DE GRUYTER OPEN Proceedings on Privacy Enhancing Technologies ..; .. (..):1–9

Zeta Avarikioti*, Roman Brunner, Aggelos Kiayias, Roger Wattenhofer, and Dionysis Zindros

The iceberg below the surface: An analysis of dark net content

https://v2.overleaf.com/project/5b6c19affe683e5bba4ed911c

Abstract: We analyze the type of content present on the “dark web”, the set of websites accessible via Tor. We create a darknet spider and crawl the darknet starting from a bootstrap list and recursively following links. The whole connected component of more than X websites and Y base URLs is explored. We publish our spider as open source software. We find that the darknet is well-connected through hub websites such as wikis and forums. We perform comprehensive measurements on the content found using machine learning to analyze and categorize the various types of content. We observe that the majority of darknet content belongs to P and Q. [1]

Keywords: anonymity, tor, machine learning, scraping, spider

DOI Editor to enter DOI
Received ..; revised ..; accepted ...

1 Introduction

When talking about the darknet, one does often think of the Tor network in particular. However, there are other so-called darknets provided by other anonymisation networks such as i2p, FreeNet or Disconnect, to name only a few. So what is the darknet and how is it defined? Let us first have a look at the whole iceberg and start with the tip of the iceberg above the surface. This part of the internet is typically referred to as the clearnet, which may be best defined as the pages that one can find directly or indirectly using their search machine of choice. The part below the surface is often called the "deep web", which contains everything that is not directly accessible without the further use of tools or knowledge. This means that only a restricted number of people can access those pages. Such areas could be a university’s internal networks (may require tools (VPN) or passwords) or a personal eMail box (may require a password or card reader - and is most of the time not public). Although this is arguably a large part of the deep web, there is also a part of the deep web which requires additional software as the Tor browser to be accessed with the goal of keeping its users anonym. These parts are often called the "darknet". In this paper, we will look exclusively at the Tor network since ...[why? do we have an explanation for this]. The Tor network can be used to access any page on the internet (that is publicly available), so it is often used to access pages from the clearnet, while remaining anonymous [2]. However, Tor also offers a mechanism to provide services anonymised. These services are called hidden services (HS). In order to access an HS one has to know its exact URI, which consists of two parts: the hash of the public key of the webpage (as the hostname) and .onion as the "top-level domain". This makes it very hard to guess [How hard - I know there are papers] hostnames and nearly impossible to test all in order to brute force access hidden services. However, some of the pages actually want to be found. Therefore we can partition the network into two subgraphs - the visible part, where pages advertise its URI on link hubs and search engines and the invisible part, which contains only pages with URIs that are not publicly listed. In this paper we will only look at the visible part of the Tor network. Why should we only look into those pages, one might wonder. The point is, that the hash of a public key is computationally hard to guess and even if there exist methods to extract such not listed onion addressed [CMP previous work], we consider the information of such HS to be private and therefore should not be accessed by anyone without the consent of the owner. In addition to that, we want to classify the part of the Tor network that everybody can access without additional information. Lets put it that way: If we access HS from the invisible part, we might find and classify contents from email inboxes or address lists. If we only look at the visible part, we only find data that is publicly available. We assume that the visible part of the darknet has a similar link structure to the clear net, where pages link to each other. Therefore we describe how a scraper can be used

1 [[maybe add another sentence on the results of content classification when we have them.]]

2 Anonymity also includes other behavioural and technical changes to be not tracked

***Corresponding Author: Zeta Avarikioti:** Affil, E-mail: zetavar@ethz.ch
Roman Brunner: Affil, E-mail: robrunne@student.ethz.ch
Aggelos Kiayias: Affil, E-mail: aggelos.kiayias@ed.ac.uk
Roger Wattenhofer: Affil, E-mail: wattenhofer@ethz.ch
Dionysis Zindros: Affil, E-mail: dionyziz@gmail.com

on the darknet, which components are necessary and which information can or cannot be found. A scraper requires only few resources and reduces the impact on the network since it only acts as a regular Tor client. We compare our results to previous results, which used attacks on the Tor network or guard nodes for data collection. This indicates if a scraper works on the darknet and if the so collected dataset is representative compared to previous studies. Further, we estimate the size of the darknet (number of HS) based on the visible HS found and compare it to the estimates provided by the Tor project [https://metrics.torproject.org/hidserv-dir-onions-seen.html?start=2018-01-01end=2018-08-15]. Previous work in this field either used a large number of guard nodes, which knows the actual address of an HS. In contrast to our approach, this method will collect any found HS and cannot fully reconstruct the link structure of the Tor network since it only sees its share of the whole network. However, it allows to test how often particular pages are accessed, a measurement we cannot provide. The other paper which targets darknet content classification used a flaw in the Tor protocol in order to collect data. The scraper was newly developed for this work and is made public under MIT license for further use by other researchers and implements a complete data analysis chain, from collection until classification. The data collected not only contains the content of the HS but also the link structure, status information, network topology information and can be configured to store the change over time of the network. [available under ... - anonimized].

