Fall Workshop on Algorithm and Computation (FWAC 2023)

포스코국제관 대회의실 | 2023.10.27-10.28 | 알고리즘연구회

[Session 1] Invited Talk

Discrete Algorithms for Hyperbolic Spaces - Antoine Vigneron (UNIST)

In this talk, we will review algorithms for hyperbolic spaces, and in particular for the Poincare half-space and for Gromov hyperbolic spaces. We will consider proximity problems such as the near-neighbor searching, embedding into tree metrics and the traveling salesman problem.

[Session 2] Contributed Talks

Guarding Points on a Terrain by Watchtowers - Byeonguk Kang (POSTECH)

We study the problem of guarding points on an x-monotone polygonal chain, called a terrain, using k watchtowers. A watchtower is a vertical segment whose bottom endpoint lies on the terrain. A point on the terrain is visible from a watchtower if the line segment connecting the point and the top endpoint of the watchtower does not cross the terrain. Given a sequence of points lying on a terrain, we aim to partition the sequence into k contiguous subsequences and place k watchtowers on the terrain such that every point in a subsequence is visible from the same watchtower and the maximum length of the watchtowers is minimized. We present efficient algorithms for two variants of the problem.

Maximum Coverage by k Lines - Chaeyoon Chung (POSTECH)

Given a set of n disks in the plane, we study the problem of finding k lines that together intersect the maximum number of input disks. We consider three variants depending on the constraints on a solution: (a) no constraint on the lines, (b) the k lines should be parallel to each other, and (c) the k lines should pass through a common point. For $k = 1$, we give an $O(n^2)$-time algorithm for all three cases and show that this bound is tight. For $k = 2$, we give $O(n^3 \log n)$-time algorithms for all three cases. For any fixed $k \geq 3$, we give an $O(n^{\frac{3k}{2}})$-time algorithm for (a). For the other variants, the running times of our algorithms vary from $O(n^4)$ to $O(n^6)$.

Universal convex covering problems under affine dihedral group actions - Mook Kwon Jung (POSTECH)

We consider the smallest-area universal convex H_k-covering of a set of planar objects, which covers every object in the set allowing the group action of the affine dihedral group $H_k = \mathrm{T} \rtimes D_k$ generated by the translation T and the dihedral group D_k. The dihedral group D_k is the group of symmetries of a regular polygon generated by the discrete rotation group Z_k and a reflection. We first classify the smallest-area convex H_k-coverings of the set of all unit segments. Then we show that a suitably positioned equilateral triangle of height 1 is a universal convex H_1-covering of the set S_c of all closed curves of length 2. We show that no proper closed subset of the covering is a H_1-covering and the covering is a smallest-area triangle H_1-covering of S_c. We conjecture that it is the smallest-area convex H_1-

covering of S_c. We also show that a suitably positioned equilateral triangle $\triangle\beta$ of height 0.966 is a universal convex H_2-covering of S_c. Finally, we give a universal convex H_3-covering of S_c whose area is strictly smaller than that of $\triangle\beta$.

Adjusting a cyclic sequence under translation - Seungjun Lee (POSTECH)

We consider the problem of finding an adjusted sequence (translation) for a given cyclic sequence of points in the plane. There are two optimization factors in this problem, one is the distance between consecutive points in the adjusted sequence and the other is the distance between adjusted sequence and cyclic sequence. This problem is special case of elastic geometric shape matching problem. We present $O(\varepsilon^{-1}\log\varepsilon^{-1}\ n\log n)$ time $(1+\varepsilon^{-1})$-approximation algorithm, where n is the number of points in the sequence.

[Session 3] Contributed Talks

BICE: Exploring Compact Search Space by Using Bipartite Matching and Cell-Wide Verification - Yunyoung Choi (Alsemy)

Subgraph matching is the problem of searching for all embeddings of a query graph in a data graph, and subgraph query processing (also known as subgraph search) is to find all the data graphs that contain a query graph as subgraphs. Extensive research has been done to develop practical solutions for both problems. However, the existing solutions still show limited query processing time due to a lot of unnecessary computations in search. In this paper, we focus on exploring as compact search space as possible by using three techniques: (1) pruning by bipartite matching, (2) pruning by failing sets with bipartite matching, and (3) cell-wide verification. We propose a new algorithm BICE, which combines these three techniques. We conduct extensive experiments on real-world datasets as well as synthetic datasets to evaluate the effectiveness of the techniques. Experiments show that our approach outperforms the fastest existing subgraph search algorithm by up to two orders of magnitude in terms of elapsed time to process a query. Our approach also outperforms state-of-the-art subgraph matching algorithms by up to two orders of magnitude.

