)	0.73	0.17	0.09	0.97	0.04	0.04	0.04	0.4	0.57	-0.12	0.36	0.04	1	1									
(11)	gasLimit	9,278	1,739	6,704	12,485	0.06	0.06	0.06	0.49	0.77	-0.21	0.12	0.06	0.51	0.54	1								
(12)	Age	415	322	1	1,280	0.05	0.08	0.08	0.23	0.33	-0.11	-0.15	0.03	0.24	0.24	0.49	1							
(13)	avgGasRequirement	322	478	21	9,900	0.04	-0.02	-0.02	-0.05	0.01	-0.02	-0.08	0.00	0.01	0.01	0.05	-0.1	1						
(14)	avgValue	366	3,656	0	99,002	0.01	0.00	0.00	0.01	0.01	0.00	0.02	0.01	0.01	0.01	0.01	0.01	0.01	1					
(15)	avgTokens	2,781	13,909	0	185,968	0.05	0.04	0.03	0.02	0.01	0.01	0.02	0.01	0.00	0.00	0.01	0.07	-0.03	0.01	1				
(16)	avgDailyTxn	893	5,695	1	71,089	0.53	0.62	0.61	0.02	0.01	0.00	0.00	0.00	0.01	0.01	0.01	0.05	-0.04	-0.01	0.06	1			
(17)	avgDailyEOA	288	1,954	1	24,975	0.5	0.61	0.62	0.02	0.01	-0.01	0.00	0.00	0.01	0.01	0.01	0.05	-0.04	-0.01	0.05	0.99	1		
(18)	avgTxnPerEOA	6	20	1	354	0.03	0.01	-0.01	0.01	0.04	-0.01	-0.02	0.01	0.02	0.02	0.05	-0.04	0.09	-0.01	-0.02	0.01	-0.02	1	
(19)	txnPerEOA	6	44	1	4,488	0.07	0.02	-0.01	0.00	0.01	0.01	-0.01	0.001	-0.01	-0.01	0.01	-0.04	0.04	-0.01	-0.01	0.01	-0.01	0.46	1
(20)	surplusGasPrice	19	30	-129	6,249	0.04	0.05	0.04	0.93	0.44	-0.06	0.23	-0.01	0.25	0.27	0.29	0.15	-0.08	0.01	0.03	0.02	0.02	0.00	-0.01

Appendix C – Network-level analysis

In addition to the dApp-level analysis, we created a second data set that aggregates all network transactions. This additional analysis allows us to estimate network-level demand curves (i.e., one demand curve for all transactions), compare the demand curves between Ether transfers between two externally owned accounts and dApp transactions, and estimate a separate demand curve for every group of dApps by filtering only transactions to a specific group of dApps. Further, it ensures comparability with other studies that conduct their analysis only on the network level (e.g., Donmez and Karaivanov 2021, Ilk et al. 2021). The variables we use in this analysis are analogous to the dApp level data set. Table 4 depicts summary statistics and correlations of this dataset.

Table 4: Descriptive statistics and correlations (network level)

Variables	N	Mean	S.D.	1	2	3	4	5	6	7	8	9	10	11	12
1. gasUsed	1,280	45.42	17.15	1											
2. gasUsed group 1	1,280	18.96	18.65	0.88	1										
3. gasUsed group 2	1,280	0.39	0.66	-0.5	-0.28	1									
4. gasUsed group 3	1,280	2.43	1.77	-0.04	-0.25	-0.23	1								
5. gasUsed group 4	1,280	0.86	0.61	-0.09	-0.27	-0.12	0.46	1							
6. gasUsed group 5	1,280	0.56	0.53	-0.21	-0.2	0.09	-0.14	-0.42	1						
7. marketGasPrice	1,280	6.75	12.29	0.73	0.86	-0.16	-0.33	-0.33	-0.15	1					
8. difficultyBomb	1,280	1.08	2.92	-0.48	-0.23	0.25	-0.25	-0.06	-0.05	-0.12	1				
9. networkUtilization	1,280	0.83	0.13	0.73	0.53	-0.6	0.01	-0.2	0.03	0.45	-0.18	1			
10. EtherPrice	1,280	327.48	218.96	0.1	0.11	-0.04	-0.19	-0.62	0.64	0.13	-0.16	0.27	1		
11. EtherVolatility	1,280	0.36	23.46	0.03	0.05	-0.01	0.04	-0.01	0.04	0.05	0.01	0.03	0.07	1	
12. gasLimit	1,280	0.01	0.002	0.93	0.9	-0.41	-0.08	-0.02	-0.29	0.75	-0.31	0.53	0.001	0.03	1

