

Graph Parsing Application for Bioinformatics Problems

Grammar-based approach for graph-structured data analysis

Semyon Grigorev¹, Artem Gorokhov¹, Rustam Azimov¹

¹Saint Petersburg State University, JetBrains, St. Petersburg, Russia

E-mail: semen.grigorev@jetbrains.com

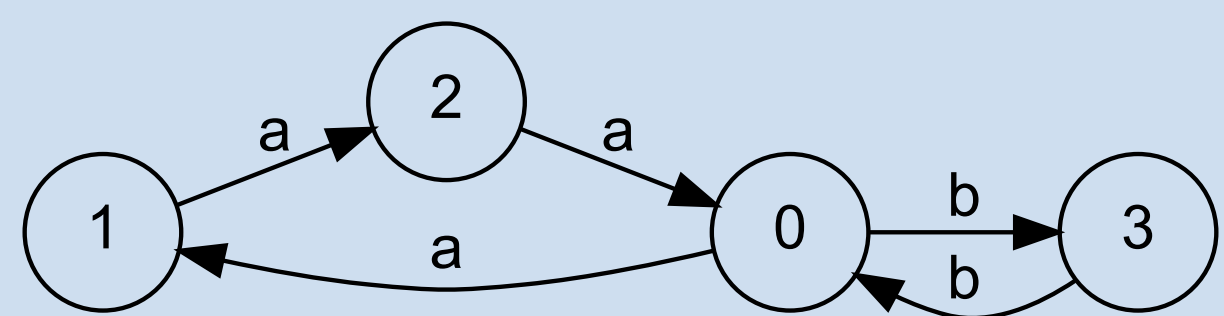


Motivation

Biomedical databases contain huge amounts of rich data which can be represented as a labelled graph. In order to investigate such data, it may be useful to extract connections with specific constraints. One natural way to provide constraints is to specify the language of paths labels which can be done by using of grammars. For example, one can use context-free grammars with productions $\{S \rightarrow aSb; S \rightarrow \varepsilon\}$ to query paths which labels should take form of $ab; aabb; aaabbb; \dots$, or, generally, should belong to the language $L = \{a^n b^n, n \geq 0\}$. This approach is named *context-free path querying* and can be applied to some problems in bioinformatics: metagenomic assemblies analysis, .

