applied sciences

MDPI

Review

The Blockchain Oracle Problem in Decentralized Finance—A Multivocal Approach

Giulio Caldarelli [1,*] and Joshua Ellul [2]

[1] Department of Business Administration, University of Verona, 37129 Verona, Italy
[2] Centre for Distributed Ledger Technologies, University of Malta, 2080 Msida, Malta; joshua.ellul@um.edu.mt
* Correspondence: giulio.caldarelli@univr.it

Abstract: Decentralized Finance (DeFi) takes the promise of blockchain a step further and aims to transform traditional financial products into trustless and transparent protocols that run without involving intermediaries. Similar to how 2017 was the year of ICOs, 2020 was the year of DeFi, with more than fifteen billion dollars of total investments. The decentralized platforms utilize oracles to retrieve asset data from the external world, but their choice and management criteria are often unknown to the end-users. If oracles are poorly selected or managed, the funds of a rising number of investors are inevitably in danger. The issue, known as "the oracle problem", which makes real-world applications controversial and debated due to the loss of decentralization, had recently drawn attention to DeFi, given the crescent number of related hacks that caused the loss of millions of dollars held in DeFi projects. Through a multivocal approach that considers academic papers, whitepapers, preprints, and opinion posts, this study aims to shed light on the pattern that identifies the oracle problem in DeFi and outline the most promising ways to overcome the related weaknesses. This research supports the view that the oracle problem in decentralized finance bears specific characteristics which require standardization and appropriate economic incentives to be addressed.

Keywords: blockchain; decentralized finance; oracles; smart contracts

Citation: Caldarelli, G.; Ellul, J. The Blockchain Oracle Problem in Decentralized Finance—A Multivocal Approach. *Appl. Sci.* **2021**, *11*, 7572. https://doi.org/10.3390/app11167572

Academic Editor: Gianluca Lax

Received: 9 July 2021
Accepted: 14 August 2021
Published: 18 August 2021

1. Introduction

Decentralized Finance (DeFi) is proving to be one of the most significant use-cases for public blockchains, with over fifty billion dollars of value locked and growing [1,2]. It has exploded from a niche market to a billion-dollar industry, bringing new applications, including stablecoins, crypto-collateralized lending, trustless margin trading, and decentralized exchanges [3]. However, in parallel with the growth of investments in DeFi, related incident and hack rates are also increasing, with over $153M stolen only in 2020 [4]. Many were made possible by exploiting smart contract vulnerabilities, such as "re-entrancy bugs", infamous for being one of the causes of the DAO attack [5]. On the other hand, a considerable amount of DeFi hacks were possible by exploiting oracle-related vulnerabilities [6,7]. While developers are familiar with smart contract vulnerabilities, price oracle manipulation is clearly not something that is often considered. Conversely, exploits based on re-entrancy have fallen over the years, while exploits based on price oracle manipulation are now on the rise [8]. Recent articles consider that almost 2/3 of DeFi hacks were possible thanks to oracle exploitation [9].

There are several well-known DeFi protocols, such as Compound, MakerDao, Uniswap, and Aave, that use oracles for fetching external data. In these use-cases, oracles are used to gather cryptocurrency exchange rate data and send them to requesting DeFi applications [10]. There is, however, a controversial point to consider. While DeFi applications are more or less decentralized, oracles are often centralized and trusted third parties who provide data through unverified and unsecured channels [11]. As Anadtiotis [12] explains, various DeFi protocols run decentralized despite potentially relying on centralized and insecure oracles. Therefore, calling those DeFi protocols "Decentralized" is a euphemism [12].

This issue, brilliantly described by Egbertz [13] and discussed in a recent book [14], is known as "the oracle problem" and affects all blockchain real-world applications. The consequences of having insecure oracles are different according to each real-world application, but in the DeFi space, considering the rising amount of investments, the potential drawbacks of a poorly designed oracle could be detrimental. Despite the criticality of the issue, this problem has not raised much interest in neither industry nor academia. A recently published article shows that only 15% of journal articles about real-world blockchain applications take into consideration oracle-related issues [15,16]. Despite being the most problematic considering the amount of capital involved, the DeFi sector, is not taking a step further to address the problem. As a recent study on DeFi oracles outlines, "Although oracles play a critical role in DeFi ecosystem, the underlying mechanics of oracles are still vague and unexplored" [17]. This paper's scope is to analyze the oracle's role in the DeFi space, focusing on the most widely used applications and outlining the issues connected to their implementation [18]. Inspired by the works of Al-Breiki [19] and Manoj Kumar [20], we aim to identify the roles oracles are playing in DeFi applications. Furthermore, risks and drawbacks are identified and discussed according to their relevance and impact. The following research questions summarize the focus of this study:

1. What is the role of oracles in decentralized finance according to the existing literature?
2. What are the current risks connected with the implementation of oracles in DeFi, and how are they being addressed?

Being an extremely recent topic, given the fact that just around a year has passed since the DeFi explosion [21], gathering enough academic papers may be unfeasible. This is probably not due to the low interest in DeFi but to the time required for publishing and indexing. However, there are many available online articles that discuss the subject and give an overview of the issues related to oracles. For that reason and for the scope of this paper, we decided to adopt a Multivocal Literature Review, including peer-reviewed resources from scholarly databases and so-called grey literature, comprised news, blogs, reports, and other official documents (whitepapers). Given the fact that grey literature is not peer-reviewed, information retrieved from such articles was double-checked by the authors with other available resources, and where not available, companies were directly contacted for clarifications. Through the analysis of recent issues, hacks, and attempts to address the oracle problem, the paper aims to outline a common pattern that can be useful for the development of present and future applications. The study supports the view that, according to the respective applications, oracle design plays a vital role in either enabling or preventing some attacks or manipulations. An approach based on a predetermined set of rules when selecting oracles may then significantly solve technical issues connected with their use in mass adoption. On the other hand, an incentive system could help reduce collusion, deliberate tampering, and data-provider manipulation. This paper aims to provide insights to both academic and professional communities. Investors can benefit from the content provided for being more informed on the risks of DeFi platforms determined by the presence of oracles. Moreover, developers can take guidance from the good practices recommended. On the other hand, academics can build on this paper to produce further reviews or empirical papers drawing on the selected material.

The paper is organized as follows. The next section outlines the literature background along with related works. Section three explains the methodology that was followed, while section four outlines the results. Section five and six summarize the material according to the research question, and section seven provides a discussion of the findings. Section eight concludes the paper by providing hints for further research.

2. Literature Background

As blockchain is traditionally blind to the real world and thought not to be able to fetch data from the outside directly, the range of applications for smart contracts and apps was very limited [13,22]. The idea of an oracle was then introduced in order to overcome this limitation [23]. Oracles are usually centralized and trusted third parties that provide

blockchain with data from the real world. Their role is, in fact, to connect those two realms making it possible for blockchain to be implemented in almost any sector [24]. While a detailed explanation of the oracle's characteristics is beyond the scope of this research, a distinction between centralized and consensus oracles is imperative to grasp all of the research content better:

Centralized oracles, regardless of their nature, are entities controlled by a single authority (or group thereof). A web API or a sensor-controlled by a company/provider is an example of a centralized oracle [25]. What a centralized oracle does is it creates a direct channel between the data source and the smart contract. Oraclize, for example, is a trusted centralized oracle provider [26] whose objective is to prove that the fetched data have not been altered after being gathered [19]. In their official documents, they specify that they are not trustless but "provably" honest. As a matter of fact, being centralized authorities, those oracles cannot operate in a trustless way. It is therefore indispensable to put trust in the centralized entity managing the data. On the other hand, it is easier to perform auditing procedures on a centralized entity. Although centralized oracles may provide any sort of data, they are more suitable for information that is not publicly available. Supply chain and product traceability are typical examples of data fetched by a centralized oracle [27].

Consensus oracles are groups of oracles that determine the data to be uploaded on the blockchain by following a predetermined set of rules [28]. The consensus between multiple oracles can be reached based on the majority of votes/thresholds or any other agreed-upon protocol [29]. The Band Protocol, for example, is a recent consensus oracle project which exploits multiple data sources and applies a mediatization of value to mitigate the effect of outliers [30]. The idea behind the introduction of a consensus mechanism in the oracle activity is to replicate the trustless nature of the blockchain. However, if the oracles and data sources belong to the same entity or group, they still require a certain degree of trust to be implemented. Consensus oracles can better replicate the blockchain trustless environment if oracles belong to different entities, i.e., if they are not centralized. Decentralized consensus, however, is feasible only when multiple devices, entities, and trusted data sources are available. The most common decentralized consensus oracle type is based on the "wisdom of the crowd", of which Augur and Pythia are known examples [31]. The principle of those oracles is that, for some events or knowledge, there are people or entities who are probably aware of their outcome and are willing to bet their reputation or their wealth to prove they are right [32]. Their proposed truth is then put under public scrutiny so that any other entity participating in the platform can confirm or deny the provided data's truthfulness [14]. Ideally, for those types of oracles, publicly available data, would be used to ensure trustworthiness [33]. Therefore, they are suitable to retrieve information that is easy to verify, such as prediction market events and exchange rates [33,34]. Given the aim of this study and targeting a broader audience, we rather not go into the details of oracle implementation and mathematics, which are, on the other hand, widely explained in other articles [35,36].

2.1. The Oracle Problem

Oracles bridge the on-chain blockchain world with the off-chain world by interfacing with external data providers [10]. Since oracles are fundamental for smart contracts, and given that they are trusted entities, they have the "privilege" to feed in data that are accepted unconditionally. This small detail is crucial, since the whole ecosystem of blockchain revolves around the concept of immutability and trustless interaction through decentralization. Connecting the blockchain with a centralized point of failure, such as an oracle, would arguably cause a loss of decentralization [37]. This conundrum is known as "the oracle problem" and affects all the real-world blockchain applications, and depending upon the sector and oracle type, different consequences may emerge [16,38,39]. There are actually many circumstances for which an oracle may fail to provide reliable data. Oracles may be poorly programmed and experience bugs or may suffer from sabotage or malfunction. Song [39] also specified that the oracle problem not only results from

technical issues but also depends on social aspects. As Paul Sztorc [38] explains, in fact, the chance for an oracle to be compromised or the administrators to collude to alter the data provided is correlated to the smart contracts' value. It means that, the higher the value within the smart contract, the higher the chance that it will be attacked through the exploitation of associated oracles. This condition leads to the conclusion that, even if the oracle is well designed and the data source is trusted, the data can still be compromised by collusion or the bribing of the administrators. Figure 1 summarizes the two dimensions of the oracle problem.

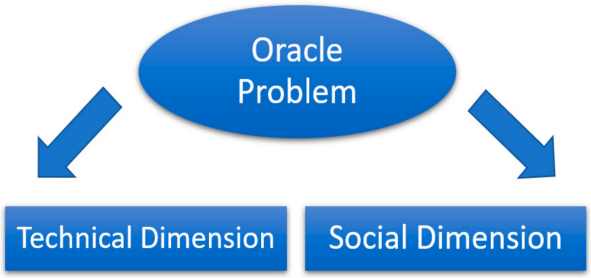

Figure 1. Dimensions of the oracle problem.

In essence, the technical dimension refers to the events in which, while the oracle behaves honestly, it fails to transmit the data as a result of bugs or malfunctions. On the other hand, the social dimension refers to the case in which, despite the oracle being well programmed, and the data source being reliable, the entity managing the oracle alters the data transfer for selfish purposes. Although ideally separated, the two dimensions are intertwined so that, most of the time, when oracles behave incorrectly, it is unfeasible to distinguish which dimension is causing the problem. Furthermore, while there are many attempts to address the oracle problem, most of them focus on the technical component, since the social dimension is inherently more controversial to detect and address [13,39,40]. The trust model is a document that tries to prove the reliability of an oracle from a social and technical point of view [41], and most of the time, it coincides with the oracle platform whitepaper [19].

2.2. Narrowing the Oracle Problem in Decentralized Finance

As explained above, the blockchain oracle problem refers to the inability to determine the veracity of data provided by oracles [39]. The uncertainty may arise from an unreliable data source, low oracle reputation, or both [42]. However, as discussed in a recent paper [16], the consequence of this condition varies according to the specific sector in which blockchain is implemented. In the supply chain, the oracle problem refers to the fact that information collected on the blockchain is filtered by the producing company so that unwanted or sensitive information may not be registered [27,43]. In the academic sector, the oracle problem does not affect the authenticity of the transcript but does question the credibility of the issuing authority [41]. A certified issue by a low-ranked university, in fact, will not gain more credibility for being stored on a blockchain [44]. In the IPRs field, the oracle determines a more social problem in which authors and certification authorities are vying for the role of the oracle, as the one who does, obtain a greater power over the other party [45]. In resource management, since the data flow is bi-directional, namely when resources are both stored and shared, there is the need for two types of oracles (inbound and outbound). This further dependency, of course, doubles the problem. Finally, in the case of health records, the oracle involvement can constitute an additional attack vector for hackers to steal or modify patient records. When multiple oracles and external databases are implemented, and there is the inability to monitor their security actively, it will also be impossible to determine the reliability of patient's data on the blockchain [14].

Decentralized finance, as a real-world application, requires oracles to operate. However, the impact of their implementation strictly depends on the extrinsic data required. In

DeFi, data such as KYC are not required (at the moment), since smart contracts manage all the transactions, and no centralized authority supervises the identity of the contractors [46]. Therefore, oracles are not implemented to collect personal data. Consequently, no GDPR or security issue arises, and no external server is needed for sensitive data management [47]. The only data that remain transparent are the transactions that only belong to pseudonym addresses.

Unlike the academic sector and IPRs management, a smart contract's authorship is guaranteed solely and exclusively by the private key that signs the agreement. Theoretically, whether the person who uses the private key is the legitimate owner of the wallet or not is not relevant for the correct execution of a DeFi contract [21,42]. Since not regulated through a KYC procedure, there is no way to enforce an unwanted operation on a crypto wallet. On the other hand, as the application is decentralized, there is also no authority to appeal. The last thing to consider to narrow the oracle involvement in DeFi environment is the fact that those applications are meant to communicate with each other (interoperability) so that data flow in both directions. This creates similar issues as those found in blockchain applications for resource management (dual oracle problem) [48].

Regardless of the specific decentralized financial application, the only data required pertains to financial assets' quantity and price. Unlike the data production of a traced bottle of wine, asset data constitute publicly available knowledge that can easily be verifiable. Oracles based on the wisdom of the crowd should then be able to "trustlessly" fetch this sort of data. However, malfunctions, tampering, and collusion can still easily alter the data provided. Being financial contracts that often manage transactions of millions of dollars also dramatically affects the incentive to alter the communication channel [38]. Thus, despite the fact that the asset data are publicly available knowledge, there are still many issues that can prevent correct information from being registered on the chain. We could argue that the oracle problem in DeFi applications, from a theoretical point of view, reflects the chance for asset data to be altered by a malfunction or deliberately manipulated for selfish purposes.

To address the oracle problem, many oracle providers such as Chainlink and Oraclize are offering their solutions. However, those are all-purpose oracles and also amongst the most discussed in papers and news articles. Therefore, an overview of oracles specifically made for DeFi projects is provided in Appendix A.

