Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

dr hab. Artur Jeż
profesor uczelni

09.01.2023 r., Wrocław

Recenzja rozprawy doktorskiej mgra Juliusza Straszyńskiego „Dokładne szablony i wyszukiwanie wzorca z dozwolonymi niedopasowaniami"

Ocena

Przedstawiona przez mgra Juliusza Straszyńskiego rozprawa doktorska „Dokładne szablony i wyszukiwanie wzorca z dozwolonymi niedopasowaniami" spełnia zarówno ustawowe (wg. Ustawy „Prawo o szkolnictwie wyższym i nauce") jak i zwyczajowe wymagania i kryteria stawiane rozprawie doktorskiej; tym samym wnoszę o dopuszczenie mgra Juliusza Straszyńskiego do dalszych etapów postępowania w sprawie nadania stopnia doktorskiego.

Ponieważ pozostali recenzenci pracy nie władają językiem polskim, pozostała część, poza podsumowanie, recenzji została napisana w języku angielskim.

Contents of the PhD thesis

The submitted thesis is based on the following 6 research articles:

[1] Jakub Radoszewski, Juliusz Straszyński, "Efficient Computation of 2-covers of a String", 28th Annual European Symposium on Algorithms (ESA) 2020, LIPIcs 173 77:1–77:17 (2020).

[2] Maxime Crochemore, Costas S. Iliopoulos, Jakub Radoszewski, Wojciech Rytter, Juliusz Straszyński, Tomasz Waleń, Wiktor Zuba, "Shortest Covers of All Cyclic Shifts of a String", *Theoretical Computer Science* 866 70–81 (2021).

[3] Maxime Crochemore, Costas S. Iliopoulos, Jakub Radoszewski, Wojciech Rytter, Juliusz Straszyński, Tomasz Waleń, Wiktor Zuba, "Internal Quasiperiod Queries", 27th International Symposium on String Processing and Information Retrieval (SPIRE) 2020, LNCS 12303 60–75 (2020).

[4] Jakub Radoszewski, Wojciech Rytter, Juliusz Straszyński, Tomasz Waleń, Wiktor Zuba, "String Covers of a Tree", 28th International Symposium on String Processing and Information Retrieval (SPIRE) 2021, LNCS 12944 68–82 (2021).

[5] Panagiotis Charalampopoulos, Costas S. Iliopoulos, Tomasz Kociumaka, Solon P. Pissis, Jakub Radoszewski, Juliusz Straszyński, "Efficient Computation of Sequence Mappability", *Algorithmica* 84(5) 1418–1440 (2022).

AJ

Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

[6] Panagiotis Charalampopoulos, Tomasz Kociumaka, Solon P. Pissis, Jakub Radoszewski, Wojciech Rytter, Juliusz Straszyński, Tomasz Waleń, Wiktor Zuba, "Circular pattern matching with k mismatches", *Journal of Computer and System Sciences* 115 73–85 (2021).

The thesis also includes a summary of results of those papers.

In general, the thesis explores variations of classic problems in string algorithms: perhaps the most important notions in string algorithms are periodicity (a string S can be represented as $S = P^k$ for some string P and $k \geq 2$) and occurrences (i.e. S can be represented as $S'PS''$, where P is the pattern). A cover is a useful variation of those two notions: P is a cover of S when for each $1 \leq i \leq |S|$ for some $i' \leq i \leq i''$ we have $S = S[1 \dots i']PS[i'' \dots |S|]$, i.e. each position of S is "covered" by some occurrence of P. The other notion investigated in the thesis are the occurrences with (at most k) errors, i.e. we represent $T = T'P'T''$, where P and P' are of the same length and differ at at most k positions.

The paper [1] considers a problem of 2-cover: given a string S we ask whether there are two strings X, Y of equal length, such that each position of S is within some occurrence of X or some occurrence of Y in S, i.e. each position is "covered" by (some occurrence of) X or Y. The paper presents an algorithm with $\mathcal{O}(n \log n \log\log n + output)$ expected running time and $\mathcal{O}(n \log n \log^2 \log n / \log\log\log n + output)$ worst-case running time; the different multipliers coming from time of performing operations in predecessor data structure; with the randomized being a classic result and deterministic — a more modern one.