Parameterized Algorithm for the Planar Disjoint Paths Problem: Exponential in k^2, and Linear in n - Kyungjin Cho (POSTECH)

We study the Planar Disjoint Paths problem: Given an undirected planar graph G with n vertices and a set $\mathrm{T} = \{(\mathrm{s}_1,\mathrm{t}_1),\dots,(\mathrm{s}_\mathrm{k},\mathrm{t}_\mathrm{k})\}$ of k pairs of vertices, the goal is to find a set P of k pairwise vertex-disjoint paths connecting s_i and t_i for all indices $i\ \in\{1,\dots,\mathrm{k}\}$. This problem is NP-complete even for grid graphs. We present a $2^{o(k^2)}n$-time FPT algorithm for the Planar Disjoint Paths problem. This improves the two previously best-known algorithms: $2^{2^{O(k)}}n$-time algorithm by Reed and $2^{O(k^2)}n^6$-time algorithm by Lokshtanov et al. Our result simultaneously achieves the best dependency on both k and n among all previous approaches. Our algorithm develops the $2^{O(k^2)}n^6$-time algorithm represented by Lokshtanov et al. We and Lokshtanov et al. both use the treewidth reduction theorem for planar graphs by Adler et al. and the cohomology classification of Schrijver. For improvement, we modify the algorithms and structures used in each stage to be specialized, and well-suited to each other.

ETH-Tight Algorithm for Cycle Packing on Unit Disk Graphs - Shinwoo An (POSTECH)

In this paper, we present an ETH-tight parameterized algorithm for the Cycle Packing problem on unit disk graphs, which runs in $\mathbf{2}^{\boldsymbol{O}(\sqrt{\boldsymbol{k}})}\boldsymbol{n}^{\boldsymbol{O}(\mathbf{1})}$ time, where $\boldsymbol{n}$ denotes the numbers of vertices of an input graph. No ETH-tight algorithm even for the non-parameterized version was known prior to this work. As a tool, we introduce a new recursive decomposition of the plane into regions with $\boldsymbol{O}(\mathbf{1})$ boundary components with respect to a unit disk graph $\boldsymbol{G}$ such that the edges of $\boldsymbol{G}$ crossing the boundary of each region deeply form a small number of cliques. We will call it an annulus cut decomposition for $\boldsymbol{G}$. It seems that it can be used for other problems such as the non-parameterized version of the Odd Cycle Packing problem, and the parameterized versions of the $\boldsymbol{d}$-Cycle Packing and $\mathbf{2}$-Bounded-Degree Vertex Deletion problems on unit disk graphs.

Degrees of Second and Higher-Order Polynomials - Donghyun Lim (KAIST)

Second-order polynomials generalize classical (=first-order) ones in allowing for additional variables that range over functions rather than values. We are motivated by their applications in higher-order computational complexity theory, extending for example discrete classes like P or PSPACE to operators in Analysis. The degree subclassifies ordinary polynomial growth into linear, quadratic, cubic, etc. To similarly classify second-order polynomials, define their degree by structural induction as an 'arctic' first-order polynomial (namely a term/expression over integer variable D and operations $+$ and $\cdot$ and $\max$). This degree turns out to transform nicely under (now two kinds of) polynomial composition. We also establish a normal form and semantic uniqueness for second-order polynomials. Then we define the degree of a third-order polynomial to be an arctic second-order polynomial, and establish its transformation under three kinds of composition.

Improved Learning-Augmented Algorithms for the Multi-Option Ski Rental Problem via Best-Possible Competitive Analysis - Changyeol Lee (Yonsei University)

In this paper, we present improved learning-augmented algorithms for the multi-option ski rental problem. Learning-augmented algorithms take ML predictions as an added part of the input and incorporates these predictions in solving the given problem. Due to their unique strength that combines the power of ML predictions with rigorous performance guarantees, they have been extensively studied in the context of online optimization problems. Even though ski rental problems are one of the canonical problems in the field of online optimization, only deterministic algorithms were previously known for multi-option ski rental, with or without learning augmentation. We present the first randomized learning-augmented algorithm for this problem, surpassing previous performance guarantees given by deterministic algorithms. Our learning-augmented algorithm is based on a new, provably best-possible randomized competitive algorithm for the problem. Our results are further complemented by lower bounds for deterministic and randomized algorithms, and computational experiments evaluating our algorithms' performance improvements.