2.3. Related Works

There is still a paucity of works on decentralized finance and even more scarcity on the role of oracles in this field. However, other authors offered perspectives that were useful for the development of this paper. Werner et al. [49], Schar [50], Harvey et al. [51], and Amler et al. [21] provided systematization of knowledge in a broad sense of the concept of decentralized Finance. They discussed many DeFi applications, as well as related opportunities and challenges, and they also provided a small overview of the role of oracles. Liu et al. [17] and Kumar et al. [20] instead focused precisely on blockchain oracles implemented in decentralized Finance. While Kumar et al. [20] offered a theoretical proposal for a DeFi oracle, Liu et al. [17], using primary data, discussed the reliability of oracles in DeFi considering the deviation rate of Ampleforth, MakerDAO, Synthetix, and Compound. Gu et al. [32] and Angeris and Chitra [52] finally produced a piece of research on blockchain oracles in DeFi for a specific application. The first discussed the role of oracles in the governance of non-custodial stablecoins, particularly analyzing the passage of MakerDAO from version V1 to version V2. The second instead focused on the role of oracles implemented for Automated Market Makers. An interesting study, while not directly related to this one by Kaleem and Shi [53], showed the increasing rate of oracle-related queries on the Ethereum platform. As expected, most are initiated by DeFi platforms.

3. Methodology

In order to answer our two research questions, an appropriate methodology was needed. A classic literature review or systemization of knowledge was, indeed, efficient in the process of summarizing the existing studies [54]. On the other hand, as Sutherland [55] explains, a systematic review should be utilized when specific research questions need to be answered and to ensure the replicability of the results guaranteeing the transparency of the data acquisition [56]. In principle, we opted for the second approach. Two important research databases (Scopus and Web of Science) were then queried to obtain the relevant articles, and specific keywords were utilized within two different research entries. While "decentralization" and "finance" were inserted in the TITLE-ABS-KEY and Scope, respectively, on Scopus and Web of Science, the "oracle" keyword was instead searched in the whole body text. The reason for this difference is that, while decentralized finance should have been the main topic of the article, it is not also requested that oracles constitute the primary area of research. Since our aim is to provide a full picture of contributions posed within the literature, regardless of weight, even a small contribution on oracles or the oracle problem in DeFi is relevant for this research. Table 1 outlines the two research strings for each database. The Scopus and Web of Science databases were both queried on 26 February 2021 and returned 14 and 1 entries, respectively. However, the article returned from the Web of Science database was, unfortunately, off-topic, which means that only articles from Scopus were included.

Table 1. Selected databases and keywords.

Database	Research String
Scopus	(TITLE-ABS-KEY (decentralized AND finance) AND ALL (oracle))
Web of Science	(TS = (decentralized finance)) AND ALL = (oracle)

Despite the restricted sample, excluding criteria were implemented to include relevant articles. First, non-English articles were excluded, which reduced the sample to 12 entries. By reading the title and abstract, another five articles were excluded since considered off-topic, despite including the three keywords, which reduced the sample to seven entries. We inspected the article's content to determine whether they discussed oracle or oracle problem issues. We realized that two of the papers were included in the sample for only having references to other articles that include the word "oracle" in the title [57,58]. One article at the end, although using the word "oracle" within the text, referred to the Oracle company and not to blockchain oracles [59]. Since oracles were not actually discussed anywhere in the text, we decided to drop the entries, leading to the final sample (displayed in Table 2) of only four papers. While these papers were very informative and well-written, the authors agreed that robust and meaningful results could not be produced based on only four articles. Considering the relevance that Decentralized Finance had in recent months, the lack of scientific papers, which require a considerable amount of time for publication and indexing procedures, is understandable. On the other hand, the web is constantly providing a plethora of articles that could have highly contributed to our research. For that reason, we first decided to include another database (Google Scholar) and further opted for a Multivocal Literature Review (MLR), which is a form of Systematic Literature Review that also includes so-called "grey literature" [60]. Whilst MLR includes other non-academic literature, the method followed remains the same as a SLR. Rodney et al. [61] describe MLR as literature "comprised of all accessible writings on a common, often contemporary topic. The writings embody the views or voices of diverse sets of authors .... They address different aspects of the topic and incorporate different research or non-research logics". On the other hand, grey literature is composed of documents "produced on all levels of government, academics, business and industry in print and electronic formats, but which is not under control of commercial publishers" [62]. Falling under that grouping is information found in news editions, blogs, websites, or whitepapers. The retrieved

documents are used as complementary material to peer-reviewed articles published and indexed in scholarly databases.

Table 2. First round of research.

Title	Author	Year	Source
"Decentralizing finance using Decentralized Blockchain Oracles"	Kumar M., Nikhil, Singh R.	2020	International Conference for Emerging Technologies, 2020
"Stablecoins 2.0: Economic Foundations and Risk-Based Models"	Klages-Mundt A., Harz D., Gudgeon L.	2020	Conference on Advances in Financial Technologies 2020
"LoC—A new financial loan management system based on smart contracts"	Wang H., Guo C., Cheng S.	2020	Future Generation Computer Systems
"Improved Price Oracles: Constant Function Market Makers"	Angeris G., Chitra T.	2020	Conference on Advances in Financial Technologies 2020

We used two different web search engines to search for relevant grey literature—Google Search and DuckDuckgo. The keywords used were the same as for the academic databases, while at this time, all three were inserted in the research string with no exclusion criteria (e.g., TITLE-ABS-KEY and Topic). We started our research on 3 March 2021 and organized our results in pages of ten entries, and decided to stop our research at the tenth (100 entries). Since using non-peer-reviewed material for academic publications may be debatable, robust exclusion criteria needed to be adopted. We decided to keep the ones with the source we were more familiar with and confident about when similar articles were found. If a piece of news was found on a source that we were not confident of, the content was then verified with further research to decide whether to keep or discard the article. The authors are aware that this condition includes a source of subjectivity. We wish to stress, however, that no articles were dropped according to their source but only when we were not able to double-check the reliability of the content provided. Therefore, we believe that replication of this study—however, not including the same articles—would probably provide the same content. Another thing that the authors wish to stress is that, despite the fact that the article sample was gathered between February and March 2021, later-published articles, such as that by Eskandari et al. [63], are also cited, due to their importance in clarifying some concepts that otherwise would have remained blurry. Furthermore, other sources are cited to expand on some concepts that were not fully explained in the corresponding article.

Despite the higher availability of grey literature with respect to academic papers, we found much redundancy with the results. In particular, we observed not only similar articles but double, triple, and n entries of the same articles within the same research, so that even summing the entries from both searches, we could only retrieve 47 different articles. We then searched for articles that did not use all three keywords in the body text, and we found then eleven articles were about oracles with surrounding ads on DeFi. Other eight articles on DeFi had oracles articles in the suggested sections. We then removed another five articles for being clearly off-topic, which reduced our grey literature sample to 23 entries.

A final Google Scholar search was performed on 26 March 2021, using the same three keywords, and, with the results organized by ten per page, we decided to stop at the fifth (50 entries) due to theoretical saturation (no new concept emerged) [64]. From a sample of 26 potential articles, we excluded 7 for being off-topic. After full-text reading, we excluded the other six articles, since even using the three keywords did not provide data for our research. This lead to a sample of 13 articles from Google Scholar, of which two articles

were already retrieved via Scopus. Therefore, the total number deemed to be retrieved from Google Scholar resulted in 11 articles.

The final sample was then composed of 4 academic articles retrieved from Scopus, 11 academic articles from Google Scholar, and 23 blog articles retrieved from web search. This makes a total of 38 entries.

Although the number of articles retrieved is higher for the grey literature, it must be stated that the main source of this work is still of an academic nature due to the depth that the academic articles delve into. Furthermore, the mean article length for GL type is 1123 words compared to the 10,255 of Academic papers (references excluded). This paper mainly draw its conclusions from academic resources and use grey literature as a complementary asset. Table 3 and Figure 2 summarize the MLR steps.

Table 3. Sources and excluding criteria.

Database	Scopus	WoS	Google Scholar	Web Search
Initial Sample	14	1	26	47
Misplaced keywords	3	0	6	19
Off-topic	5	1	7	5
Non-English	2	0	0	0
Duplicates	0	0	2	0
Final Sample	4	0	11	23

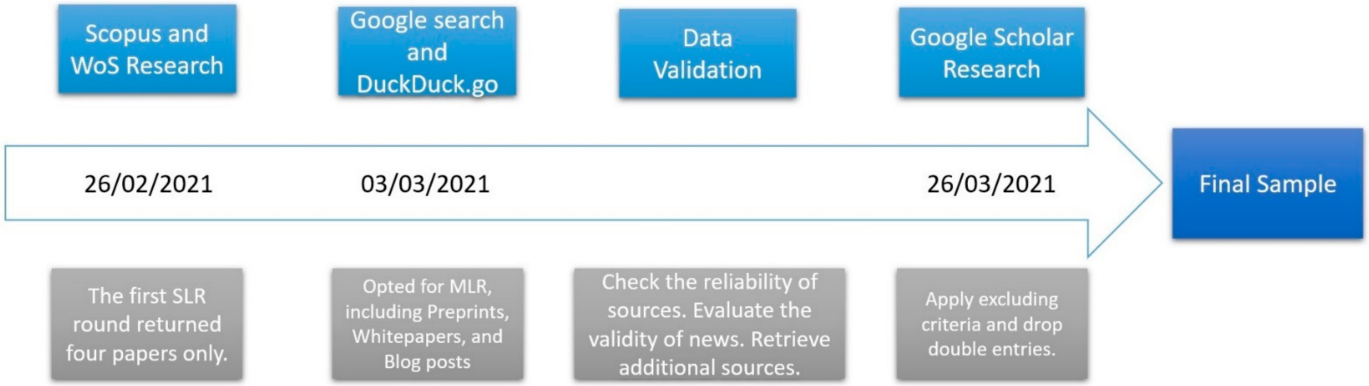

Figure 2. Review steps.

3.1. Data Acquisition

Although some interesting reviews and systemization of knowledge articles were recently published on the DeFi space, to the best of the author's knowledge, none of these provide reproducible patterns and neither address specific questions. On the one hand, this limitation implies that this work is novel; on the other hand, it does not grant the luxury of building upon an existing framework. The data-extraction model had to be perfected from scratch. To reduce the arbitrariness of the research, the author decided to inspire the data extraction to other systematic reviews and the few works of literature produced on the oracle problem within other domains [15,44,56]. Common data attributes extracted include title, year, resource type, and source. Those data fields help contextualize the MLR in a specific timeframe and provide information on the balance between academic and grey literature. Drawing upon Wang et al. [65], we decided to check for risks and attacks, given the oracle's impact on the security of blockchains [13]. A survey and illustration of the most used DeFi applications as discussed in Harvey et al. [51] is also perceived useful to narrow the area of interest for this research. Furthermore, as performed by Al-Breiki et al. [19],

research of the oracle providers with solutions to the oracle problem in DeFi was fulfilled. The complete list of extraction keywords is provided in Table 4.

Table 4. Data-extraction items.

Data Item	Description
Title	Title of the paper
Year	Time of Publication
Source Type	Academic Article or Grey Literature
Source Name	The name of the Site, Journal, or Proceedings
DeFi Applications	Discussed DeFi Applications
Applications type	Which application was described
DeFi Companies	Presented DeFi Projects
Risks and Attack vectors	Discussed risks connected with the use of oracles in DeFi
Issues with oracles	Discussed real incidents in DeFi connected with the use of oracles.
Oracle Providers	Described Oracle providers who operate in the DeFi Domain

3.2. Results

As described in the Methodology section, the exclusion criteria did not restrict articles to those published within a specific timeframe, and this resulted in articles retrieved for only the past three years. Figure 3 depicts the results. First, only articles between 2019 and 2021 were retrieved due to the subject's nascency. As shown, there is an increase in 2020, and as expected, the numbers reported for 2021 are lower, since data gathering was finished by March 2021. However, if the trend remains constant, the number of published articles of 2021 may exceed double of 2020. As discussed in the Methodology section, a majority of articles are grey literature, while the others are academic papers drawn from Scopus (4) and Google Scholar (11). Figure 4 displays this result.

To provide a better breakdown of the results, an overview of topics was then first distinguished into available (Y) and non-available (N), and according to the source between academic papers (AP) and grey literature (GL). The results are shown in Figure 5. The figure already gives an idea of why the MLR was strictly necessary to address the questions of the paper. While both academic and grey literature discussed risks relating to oracles, mainly grey literature outlined "real" incidents with oracles in DeFi. Similarly, while both equally described DeFi applications, mainly grey literature outlined actual DeFi companies and Oracles providers. The paucity of these data in the academic literature gives the idea that the analysis is still at a general level and requires more practical case studies. Furthermore, if not strictly necessary for the empirical analysis, reviewed academic research did not mention real companies (likely due to blind research) and discussed the phenomenon in a broad sense. In other words, it is common to read discussions about Automatic Market Makers and not specific analyses of Uniswap or Curve. In our case, however, it is important to gather data on real projects to understand state-of-the-art with respect to oracle implementation in the DeFi space. For that reason, again, the involvement of grey literature was considered necessary. Finally, we provide a chart in Figure 6, displaying the availability of data for each DeFi sector. The graph displays the number of papers that dedicated a paragraph or a section to a specific application type. Of course, the same paper may have a dedicated paragraph to multiple applications while other may not have dedicated paragraph or sections to a specific application.

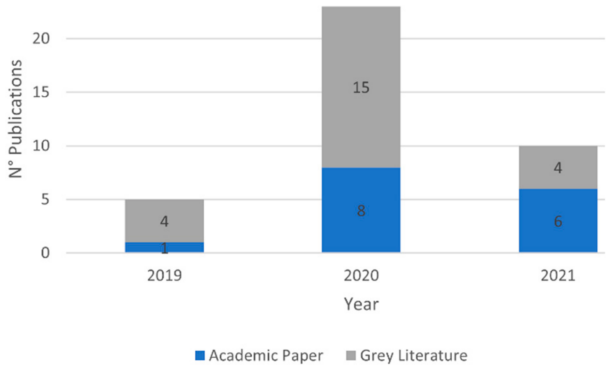

Figure 3. Publications/year.

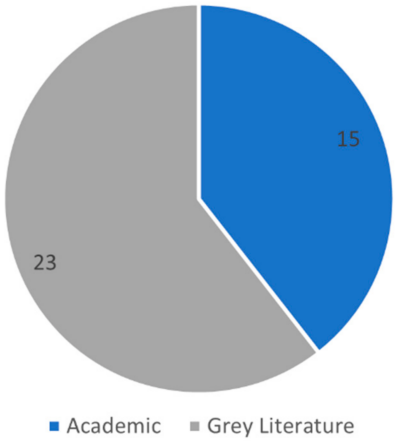

Figure 4. Article type/number.