2 Scraping the Darknet

In this section, we describe in detail the structure of the spider we built to crawl the darknet. To understand some of our design choices, we will present the problems faced and the different strategies applied to resolve those issues. [3]. Furthermore, we present some preliminary results on the darknet structure we collected from the crawling process.

To get a complete scrape of the visible darknet, we needed at least the following functionalities:

- Network module: The functionality of this module is to download any content and handle any network related exceptions and issues.
- URI extractor: It is used to parse the downloaded content and extract URIs that will be used recursively to download further content.
- Database: Stores content downloaded as well as the network's structure and URIs to download.
- Tor proxy: Since the scraper should download hidden services (HS), a Tor proxy is required.

The modules depicted in Fig [] provide this functionality and are explained below in more detail.

2.1 Downloading the content

The network module needs to connect to the Tor network, therefore needs a Tor proxy. Since the Tor network is slower than a typical network connection, we used multiple Tor instances, to increase the download performance of the scraper. The Tor proxy spins up a configurable number of Tor instances and uses a round-robin scheduler to distribute download tasks across the instances. Additionally, the network module has to keep track of how many connections are made to a specific host in order to prevent a DoS-like behaviour of the scraper. If it detects that there are already four requests pending to a given host, it puts the new download task into a waiting queue, which will be consumed if a connection to this host is freed up. However, the other modules have to ensure the network module is not overloaded with too many requests to the same host, since the memory usage of the network module increases with every request in the waiting queue. The value of four concurrent requests was chosen to be lower than the default Tor setting[4] to be less detectable by scraping countermeasures some hosts apply.

2.2 Orchestrating the scraping

The conductor balances the needs of the other modules. It is the responsibility of the conductor to dispatch URIs for download to the network module and get new URIs from the DB. Since the network is typically slower than local storage (and this is especially true for the Tor net-

3 [[can you rewrite this sentence? i cant seem to make it look nice or precise!]]

4 Tor 7.5.6, setting network.http.max-persistent-connections-per-server

work), the goal of the conductor is to ensure that the network (as the scarce resource) never has to wait for local storage to deliver new URIs for download. To balance out short peaks in access time to the local storage, we use a pool of URIs ready to download. The size of the pool is configurable to adapt to different setups, e.g. an SSD can deliver data much faster than an HDD; therefore the pool size can be smaller, which also results in a smaller memory footprint. We noticed early in this work that deciding which URI to schedule next for download is important. We also observed that at some point during the scraping process most of the available URIs only stem from a few hosts. Since we do not want to DoS any hosts and still be able to scrape fast, we require a way to prioritise URIs. The prioritisation scheme should ensure fast discovery of new hosts and paths while it avoids bombarding single hosts with tons of requests. These requirements lead to the definition of the following prioritisation schemes:

- Random: When scheduling the URIs in insertion order, it happens almost immediately, that more than the allowed number of URIs is dispatched for download. One easy and fast way to mitigate this issue is randomising the order in which the URIs are scheduled. However, this strategy breaks down when large amounts of the available URIs are pointing to only a few hosts. In such a scenario, randomised access fails, since even if the URIs are reshuffled, they still stem from the same host.
- New: This strategy only gets paths from hosts, for which yet no of the available paths were scraped before. This works well if the collection of not yet scraped hosts is huge since it advances very fast through the network. However, the first page downloaded from a host is not always useful for classification or collection of further links. Therefore this only works well for seed generation but not for data collection.
- Prioritised: Along each URI, two counters are stored. One counts the total number of links pointing to this URI, the other only counts links from distinct hosts that point to this URI. These counters then are used as a proxy for the importance of a particular page. In general, this works well and gives power law distribution [– see powerLawAbnormality.png], however, we see a few abnormalities. The most obvious one stems from bitcoin scamming pages linking to a bitcoin explorer, trying to prove that they are credible. Such tightly connected hosts can break this prioritisation scheme.
- Recursive: One of the most advanced strategies, therefore it also is one of the most computationally intensive. It gets one path per host for which has to hold that not yet a path is in dispatched for download and it has to have unscraped paths. Therefore this strategy will make equally progress on all available hosts, and newly available hosts are considered as soon as the pool is repopulated. Since a chosen host may have multiple paths available for download, we have to specify how we chose this single path. We decided to again look at the incoming link count of a path. The recursive strategy always chooses the path of a host which was not scraped and has the highest incoming link count.
- Inverse recursive: This strategy is the same as the recursive strategy, except it chooses the path with the lowest incoming link count per host. One can choose this option to try and make faster progress on the network scraping since a path that has a high incoming link count already is part of a well-scraped area, whereas the one with the lowest incoming link count is of a not yet very well discovered part.
- Combined: This strategy combines the new, prioritised and randomised strategy from above, to mimic the recursive strategy with less computational resources needed. First, it applies the new strategy until we are not able to repopulate the pool anymore with only not yet scraped hosts. Then the prioritised strategy is used. If this one gets stuck (e.g. because it hits a region of interlinking hosts), it will not be able to return enough paths to repopulate the pool. In this case, this strategy falls back to the random strategy, getting entries at random. The combined strategy is an approximation to the recursive strategy since we use the new strategy first, we have multiple starting points for the prioritised strategy, therefore it should not immediately run into the issue of lots of interlinking hosts. Since this still can happen, the fallback to random is important, in order to keep the scraper running. Note that always the lower strategies are applied first, so as soon as a new host becomes available for download, it will be dispatched. Of course, this is not a perfect replication of the recursive strategy, but it is less computational expensive; however, it is not as precise as the recursive strategy.

2.3 Extract URI

From each of the downloaded HTML files, we need to extract the links that point to another page. A first approach here is to use regular expressions. However, a hidden service URI can still be very complicated, which also leads to a quite complex regex. If the HTML string received is very large, the RegEx solution produced high runtimes, therefore another solution was needed for the collection of URIs along the scraping process. For this purpose, cheerio [], an HTML parser, was used to extract links from link tags, instead of scanning through the whole string looking for URIs. However, we found that some URIs are not enclosed in link tags, thus another process runs with the slower RegEx based extraction to ensure all URIs are found as well as the cheerio based in the scraper process to provide the scraper with new URIs to download.

2.4 Store

FORMATTING To speed up the scraping process, we ran 100 concurrent requests to the Tor network. In order to prevent overload on the Tor nodes within the circuit, we used a pool of Tor of equal size, such that per request one Tor instance was used. The Tor instances were scheduled in a round robin manner

Describe the architectural decisions put in designing the spider. Mention the technologies we're using (node.js, postgres, the tor libraries). Include a diagram for our software architecture. Discuss how we bypass rate limits (e.g. by randomization of visits and by using multiple tor circuits in parallel) and what the rate limiting problems are (including rate limits by individual websites as well as by the tor circuit / network itself). How we avoid downloading illegal content (whitelists and blacklists, content types). Talk about bootstrapping the list, exploration depth. Show the numbers such as how many sites exist, how many base URLs exist, how many links exist, summarize them nicely in a table. Try to aggregate data from our database to create insights for the reader. Extract the "hubs" and state that, contrary to popular opinion, the darknet is well connected; specifically discuss which this hubs are, what their purpose is, and how many links they have for the top 10 hubs.

3 Analsys methodology

Describe the types of content that we found. Talk about our methodology in how we analyzed the content using machine learning and NLP. How were the categories chosen? Discuss how we rank base URLs as well as individual paths. Show diagrams (a pie chart or bars?) with the popularity of various content categories.

4 Darknet contents

5 Political and Ethical Discussion

Discuss the political implications of tor. Is it mostly illegal content? Does it matter? Discuss the ethical considerations of our research and make sure you mention that we didn't subvert the protocol itself but only collected public data.

6 Conclusion

Summarize our results and discuss directions for future work. One issue with the proposed method is the high volatility of the network. We found that up to 30% of the found hosts switch from being online to offline and vice versa in only a week. This circumstance could be faced by adding a temporal resolution to the data gathering process. The provided code already supports temporal resolutions for the hosts and paths, however, one has to match the links by timestamp.One addition could be that along the existing data collection, one also stores to which round of rescraping any entry (whether it now be a host, a path or a link) belongs.