Context-free path querying

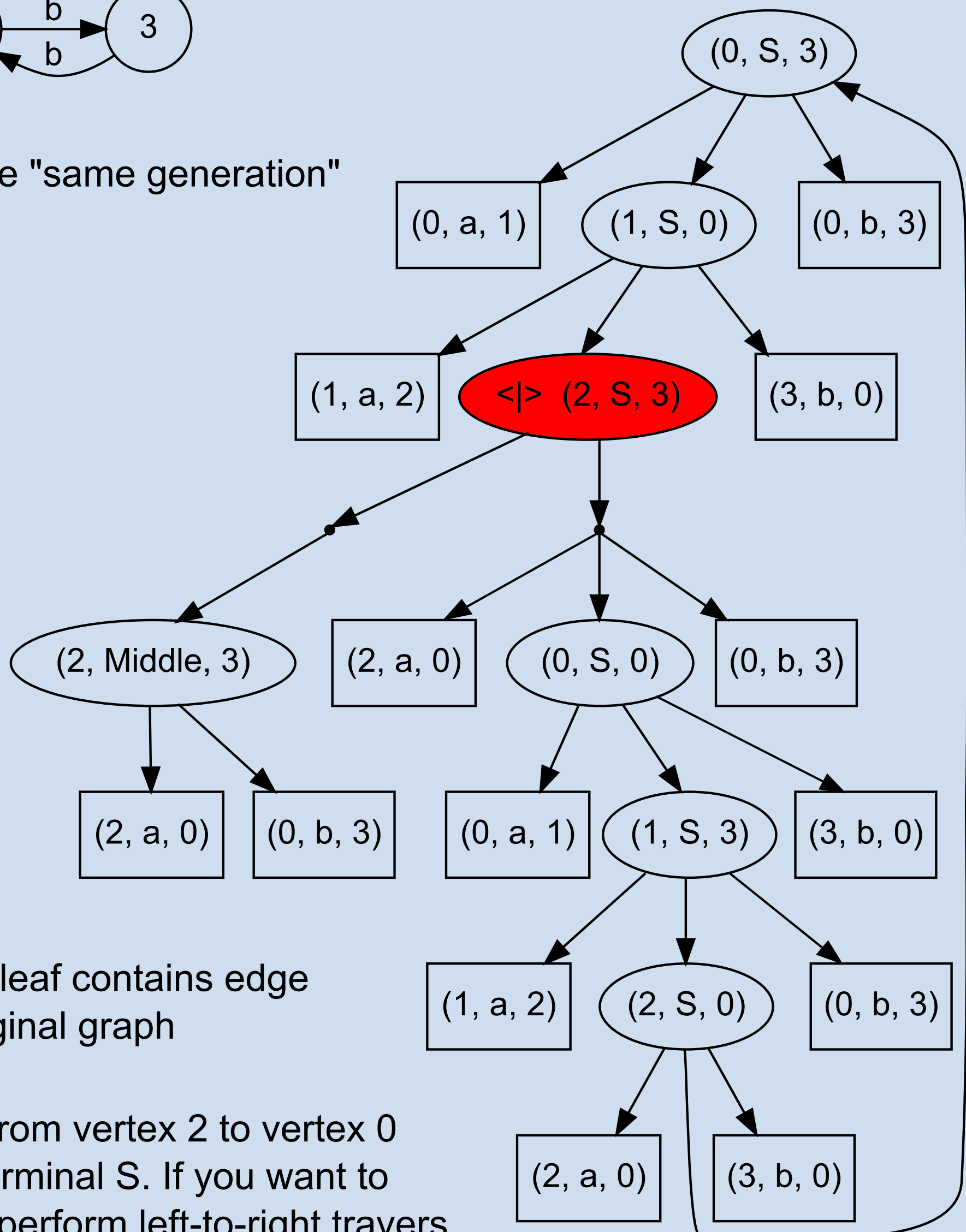
Graph-structured data



Context-free grammar for the "same generation" query with marked middle:

$S \rightarrow a S b$
 $S \rightarrow \text{Middle}$
 $\text{Middle} \rightarrow a b$

Query result: compressed parsing forest for all matched paths. Paths can be extracted by forest traversal.



Each leaf contains edge of original graph

There exists path from vertex 2 to vertex 0 derived from nonterminal S. If you want to get this path, then perform left-to-right travers from this vertex

Possible application

One of the examples is an analysis of graphs where vertices correspond to entities and concepts such as gene or phenotype while edges represent known relationships such as “codes for”, “interacts with”, etc (UniProt dataset [5]). Querying paths with special constraints may shed light upon unknown before links between vertices, forming the basis for new hypotheses.

Acknowledgments

This work is supported by grant from JetBrains Research.

Results

Tasks described above can be solved by using common technique named graph parsing — application of classical parsing techniques for graphs; and we have some experience in this field [1, 2, 3]. Our results solve some problems of existing approaches (such as cycles processing problem in [4]), and provide an ability to use GPGPU and multi-core systems for graph parsing which can be useful for huge biological data analysis.

- Graph parsing framework
- !!!
- !!!

Future Research

- Currently we are working on long subsequences of 16s rRNA reconstruction from metagenomic assembly and on entities connections detection.
- Our aim is to present current results and also to find other applications for this techniques.
- Context-free compressed data processing

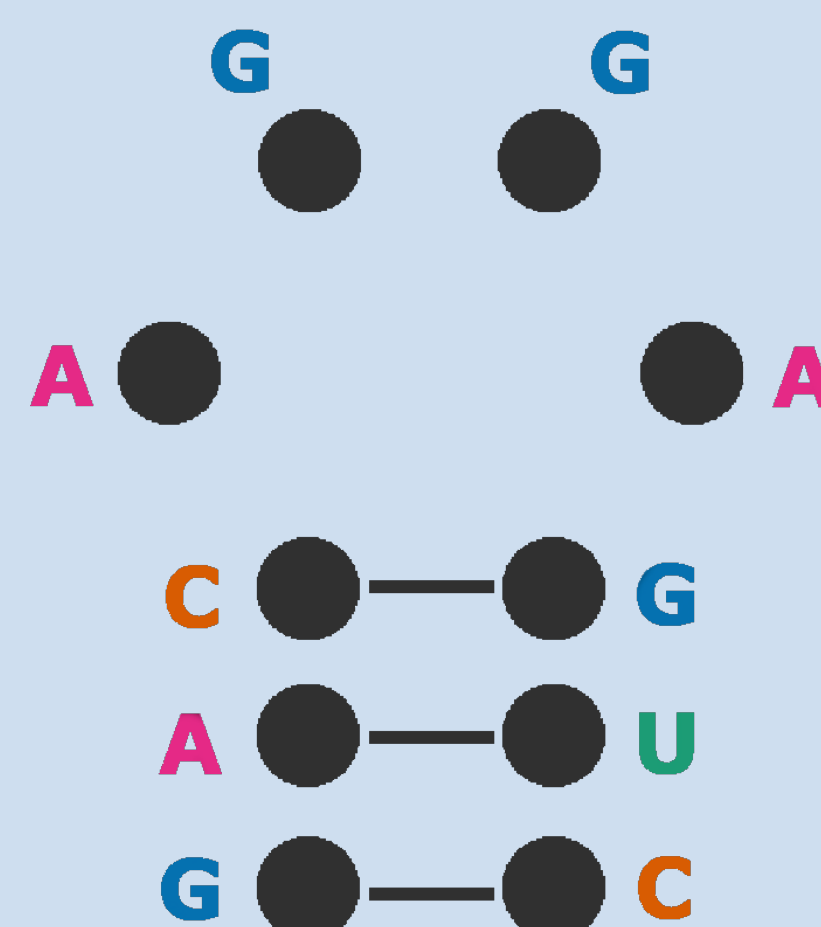
Metagenomic assemblies analysis

Another example of graph structured data is metagenomic assemblies, and the problem is long subsequences detection and reconstruction. Some sequences have specific secondary structure, which can be described in terms of context-free grammar, and this grammar can be used for searching and classification. A lot of research in this area and tools are based on this approach, but most of them are only aimed at linear data processing. Despite the existence of tools for metagenomic assemblies analysis, context-free search in graph-structured assembly is still a challenge.