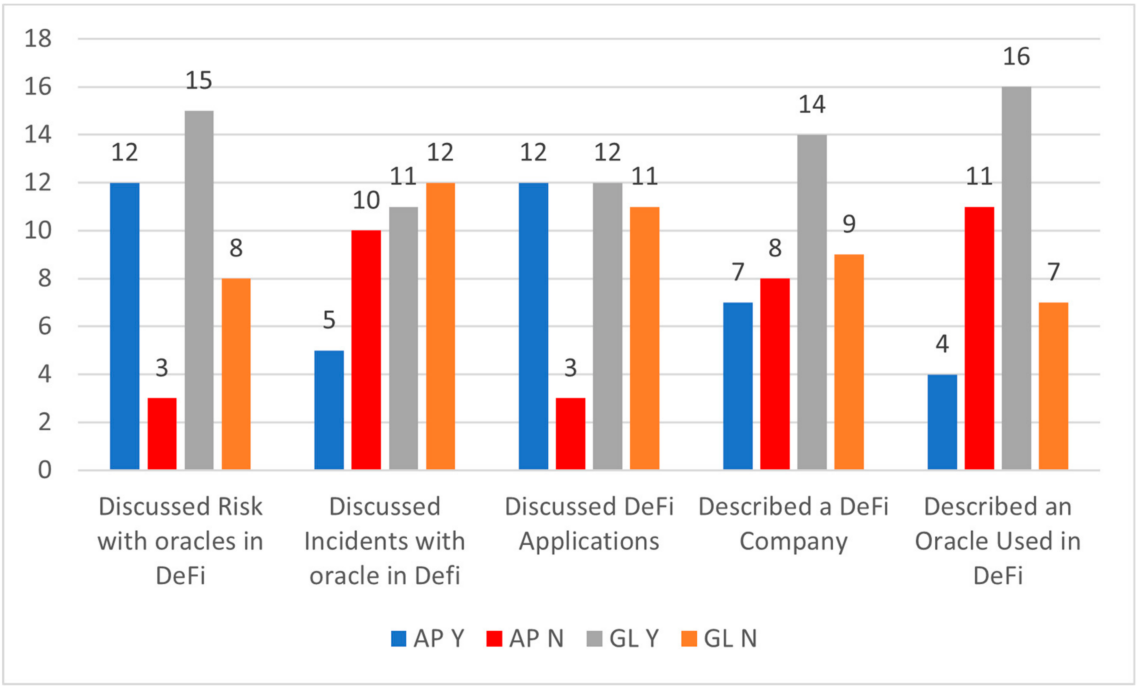

Figure 5. Data-acquisition results.

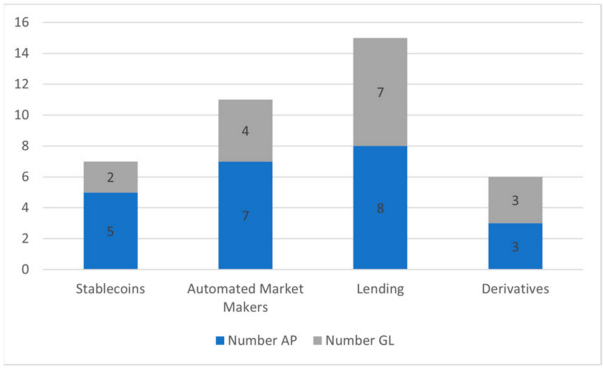

Figure 6. Applications/entries.

From the result, we can observe that academic literature, as well as grey literature, focused mainly on lending and liquidity pools while fewer focused on derivatives. This also reflects the lower availability and early stage of decentralized derivatives services. This, unfortunately, also affects the availability of data on oracles in the derivatives sector. Therefore, while more examples are available in the lending, liquidity pools, and stablecoins field, lesser are provided for derivative ones. Appendix B provides a sample of AP and GL divided by subject to facilitate the reproduction of results. Furthermore, in Appendix C of this paper, two graphs display the most discussed DeFi companies and oracle providers.

4. Oracles in Decentralized Finance

Since DeFi applications differ according to the provided service, related oracle systems vary accordingly. Discussing leading applications in the DeFi space and oracle's role, this section aims to answer our first research question.

4.1. Lending Pools

Not to be confused with Liquidity Pools; lending pools are financial applications that create a crypto-asset loan market. Managed by a system of smart contracts and incentive mechanisms, they create advantages for lenders as well as for borrowers [66]. Loans are an essential part of the financial ecosystem, and in the DeFi platforms, it is possible to lend and borrow crypto assets without KYC procedures [50]. Users can lend their assets to the LPs, and receive an LP token which serves as a sort of receipt of their lending. The LP tokens can then be transferred, and their exchange negotiated like any other token allowing for a sort of crypto securitization of the loan [67]. However, if the token is lost or stolen, the amount lent will also be forfeited.

It is important to note that the main risk for a lender is that a borrower could run away with the money (crypto), and it is crucial in an environment in which the borrower cannot be identified. There are two mechanisms that aim to solve this problem:

Collateralized loans are loans given after the borrower locks a predetermined amount of assets. Usually, the locked amount should exceed the borrowed amount. The smart contract then decides the ratio between collateral and the borrowed amount and the interests to be paid. Of course, in order to recover the locked assets, it is sufficient to refund the loan [50]. Due to the fact that those loans run on a decentralized system, there is no way to renegotiate terms after its execution, and if the contractor is unable to repay the debt, the collateral is liquidated (sold at a discounted rate). Although P2P lending exists, the most common lending type is organized in pools of which Compound, Aave, and dYdX are among the leading dApps [68–70].

Flash loans are a particular form of loans that are only possible thanks to the implementation of smart contracts. The name was coined by Max Wolff, the creator of Marble protocol, in 2018 [71]. It consists of a particular form of contract in which the loan is taken and repaid simultaneously [72]. That way, the illiquidity, and the default risks are denied. In a globalized crypto market, flash loans are beneficial when performing arbitrage. An

investor can borrow an amount of money to buy the crypto from a market, sell them in a market where the price is higher and repay the loan making a profit. All can and should happen in transactions within the same block. Since the loan is taken and repaid simultaneously, there is virtually no limit to the amount of money borrowed (subject to availability of the requested token).

Oracles in lending pools and flash loans are important to determine the price of assets and, in particular, to those held as collateral. The value of a deposited digital asset determines the amount of loan that can be borrowed, and for lenders, the interest that has to be accrued. Unlike Aave and dYdX that outsource oracle-related tasks to Chainlink, Compound has its own price oracle [73–75]. Oracles on Compound are managed by administrators, who are COMP token holders. The administrator deploys a price aggregator contract in which it specifies *min*, *anchor,* and *tolerance* sets. The *min* is the minimum number of reports necessary to calculate a new median value. The *anchor* is the address of the contract that requires the price feeds. Finally, the *tolerance* is the maximum deviation accepted by the contract, which is usually set at 10%. Price oracles on Compound protocol could be represented by major exchanges, other DeFi projects, and Over-the-Counter (OTC) services. If the reports are under the min or if the median value calculated by the aggregator exceeds the tolerance, the value is rejected, and the asset's price will not be updated [17,76]. If oracles fail to provide the right collateral price for lending contracts, there is the risk for it to be under-collateralized. That poses severe threats for the lender to recover the investment if the borrower cannot repay the debt. In flash loans, oracle failures are even more dangerous as they can even damage the whole platform. Details on that circumstance are provided in the next sections.

4.2. Automated Market Makers (AMM)

Automated Market Makers, often referred to as decentralized exchanges, are smart contracts that hold both assets of a trading pair. For example, in the case of ETH/USDT, the smart contracts hold a certain amount of Ether and Tether in what is called a liquidity pool [77]. The price of each asset is derived as a function of availability and, of course, is stabilized by arbitrage. Unlike centralized exchanges, the companies that manage AMMs, have the role of developing contracts, minimize the chance of bugs and malfunctions, but they do not directly provide the assets [3]. The fees paid in the exchange are then shared among the liquidity providers and service providers. Uni-swap, pancake-swap, and just-swap are the most known AMMs in the respective ecosystems: Ethereum, BinanceSmartChain, and Tron [78]. It is important to point out that, as stated by Schär [50], smart-contract-based liquidity pools do not rely on price oracles to operate. According to the author, the product model of a liquidity pool can be expressed as XY = K in its simplest form. Where x and y are the token reserves and k is a constant. If an agent wants to buy Δ' coins of token "y" must put in the swap contract, enough "x" such that the product of the reserves remains constant. Angeris [79] formalize this concept with the following function:

$$(R' - \Delta')(R + \Delta) = R' R$$

Furthermore, the price of assets is derived with a similar principle. It is constantly adjusted so that if the asset X is deposited to take Y, then the price of Y raises, as it will be less and less convenient to keep buying the same asset. That way, it would be indeed profitable the opposite swap (deposit Y to take X) so that the pool should never be drained. Despite not relying on oracles or exchanges to price their assets, liquidity pools, thanks to that mechanism, are sometimes more efficient than centralized exchanges in determining the price of assets. For that reason, DEXes are often selected as price oracles. Uniswap, for example, is being lately chosen as a reliable price oracle by Aave, bZx, Debank, and others [80]. Consequently, the developers have implemented specific features to serve that particular purpose [81]. The Uniswap price oracle evolved, in fact, from V1 to V2, changing from the last swap price feed to a time-weighted average price feed. While the first offered chances for flash-loan attacks, the second type was less exploitable with flash loans. More

technically, while with the V1, every token transacted price was registered in the block and immediately used as a feed, with the V2 version, the feed is extracted as a mean value of 24-h transactions for that specific rate [82]. It is essential to notice that, although liquidity pools do not require oracles to operate, they expose the liquidity providers to the risk of "Impermanent Loss" [83]. This risk arises when one asset significantly changes its price with respect to the other in the contract. This provides an opportunity for arbitrageurs to drain the asset unbalancing the pool. Given the lack of one of the assets, when the LP provider withdraws its liquidity, it will then receive an amount with a lower value than that provided, experiencing a "permanent" loss [84]. Given the seriousness of the issue, lately, platforms such as Bancor are implementing oracles to limit the action of arbitrageurs. On the other hand, other approaches such as the one followed by Balancer include the chance to provide assets also with an unbalanced rate (e.g., 40/60, 90/10, etc.)

4.3. Stablecoins

To better exploit the advantages of the new financial services offered by DeFi, it is crucial to rely on means of exchange with a stable value [85]. Unlike common cryptocurrencies (e.g., Bitcoin and Ethereum), which are extremely volatile, stablecoins maintain almost constant value over time. Stablecoins are usually pegged to the value of an external asset such as gold, but a majority are linked to the US Dollar. Depending on how the system is linked to the stable value, different kinds of stablecoins can be distinguished.

Custodial stable coins are crypto-assets whose stable value is guaranteed by an external authority. The most known stable coins are Tether (USD-T) and USDC, which mainly operate on Ethereum blockchain and are managed respectively by Tether Operations and Centre Organization [86,87]. Companies that manage stablecoins are generally in charge of guaranteeing the asset's value by the deposit (through a bank or another trusted entity) of the equivalent in dollars, gold, or other financial assets [85]. For example, to mint one million dollars of USDC, the same amount in dollars or assets must be locked within the trusted entity. Those stable coins are recognized as crypto assets in the sense that they can actively interact with other cryptocurrencies through smart contracts and exchanges, but like fiat currencies, their use can be censored, seized, and limited by the issuing authority [46]. By definition, custodial stable-coins require trust in an institution that guarantees the pegs to a certain asset. For that reason, blockchain oracles are not required to derive the price of custodial stable-coins. On the other hand, market congestions or downturns may determine temporary deviance from the pegged value.

Non-custodial collateralized stable coins are crypto-assets whose value is not guaranteed by a centralized entity and, most of the time, are managed by a Decentralized Autonomous Organization [17]. DAI is, for example, a non-custodial stable-coin whose value is guaranteed by the deposit of a collateral (mainly Ethereum) whose volatility is exploited to stabilize the value of the asset. For example, after the deposit of $150 of ETH in the appropriate smart contract, it is possible to mint $100 of DAI. If ETH rise in price, then more DAI are minted to stabilize their value. On the other hand, if ETH price decreases, a proportional amount of DAI is burnt [88]. Unlike custodial stable-coins, non-custodial are open and censorship-resistant, but due to the over-collateralization rules (usually 150%), the total issued amount is generally lower [46]. Other examples of non-custodial stable-coins are sUSD and USDJ. Non-custodial stable coins, being untied by external entities who guarantee the asset's value, require oracles to verify the exchange rate between the stable coin and the collateral. Surely, the most interesting case is the MakerDAO Oracle. As extensively explained in Gu et al. [32], the MakerDAO oracle had a major change for which it can be distinguished in V1 and V2. In MakerDAO V1, the collateral for DAI stable-coin was only ETH, so that the oracle had to update ETH/DAI price in real time to enforce the collateralization ratio properly. The Maker V1 Governance whitelisted 14 independent, and anonymous price feeds "to monitor the reference prices across a number of external sources" [89]. When the DAI/ETH price is to be updated, a price oracle calls a function that indicates *value*, *valid_unitl*, and *medianizer.addr*. The *value* is the DAI/ETH claimed exchange

rate, the *valid_until* indicates its expiration time, and *medianizer.addr* is the contract address of the *medianizer*. The *medianizer* then aggregates all the value from the price feeds. Of course, the *medianizer* updates the prices independently on when it receives the prices from the feeds, so it happens that aggregators use price feeds with different expiration times.

The MakerDAO V2, on the other hand, brought many innovations to the Maker Protocol. First, it enabled a multi-collateral feature. Second, it counts on a broader source of price feeds [90]. Unlike V1, in this new version, the identities of price oracles are disclosed (0x, dYdX, and Gnosis), and the contract also introduces a novel *medianizer* mechanism. The new protocol requires the presence of an Oracle Security Module (OSM) for each collateral type. In addition to the V1 functions, a poke function is introduced, which excludes feeds that lack three critical requirements [91]. First, feeds should be provided by a minimum number of sources. Second, the values should be all positive and presented in ascending order. Finally, signatures must be verified and belong to all different whitelisted feeds.

Non-custodial algorithmic stable-coins constitute a complex experiment in which the pegged value is not ensured by collateral but relies only on a system of algorithms and smart contracts [51]. Those projects employ a model in which the token holder receives new coins when demand increases. On the other hand, if the demand decreases, the amount is automatically deducted from the market to limit the loss of value. Although simple in principle, algorithmic stable coins are challenging projects to realize, and some, such as Basis, already shut down due to regulatory hurdles [92]. Ampleforth is a project which is still active and employs the algorithmic principle; however, the stability of its value still represents a challenge [93]. Ampleforth utilizes a system of oracles trusted by the platform that reports price feeds cyclically. The platform administrator sets min, delay, and expire parameters, where min is the minimum number of reports. Delay indicates the time from which the reports can be used and expire the time in which the report becomes unreliable.

CeloUSD is another algorithmic stable coin that implements a quite complex oracle type [94]. Celo protocol has a smart contract called SortedOracles that recognizes only four trusted price sources (Binance, Bittrex, Coinbase, and OKCoin). The Celo Oracle data aggregator, other than deploying the mean value, also checks if a minimum number of exchanges were queried [95]. The reporter then transfers the feed from the aggregator to the SortedOracle contracts, ideally on a stable time basis. For example, if the maximum age of a report is 5 min and there are ten participating oracles, then, every 30 s, an oracle should send a report. Celo also employs a "MetricCollector" that checks on the performance of oracles to detect anomalies in their behavior. The most attractive feature of Celo oracle is the so-called "Circuit Breaker". The circuit breaker basically shut down the oracle service in case of high volatility. Once shut down, the system should be restarted manually and revised by the platform expert before being operative again. During the shutdown, a trusted provider will adjust the price dynamically until the oracle restarts [96].