The baseline specification for our network level is analogous to our dApp level specification but without dApp-level fixed effects:

$$\log(gasUsed_t) = \alpha_0 + \alpha_1 \log(marketGasPrice_t) + \alpha_2 networkUtilization_t + \alpha_3 networkUtilization_t^2 + \alpha_4 \log(EtherPrice_t) + \alpha_5 \log(EtherVolatility_t) + \alpha_6 \log(gasLimit_t) + \mu_{dayofweek} + \mu_{year} + trend + u_t$$

where gas used is the equilibrium gas demand aggregated over all executed transactions on the network or per group of dApps in the period t (day), $\mu_{\text{dayofweek}}$ denotes the day of week fixed effects, μ_{year} the year fixed effects, and u_{t} is the error term. We chose a log-log specification for gas used and market gas price to be able to interpret α_1 as the price elasticity of the demand. Due to the skewed distributions of Ether price, Ether volatility, and the gas limit, we use log-transformed versions of these variables in our specification. In addition, we also control for the level of network utilization. This allows us to control for the degree to which miners use the available block gas limit on a given day and has been used by prior scholars as a measure of network congestion (Donmez and Karaivanov 2021). We also add a quadratic term to account for the nonlinear relationship between gas price and network utilization.[9]

Baseline network-level results

Following the network-level specification, Table 8 reports the results of our 2SLS demand curve estimation. Column 1 presents the first stage results, where we predict the gas price (*log(Market gas price)*) with our IV (*difficulty bomb*). Column 2 presents the second stage results, where we use

[9] We also compute the same model with a threshold specification where we added only the linear term and dummy variable that takes on the value one if the utilization level exceeds 90%. The were qualitatively the same regarding the magnitude and significance of the coefficients we obtained.

the predicted gas price to estimate the price elasticity of the gas demand (*log(Gas used)*). Finally, column 3 provides an OLS model for comparison.

Table 5: 2SLS model with 1st and 2nd stage and OLS benchmark (network level)

	(1) 2SLS 1st stage log($marketGasPrice$)	(2) 2SLS 2nd stage log($gasUsed$)	(3) OLS log($gasUsed$)
difficultyBomb	0.10*** (0.02)		
log($marketGasPrice$)		-0.69*** (0.16)	-0.04** (0.02)
networkUtilization	-3.03*** (0.35)	-1.58*** (0.43)	0.20 (0.19)
networkUtilization2	17.51*** (1.85)	10.38*** (2.60)	-0.33 (0.87)
log($EtherPrice$)	0.09 (0.13)	0.06 (0.08)	0.12** (0.05)
log($EtherVolatility$	-0.02 (0.02)	-0.01 (0.01)	0.001 (0.003)
log($GasLimit$)	3.08*** (1.11)	3.02*** (0.99)	0.53* (0.28)
DThursday	-0.04 (0.03)	-0.03 (0.02)	-0.001 (0.002)
DFriday	0.01 (0.03)	0.005 (0.02)	-0.001 (0.003)
DWednesday	-0.02 (0.02)	-0.01 (0.02)	0.0002 (0.002)
DMonday	-0.05 (0.03)	-0.03 (0.02)	-0.00004
DSaturday	-0.02 (0.04)	-0.01 (0.02)	-0.01 (0.01)
DSunday	-0.03 (0.04)	-0.02 (0.02)	-0.01 (0.01)
D2018	-1.21*** (0.20)	-0.85*** (0.26)	0.13 (0.19)
D2019	-1.61*** (0.29)	-1.11*** (0.30)	-0.005 (0.24)
D2020	-1.30** (0.62)	-0.90** (0.40)	-0.03 (0.27)
Trend	0.001 (0.001)	0.001* (0.0005)	0.001*** (0.0003)
Constant	-13.30 (18.66)	-2.97 (12.00)	-7.81 (6.25)
Observations	1,279	1,279	1,279
R2	0.79		0.94
F Statistic (df = 16; 1262)	305.20***		1,220.08***
C-D Wald F Stat.		85.06	
Stock-Yogo Critical Value		16.38	
Kleibergen-Paap LM Stat.		4.18**	