[Session 4] Invited Talk

Learning to Understand 3D Point Clouds - Jaesik Park (Seoul National University)

3D sensors are getting cheaper and more accurate and widely adopted for intelligent systems. As 3D data has become more prevalent, understanding the shape or structure of the point clouds is a crucial task. This talk addresses recent approaches that apply machine learning techniques for point cloud understanding. The first part of the talk introduces the general characteristics of the point clouds, and the second part introduces the generalized convolution for the fast processing of the point clouds. The last part of the talk introduces the advanced techniques to adopt an attention mechanism for point cloud learning.

[Session 5] Contributed Talks

Smaller Representation of Compiled Regexes - Sicheol Sung (Yonsei University)

We consider the problem of running the regex pattern matching in a space-efficient manner. Given a regex, we suggest a bit-packing scheme for representing a compiled regex in a compressed way, which is its position automaton. Our scheme reduces its representation size further by relying on the homogeneous property of position automata and practical features of regexes. We implement the proposed scheme and evaluate the memory consumption using a practical regex benchmark dataset. Our approach produces a much smaller representation compared to two common FA representations. In addition, experimental results show that our bit-packing regex engine is effective for matching regexes that have large compiled forms, by showing less memory consumption compared to the current state-of-the-art regex engine (RE2).

Weak Inverse Neighborhoods of Languages - Hyunjoon Cheon (Yonsei University)

While the edit-distance neighborhood is useful for approximate pattern matching, it is not suitable for the negative lookahead feature for the practical regex matching engines. This motivates us to introduce a new operation. We define the edit-distance interior operation on a language L to compute the largest subset $I(L)$ of L such that the edit-distance neighborhood of $I(L)$ is in L. In other words, L includes the edit-distance neighborhood of the largest edit-distance interior language. Given an edit-distance value r, we show that the radius-r edit-distance interior operation is a weak inverse of the radius-r edit-distance neighborhood operation, and vice versa. In addition, we demonstrate that regular languages are closed under the edit-distance interior operation whereas context-free languages are not. Then, we characterize the edit-distance interior languages and present a proper hierarchy with respect to the radius of operations. The family of edit-distance interior languages is closed under intersection, but not closed under union, complement and catenation.

M-equivalence of Parikh Matrix over a Ternary Alphabet - Joonghyuk Hahn (Yonsei University)

The Parikh matrix, an extension of the Parikh vector for words, is a fundamental concept in combinatorics on words. We study how the Parikh matrix is used instead of the word itself to characterize the certain class of words, in particular, M-unambiguity. We study the problem of M-unambiguity that identifies words with unique Parikh matrices. While the problem is solved for binary alphabets using the palindromicly amicable relation, it is open for larger alphabets. For a ternary alphabet, we propose substitution rules that establish M-equivalence and solve the problem of M-unambiguity. Our rules build on the principles of the palindromicly amicable relation and enable tracking of the differences of length-3 ordered scattered-factors. We characterize the set of M-unambiguous words and obtain a regular expression for the set.

On the Simon's Congruence Neighborhood of Languages - Sungmin Kim (Yonsei University)

Given an integer k, Simon's congruence relation says that two strings u and v are $\sim k$-congruent if they have the same set of subsequences of length at most k. We extend Simon's congruence to languages. First, we define the Simon's congruence neighborhood of a language L to be a set of strings that have a $\sim k$ -congruent string in L. Next, we define two languages L_1 and L_2 to be $\equiv k$-congruent if both have the same Simon's congruence neighborhood. We prove that it is PSPACE-complete to check $\equiv k$ -congruence of two regular languages and decidable up to recursive languages. Moreover, we tackle the problem of computing the maximum k that makes two given languages $\equiv k$ -congruent. This problem is PSPACE- complete for two regular languages, and undecidable for context-free languages.