4.4. Derivatives

As known, derivatives are financial assets that derive their value from represented assets' performance [97]. Derivatives in DeFi are extremely important due to the lack of interoperability between blockchains. As Larsen [98] explains, "Bitcoin can't speak the language of Ethereum and vice versa". This means that we cannot spend bitcoin on the Ethereum network, and we cannot operate Ethereum smart contract on the bitcoin network. Wrapped tokens were specifically made to overcome this limitation, in particular for DeFi applications. WBTC, as an example, is an ERC-20 version of bitcoin and can be spent on the Ethereum network and managed by Ethereum smart contracts. In order to issue WBTC on Ethereum, an equal amount of BTC has to be locked on the Bitcoin blockchain. An oracle service should then ensure that as long as WBTC is used on the Ethereum network, the corresponding amount on the bitcoin network is not spent. Being WBTC traded for other tokens and used as collateral for loans, failure in communicating the exact quantity of locked tokens can be fatal [98]. Due to the complexity of derivative contracts, their management is mainly left to centralized exchanges (Coinbase, Poloniex) [46]. Lately, however, some

platforms such as Synthetix are also offering DeFi solutions in the derivatives field [99]. In those platforms, it is possible to find tokens representing all sorts of fiat currencies (such as GBPN representing Pounds), stocks or other crypto assets (wrapped tokens). The price of external assets offered on Synthetix is determined through a system of oracles provided through Chainlink. However, the requirements of Synthetix platforms are that the oracles should be distinct for each asset and that prices are updated every 5 or 10 min.

Universal Market Access (UMA) is another recently born platform that aims to create fast, efficient, and secure derivatives on the Ethereum platform [32]. Unlike Synthetix, it has its own oracle system made of two distinct modules called Optimistic Oracle and Data Verification Mechanism (DVM), respectively. UMA's oracles are not mandatory to use in principle, but if the contract needs a secure price feed, it can quickly require it with the Optimistic Oracle [100]. This oracle is mainly automated and provides an off-chain price feed within a pre-defined length of time, without the need to pay any on-chain fees. If the price is disputed, then the second UMA's oracle, the DVM, comes into play. The DVM works mostly like the Tellor and Razor oracle systems, and it takes two days to solve a dispute. Those who misreport price lose their bond in favor of those who reported a correct price. It is also possible to require a price directly to the DVM, but it will take two days to resolve anyway. UMA's whitepaper is also interesting because it has an appropriate section in which they explain how they claim to solve the oracle problem. They explain that if the price of the contract is high, then the oracles may be incentivized to provide imprecise price feeds for personal purposes. In particular, they distinguish between Profit-from-Corruption (PfC) and Cost-of-Corruption (CoC). The Cost-of-Corruption is the total amount of UMA tokens needed to perform a 51% attack on the platform. The Profit-from-Corruption is the total asset value of the derivative for which the price is requested. The UMA protocol then requires that the CoC is always greater than 200% of the PfC. In that way, although possible to perform a 51% attack, it will always be unprofitable. However, according to their website, this prevention system is not yet automated [101]. Furthermore, collusion is not the only risk of using oracles in DeFi.

5. Risks and Attack Vectors

The presence of oracles in DeFi naturally determines many vulnerabilities that unfortunately cannot be eliminated. Drawing upon the selected sample, we focus on answering our second research question, distinguishing between social (S) and technical (T) risks/vulnerabilities.

5.1. Front Running (S)

Front-running attacks are due to the natural condition in which the oracle is the first to be aware of critical data uploaded to the blockchain to run a smart contract. Arguably, the early awareness of data gives an important advantage to the oracle data provider [66]. Knowing that an asset price will be sent to a specific platform at a precise time may enable fraudulent operations. Furthermore, block-time, congestions, the exploitation of bots, and automated procedures make front-running attacks more and more feasible [85]. In legacy finance, the front-running attack is quite known, but it can be performed only by a restricted number of people (insiders) aware of pieces of knowledge that are not of the public domain [102]. In decentralized finance, as information is transparent, anyone that realizes the potential of information to be exploited for a front-running attack may be a potential attacker. A known example of a front-running attack is the one exploited on Terra Money. Terra is a stable coin whose potential is to be available on multiple blockchains such as Ethereum and Solana [103].

In 2019, Terra witnessed a front-running attack on its oracle, specifically designed to be secure and reliable. The Terra oracle price feed registers the price of external assets in three phases. The first is the pre-vote phase (N-2), in which the price of an external asset is proposed as a feed. The second is the voting phase (N-1), in which, in case of insufficient votes, the proposal can be rejected. The third is the confirmation phase (N), in which the

proposal is finally registered in the blockchain. Every step of this system lasts for 12 blocks so that from the pre-voting phase to the confirmation phase, there is a 24 blocks distance. This means that at time N, the price accepted is that of time N-2. What helped the hack was also that, by the time of the attack, to incentivize trades on the platform, the swap between Terra coins was offered without fees. It happened then in August 2019 (Figure 7) an attacker spotted a little discrepancy (2%) between the spot price and the oracle price. Exploiting the zero fees policy was then able to trade assets at a discounted rate. From that point, Terra Money was then obliged to introduce two fees: fixed and proportional. A little discrepancy of 1 to 2% of the price was then completely absorbed by the fees, making front-running attacks inconvenient [104].

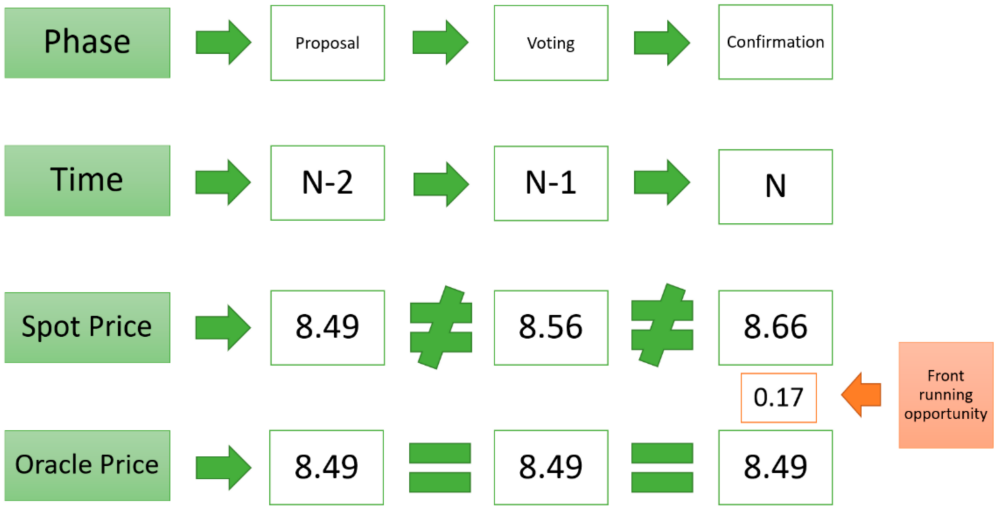

Figure 7. Example of a front-running opportunity.

The front-running attack can also be performed by exploiting the blockchain fee mechanism, which prioritizes transactions with higher fees. Since blockchain is publicly auditable, it is also possible to have a look at pending transactions that have still to be confirmed. If, for example, in the transaction pool, we observe a swap of thousands of DAI for another asset, this will surely impact prices. If we then perform a transaction and pay a higher fee so that our transaction is mined before the swap, then we can benefit from that prioritized action. This is known as "Miners Extractable Value (MEV) Problem", but although it involves oracles, it cannot be considered an oracle manipulation [105].

5.2. Sybil Attacks (S)

Sybil attacks are means to manipulate decentralized oracle platforms. To obtain the majority of votes, the Sybil replicates its vote to gain a higher weight when compared to others [14]. Equally, the Sybil can also manipulate other oracles for them to display the same choice as its own. Sybil schemes are only possible in systems where the oracle identity is anonymous, and their choice is private. In an interesting paper, Douceur [106] explains that the only way to prevent a Sybil attack is to make the voters' identity public and their choices transparent. In decentralized systems, influencing the majority of voters can also be identified as a 51% attack. Unlike those that can be performed on a Proof-of-work protocol, it does not involve computing power but refers to the corruption of enough voters to be able to control the outcome of an oracle feed. Alternatively, it may refer to a condition in which agents with enough voting power collude to change the rules for selfish purposes. On MakerDAO, for example, due to the fact that a restricted number of Maker holders take decisions on oracles, Liu et al. [17], hypothesized that a 51% attack was then possible and expectable. Basically, due to the concentration of Maker owners and their low participation in the voting process, collusion to manipulate governance and price oracles is perceived, extremely likely. The thesis was also supported by other studies [107–109],

which confirms the chances of performing governance attacks on MakerDAO, as well as other DeFi platforms, eventually pushing the entire DeFi ecosystem to a crisis. However, the thesis is confirmed mainly by the fact that, on October 2020, an attacker used a flash loan to borrow enough MKR to pass a governance proposal to change the MKR oracle whitelist parameter and managed to push his feeder to the allowlist [63,110]. Therefore, Sybil attacks cannot be underestimated. Those attacks can also be performed through mirroring and freeloading.

Mirroring: In mirroring attacks, the Sybil honestly works for the platform executing the data collection as requested. However, to guarantee the data collection's major weight, it "mirrors" the data collected into different oracles under its control. That technique ensures the lowest cost of data collection and the highest chance of selling the data to the customer platform. Again, in an anonymity and privacy condition, the chance of spotting a mirroring attack is quite low [6].

Freeloading: Extensively explained in the Augur whitepaper [33], the freeloading attack is a technique thanks to which an oracle replicates the data fetched by another entity, without making the effort of performing the data collection itself. Since the data are obtained effortlessly, the Sybil can provide the service at a lower price than the honest oracles, eventually pushing them out of the market. Once the honest oracles are out of play, the system is solely managed by malevolent or corrupt oracles [20]. In off-chain acquisition, this vulnerability can easily be overcome using a Commit/Reveal design pattern (e.g., Razor Network), which ensures that cleartext data is only revealed when the transaction is ready to be finalized. This pattern also provides a guarantee that the data eventually revealed cannot be changed. If the results are always transparent, then it is harder to counter. Although possible, this type of attack is quite rare, and, as explained in the Augur whitepaper, it never constituted a real threat.

5.3. Selection Bias (S)

Selection bias refers to the situation in which a DeFi service selects a data source that is not explicitly meant to serve as an oracle [25]. To give an example, to measure the temperature, we generically employ a thermometer. However, to measure a patient's temperature during surgeries, for instance, special thermometers are needed to rapidly check for any drastic temperature change and warn the surgeon to intervene. In the DeFi space, the role of price oracle should be entrusted to entities specifically made to serve this purpose. Price oracles should, in fact, report price changes avoiding distortions and deviances. Arguably, the higher the value locked into DeFi smart contracts, the higher the accuracy and security of the selected price oracle should be [111]. To better understand the problem of price oracle source selection, the example of Curve Finance fits perfectly. In 2020, Curve Finance was considered responsible for many flash-loan attacks for an amount that exceeds one hundred million dollars [112]. The attackers exploited the fact that many DeFi platforms arbitrarily decided to adopt Curve Finance LP's price feeds. LP price feeds, of course, are characterized by high volatility to rapidly adapt to the size and usage of the selected pool. Exploiting the LP price volatility, flash-loan attacks were incredibly easy and rewarding. Hence, despite the fact that Curve Finance data were publicly available, their LP price feeds were not intended to provide pricing for collaterals in the first place. The Curve Finance team declared in many articles, in fact, that they were not aware that other DeFi projects utilized their LP price feed [112]. Therefore, they publicly suggested relying on other and specialized oracle services.

5.4. Data Manipulation (T)

Data manipulation prior to the oracle transmitting information constitutes the hardest problem to spot. If oracles work well, there is still the chance that the data they collect have been tampered with at the source [24]. If this happens, the only way to spot that there is a problem is to compare the price feeds with other oracles or price feeds. For that reason, in a centralized oracle service, data manipulation constitutes an important risk to consider.

Centralized-oracle providers (e.g., Oraclize) may provide different levels of guarantees that the data have not been altered "after" the collection, but not that they were genuine at the source [10]. In the so-called "Compound incident" that led to the liquidation of more than eighty-five million dollars, for example, it is still not clear if data were manipulated [113]. To understand what happened, it is important to know that by the time of the incident (November 2020), the price of DAI/USDC was pushed by CoinbasePRO, as an oracle, into Compound. For an unknown reason, the price of DAI was reported by CoinbasePRO at $1.3. Due to Compound's mechanics that "unfortunately" worked perfectly, many of the loans were considered under-collateralized [114]. Therefore, the platform proceeded to undertake a massive liquidation. Per contra, in his article, Chipolina [115] supports the view that there was a malevolent manipulation of data behind the incident. However, there is still no clear evidence of that. According to an official blog announcement, the Compound developers then decided to engage Open Zeppelin, a blockchain-focused consulting firm, to audit the proposed price feed and Certora to implement a specification for formal verification of properties of the contracts as a part of the migration to the Open Price Feed [76,116].

5.5. Malfunctions (T)

Being software (or hardware-based), oracles may suffer from bugs or malfunction. A solution to this problem is to periodically check the status of oracles and undertake tests and maintenance [23]. The Synthetix issue that happened in 2019 [117] constitutes a clear example of malfunction. As previously explained, the Synthetix platform is a specialized DeFi that allows users to be exposed to the price of assets that do not typically belong to the world of cryptocurrencies (e.g., fiat currencies $£¥€). By the time of the incident, the platform relied on an aggregate price derived from a set of off-chain oracles that was updated on-chain at fixed intervals. The users of the platform could then speculate over the supported assets by using long and short positions. The system worked very similar to common trading platforms, such as eToro or Plus500, but exploited decentralization provided by blockchain [8]. On 25 June 2019, however, one of the systems of off-chain oracles on which Synthetix relied for the price of Korean Won (sKRV), misreported the price to be 1000 times higher than the actual rate. To prevent misrepresentation of price feeds given by outliers, the Synthetix platform performed a mean value between the price feeds; however, for sKRV, they could only rely on two oracles. Supposing that at least one of the oracles was performing correctly, the mean between two values was, in any case, not sufficient to outweigh the error. The misreported price was then accepted by the system and exploited by a bot specifically programmed for spotting price anomalies. The bot performed multiple transactions for a total turnover of over one billion dollars. Luckily, the platform was able to identify the proprietary of the bot, which was not a malevolent operator, and used the bot only for ordinary trading procedures. He agreed then to return the money in exchange for a so-called "bug bounty" (reward for spotting bugs before they harm other users or the platform itself). In the end, it is arguable that two main factors created the problem. First, the scarcity of price feeds, and second, a malfunction or tampering of one of the oracles.