Note: Heteroskedastic and autocorrelation consistent (HAC) standard errors are shown in parentheses, where the optimal bandwidth (23) is calculated following Newey and West (1987).

Signif. Codes:
**: 0.01, **: 0.05, *: 0.1

To establish robustness, we ran a series of alternative models of the network-level analysis similar to the robustness checks reported in the main paper. Table 6 reports the results of these robustness checks.

Table 6: Robustness checks

	(1) Baseline	(2) Alternative Dependent variable	(3) Alternative market gas price (25th percentile)	(4) Alternative market gas price (average gas price)	(5) Alternative market gas price (normalized by ETH supply)	(6) Alternative instrument (block difference)	(7) Outliers (5th-95th percentile gas used)	(8) Subsample (specific difficulty bomb period)
	log($gasUsed$)	log($txnCount$)	log($gasUsed$)	log($gasUsed$)	log($gasUsed$)	log($gasUsed$)	log($gasUsed$)	log($gasUsed$)
log($marketGasPrice$)	-0.69*** (0.16)	-0.63*** (0.15)	-0.80*** (0.20)	-1.83** (0.61)	-0.57** (0.24)	- -0.75** (0.24)	-0.69** (0.19)	-2.70 (2.85)
Observations	1,279	1,279	1,279	1,279	1,279	1,279	1,279	101

HAC standard-errors in parentheses.
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Differing Demand Curves per Group

In addition to estimating a demand curve for all transactions on Ethereum, we also estimate a specific demand curve for every group of dApps along with their confidence intervals. Table 11 reports the second stage result of this estimation. Each of these models uses the aggregated daily gas used by all dApps within the respective group as the dependent variable. Columns 2-6 depict that the coefficients of log(*Market gas price*) significantly vary between the groups of dApps and thus signal that the groups differ regarding their sensitivity to changes in the gas price.

Table 7: 2SLS model with 1st and 2nd stage and OLS benchmark (network level)