7 Editorial Policy

7.1 Choice of reviewers

The Editor responsible for a given area of physics, turns to experts of the subject for opinion. Research articles and communications are reviewed by minimum two reviewers, review papers by at least three.

7.2 Suggestions from authors

Authors are requested to suggest persons competent to review their manuscript. However, please note that this will be treated only as a suggestion, the final selection of reviewers is exclusively the Editor's decision. The reviewers remain anonymous in any case.

The Editor is fully responsible for decision about the manuscript. The final decision, whether to accept or reject a paper, rests with him/her. The Managing Editor only communicates the final decision and informs the author about further processing.

7.3 Revised manuscript submission

When revision of a manuscript is requested, authors should return the revised version of their manuscript as soon as possible. Prompt action may ensure fast publication, if the paper is finally accepted for publication in .... If it is the first revision of an article authors need to return their revised manuscript within 60 days. If it is the second revision authors need to return their revised manuscript within 14 days. If these deadlines are no met, and no specific arrangements for completion have been made with the Editor, the manuscript will be treated as a new one and will receive a new identification code along with a new registration date.

7.4 Final proofreading

Authors will receive a PDF file with the edited version of their manuscript for final proofreading. This is the last opportunity to view an article before its publication on the journal's web site. No changes or modifications can be introduced once it is published. Thus authors are requested to check their proof pages carefully against manuscript within 3 working days and prepare a separate document containing list of all the changes that should be introduced. Authors are sometimes asked to provide additional comments and explanations in response to remarks and queries from the language and technical editors. In case the authors do not deliver the list of corrections to proofs in the requested time the manuscript will be published as is.

7.5 Reprints

Because the journal is published in an Open Access model, and has no printed version, the authors receive no reprints.

7.6 Erratum

If any errors are detected in the published material they should be reported to the Managing Editor. The corresponding authors should send appropriate corrected material to the Managing Editor via email. This material will be considered for publication in form of erratum in the earliest available issue of ....

7.7 Copyright

All authors retain copyright, unless – due to their local circumstances – their work is not copyrighted. The non-commercial use of each article will be governed by the Creative Commons Attribution-NonCommercial-NoDerivs license. The corresponding author grants De Gruyter Open the exclusive license to commercial use of the article, by signing the License to Publish. Scanned copy of license should be sent by e-mail to the Managing Editor of the journal, as soon as possible.

8 Paper writing guide

8.1 Paper elements

1. title page with:
   (a) title (short title),
   (b) full name(s) of author(s),
   (c) name and address of workplace(s),
   (d) personal e-mail address(es),
2. abstract,
3. up-to five keywords,
4. text,
5. reference lists.

8.1.1 Abstract

An abstract must accompany every article. It should be a brief summary of the significant items of the main paper. An abstract should give concise information about the content of the core idea of your paper. It should be informative and not only present the general scope of the paper but also indicate the main results and conclusions. An abstract should not normally exceed 200 words. It should not contain literature citations or allusions to the tables or illustrations. All non-standard symbols and abbreviations should be defined.

In combination with the title and key-words, the abstract is an indicator of the content of the paper. Authors should remember that on-line systems rely heavily on the content of titles and abstracts to identify articles in electronic bibliographic databases and search engines. They are therefore requested to take great care in preparing these elements.

8.1.2 Text

8.1.2.1 General rules for writing

- use simple and declarative sentences, avoid long sentences, in which the meaning may be lost by complicated construction;
- be concise, avoid idle words;
- make your argumentation complete; use commonly understood terms; define all non-standard symbols and abbreviations when you introduce them;
- explain all acronyms and abbreviations when they first appear in the text;
- use all units consistently throughout the article;
- be self-critical as you review your drafts.

8.1.2.2 Structure of a paper

Research papers and review articles should follow a strict structure. Generally a standard scientific paper is divided into:

- introduction: you present the subject of your paper clearly, you indicate the scope of the subject, you present the goals of your paper and finally the organization of your paper;
- main text: you present all important elements of your scientific message;
- conclusion: you summarize your paper.

Experimental part and/or calculations should be presented in sufficient details to enable reader to repeat the original work.

Fig. 1. A figure caption should be placed **below** the figure.

Fig. 2. A figure caption for Fig. 2.

8.1.2.3 Footnotes/End-notes/Acknowledgments

We encourage authors to restrict the use of footnotes. If necessary, please make end-notes rather than footnotes. Allowable footnotes/end-notes may include:

- the designation of the corresponding author of the paper;
- the current address of an author (if different from that shown in the affiliation);
- traditional footnote content.