Another interesting example of oracle malfunction is the MakerDAO Black Swan Incident [118]. The outcome was very similar to the previous Compound example, which witnessed an important amount of collateral liquidated due to unexpected price change. However, in the MakerDAO incident, the problem originated with an unforeseen event (COVID-19 Crisis) that made cryptocurrency prices fall unexpectedly. The price of ETH fell from $200 to roughly $80 in a matter of hours. This generated two unparalleled consequences. First, the vaults were under-collateralized, and that obliged users to return some DAI or to fund the vault with more ETH to balance with the loss value. Second, the market congestion made transaction fees explode, also slowing the time of settlements. So, most users were unable to access their vaults and save their collaterals. Meanwhile, oracles were unable to update prices timely, and this made it possible for some bots and

users to liquidate vaults nearly for free. The total amount of assets siphoned from the Maker vault exceeded eight million dollars. Brilliantly explained by Eskandari et al. [63], the issue, in this case, was that the oracle contract had a predetermined (and fixed) amount of fees to spend for price updates. Due to the congestion then (and the higher price of fees), every price-updating transaction failed. When the price of fees matched again those predetermined in the oracle contracts, the reported price was then too different from the previous one, creating a "jump" in the ETH price. This caused the liquidation of collaterals in the MakerDAO vault. From an oracle perspective, this event is interesting because even if the oracle contract correctly reported the price, the time of update created an unprecedented discrepancy in the ETH price, making the timeliness of transactions another critical characteristic to take into account [119,120].

5.6. Flash-Loan Attacks

As previously explained, flash loans are crypto loans repaid within the same block. This condition determines that there is virtually no limit to the amount of money borrowed [121]. As the transaction can influence the prices instantaneously, huge transactions can exponentially affect prices. If the oracles are not prepared to face the event, this condition can be exploited to manipulate the price of assets and collaterals [71]. Two articles from Peaster [7] and Tarasov [122], display a list of the most famous attacks in the DeFi space. In accordance with Thompson [4], they show that the most common and rewarding DeFi hack is the flash-loan attack. Due to that reason, Qureshi [71], commenting on the flash loan, declares that to date, the major outcome of flash loans is the enabling of flash-loan attacks. The bZx incident constitutes a typical example of the oracle problem in DeFi, and it is quite famous for being implemented multiple times on the same platform with success and even higher turnover. Basically, bZx smart contracts utilized the Kyber Decentralized Exchange price feed as an oracle. This dependency was exploited during the attack. The attack that happened on 18 February 2020 is, therefore, described and displayed in Figure 8.

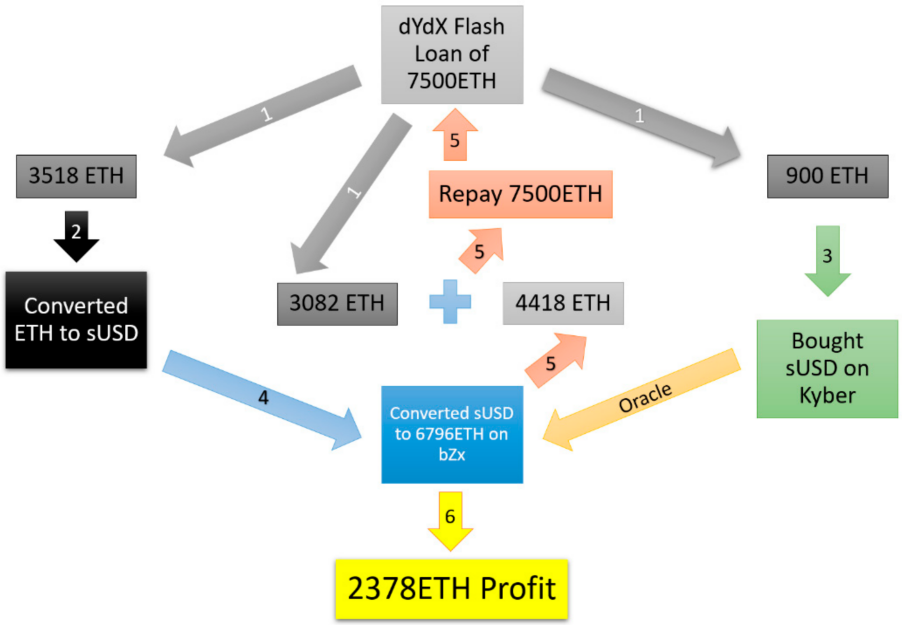

Figure 8. Flowchart of 18 February 2020 bZx flash-loan attack scheme.

An attacker obtained a flash loan on the bZx of 7500 ETH, which was split into three parts (3518, 900, and 3082). The first part of 3518ETH, was used to buy sUSD (on Synthetix), a synthetic USD token enabled by the Syntetix protocol and which should always keep the price of $1. The sUSD were actually bought at the price of $1. However, a second part of the loaned ETH (900 ETH) was used to buy sUSD on Kyber, artificially inflating the price over $2. The bZx used Kyber as a price oracle, so on their platform, the sUSD was then

priced at $2. The overpriced sUSD were then used as collateral on bZx to borrow ETH. Due to the inflated price of sUSD, the hacker was able to borrow 6796 ETH instead of 3518ETH. The attacker then had a total pot of 6796 ETH plus 3082 ETH from the original loan for a total of 9878 ETH. In order for all these transactions to be validated, the initial loan had to be paid back so that the attacker was obliged to send 7500 ETH back to the smart contract. However, he could run away with the surplus of 2378 ETH, equal to $636,000 by that time [111]. The flash-loan attack is an important issue with oracle in DeFi, since the very presence of oracles causes it. As described, no collusion, bug, or data tampering was exploited, but only the oracles and their refresh rate. All the flash loans are performed more or less the same way as the one described here [123]. They are indeed used a lot because of their simplicity and, most of all, because they come basically at no cost. If they fail, only the transaction fees have to be paid, and if they succeed, then the turnover is priceless.

6. Discussion

This paper provided an overview of the role of oracles in Decentralized Finance and the issues related to their usage. Depending on the applications, we present the main types of oracles proposed. While some applications, such as MakerDAO and Compound, have built-in oracles systems, others, such as Aave and Syntetix, rely on third-party DeFi platforms, e.g., Uniswap, or specialized oracle providers, e.g., Chainlink [68,69,90,99]. All the solutions displayed advantages and drawbacks due to intrinsic oracle characteristics and trust model design [19]. The first thing to note is that there is always a deviation between the oracle's price and the spot prices of assets [17]. While a deviation of 1 to 2% is acceptable and balanced with operating fees, as in the case of Terra Money, the extreme deviation can lead to severe consequences, as in the case of Synthetix [104,124]. Deviation in oracle reporting has been monitored in recent research by Liu et al. [17]. While analyzing deviation, they discovered that, while Synthetix relied on external oracle providers, it had a mean deviation of $\pm$2%. Other platforms, such as Maker, AmpleForth, and Compound, had a deviation from $\pm$5% to peaks of over $\pm$200%. It is important to note that the analysis excludes episodes of oracles malfunction or manipulation and displays the deviation of oracles on a regular basis. Considering that oracles' normal behavior includes a deviation of more than 5%, it is important to design DeFi platforms and products to counterbalance this situation efficiently. Many have suggested, due to that reason, that the feeds are to be selected from multiple data sources and oracles, as well as that accepted price should be mean values of those extracted in the previous feeds [111].

What emerges also is that there is an evident lack of standardization. Heterogeneous systems are to be welcome, in the sense that they can meet the different needs of customers. However, the oracle choice and design appear to be arbitrary. From this and related studies, it emerges that there are no rules or frameworks, thanks to which a DeFi project selects an oracle type in place of another [8,32]. Without a set of accepted rules, the choice over oracle may be simply derived by cost or availability. Moreover, even if the oracle mechanic is made public, there is no way to objectively determine the quality of service as it is instead possible with smart contract auditing [125]. The case of MakerDAO, for example, shows the problem of discrepancies between what is perceived as a working oracle and a faulty one [109]. The fact that the oracles were providing prices with a delayed time with respect to the spot price was perceived as a normal feature by the company, while it was described as one of the problems by articles describing the incident [118–120,126–128]. If there is no agreement on the fact that an oracle should provide prices with a delay and how much deviance should have from the spot price, it is indeed impossible to distinguish between reliable and unreliable platforms.

Another important thing to consider is that there is not a clear framework that determines which source can be used as a blockchain oracle or not. Therefore, it is impossible for a service provider to foresee if its data can be used (or misused) on a decentralized platform. This is the case of Curve Finance that was selected (without its knowledge) as a price oracle for many DeFi projects [112]. Being unaware of the third-party data exploitation, it did not

pay attention to the outcome that the price feed could have determined. This, unfortunately, led to many flash-loan attacks and losses determined by Curve price feeds, but of which Curve Finance was probably not accountable. The Curve team, of course, replied to the incident, asking to rely on specific oracle services—Chainlink, in that case.

A recent study that wanted to show the crescent usage of oracles on Ethereum blockchain also utilizes Chainlink data as a benchmark [53]. Of course, Chainlink is one of the first oracle providers; however, focusing and querying only a precise oracle service can also lead to a problem of centralization. From the Chainlink whitepaper, we also find out that they are willing to make a sort of certification of oracles that distinguish between reliable and unreliable sources [6,129]. If, on the one hand, this can help platforms in selecting reliable oracles, this can also create an oligarchy circle in which new providers are prevented from playing. Another aspect to consider is the transparency of oracle choice. Unless the client does appropriate research, it is not always transparent the oracle service that is used in the DeFi ecosystem. A recent AMM project, named "SushiSwap", added in its trading interface information about the oracle that communicates the price for every trading pairs. However, when Chainlink is declared as an oracle, it does still leave with some doubts. If, on the one hand, it means that the platform relies on a third-party oracle, it does not explain the type of contract that has been made with that provider. As an oracle marketplace, Chainlink, depending on the agreed price, may query more or less trusted oracle or update the price more or less frequently [6].

In the DeFi sector, the oracle problem can also lead to the condition in which the unreliability of the oracle and the data source could determine an unfair risk allocation. That oracles can be imprecise, slow, and may undergo temporary malfunction should be a pretty known condition by the company as well as traders [51]. However, if insiders can influence the work of oracles or be aware of their data before others, they may obtain an unjust revenue [18,111]. As in the case of Terra Money, the early oracle data were exploited to perform a front-running attack [104]. The bZx case and many of the flash-loan attacks instead were sometimes not due to a malfunction but to knowledge about the price oracle's poor design [122]. This creates then a paradox of transparency. It is true that oracles have to be transparent to be trusted, but if their functioning is completely exposed, it can be easily exploited by malevolent actors.

The governance of oracles also raises concerns. In fact, although the chance of a Sybil attack on MakerDAO was widely explained in academic papers and blog articles [107,109], this vulnerability was exploited anyway, posing severe doubts on the security and feasibility of decentralized governance [63,110]. Razor and UMA proposed and employed an economic incentive to counterbalance the possibility of 51% attacks. As written in the UMA whitepaper, "any on-chain oracle can be corrupted—for a price. Because there is no "rule of law" on blockchains outside of economic incentives" [101]. Therefore, it is arguable that it is possible to prevent oracle corruption with the right economic incentive [34].

From a technical point of view, it is evident that a mechanism to identify faulty oracles quickly and the availability of multiple data sources would increase the reliability of DeFi platforms. Moreover, the DeFi platforms should provide a detailed explanation of the oracle service selection process and trusted data sources. Again, widely accepted selection criteria may help to set the timeliness of oracles in a way that is optimal to prevent attacks, such as front running (too delayed) and flash loans (too precise). Table 5 reorganizes and summarizes the main issues of using oracles in DeFi.

Table 5. Challenges and mitigation factors of oracle use in DeFi.

Dimension	Name	Description	Mitigation	Challenges	Source
Technical	Malfunction	An unpredictable condition in which oracles provide biased data, although the source is reliable and trustworthy	Enable a consensus mechanism to include more oracles to spot faulty ones.	Bear the costs of multiple oracles and maintenance services.	Curran [23], Sun [8], Campbell [119], Harvey et al. [51]
	Biased data	Despite the genuineness of the oracle design and its reliability, data are biased at their origin.	Query different data sources and monitor their reliability in time.	Find multiple but equally trusted data sources at an affordable price that remains provably honest.	Mlinaric [113], Jared [116], Omelchenko [76]
	Timeliness	The oracle provides trustworthy data but at an unwanted time.	Adapt the timeliness of the oracle to the specific application. Apply a dynamic fees model.	Lack of standards to understand the exact delay required for an oracle to be perceived as a good oracle.	Liu et al. [17], Kain [124], Lu [111], Eskandari et al. [63]
Social	Sybil Schemes	Act of one or multiple entities to modify the governance of oracles for selfish purposes.	Decentralize the governance by ensuring a fairer distribution of voting power. Provide economic incentives.	Irrational behavior. Deliberate destruction of the platform.	Gudgeon et al. [107], Kelso [109], Zoltu [108]
	Front Running	Exploit of the transparency of data fetched by oracles for selfish purposes.	Apply a Commit/Reveal scheme so that data are disclosed at the last time. Apply fees to counterbalance small deviations.	Reduce the transparency of oracles. Increase in the cost of service for customers.	Lu [111], Morselli [18]
	Selection Bias	Selection of an oracle whose scope is different from the one required by the application.	Select data feeds specifically created for the attended purposes.	Risk of centralization of power by the early players in the oracle industry.	Stevens [112], Kaleem and Shi [53], Gu et al. [32]

7. Conclusions

Oracles constitute the interface between the blockchain and the real world [24]. However, as centralized entities, they reintroduce the concept of trust and single point of failure in a decentralized environment [42]. Decentralized Finance as a real-world blockchain requires oracles to fetch data about prices and exchange rates of crypto assets [49]. A biased or imprecise communication channel can result in an unwanted operation and the consequent loss of millions of dollars [114,115]. The large amount of hacks that were performed, exploiting oracle vulnerabilities solely in 2020, are proof of this weakness [7,113,120,122]. Utilizing an MLR, this study investigated the use of oracles in DeFi, summarizing and codifying the main issues arising from their implementation. Since the oracle problem includes technical and social aspects, this works also distinguishes these two areas, providing examples of failures and solutions. From the analysis of data provided, it emerges that standardization of oracle design and patterns is required to face the most recurrent hacks,

such as front running and flash loans [8,104]. On the other hand, social issues, such as Sybil schemes, due to their consequences for the performers themselves, can be considered less worrying than technical issues, such as malfunction or data tampering. For that reason, addressing the oracle problem in DeFi may be easier by prioritizing standardization to address the technical issues and providing economic incentives to limit the social ones.

Interestingly, there are newborn blockchain protocols that propose different blockchain interactions with the external world. Mina, as an example, presents "snarks" to privately verify external data on-chain, without the need for external oracles [130,131]. Another recent proposal enables blockchains to make external calls to the real world directly [22]. If blockchain systems will someday be able to interact with the real world without the need for oracles, this will surely determine a great change for Decentralized Finance (and other DLT applications).

This study contributes to the literature by providing the first structured review on oracles in DeFi. Furthermore, it focuses on the oracle problem and outlines a framework for more robust oracles to be developed. The results presented herein can be a starting point for academics to build other structured reviews, or they can be used as the background for empirical works. Practitioners may also draw on the conclusions to enhance their oracle's security or develop new projects. Furthermore, in this paper, insight was provided on the following:

- What is the oracle problem, and how does it impact DeFi applications?
- How do price oracles work?
- How do different DeFi applications rely on oracles?
- What are the risks associated with oracles in DeFi?
- What was the impact of past oracle failures in DeFi?
- What are possible improvements to oracle weaknesses?