	(1) 2SLS 2nd stage log($gasUsedbyalldApps$)	(2) 2SLS 2nd stage log($gasUsedbygroup1$)	(3) 2SLS 2nd stage log($gasUsedbygroup2$)	(4) 2SLS 2nd stage log($gasUsedbygroup3$)	(5) 2SLS 2nd stage log($gasUsedbygroup4$)	(6) 2SLS 2nd stage log($gasUsedbygroup5$)
log($marketGasPrice$)	-0.45*** (0.14)	-0.0464	0.09 (0.19)	-2.09*** (0.63)	-0.59*** (0.13)	-0.48*** (0.17)
networkUtilization	-1.04*** (0.36)	-0.27 (0.41)	-0.84 (0.61)	-2.37 (1.67)	-0.4368	-1.05** (0.51)
networkUtilization2	6.61*** (2.25)	2.51 (2.58)	2.89 (3.60)	17.04* (10.24)	5.44* (2.81)	7.20** (3.04)
log($EtherPrice$)	0.20** (0.08)	0.39*** (0.08)	0.03 (0.09)	-0.02 (0.23)	-0.93*** (0.09)	0.37*** (0.10)
log($EtherVolatility$)	-0.0000 (0.01)	0.01 (0.01)	-0.02 (0.02)	-0.005 (0.03)	0.02 (0.02)	-0.02 (0.01)
log($gasLimit$)	2.49*** (0.92)	1.56 (1.05)	-0.75 (1.07)	7.61*** (2.28)	1.88** (0.86)	2.68*** (0.91)
DThursday	-0.03 (0.02)	-0.02 (0.02)	0.02 (0.04)	-0.12 (0.08)	-0.0015	-0.09** (0.04)
DFriday	0.01 (0.02)	0.01 (0.02)	-0.04 (0.04)	0.03 (0.07)	-0.02 (0.03)	-0.13*** (0.04)
DWednesday	-0.002 (0.02)	0.004 (0.01)	-0.02 (0.03)	-0.06 (0.05)	-0.03 (0.02)	-0.0024
DMonday	-0.02 (0.02)	-0.01 (0.02)	-0.03 (0.04)	-0.10 (0.07)	-0.06** (0.03)	-0.12*** (0.03)
DSaturday	-0.04 (0.03)	-0.07*** (0.03)	-0.09** (0.04)	0.13* (0.07)	-0.0018	-0.13*** (0.05)
DSunday	-0.04 (0.02)	-0.08*** (0.02)	-0.004	0.14* (0.07)	-0.07** (0.03)	-0.13*** (0.05)
D2018	-1.25*** (0.28)	-1.36*** (0.35)	-0.26 (0.31)	-1.29 (1.15)	-0.66** (0.28)	-0.23 (0.30)
D2019	-1.53*** (0.32)	-1.80*** (0.40)	-0.23 (0.38)	-1.69 (1.43)	-0.41 (0.35)	0.22 (0.38)
D2020	-1.35*** (0.38)	-1.61*** (0.42)	-0.29 (0.44)	-1.90 (1.35)	-0.34 (0.40)	1.37*** (0.42)
Trend	0.002*** (0.0004)	0.003*** (0.0005)	-0.001** (0.001)	0.002 (0.001)	0.0004 (0.001)	-0.003*** (0.001)
Constant	-0.03 (10.36)	-18.54 (12.14)	35.66** (14.67)	16.61 (30.89)	24.97* (13.23)	83.40*** (12.13)
Observations				1,279		
C-D Wald F Stat.				85.06		
Stock-Yogo Critical Value				16.38		
Kleibergen-Paap LM Stat.				4.19**		

Heteroskedastic and autocorrelation consistent (HAC) standard errors are shown in parentheses, where the optimal bandwidth (23) is calculated following Newey and West (1987). All models use the first-stage regression reported in Table 8.

Signif. Codes: ***: 0.01, **: 0.05, *: 0.1

To compare the different gas price elasticities, we also compute their 95 percent confidence intervals. Figure 3 depicts these intervals and shows that not all elasticities can be distinguished with enough confidence, but some significant differences are still noticeable. Especially games and marketplaces (group 3) seem to be far more sensitive to changes in gas prices than dApps in group 1 and group 2. Considering that group 3 mainly comprises collectible games, such as crypto kitties, where the timing of the transaction does not matter as much as, for example, finance or cryptocurrency exchange dApps, where the timing often matters due to swift changes in prices of cryptocurrencies, this result seems plausible. Further, the one-time nature and relatively high transaction values in group 2 (identify and property dApps) can explain why users are relatively insensitive to changes in the gas price.

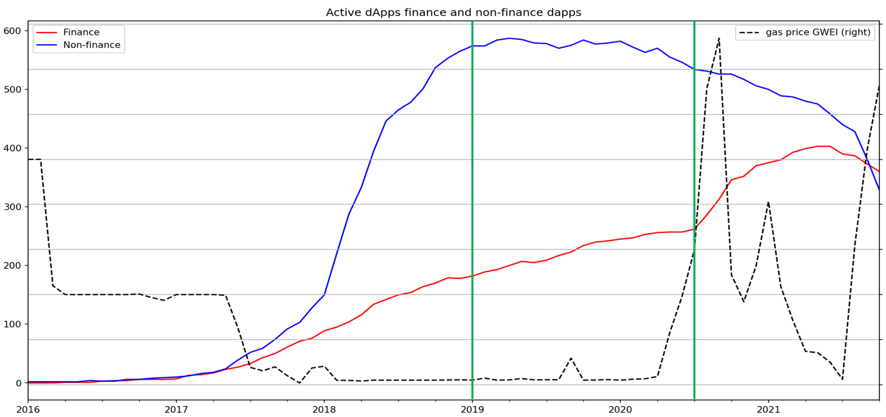

Figure 3: Price elasticities of demand per group of dApps

Appendix D – Supplementary survival analysis

To investigate the impact of Ethereum's transaction validation mechanism on platform complements' heterogeneity, we examine our explanatory variables' simultaneous effect on the overall hazard-rate function by using the semi-parametric Cox proportional-hazards regression analysis (Cox 1972). Previous scholars have used Cox-proportional hazard models to study market exit or entry (e.g., Agarwal and Gort 2002, Huang et al. 2013). In our benchmark specification, we estimate the hazard of dApp d leaving the market on day t as:

$$h_{dt} = h_o(t)exp\{\beta_x^{'} x_t\}$$

Where h_0(t) is the baseline hazard, $x_t$is a vector of explanatory and control variables pertaining to time t. With this model, we are not interested in predicting the exit time but the effect of gas price as a time-dependent covariate. For the analysis, we cluster the standard errors on the dApp level to control for heteroskedasticity and nonindependence of observations. Further, we stratify our observations by the group of the dApp. This allows us to account for different baseline hazard rates between the groups of dApps. To measure market exit, we leverage the fact that stateofthedapps.com reports the status of dApps and classifies discontinued dApps as "abandoned." For the exact timing of the market exit, we take the date of the last transaction a dApp has received. Table **??** reports the results of our analysis. Column 1 shows our benchmark specification. Column 2 depicts the gas price interacted with the group of the dApp.

Table 8: 2SLS model with 1st and 2nd stage and OLS benchmark (network level)

	(1) all dApps stratified by group	(2) all dApps stratified by group
log($MarketgasPrice$)	0.02 (0.09)	-0.187
log($MarketgasPrice$) $\times$ $group2$	0.49** (0.23)	
log($MarketgasPrice$) $\times$ $group3$	0.15 (0.10)	
log($MarketgasPrice$) $\times$ $group4$	0.21** (0.09)	
log($MarketgasPrice$) $\times$ $group5$	0.22* (0.12)	
networkUtilization	-6.68 (8.24)	-6.89 (8.18)
networkUtilization2	4.01 (5.32)	4.15 (5.28)
log($EtherPrice$)	-0.04 (0.14)	-0.02 (0.14)
log($EtherVolatility$)	0.01 (0.04)	0.01 (0.04)
log($gasLimit$)	1.07 (0.71)	1.11 (0.71)
Year of entry dummies	YES	YES
Observations	783,619	783,619
Market exit events	399	3991
Log-likelihood	-2,088.39	-2,083.79

Note: Robust standard errors are clustered at the group level and reported in parentheses. Hazard ratios can be calculated by exponentiating the coefficients reported for each variable.
Signif. Codes: ***: 0.01, **: 0.05, *: 0.1

Our benchmark specification shows no significant impact of the gas price on the survival of a dApp. However, after interacting the gas price with the group of a dApp (Column 2), we find that a 10% increase in the Market price (∼0.095 increase in log(Market price) is associated with a reduction of the hazard rate (β = -1.7; hazard rate = exp(0.095×-1.7) = 0.851) by around 16.9% for our base category (group 1, finance dApps). The positive and (except for group 3) significant interactions indicate that all other groups of dApps profit less from a higher gas price and face a higher likelihood of market exit. For instance, for group 2, the hazard rate decrease only equals 10.9% (exp((-1.7 + 0.49) $\times$0.095)=0.891).

The results of our hazard model suggest that an increase in the market gas price reduces the likelihood of a market exit on a given day, but groups differ significantly regarding this effect. Especially when considering that the gas price fluctuates quickly and sometimes doubles or even triples within a month (e.g., January 2018, June 2020 at the start of the Defi hype), these results can be of economic significance. Further, the result seems plausible as an increase in the gas price is typically the consequence of increased demand for gas caused by more transaction activity with dApps. Again, however, we can see that dApps from group one benefit more from this effect than other dApps and thus have an overall higher likelihood of staying in this market. This differentiating effect is problematic as it corroborates our main argument by showing that a market for transactions disproportionately favors a specific type of dApps and thus leads to a long-run reduction of the heterogeneity of dApps offered on the Ethereum platform.