8.1.2.4 Tables

Authors should use tables only to achieve concise presentation, or where the information cannot be given satisfactorily in other ways. Tables should be numbered consecutively using Arabic numerals and referred to in the text by number. Each table should have an explanatory caption which should be as concise as possible.

8.1.2.5 Figures

Authors may use line diagrams and photographs to illustrate theses from their text. The figures should be clear, easy to read and of good quality. Styles and fonts should match those in the main body of the article. All figures must be mentioned in the text in consecutive order and be numbered with Arabic numerals.

8.1.2.6 Typesetting

Type main text in roman (upright) font. The chemical symbols and compounds, units of measure, most multi-letter operators and functions should are written in roman upright as well. The variables, constants, symbols for particles, most single-letter operators, axes and planes, channels, types (e.g., n, p), bands, geometric points, angles, lines, chemical prefixes, symmetry designations, transitions, critical points, color centers, quantum-state symbols in spectroscopy, and most single-letter abbreviations should be written in roman italic. Boldface roman type is reserved for indicating vectors and in some special cases matrices.

8.1.2.7 Mathematical symbols

The multiplication signs are reserved for a vector product ($\mathbf{A} \times \mathbf{B}$) and simple dot product ($\mathbf{A} \cdot \mathbf{B}$). The only exception are numbers expressed in scientific notation (9.7×10^3 MeV).

8.1.2.8 Units

Units and dimensions should be expressed according to the metric system and SI units. This system is based on: meter (m), second (s), kilogram (kg), ampere (A), kelvin (K), mole (mol), and candela (cd). Most units are spaced off from the number, e.g. 12 mV. The only exceptions are:

1%, 1‰, 1°C, 1°, 1', 1".

Decimal multiples or sub-multiples of units are indicated by the use of prefixes

$$\mu = 10^{-6},\ \mathrm{m} = 10^{-3},\ \mathrm{c} = 10^{-2},\ \mathrm{d} = 10^{-1},\ \mathrm{da} = 10^{1},$$
$$\mathrm{h} = 10^{2},\ \mathrm{k} = 10^{3},\ \mathrm{M} = 10^{6},\ \mathrm{G} = 10^{9},\ \textit{etc.}$$

Compound units are written as

$$4221.9\ \mathrm{J\,kg^{-1}\,K^{-1}} \text{ or } 4221.9\ \mathrm{J/(kg\,K)},$$

with a thin space between unit parts.

Authors should indicate precisely in the main text **where tables and figures should be inserted**, if these elements are given at the end in the original version of the manuscript (or supplied in separate files). If this information is not provided along with the manuscript, we will assume that the figures and/or tables should be insert at the closest position to first reference to them in the published paper.

8.1.2.9 Multimedia and images

Authors can attach files in most popular formats, including (for example):

- images in BMP, GIF, JPEG formats,
- multimedia files in MPEG or AVI formats.

However please keep to file types that are read by standard media players (e.g. RealPlayer, Quicktime, Windows Media Player) and/or standard office applications (Adobe Acrobat Reader, Microsoft Office etc.).

Your attachments may be accessible through links to external locations or to our internal locations (if you choose the second option, please remember to send us your attachments).

Please remember that your images, video and animation clips are intended for Internet use and we need to consider the needs of users with slow Internet connections. Please try to minimize file sizes by using a lower resolution or number of colors for images and animations (as long as the material is still clear). To help you in formatting your images (including tables and figures) or multimedia files, please submit your paper with separate attachments, which are used in your paper.

8.1.2.10 English language

Journal is published only in English. Make sure that your manuscripts are clearly and grammatically written. Please note that authors who are not native-speakers of English can be provided with help in rewriting their contribution in correct English. Try to prepare your manuscript in an easily readable style; this will help avoid severe misunderstandings which might lead to rejection of the paper.

8.1.3 Reference list

A complete reference should give the reader enough information to find the relevant article. All authors (unless there are six or more) should be named in the citation. If there are six or more, list the name of the first one followed by “et al”. Please pay particular attention to spelling, capitalization and punctuation here. Completeness of references is the responsibility of the authors. A complete reference should comprise the following:

8.1.3.1 Reference to an article in a journal

Elements to cite: Author’s Initials. Surname, – if more authors, see examples below, Title of journal – abbreviated according to the ISI standards[5], volume number, page or article number (year of publication). Please supply DOI or URL for e-version of the papers. See Refs. [1–8] for example.