This study has, of course, limitations that are derived first from the scarcity of data to draw upon and then by the lack of previous systematic reviews in the field. Furthermore, the present study is biased by the authors' personal perspective and background, which may have influenced the selection and excluding criteria. Further studies may build on this, as well as on Liu et al.'s paper [17], and check whether different oracle designs affect the deviation index of data feeds. Furthermore, a comparison between UMA and Razor Network economic incentives could shed light on their effectiveness to prevent 51% of attacks. Other reviews may also focus on a specific DeFi application (lending, flash loans, derivatives, etc.) and further discuss the role and risks of the implemented oracles.

Author Contributions: Conceptualization, G.C.; methodology, G.C.; validation, J.E., data curation, G.C.; writing—original draft preparation, G.C.; writing—review and editing, G.C. and J.E.; supervision, J.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by UniCredit Foundation "Fondo Emma Gianesini".

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Appendix A.1. Tellor Oracle

Tellor is an oracle system specifically made to fetch data about exchange rates. On the Tellor platform, miners compete to add data to an on-chain data bank. Users that need updated data about exchange rates make queries on the data bank, and for each query, they leave a tip in "Tribute" (TRB), the Tellor native token. Every 5 min, the Tellor smart contract groups the five queries with the higher tips and creates a PoW for miners to solve. The first five miners to provide the off-chain data and the solution to the PoW are rewarded with newly minted TRB and tips. The oracle resolves the value once five reports for the same feeds arrive. To ensure fair competition, in order to participate in the PoW,

the miners must freeze in the Tellor smart contract a deposit (500 units of TRB, around $35,000) which is lost in case of a successful dispute by TRB holders. Data-submission cycles last 5 min and are called blocks. Since only one feed is provided by the oracle every 5 min, the Tellor oracle does only provides 288 price updates per day. After prices are settled, a dispute round begins, which lasts for two days. During that time, other TRB holders can submit a different value for an already confirmed price feed depositing double of the stake (1000 TRB). When a dispute is initiated, the dispute round is prolonged for an additional day until no more disputes for the same feeds are received. To ensure also the fairness of wealth distribution, on the Tellor ecosystem, each miner cannot win two consecutive rounds (blocks), and the minimum waiting time is 15 min (three blocks). Security mechanisms on the Tellor ecosystem are mainly toward the exclusion of faulty oracles in favor of trusted ones. The PoW also proves to be efficient against Sybil attacks that would be nearly infeasible due to the computing power required. The Tellor dispute mechanism is summarized in Figure A1 [132].

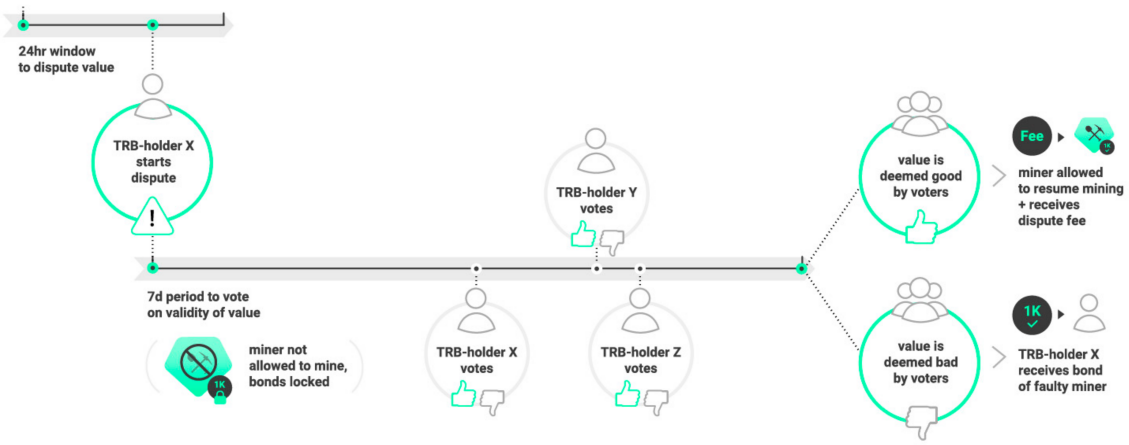

Figure A1. Tellor dispute mechanism. Source: https://docs.tellor.io/tellor/whitepaper/tellor-oracle-overview/disputes (accessed on 12 March 2021).

Appendix A.2. Razor Network

Razor Network is an oracle service inspired by oracle protocols such as Augur and Kleros. It is specifically made to be resistant to social attacks, such as collusion, takeover, griefing, bribing, etc. The Razor oracle is made of stakers who process queries and provide the results to the client. Stakers are those who deposit Razor Token in the platform as a stake. The job manager then orders the queries in relation to the paid fees and delays (not discard) those with lower fees.

Any application can be a client for the Razor Network, as it is a permissionless platform; however, only one query and feed can be executed at a time. Since working in the background, the final user of the DeFi platform is not aware when the Razor oracle is being used. Depending on the request made by the client, the oracle may perform an automated round or a manual round. The automated round resolves relatively quickly and, the staker only submits an URL that provides the required data. Since the URL may be unreachable or fails to load, the required stake, called "validity bond", is also lower than the amount required for the manual round.

On the other hand, the manual round requires more fees, and the stakers manually provide the feed. The manual round may take few days to resolve. In both cases, the result can be disputed, but again, the dispute is quicker for the automated round. The Razor Network employs a Commit/Reveal scheme in which stakers encrypt their reply until it is received by the network. That way, other stakers or oracles cannot copy the answers. Unlike Tellor, there is no minimum stake to participate in the network, and the contract randomly chose validators among the stakers. However, chances, as well as rewards, are proportional to the staked amount. If a result is successfully disputed, 100% of the stake is

slashed, and of this amount, 50% is burned, while the other half is distributed among the disputers. Proposing a faulty data feed led to the severest punishment, but other behavior, such as failure to commit a result or delay in providing a feed, also results in a slash of 5% and 1%, respectively. Every oracle round takes the name of "Epoch". Figure A2 describes the steps followed in the Razor Network epoch. Finally, it is interesting how the Razor Network exploits economic incentives to prevent social attacks. If the value of the data feeds is worth more than any incentive mechanism offered by the platform, the oracle may deliberately provide false data for its self-interest [38]. For that reason, the Razor Network only works for applications whose total market cap equals half of the total staked amount [34].

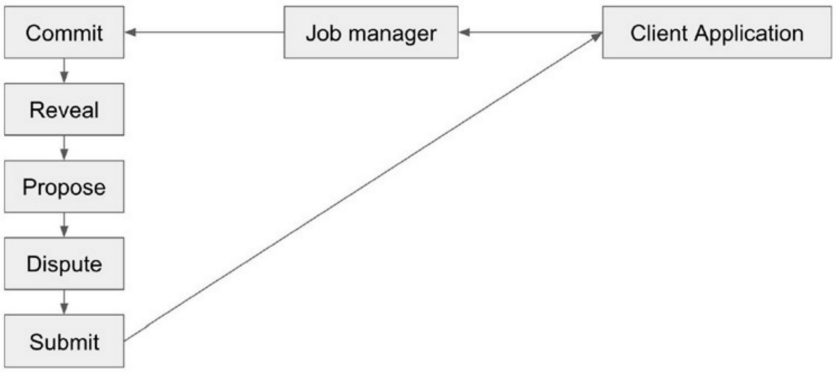

Figure A2. Razor epoch description. Source: Razor whitepaper.

Appendix A.3. Bluzelle DeFi Oracle

Bluzelle is a project born in 2017 described in its whitepaper as the "Decentralized Database for the future". As known, the blockchain stores hashed data of transactions, and if documents are signed or registered with blockchain, "only" the hash is uploaded on the chain [133]. This is to limit the size of the blockchain, which is replicated in every node. An excessive increase of the blockchain may, in fact, determine a problem of centralization [41]. For that reason, blockchain applications involving data such as photos, documents, videos, and audio rely on external databases to balance immutability and blockchain size growth. This is where Bluzelle comes into play. Bluzelle offers a system of decentralized databases to supply data for blockchain applications. They claim to be the alternative to Oracle (the database management company) in the decentralized world [134]. As the Bluzelle database system provides any sort of data for blockchain applications, it is lately being used in Decentralized Finance to fetch the exchange rate of assets. However, as the company realized that they were not prepared to be price oracles, they decided to build an appropriate section to serve that purpose [135]. Considering the issues that may arise with unreliable price feeds, the Bluzelle oracles organize the work in five phases, which are described in Figure A3.

First, the price feeders are chosen among the network contributors (validators). Those entities are the ones that contribute to the Bluzelle Network, providing storage and staking BLZ (the network native token). Thus, the position of price feeders is open only to those who actively contribute to the network. Unlike other platforms, Bluzelle does not impose a trusted data-source list and leaves the feeders the choice of the reliable source to draw upon. The voting power of the feeders is based on their storage supply and staking amount. The second phase consists of eliminating feeds of validators whose value exceeds the platform's maximum deviation index. The stake of outliers is then slashed. The third phase consists of a recalculation of the voting power of validators due to the exit of outliers in phase two. In the fourth phase, the feeds that stay inside the tolerance limits are blended to create a median value that is then passed to the fifth phase for the final evaluation. The last phase of the Bluzelle price oracle consists of putting the blended value under a statistical analysis that can discover price anomalies by confronting the value with past

feed processes. If the price is confirmed, the system transfers the value to the Bluzelle database and rewards the validators that provided the correct price [136]. Compared to other price oracles, Bluzelle has the characteristics of being updated every sixty seconds, which makes its resource adaptable to many DeFi projects. Furthermore, data stay on the DB, and they are transmitted on-chain only if requested by clients. At the time of writing the paper, Bluzelle offers its service for free and also free access to price history on their database [137].

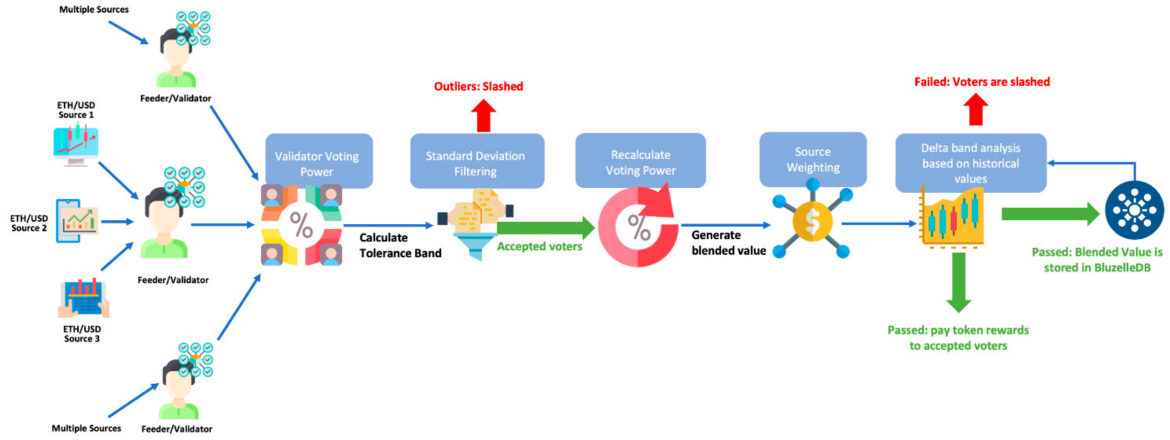

Figure A3. Phases of Bluzelle DeFi oracle. Source: https://oracles.bluzelle.com/ (accessed on 18 March 2021).

Appendix B

Table A1. Article classification sample.

Title	Author	Year	Source	DR	DI	DA	DC	DO
"SoK: Decentralized Finance (DeFi)"	Werner et al. [49]	2021	AP	Y	N	Y	N	N
"DeFi and the Future of Finance"	Campbell et al. [51]	2021	AP	N	N	Y	Y	N
"A First Look into DeFi Oracles"	Liu et al. [17]	2020	GL	Y	Y	Y	Y	Y
"So you want to use a price oracle"	SamCzun [8]	2020	GL	Y	Y	Y	N	Y
"Biggest DeFi Hacks in 2020"	Peasteron W.M. [7]	2021	GL	Y	Y	N	N	N
"Automated Market Makers for Decentralized Finance (DeFi)"	Wang Y. [77]	2020	GL	Y	N	Y	N	N
"Decentralized Finance: On Blockchain- and Smart-Contract-based Financial Markets"	Schär F. [50]	2021	AP	N	N	Y	Y	N
"Decentralising Finance using Decentralised Blockchain Oracles"	Manoj Kumar et al. [20]	2020	AP	Y	N	N	N	N
"Improved Price Oracles: Constant Function Market Makers"	Angeris G. and Chitra T [52]	2020	AP	Y	N	Y	N	N
"Stablecoins 2.0: Economic Foundations and Risk-based Models"	Klages-Mundt et al. [85]	2020	AP	Y	N	Y	Y	Y
"Flash Loans: Why Flash Attacks will be the New Normal"	Qureshi H. [71]	2020	GL	Y	Y	N	N	N
"When is Uniswap a good oracle?"	Angeris G. [79]	2020	GL	Y	N	Y	Y	Y
"Analysis of 8/5, 8/12 Front-Running Attack"	EJ [104]	2019	GL	Y	Y	N	Y	Y
"Bluzelle Reveals Decentralized Oracle to Enhance DeFi Project Security and Price Reliability"	Edelstein D. [135]	2020	GL	Y	N	N	N	Y
"Razor Network: A decentralized oracle platform"	Huilgokar H. [34]	2021	GL	Y	N	Y	N	Y

* DR = discussed (oracle) risks, DI = discussed (oracle) incidents, DA = discussed (DeFi) applications, DC = discussed (DeFi) companies, DO = discussed oracle (company/service), Y = yes, N = no.

Appendix C. Retrieved DeFi/Oracle Projects

As specified in Section 3, Figures A4 and A5 display the list of DeFi and oracle projects encountered in the sample organized by the number of occurrences in crescent order.

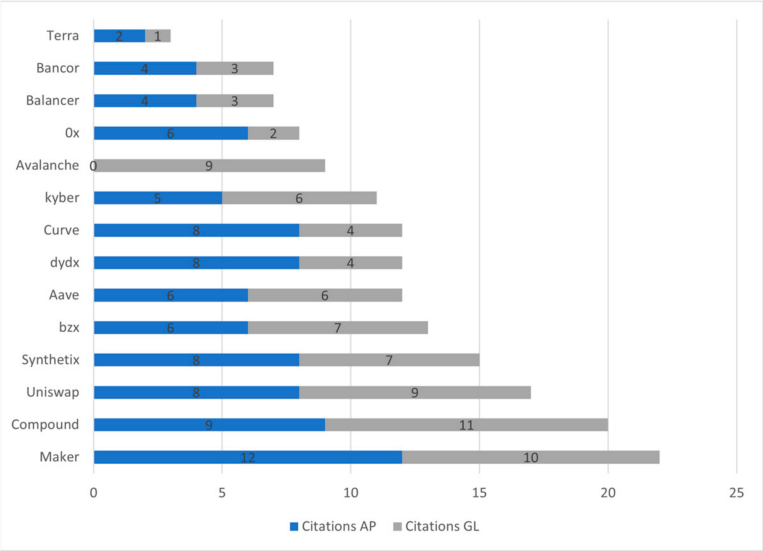

Figure A4. DeFi projects/citations.

