Master internship proposal : Lexicographic Enumeration of Maximal Independent Sets

Advisors: Oscar Defrain and Arnaud Labourel
DALGO, LIS, Aix-Marseille Université, CNRS
Mails : oscar.defrain@lis-lab.fr and arnaud.labourel@lis-lab.fr
Websites : http://pageperso.lif.univ-mrs.fr/~oscar.defrain/ and
http://pageperso.lif.univ-mrs.fr/~arnaud.labourel/

1 Context

An independent set is a set of nodes in a graph such that no two of them are adjacent. Finding a maximum independent set, i.e. an independent set of maximal size is a fundamental problem in computer science and was one of the first problems to be shown to be NP-complete [6]. Since producing a maximum independent set is computationally hard, it may be useful to produce independent sets that are only maximal, i.e., independent sets for which there is no node outside the set that may join it. By definition, Maximal Independent Sets (MIS) are dominating sets (set for which every node of the graph is in the set or adjacent to a node of the set). MIS is an important notion in graph theory and other fields. For instance, one possible use of MIS is for networks analysis, such as social science, where a MIS identifies a set of nodes that can be used as a representative sample where each node comes from a different cluster. For some applications, producing just one MIS isn't totally satisfactory, as it may not take into account some constraints that are not fully formalized. In this case, one of the best solution is to produce all the MIS and select the good ones.

Listing MIS is a classical problem in enumeration which dates back to the 70s with the paper of Tskuiyama et al. [11]. This was the first paper to give an algorithm enumerating all maximal independent sets with a *polynomial delay*, i.e., with the times spent by the algorithm before the first output, between any two consecutive outputs, and after the last output bounded by a polynomial in the size of the input. The notion of polynomial delay is useful to study the complexity of numerous enumeration problems since in most of the cases the number of enumerated objects is exponential in the size of the input. Hence, the total time complexity of the enumerating algorithm must be at least exponential in order to produce the output objects. This is the case for MIS, since a graph can contain up to $3^{n/3}$ MIS [9]. Since the paper of Tskuiyama, several papers have improved the bounds of $O(nm\mu)$ for processing time and $O(n+m)$ for memory space, where n, m, and μ are the numbers of vertices, edges, and maximal independent sets of a graph. Some of these papers have considered the general case of all possible graphs [3, 2] while others have focused on specific families of graphs such as claw-free graph [8], interval graphs, circular-arc graphs and chordal graphs [7, 10] and permutation graphs [12].

One important paper for enumeration problems is the paper of Johnson, Yannakakis and Papadimitriou [5] which is the first to explicit state the three main notions of complexity for enumeration problems: output-polynomial time, incremental-polynomial time and polynomial delay. They give in this paper an enumeration algorithm with polynomial delay for listing all independent sets in lexicographic order. Intuitively, imposing a lexicographic order means that, given a total order on all vertices, we first enumerate MIS using the smallest nodes in the order. More formally, one gives numbers from 1 to n to the nodes, giving us an ordered set, and to each MIS, one associates the tuple containing all the numbers of its nodes in increasing order. The lexicographic order of the MIS corresponds to the lexicographic order of the tuples thus constructed.

2 Goals of the internship

The goal of the internship is to study questions left open by the paper of Johnson et al. [5]. Their algorithm uses a memory space exponential in the size of the input and to date the problem is open whether this is really necessary. This question has been left open for thirty-five years and thus may prove to be difficult to solve. However, maybe specific cases such as restricting to a specific family of graphs can be solved. This is already known for trees, for which it is possible to design a stable enumeration algorithm with polynomial delay and linear space in tree size [1]. One can wonder is the same can be said for other families of graphs such as chordal graphs, planar graphs, ...

Another open problem to consider is that of enumerating MIS in reverse lexicographic order for specific families of graphs (start by enumerating the last, then the penultimate and so on). Somewhat surprisingly, this problem has been shown to be computationally hard to solve for general graphs [5] (no polynomial delay algorithm under classical complexity assumptions). For example, we could investigate whether, for certain classes of graphs (such as trees, chordal graphs, planar graphs, ...), there might be a polynomial delay MIS enumeration algorithm for reverse lexicographic order.

In order to attack these problems, one tool that can be used is the meta-theorem of [4] that gives an algorithm with polynomial delay for any output that can be expressed in Monadic Second Order logic. It may give a positive answer for both problems for bounded clique-width and bounded tree-width graphs.

References

[1] Y.H. Chang, J.S. Wang, and R.C.T. Lee. Generating all maximal independent sets on trees in lexicographic order. *Information Sciences*, 76(3):279–296, 1994.

[2] Alessio Conte, Roberto Grossi, Andrea Marino, Takeaki Uno, and Luca Versari. Listing maximal independent sets with minimal space and bounded delay. In *String Processing and Information Retrieval: 24th International Symposium, SPIRE 2017, Palermo, Italy, September 26–29, 2017, Proceedings 24*, pages 144–160. Springer, 2017.

[3] Alessio Conte, Roberto Grossi, Andrea Marino, and Luca Versari. Sublinear-space bounded-delay enumeration for massive network analytics: Maximal cliques. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, volume 148, pages 1–148, 2016.

[4] Bruno Courcelle. Linear delay enumeration and monadic second-order logic. *Discrete Applied Mathematics*, 157(12):2675–2700, 2009. Second Workshop on Graph Classes, Optimization, and Width Parameters.

[5] David S Johnson, Mihalis Yannakakis, and Christos H Papadimitriou. On generating all maximal independent sets. *Information Processing Letters*, 27(3):119–123, 1988.

[6] Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972.

[7] Joseph Y-T Leung. Fast algorithms for generating all maximal independent sets of interval, circular-arc and chordal graphs. *Journal of Algorithms*, 5(1):22–35, 1984.

[8] George J Minty. On maximal independent sets of vertices in claw-free graphs. *Journal of Combinatorial Theory, Series B*, 28(3):284–304, 1980.

[9] John W Moon and Leo Moser. On cliques in graphs. *Israel journal of Mathematics*, 3:23–28, 1965.

[10] Yoshio Okamoto, Takeaki Uno, and Ryuhei Uehara. Counting the number of independent sets in chordal graphs. *Journal of Discrete Algorithms*, 6(2):229–242, 2008. Selected papers from CompBioNets 2004.

[11] Shuji Tsukiyama, Mikio Ide, Hiromu Ariyoshi, and Isao Shirakawa. A new algorithm for generating all the maximal independent sets. *SIAM Journal on Computing*, 6(3):505–517, 1977.

[12] Chang-Wu Yu and G. H. Chen. Generate all maximal independent sets in permutation graphs. *International Journal of Computer Mathematics*, 47:1–8, 1993.