There are two distinctive cases: either X is prefix of S and Y a suffix — this case is much easier, or X is both a prefix and a suffix of S, so in particular there is no "canonical" candidate for Y. A natural observation is that once $\ell = |X| = |Y|$ is fixed, then allowing $\mathcal{O}(n/\ell)$ running time for finding Y yields in total $\mathcal{O}(n \log n)$ time. At the same time if there is a 2-cover (X, Y) then there is one using $\mathcal{O}(n/\ell)$ occurrences of Y. A data structure for representing (appropriate subset of) such occurrences is proposed; in particular it uses recent results on IPM (internal pattern matching) queries

The results generalize to the case of λ-covers, where we consider λ strings $X_1, X_2, \dots, X_\lambda$ and their occurrences in S; this variant is known to be NP-complete. An extension of the algorithm with $n^{\lambda-2}$ running time multiplier is given: in essence, we choose X_1 as X in the 2-cover case then consider all valid $X_2, \dots, X_{\lambda-1}$ and compute X_λ similarly as Y in the 2-cover.

This is an example of good paper in string algorithms, which on one hand proposes deeper insight into the combinatorial structure and on the other applies advanced data structures and recent results to algorithmically exploit those properties. To the best of my knowledge this problem was not addressed directly before (the λ-cover was), so one cannot directly compare to known results. The author statement about his involvement clearly makes his input substantial.

The second paper [2] considers a connected problem, in which given a string S we want to find the shortest covers of all cyclic shifts of S, i.e. for each $1 \leq k \leq |S|$ a cover of $S[k \dots n]S[1 \dots k-1]$. The main combinatorial observation relates the covers (of a cyclic shift of S) with occurrences of squares in T^3. While considering occurrences of squares is a known technique and occurrences within powers are used in terms of cyclic shifts, the usage of square within cubes is unusual. Using this approach a relatively standard, not meaning easy, combinatorial arguments and algorithms are used to obtain the $\mathcal{O}(n \log n)$ running

AJ

Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

time to compute all covers and $\mathcal{O}(n)$ for the shortest one. The input of Juliusz Straszyński is again important. This result has a "classic" feel, in the sense that it is much more based on combinatorial properties of the strings, which make it nice, and at the same time not as technically involved as, say, results in [1]. Again, this problem was not considered before, however, extending known results (computation of covers of all prefixes) to such related setting is a common practice.

Apart from that there is a much simpler and faster algorithm for computing the shortest covers of Fibonacci words. It is based on combinatorial properties of Fibonacci words. I have more mixed feelings about those results: the resulting algorithm is much simpler and "cleaner", yet it is not clear why one would like to have such an algorithm; I would value it much higher if an explicit precise description were obtained.

The third paper [3] considers covers in a modern setting of internal queries: given a string S we preprocess it and then want to answer queries: given i, j return the shortest cover of $S[i \dots j]$ and (compact) representation of all covers of $S[i \dots j]$. The approach of internal queries, in which we preprocess a given string and want to answer some queries on its substrings, is a relatively new one, with some important results and applications.

The main idea of the proposed solution is to represent the tree as $n/2^k$ substrings of length 2^k for each k and organize it into a tree. Then each substring S' can be represented as at most $2 \log n$ such substrings and covers of S' are seeds of each of those $2 \log n$ substrings. On the technical side, internal periodicity queries as well as much new combinatorial insight is used.

While this problem was not directly addressed before, it generalizes the internal periodicity queries (which, essentially, were focused on perdiods and related notions) to covers and the obtained bounds are similar as the ones for more restricted queries, which presumably makes them hard to beat. The authorship statement makes Juliusz Straszyński as an author of at least one important technical part of the paper. My opinion on the paper is similar as for paper [1], i.e. a good paper with combinatorial insight and using advanced data structures, though I think that the former is more involved.

The last paper on covers [4] considers covers of trees. The notions of occurrences, covers, *etc.* in case of labelled trees are less standard and may vary significantly depending on technical choices. Here the tree T is edge-labelled and a string S occurrence corresponds to a directed path whose labels form S; if T is rooted then we allow only paths that are subpaths of leaf-to-root paths. Similarly as for string, S is a cover of T when each edge in T appears in some occurrence of S.