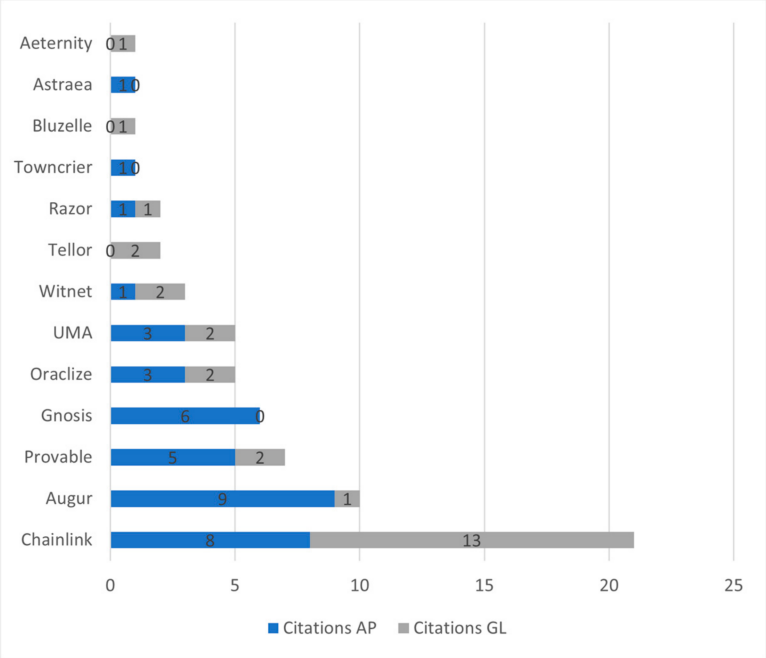

Figure A5. Oracle provider/citations.

References

1. Huilgolkar, H. Designing the Most Secure Oracle for the Decentralized Finance. Available online: https://medium.com/coinmonks/designing-the-most-secure-oracle-for-the-decentralized-finance-9853237f0c37 (accessed on 11 January 2021).
2. DefiPulse DexGuru-Real-Time Data. Analytics and Trading for AMMs in One Place. Available online: https://defipulse.com/blog/dexguru/ (accessed on 19 April 2021).
3. Leybold, M. Decentralized Finance (DeFi) in 2020 and Its Future Trajectory. Available online: https://www.linkedin.com/pulse/decentralized-finance-defi-2020-its-future-trajectory-matthew-leybold/ (accessed on 12 March 2021).

4. Thompson, P. The DeFi Hacks of 2020. Available online: https://coingeek.com/the-defi-hacks-of-2020/ (accessed on 20 March 2021).
5. Thomson, C. The DAO of ETHEREUM: Analyzing the DAO hack, the Blockchain, Smart contracts, and the Law. Available online: https://medium.com/blockchain-review/the-dao-of-ethereum-e228b93afc79 (accessed on 3 April 2020).
6. Ellis, S.; Juels, A.; Nazarov, S. ChainLink: A Decentralized Oracle Network. *Retrieved March* **2017**, *11*, 2018.
7. Peaster, W.M. Biggest DeFi Hacks in 2020. Available online: https://defiprime.com/hacks2020 (accessed on 12 March 2021).
8. Sun, S. So You Want to Use a Price Oracle. Available online: https://samczsun.com/so-you-want-to-use-a-price-oracle/ (accessed on 24 March 2021).
9. Staff, F.M. DeFi Startup Acala to Restructure Oracle Network—For the Better. Available online: https://www.financemagnates.com/thought-leadership/defi-startup-acala-to-restructure-oracle-network-for-the-better/ (accessed on 3 February 2021).
10. Sharma, T.K. Centralized Oracles vs. Decentralized Oracles. Available online: https://www.blockchain-council.org/blockchain/centralized-oracles-vs-decentralized-oracles/ (accessed on 11 February 2021).
11. Zheng, S. Compound Launches an Open Oracle System for Decentralized Pricing Data. Available online: https://www.theblockcrypto.com/post/36460/compound-launches-an-open-oracle-system-for-decentralized-pricing-data (accessed on 19 April 2020).
12. Anadiotis, G. Off-Chain Reporting: Toward a New General Purpose Secure Compute Framework by Chainlink. Available online: https://www.zdnet.com/article/off-chain-reporting-towards-a-new-general-purpose-secure-compute-framework-by-chainlink/ (accessed on 19 March 2021).
13. Egberts, A. The Oracle Problem—An Analysis of how Blockchain Oracles Undermine the Advantages of Decentralized Ledger Systems. *SSRN Electron. J.* **2017**. [CrossRef]
14. Caldarelli, G. *Blockchain Oracles and the Oracle Problem*, 1st ed.; Amazon Publishing: Seattle, WA, USA, 2021; ISBN 979-1220083386.
15. Caldarelli, G. Real-World blockchain applications under the lens of the oracle problem. A systematic literature review. In Proceedings of the 2020 IEEE International Conference on Technology Management, Marrakech, Morocco, 25–27 November 2020.
16. Caldarelli, G. Understanding the Blockchain Oracle Problem: A Call for Action. *Information* **2020**, *11*, 509. [CrossRef]
17. Liu, B.; Szalachowski, P.; Zhou, J. A First Look into DeFi Oracles. *arXiv* **2020**, arXiv:2005.04377.
18. Morselli, G. Oracles as Decentralized Data Sources for the Blockchain. Available online: https://en.cryptonomist.ch/2020/01/12/oracles-decentralized-data-blockchain/ (accessed on 6 January 2021).
19. Al-Breiki, H.; Rehman, M.H.U.; Salah, K.; Svetinovic, D. Trustworthy Blockchain Oracles: Review, Comparison and Open Research Challenges. *IEEE Access* **2020**, *8*, 85675–85685. [CrossRef]
20. Kumar, M.; Nikhil, N.; Singh, R. Decentralising finance using decentralised blockchain oracles. In Proceedings of the 2020 International Conference for Emerging Technology, Belgaum, India, 21–23 December 2020; pp. 1–4.
21. Amler, H.; Eckey, L.; Faust, S.; Kaiser, M.; Sandner, P.; Schlosser, B. DeFi-ning DeFi: Challenges & Pathway. *arXiv* **2021**, arXiv:2101.05589.
22. Ellul, J.; Pace, G.J. Towards External Calls for Blockchain and Distributed Ledger Technology towards External Calls for Blockchain and Distributed Ledger Technology. *arXiv* **2021**, arXiv:2105.10399.
23. Curran, B. What are Oracles? *Smart Contracts, Chainlink & "The Oracle Problem"*. Available online: https://blockonomi.com/oracles-guide (accessed on 12 April 2019).
24. Damjan, M. The interface between blockchain and the real world. *Ragion Prat.* **2018**, *2018*, 379–406. [CrossRef]
25. Beniiche, A. A study of blockchain oracles. *arXiv* **2020**, arXiv:2004.07140.
26. Park, J.; Kim, H.; Kim, G.; Ryou, J. Smart contract data feed framework for privacy-preserving oracle system on blockchain. *Computers* **2021**, *10*, 7. [CrossRef]
27. Caldarelli, G.; Rossignoli, C.; Zardini, A. Overcoming the blockchain oracle problem in the traceability of non-fungible products. *Sustainability* **2020**, *12*, 2391. [CrossRef]
28. Wang, S.; Lu, H.; Sun, X.; Yuan, Y.; Wang, F.Y. A Novel Blockchain Oracle Implementation Scheme Based on Application Specific Knowledge Engines. In Proceedings of the 2019 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), Zhengzhou, China, 11–13 October 2019, IEEE: Piscataway, NJ, USA, 2019; pp. 258–262. [CrossRef]
29. Dale, O. What Is Chainlink? Guide to the Decentralized Oracle Network. Available online: https://blockonomi.com/chainlink-guide/ (accessed on 12 March 2020).
30. Nattapatsri, P. Guide to Building DeFi Using Band Protocol Oracle and Cosmos IBC. Available online: https://blog.cosmos.network/guide-to-building-defi-using-band-protocol-oracle-and-cosmos-ibc-fa5348832f84 (accessed on 19 January 2021).
31. Delphi Delphi Systems Whitepaper. Available online: https://delphi.systems/whitepaper.pdf (accessed on 12 July 2020).
32. Gu, W.C.; Raghuvanshi, A.; Boneh, D. Empirical measurements on pricing oracles and decentralized governance for stablecoins. *SSRN Electron. J.* **2020**. [CrossRef]
33. Peterson, J.; Krug, J.; Zoltu, M.; Williams, A.K.; Alexander, S. Augur: A decentralized oracle and prediction market platform. *arXiv* **2015**, arXiv:1501.01042.
34. Huilgolkar, H. Razor Network: A Decentralized Oracle Platform. 2021. Available online: https://razor.network/whitepaper.pdf (accessed on 18 February 2021).
35. Murimi, R.M.; Wang, G.G. On elastic incentives for blockchain oracles. *J. Database Manag.* **2021**, *32*, 1–26. [CrossRef]

36. Yutaka, K.; Fujihara, A. Ken-System: Experimental performance evaluation on avoidance of blockchain oracle problem using collective intelligence. In *Proceedings of the IEICE Technical Report*; No. 414; IN2020-74; The Institute of Electronics, Information and Communication Engineers: Tokyo, Japan, 2021; Volume 120, pp. 120–125.
37. Wöhler, S. Blockchain Oracles I of II. Available online: https://www.anyblockanalytics.com/blog/blockchain-oracles-part-1/ (accessed on 5 March 2021).
38. Frankenreiter, J. The Limits of Smart Contracts. *J. Inst. Theor. Econ. JITE* **2019**, *175*, 149–162. [CrossRef]
39. Low, K.F.K.; Mik, E. Pause the blockchain legal revolution. *Int. Comp. Law Q.* **2020**, *69*, 135–175. [CrossRef]
40. Song, J. The Truth about Smart Contracts. Available online: https://medium.com/@jimmysong/the-truth-about-smart-contracts-ae825271811f (accessed on 2 March 2020).
41. Sztorc, P. The Oracle Problem. Available online: https://www.infoq.com/presentations/blockchain-oracle-problems (accessed on 3 March 2020).
42. Antonopoulos, A.M.; Woods, G. *Mastering Ethereum—Building Smart Contracts and DAPPS*; O'Reilly Media, Inc.: Newton, MA, USA, 2018.
43. Kumar, A.; Liu, R.; Shan, Z. Is Blockchain a Silver Bullet for Supply Chain Management? Technical Challenges and Research Opportunities. *Decis. Sci.* **2020**, *51*, 8–37. [CrossRef]
44. Caldarelli, G.; Ellul, J. Trusted academic transcripts on the blockchain: A systematic literature review. *Appl. Sci.* **2021**, *11*, 1842. [CrossRef]
45. Finck, M.; Moscon, V. Copyright Law on Blockchains: Between New Forms of Rights Administration and Digital Rights Management 2.0. *IIC Int. Rev. Intellect. Prop. Compet. Law* **2019**, *50*, 77–108. [CrossRef]
46. Zetzsche, D.A.; Arner, D.W.; Buckley, R.P. Decentralized Finance (DeFi). *IIEL Issue Brief* **2020**. [CrossRef]
47. Arndt, T. Towards an Implementation of Blockchain-Based Transcripts with Nosql Databases. In Proceedings of the 17th International Conference on E-Society 2019, Utrecht, The Netherlands, 11–13 April 2019; IADIS Press: Lisbon, Portugal, 2019; pp. 309–312.
48. Mühlberger, R.; Bachhofner, S.; Castelló Ferrer, E.; Di Ciccio, C.; Weber, I.; Wöhrer, M.; Zdun, U. Foundational Oracle Patterns: Connecting Blockchain to the Off-Chain World. In *Proceedings of the International Conference on Business Process Management*; Springer: Cham, Switzerland, 2020; pp. 35–51.
49. Werner, S.M.; Perez, D.; Gudgeon, L.; Klages-Mundt, A.; Harz, D.; Knottenbelt, W.J. SoK: Decentralized Finance (DeFi). *arXiv* **2021**, arXiv:2101.08778.
50. Schär, F. Decentralized Finance: On Blockchain- and Smart Contract-based Financial Markets. *FRB St. Louis Rev.* **2020**. [CrossRef]
51. Harvey, C.R.; Ramachandran, A.; Santoro, J. *DeFi and the Future of Finance*; John Wiley & Sons: New York, NY, USA, 2021. [CrossRef]
52. Angeris, G.; Chitra, T. Improved Price Oracles: Constant Function Market Makers. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*; Association for Computing Machinery: New York, NY, USA, 2020; pp. 80–91. [CrossRef]
53. Kaleem, M.; Shi, W. Demystifying Pythia: A Survey of ChainLink Oracles Usage on Ethereum. *arXiv* **2021**, arXiv:2101.06781.
54. Robert, R. What Do You Look for in a Classic Literature Review? Available online: https://classic-literature.yoexpert.com/classic-literature-general/what-do-you-look-for-in-a-classic-literature-revie-30604.html (accessed on 12 January 2021).
55. Sutherland, S.E. An introduction to systematic reviews. *J. Evid. Based. Dent. Pract.* **2004**, *4*, 47–51. [CrossRef]
56. Alammary, A.; Alhazmi, S.; Almasri, M.; Gillani, S. Blockchain-Based Applications in Education: A Systematic Review. *Appl. Sci.* **2019**, *9*, 2400. [CrossRef]
57. Ahmed, U. The Importance of Cross-Border Regulatory Cooperation in an Era of Digital Trade. *World Trade Rev.* **2019**, *18*, S99–S120. [CrossRef]
58. Kursawe, K. Wendy, the Good Little Fairness Widget: Achieving Order Fairness for Blockchains. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*; Association for Computing Machinery: New York, NY, USA, 2020; pp. 25–36. [CrossRef]
59. Elghaish, F.; Abrishami, S.; Hosseini, M.R. Integrated project delivery with blockchain: An automated financial system. *Autom. Constr.* **2020**, *114*, 103182. [CrossRef]
60. Fedorova, E.P.; Skobleva, E.I. Application of blockchain technology in higher education. *Eur. J. Contemp. Educ.* **2020**, *9*, 552–571. [CrossRef]
61. Ogawa, R.T.; Malen, B. Towards Rigor in Reviews of Multivocal Literatures: Applying the Exploratory Case Study Method. *Rev. Ed. Red.* **2016**, *61*, 265–286. [CrossRef]
62. Schöpfel, J.; Farace, D.J. Grey Literature. *Encycl. Libr. Inf. Sci.* **2010**, *3*, 2029–2039.
63. Eskandari, S.; Salehi, M.; Gu, W.C.; Clark, J. *SoK: Oracles from the Ground Truth to Market Manipulation*; Association for Computing Machinery: New York, NY, USA, 2021; Volume 1.
64. Kitchenham, B.; Charters, S. Guidelines for Performing Systematic Literature Reviews in Software Engineering. 2007, Volume 1. Available online: https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf (accessed on 2 April 2021).
65. Wang, H.; Wang, Y.; Cao, Z.; Li, Z.; Xiong, G. An overview of blockchain security analysis. *Commun. Comput. Inf. Sci.* **2019**, *970*, 55–72. [CrossRef]
66. Bartoletti, M.; Chiang, J.H.; Lluch-Lafuente, A. SoK: Lending Pools in Decentralized Finance. *arXiv* **2020**, arXiv:2012.13230.