5 http://images.isiknowledge.com/WOK46/help/WOS/0-9_abrvjt.html

8.1.3.2 Reference to a book

Elements to cite: Author's Initials. Surname, Title, Edition – if not the first (Publisher, Place of publication, Year of publication) [9].

8.1.3.3 Reference to a part/chapter book

Elements to cite: Author's Initials. Surname, In: Editor's Initials. Editor's Surname (Ed.), Book Title, Edition – if not the first, (Publisher, Place of publication, Year of publication) page number [10].

8.1.3.4 Reference to a preprint

Elements to cite: Author's Initials. Surname, arXiv:preprint-number and version [11, 12].

8.1.3.5 Reference to a conference proceedings

Elements to cite: Author's Initials. Surname, In: Editor's Initials. Editor's Surname (Ed.), Conference, date, place (town and country) of conference (Publisher, place of publication, year of publication) page number [13].

8.1.3.6 Reference to a thesis

Elements to cite: Author's Initials. Surname, D.Sc./Ph.D./M.Sc./B.Sc. thesis, University, (town, country, year of publication) [14].

8.1.3.7 Reference to an article in a newspaper

Elements to cite: Author's Initials. Surname, Newspaper Title, date of publication, page number [15, 16].

8.1.3.8 Reference to a patent

Elements to cite: Originator, Series designation which may include full date [17].

8.1.3.9 Reference to a standard

Elements to cite: Standard symbol and number, Title [18, 19].

Please add language of publication for materials which are not written in English. Indicate materials accepting for publications by adding "(in press)". Please avoid references to unpublished materials, private communication and web pages.

You should make sure the information is correct so that the linking reference service may link abstracts electronically. For the same reason please separate each reference from the others.

Before submitting your article, please ensure you have checked your paper for any relevant references you may have missed.

8.1.4 Submission formats

Manuscripts for ... should be submitted in the LaTeX format with figures in EPS, PDF or PNG format. Authors are strongly encouraged to register their manuscript in arXiv preprint server and submit it to our Editorial Manager using arXiv's paper ID.

8.1.5 Supplementary data

You can also submit any supplementary data files as well. These may include long tables (in HTML or plain TXT format) or movies (preferably in AVI format).

References

[1] A. P. Raposo, H. J. Weber, D. E. Alvarez–Castillo, M. Kirchbach, Cent. Eur. J. Phys. 5, 253 (2007)

[2] J. Barth et al. (SAPHIR Collaboration), Phys. Lett. B 572, 127 (2003)

[3] S. Chekanov et al., Eur. Phys. J. C 51, 289 (2007)

[4] K. Malarz, Postepy Fizyki 57, 235 (2006) (in Polish)

[5] G. Meng, Cent. Eur. J. Phys., DOI:10.2478/s11534-007-0038-1

[6] R. Hegselmann, U. Krause, Journal of Artificial Societies and Social Simulation (2006), http://jasss.soc.surrey.ac.uk/9/3/10.html

[7] A. Dybala, Cent. Eur. J. Chem. (in press)

[8] A. Dybala, Przeglad chemiczny (in Polish, in press)

[9] M. Lister, Fundamentals of Operating Systems, 3rd edition (Springer-Verlag, New York, 1984)

[10] C. K. Clenshaw, K. Lord, In: B. K. P. Scaife (Ed.), Studies in Numerical Analysis (Academic Press, London and New York, 1974) 95

[11] M. Majewski, K. Malarz, arXiv:cond-mat/0609635v2 [cond-mat.stat-mech]

[12] J. A. C. E. Solano, arXiv:0707.1343v1 [astro-ph]

[13] A. Kaczanowski, K. Malarz, K. Kulakowski, In: T. E. Simos (Ed.), International Conference of Computational Methods in Science and Engineering, Sep. 12-16, 2003, Kastoria, Greece (World Scientific, Singapore 2003) 258

[14] A. J. Agutter, Ph.D. thesis, Edinburgh University (Edinburgh, UK, 1995)

[15] A. Sherwin, The Times, Jul. 13, 2007, 1

[16] M. Dzierzanowski, Wprost, Jul. 8, 2007, 18 (in Polish)

[17] Philip Morris Inc., European patent application 0021165 A1, Jan. 7, 1981
[18] ISO 2108:1992, Information and documentation — International standard book numbering (ISBN)
[19] ISO/TR 9544:1988, Information processing — Computer-assisted publishing — Vocabulary