The case of directed and undirected tree differ greatly. For the directed case the algorithm is somewhat similar to the string case: as each leaf needs to be covered, a cover of length ℓ needs to cover each path of length ℓ that begins in a leaf; in particular, there is at most one cover of length ℓ. The algorithms stores appropriate representation of occurrences of such candidates, with the main technical tool being on one hand an analogue of suffix tree, i.e. the suffix tree of a rooted tree (a rather involved data structure on its own) and marked ancestor data structure, a classic result on its own. The analysis also involves a nice combinatorial property — it is well known that the sum of heights of all nodes of a tree (which can be seen, up to additive linear term, as the sum of heights of highest sons) can be quadratic, yet it is shown that the the sum of heights of second-highest son is at most linear. In the end the running time is $\mathcal{O}(n \log n / \log \log n)$.

The case of unrooted trees is substantially different in flavour, and it seems more complex:

Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

there could be many different covers of the same length. There is a simple linear bound on the number of covers, and an algorithm testing them in quadratic time follows relatively naturally, yet it requires quadratic space. The space can be reduced to linear, at the expense of logarithmic time increase; in essence a recursive approach based on centroid decomposition is us.

What I personally like in the paper is that the combinatorics of repetitions in trees is different than the one for strings and so some fresh perspective and new arguments are needed. This means that even more basic results need to be discovered and neat results can be showed on the way. This is an advantage of results in a less developed field. Still, the results use the strong tools from string algorithms.

The declaration of the input of Juliusz Straszyński marks him as the main author of the results of the unrooted trees, which, in my opinion, constitute a smaller part of the paper.

The next considered problem is the sequence mappability [5]. The problem comes from computational genomics: given a (long) string we want to compute for each of its substring of length m the list of all its "approximate" occurrences, so copies, within this string, where approximate means that there differences on at most k positions between the pattern and the occurrence. This problem is known to be hard, and so exponential algorithms are investigated. The paper proposes a solution which runs in $\mathcal{O}(n \cdot \min(m^k \log^k n))$ (times a function of k, k is usually assumed to be small) time and linear space, with high probability. As usual (and perhaps necessary) for such algorithms, we perform an exhaustive search, however, appropriate organization of the data allows to limit the space and time consumption. The paper also investigates some particular cases, say $k = 1$, in which the analysis can be improved, as well as as some applications. A conditional lower bound is also given, it follows rather easily from earlier known conditional lower bounds.

This is a technically involved results, which specialises and improves a known technique, the authorship statement makes Juliusz Straszyński as the leading author. The investigation of exponential algorithm for computationally hard problems is one of the actively researched branches of computer science, and this works follows nicely within this framework. Those are particularly investigated for problems with real-life applications. Similar thing can be said about the conditional lower bound.

I have some reservations about the condition $k = \mathcal{O}(1)$. One would rather expect that the number of errors is a (linear?) function of m. Also, since this is advertised as an applied problem, some comments on what are the usual ranges of values of n, m, k would be in place.

The last paper [6] from the thesis considers a problem of circular pattern matching with mismatches, i.e. given a string S (of length n and pattern P (of length m) and a bound k we look for substrings $T[i \dots i + |P| - 1]$ which differ at at most k positions from $P[\ell \dots m]P[1 \dots \ell - 1]$ for some ℓ. The paper first presents a rather simple algorithm of running time $\mathcal{O}(nk)$, based on standard stringology tools. This algorithm is credited to Juliusz Straszyński. It is claimed that due to its simplicity of implementation, it is practical.

The more involved algorithm essentially employs a novel technique by Bringmann, Wellnitz, Künnemann, called "few mismatches or almost periodic", which was developed in context of approximate pattern matching but with much different application in mind. Ideas for analysis of this approach were also used, though the different setting required many new ideas and combinatorial analysis. The final complexity is $\mathcal{O}(n + \frac{n}{m}k^4)$. Yet again, those results combine combinatorial insight and tools (the mentioned technique) with data structures.

MJ

Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

While Juliusz Straszyński contribution in this part is not listed, he was the one who familiarized the authors with this technique.

Evaluation

The obtained results are typical, which does not mean simple, examples of good results in string algorithms, merging, in various proportions depending on the exact paper, good knowledge and ability to employ modern and advanced data structures, good skill in string combinatorics and general combinatorial algorithms design. They employ recent techniques and results, like the internal pattern matching queries or "few mismatches or almost periodic" technique, and are up to date with modern results and trends in string algorithms. At the same time, say in [4], there are examples of purely combinatorial, elegant arguments. I want to stress that the technical content of the papers is of very high quality, in each aspect: combinatorial arguments, utilizing advanced techniques and employing data structures.