67. Harwick, C.; Caton, J. What's holding back blockchain finance? On the possibility of decentralized autonomous finance. *Q. Rev. Econ. Financ.* **2020**. [CrossRef]
68. Boado, E. AAVE Protocol Whitepaper. Available online: https://github.com/aave/protocol-v2/blob/master/aave-v2-whitepaper.pdf (accessed on 2 April 2021).
69. Leshner, R.; Hayes, G. Compound: The Money Market Protocol. White Paper. 2018. Available online: https://www.digitalcoindata.com/whitepapers/compound-whitepaper.pdf (accessed on 2 April 2021).
70. Juliano, A. dYdX: A Standard for Decentralized Margin Trading and Derivatives. 2018. Available online: https://whitepaper.dydx.exchange (accessed on 2 April 2021).
71. Qureshi, H. Flash Loans: Why Flash Attacks Will Be the New Normal. Available online: https://medium.com/dragonfly-research/flash-loans-why-flash-attacks-will-be-the-new-normal-5144e23ac75a (accessed on 19 August 2020).
72. Wolff, M. Introducing Marble: A Smart Contract Bank. Available online: https://medium.com/marbleorg/introducing-marble-a-smart-contract-bankc9c438a12890 (accessed on 12 March 2021).
73. Chainlink The Aave Oracle Network Powered by Chainlink Is Now Live! Available online: https://chainlinkecosystem.com/ecosystem/aave/ (accessed on 2 April 2021).
74. DYdX dYdX Chooses Chainlink as its Oracle Provider for New Market. Available online: https://integral.dydx.exchange/dydx-chooses-chainlink-as-its-oracle-provider-for-new-market/ (accessed on 2 April 2021).
75. Tiwari, A. DeFi Protocol Compound (COMP) Releases Decentralized Price Oracle. Available online: https://btcmanager.com/defi-protocol-compound-comp-decentralized-price-oracle/ (accessed on 5 April 2021).
76. Omelchenko, D. Compound Launches Decentralized Price Oracle. Available online: https://ihodl.com/topnews/2020-08-09/compound-launches-decentralized-price-oracle/ (accessed on 2 January 2021).
77. Wang, Y. Automated market makers for decentralized finance (DeFi). *arXiv* **2020**, arXiv:2009.01676.
78. Coin Market Cap Coin Market Cap-Defi. Available online: https://coinmarketcap.com/view/defi/ (accessed on 5 April 2021).
79. Angeris, G. When Is Uniswap a Good Oracle? Available online: https://medium.com/gauntlet-networks/why-is-uniswap-a-good-oracle-22d84e5b0b6c (accessed on 2 April 2021).
80. Enclave Projects Utilizing Uniswap Oracle. Available online: https://enclaveresearch.com/projects-utilizing-uniswap-oracle/ (accessed on 5 April 2021).
81. Waas, M. Using the New Uniswap v2 as Oracle in Your Contracts. Available online: https://soliditydeveloper.com/uniswap-oracle (accessed on 2 April 2021).
82. Angeris, G.; Kao, H.T.; Chiang, R.; Noyes, C.; Chitra, T. An analysis of Uniswap markets. *arXiv* **2019**, arXiv:1911.03380.
83. Lielacher, A. What Is Impermanent Loss? Available online: https://trustwallet.com/blog/what-is-impermanent-loss (accessed on 3 April 2021).
84. Jakub What Is Impermanent Loss? DEFI Explained. Available online: https://finematics.com/impermanent-loss-explained/ (accessed on 13 April 2021).
85. Klages-Mundt, A.; Harz, D.; Gudgeon, L.; Liu, J.Y.; Minca, A. Stablecoins 2.0: Economic Foundations and Risk-based Models. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*; Association for Computing Machinery: New York, NY, USA, 2020; pp. 59–79. [CrossRef]
86. Tether. Tether: Fiat Currencies on the Bitcoin Blockchain. Available online: https://assets.ctfassets.net/sdlntm3tthp6/SevuOTDbYiCcoaeKEoQAO/828cba8e4e76f9c3075594a98ba807df/TetherWhitePaper.pdf (accessed on 3 April 2021).
87. *Centre USDC Centre whitepaper. Self-Published White Paper.*; 2018.
88. Coinbase DAI Stablecoin. Available online: https://www.coinbase.com/it/earn/dai (accessed on 9 April 2020).
89. Maker Team the Dai Stablecoin System. Whitepaper. 2017. 21p. Available online: https://makerdao.com/whitepaper/DaiDec17WP.pdf (accessed on 9 April 2020).
90. Maker Introducing Oracles V2 and DeFi Feeds. Available online: https://blog.makerdao.com/introducing-oracles-v2-and-defi-feeds/ (accessed on 14 February 2020).
91. Maker MakerDAO Oracle Module. Available online: https://docs.makerdao.com/smart-contract-modules/core-module (accessed on 7 April 2021).
92. Al-Naji, N. Basis. Available online: https://www.basis.io/ (accessed on 5 April 2021).
93. Kuo, E.; Iles, B.; Cruz, M.R. Ampleforth: A New Synthetic Commodity. Ampleforth White Paper. 2019. Available online: https://www.ampleforth.org/papers/ (accessed on 7 April 2021).
94. Slawson, A. CELO Holders: Make Your Voice Heard through On-Chain Governance. Available online: https://medium.com/celoorg/celo-gold-holders-make-your-voice-heard-through-on-chain-governance-96cb5a1e8b90 (accessed on 7 April 2021).
95. Croessmann, R. Zooming in on the Celo Expansion & Contraction Mechanism. Available online: https://medium.com/celoorg/zooming-in-on-the-celo-expansion-contraction-mechanism-446ca7abe4f (accessed on 3 April 2020).
96. Celo-org Celo-Oracle. Available online: https://github.com/celo-org/celo-oracle (accessed on 7 April 2021).
97. Hull, J.C. *Options, Futures and Other Derivatives*, 1st ed.; Pearson Education India: New York, NY, USA, 2017; ISBN 9781292212890.
98. Larsen, A. A Primer on Blockchain Interoperability. Available online: https://medium.com/blockchain-capital-blog/a-primer-on-blockchain-interoperability-e132bab805b (accessed on 9 December 2019).
99. Brooks, S.; Jurisevic, A.; Spain, M.; Warwick, K. Synthetix: A decentralised payment network and stablecoin v0.8. 2018.
100. UMA. *UMA: A Decentralized Financial Contract Platform*; UMA: New York, NY, USA, 2018.

101. UMA. How UMA Solves the Oracle Problem. Available online: https://docs.umaproject.org/oracle/econ-architecture (accessed on 3 April 2021).
102. Mitchell, C. Front-Running. Available online: https://www.investopedia.com/terms/f/frontrunning.asp (accessed on 9 April 2021).
103. Kereiakes, E.; Kwon, D.; Di Maggio, M.; Platias, N. Terra Money: Stability and Adoption. 2019. Available online: https://res.tuoluocaijing.cn/20190705164535-yagq.pdf (accessed on 19 March 2021).
104. EJ Analysis of 8/5, 8/12 Front-Running Attack (Korean). Available online: https://medium.com/terra-money/analysis-of-8-5-8-12-front-running-attack-%ED%94%84%EB%A1%A0%ED%8A%B8-%EB%9F%AC%EB%8B%9D-%EA%B3%B5%EA%B2%A9-%EB%B3%B4%EA%B3%A0%EC%84%9C-4f150ae0bf26 (accessed on 6 April 2021).
105. Foxley, W. Bad Sandwich: DeFi Trader "Poisons" Front-Running Miners for $250K Profit. Available online: https://www.coindesk.com/bad-sandwich-defi-trader-poison-front-running-ethermine-miners (accessed on 9 April 2021).
106. Douceur, J.R. The Sybil Attack. In *Peer-To-Peer Systems*; MIT Faculty Club: Cambridge, MA, USA, 2002; pp. 251–260. [CrossRef]
107. Gudgeon, L.; Perez, D.; Harz, D.; Livshits, B.; Gervais, A. The Decentralized Financial Crisis. In *2020 Crypto Valley Conference on Blockchain Technology (CVCBT)*; IEEE: Piscataway, NJ, USA, 2020; pp. 1–15. [CrossRef]
108. Zoltu, M. How to Turn $20M into $340M in 15 s. Available online: https://medium.com/coinmonks/how-to-turn-20m-into-340m-in-15-s-48d161a42311 (accessed on 1 July 2020).
109. Kelso, E. $340,000,000 Worth of ETH Locked in Maker DAO Can Be Stolen in Seconds, Right Now, Analyst Claims. Available online: https://coinspice.io/news/340000000-worth-of-eth-locked-in-maker-dao-can-be-stolen-in-seconds-right-now-analyst-claims/ (accessed on 9 November 2020).
110. LongForWisdom Flash Loans and Securing the Maker Protocol. Available online: https://forum.makerdao.com/t/updates-flash-loans-and-securing-the-maker-protocol/4923 (accessed on 12 June 2021).
111. Lu, K. Why DeFi Needs Real Oracles. Available online: https://medium.com/bandprotocol/why-defi-needs-real-oracles-beyond-dex-9c80cf192883 (accessed on 19 December 2020).
112. Stevens, R. After DeFi Lost $100 Million to Flash Loan Attacks, Curve Pushes Chainlink. Available online: https://decrypt.co/49758/after-100-million-lost-to-flash-loan-attacks-curve-pushes-chainlink (accessed on 12 February 2021).
113. Mlinaric, N. DeFi Hacks-Million Lost in 2020. Available online: https://nodefactory.io/blog/defi-hacks-millions-lost-in-2020/ (accessed on 9 April 2021).
114. WIlliams, C. Compound User Liquidated for $49 Million, Price Oracle Blamed. Available online: https://cryptobriefing.com/compound-user-liquidated-49-million-price-oracle-blamed/ (accessed on 11 March 2021).
115. Chipolina, S. Oracle Exploit Sees $89 Million Liquidated on Compound. Available online: https://decrypt.co/49657/oracle-exploit-sees-100-million-liquidated-on-compound (accessed on 9 April 2021).
116. Jared Compound: Open Price Feed Live.
117. Carl, T. Synthetix Users Returns 1$ Billion After an Oracle Issue with Their Exchange Platform. Available online: https://bitcoinexchangeguide.com/synthetix-users-returns-1-billion-after-an-oracle-issue-with-their-exchange-platform/ (accessed on 18 April 2021).
118. Redman, J. ETH Price Strains Defi Collateral Loans as "Black Swan" Event Strikes Makerdao. Available online: https://news.bitcoin.com/eth-price-dai-collateral-loans-makerdao/ (accessed on 3 March 2021).
119. Campbell, L. MakerDAO Systems Struggles Amid Volatility-$4M in Dai Left. Available online: https://defirate.com/makerdao-shutdown-concerns/ (accessed on 19 January 2021).
120. HacKen Biggest DeFi Hacks of 2020 Report. Available online: https://hacken.io/researches-and-investigations/biggest-defi-hacks-of-2020-report/ (accessed on 2 March 2021).
121. De'Shazer, M. Flash Loan, Re-Entrancy Attack and DEX Oracle Manipulation Exploit on Ethereum (Trojan Coin Bricks Project Instructions). Available online: https://medium.com/@mikedeshazer/flash-loan-reentry-attack-and-dex-oracle-manipulation-exploit-on-ethereum-trojan-coin-bricks-7969820589d7 (accessed on 14 February 2021).
122. Tarasov, A. Millions Lost: The Top 19 DeFi Cryptocurrency Hack of 2020. Available online: https://cryptobriefing.com/50-million-lost-the-top-19-defi-cryptocurrency-hacks-2020/ (accessed on 9 March 2021).
123. Cao, Y.; Zou, C.; Cheng, X. Flashot: A Snapshot of Flash Loan Attack on DeFi Ecosystem. *arXiv* **2021**, arXiv:2102.00626.
124. Kain, W. Synthetix Response to Oracle Incident. Available online: https://blog.synthetix.io/response-to-oracle-incident/ (accessed on 21 March 2021).
125. Davies, A. How to Audit a Smart Contract? A Guide. Available online: https://www.devteam.space/blog/how-to-audit-a-smart-contract-a-guide/ (accessed on 9 January 2021).
126. Osato, A.-N. MakerDAO Takes New Measures to Prevent Another "Black Swan" Collapse. Available online: https://cointelegraph.com/news/makerdao-takes-new-measures-to-prevent-another-black-swan-collapse (accessed on 9 February 2021).
127. Dramaliev, V. The 7 Biggest Threats to Decentralized Finance. Available online: https://www.trendingtopics.eu/the-7-biggest-threats-to-decentralized-finance-part-2/ (accessed on 3 January 2021).
128. Swami, G. Managing MakerDAO Vault Liquidations with Notifications. Available online: https://www.covalenthq.com/blog/makerdao-liquidation-notifications/ (accessed on 12 February 2021).

129. Breidenbach, L.; Cachin, C.; Coventry, A.; Ellis, S.; Juels, A.; Miller, A.; Magauran, B.; Nazarov, S.; Topliceanu, A.; Chan, B.; et al. Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks. 2021, pp. 1–136. Available online: https://research.chain.link/whitepaper-v2.pdf (accessed on 7 April 2021).
130. MINA Snarks and Snarky. Available online: https://docs.minaprotocol.com/en/snarks (accessed on 7 April 2021).
131. Shapiro, E. HTTPS and Snapps: Bridging Cryptocurrency and the Real World. Available online: https://medium.com/minaprotocol/https-and-snapps-bridging-cryptocurrency-and-the-real-world-962beb21cf2b (accessed on 21 April 2021).
132. Tellor Tellor: A Decentralized Oracle Network. Available online: https://docs.tellor.io/tellor/whitepaper/introduction (accessed on 29 March 2021).
133. Curmi, A.; Inguanez, F. *Blockchain Based Certificate Verification Platform*; Springer International Publishing: Cham, Switzerland, 2019; Volume 339, ISBN 9783030048488.
134. Bains, P.; Murarka, N. *Bluzelle: A Decentralized Database for the Future*; 2017; pp. 1–42.
135. Edelstein, D. Bluzelle Reveals Decentralized Oracle to Enhance DeFi Project Security and Price Reliability. Available online: https://finance.yahoo.com/news/bluzelle-reveals-decentralized-oracle-enhance-132000129.html (accessed on 9 November 2020).
136. Bluzelle Introducing Bluzelle Oracles. Available online: https://blog.bluzelle.com/introducing-bluzelle-oracles-6b06c547d830 (accessed on 12 November 2020).
137. Bluzelle Bluzelle Oracles Update-Ryu Release. Available online: https://blog.bluzelle.com/bluzelle-oracles-update-ryu-release-41b0116796d1 (accessed on 4 July 2021).