stem<s> ->

A stem<s> U
| U stem<s> A
| G stem<s> C
| C stem<s> G
| s

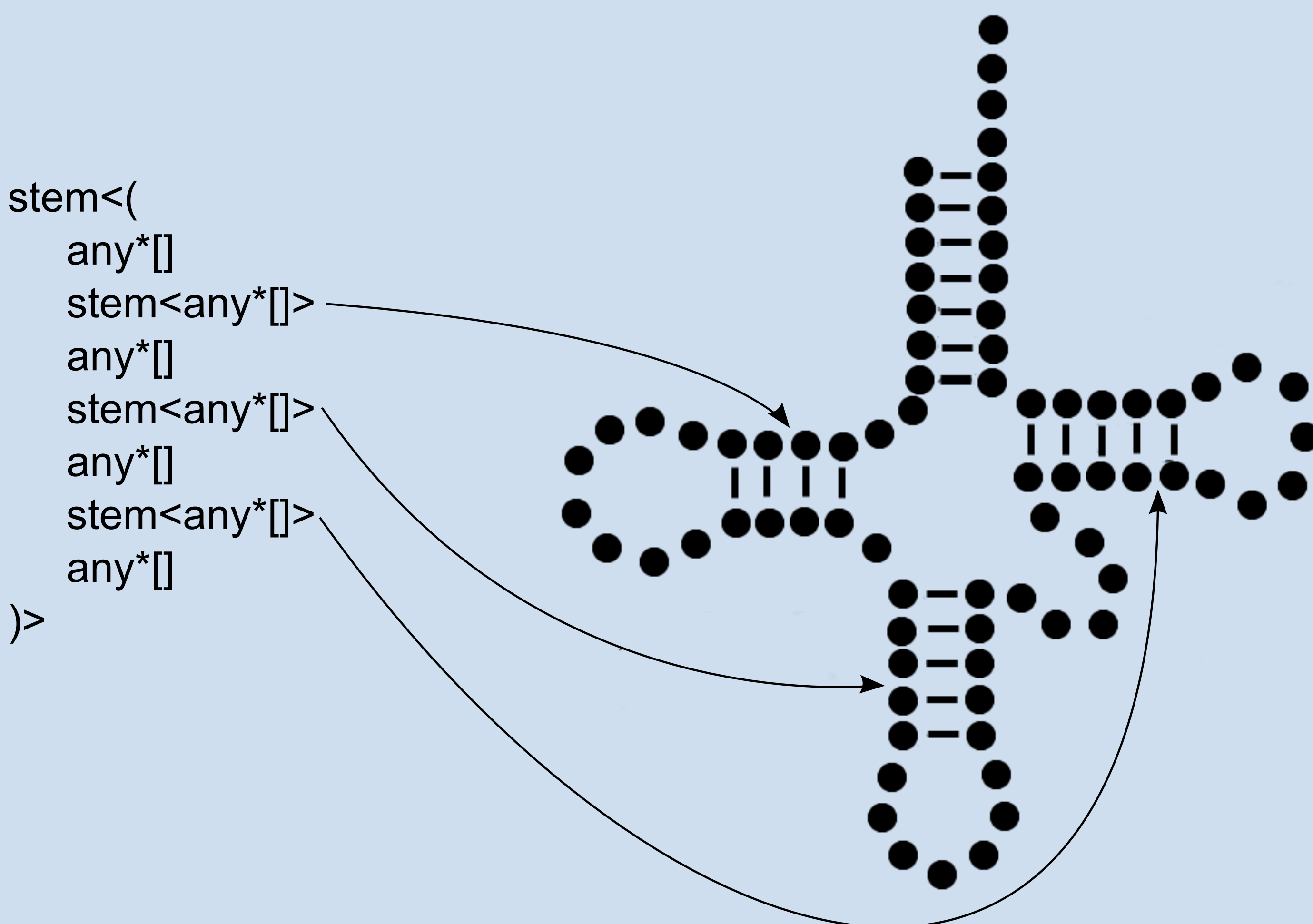
stem<(A G G A)>



stem<

any*[]
stem<any*[]>
any*[]
stem<any*[]>
any*[]
stem<any*[]>
any*[]

>



References

[1] Semyon Grigorev and Anastasiya Ragozina. Context-free path querying with struc-