Most of the problems considered in the paper, with a notable example of k-mappability, are "new" in the sense that those problems were not considered before, they are extensions or variations on some well-known previously considered problems. This is how usually research is done in developed fields. This shows an ability of Juliusz Straszyński to search for new problems. Moreover, the obtained results are technically strong and they seem difficult to improve.

On the negative side, most of the problems and results are rather specialised and are of less interest to a general audience (this does not apply to [4]). This unfortunately often applies to results in more developed fields.

The writing is in general good and understandable. However, it is rather compressed and quick in the presentation. This is common among conference papers, which have tight space limit, yet a thesis would be a good place to make the presentation more approachable.

The results, except [1], which is the most technically advanced in my opinion, were obtained in larger groups of coauthors, who are known in the field. This makes input of each individual author proportionally smaller, but in my opinion the part acknowledged to Juliusz Straszyński is more than required for a PhD thesis. Moreover, it would be unfair to use a fact that a PhD student works in a good research group as an argument against him. On the contrary, this shows that Juliusz Straszyński is part of a larger scientific community and he is recognized within it. On the other hand, Juliusz Straszyński has coauthored substantially more papers in string algorithms and some of them could easily fit in to the thesis as well. In particular, there are no doubts that Juliusz Straszyński has a good knowledge of the field of string combinatorics and string algorithms and can obtain new results in it. Summarizing, I have no doubt that the technical criteria for a PhD thesis are met in this case.

Smaller remarks

I have some smaller remarks, which do not affect my overall judgment on the thesis.

Randomised subroutines, encapsulated as data structures or hash functions, are often used in the papers forming the thesis, sometimes without clear distinction, what is randomised and what deterministic. I have no objection against randomized algorithms in general, but once randomisation is employed, several things could be done in randomised way and the authors do not seem to try to do this — the main body of developed algorithms

AM

Uniwersytet Wrocławski

INSTYTUT INFORMATYKI
ZAKŁAD ZŁOŻONOŚCI OBLICZENIOWEJ I ALGORYTMÓW
ul. F. Joliot-Curie 15
50-383 Wrocław
tel. +48 71 375 70 35
www.uni.wroc.pl

is deterministic. I find such an approach problematic. On the other hand, the time increase when deterministic subprocedures are used, is usually small, so I do not think that those results would be weaker if deterministic subprocedures were used instead.

Simple techniques, which became folklore, are referred by names and not explained, but rather linked to some usage in the literature, or called principles, etc. I find this difficult to read — I propose to explain at least what is done and refer to the analysis or just declare it to be folklore.

The thesis is formally built upon published work, so simply including them in the form in which they were accepted looks reasonable. Yet, having statements of the form "the proof is left for a full version" or similar looks strange in the PhD thesis.

The PhD thesis begins with a 15-page summary of articles on which it is based. This includes the definitions of necessary technical terms, summary of previous results, statement of the main results of those papers and a precise description of the input of Juliusz Straszyński in each of the papers. I am thankful for the precise descriptions of the contribution, which are much more detailed than the typical ones. However, otherwise I find this summary a bit redundant: it does not contain enough details to evaluate the techniques, nor even the results. It does not contain any comments that would constitute some added value when compared to the papers alone: some afterthoughts, evaluation of the research plan, hindsight, common introduction, finding common themes... This is close to concatenation of abstracts of the listed papers.

Summary

I find the PhD thesis to be of high quality, some smaller remarks do not affect my general opinion. Juliusz Straszyński has good understanding of modern results and trends in string algorithm and can apply advanced data structures and algorithms in his work. The papers forming the thesis contain new, advanced and involved results in the field of string algorithms. I have no doubts the Juliusz Straszyński should be granted the PhD title in computer science.

Podsumowanie

Pracę ogólnie oceniam bardzo dobrze, drobne uwagi redakcyjne nie wpływają na moją ocenę. Juliusz Straszyński dobrze rozumie współczesne wyniki i trendy w algorytmice tesktów i potrafi zastosować w swojej pracy zaawansowane algorytmy i struktury danych. Prace wchodzące w skład rozprawy doktorskiej zawierają nowe, zaawansowane i trudne wyniki z algorytmów tekstowych. Nie mam wątpliwości, że rozprawa jest podstawą do nadania tytułu doktora w dyscyplinie naukowej informatyka i wnioskuję o dalsze procedowanie przewodu doktorskiego.