Synopsis for Bachelorproject

Regular Expression Matching In Genomic Data

Rasmus Haarslev - nkh877
Troels Thomsen - qvw203

Supervisiors: Rasmus Fonseca & Fritz Henglein

23. Februar 2015

Department of Computer Science
University of Copenhagen

1 Problem definition

We wish to determine the possibility of converting sequence analysis patterns used for scan-for-matches[9], into regular expressions[6] and test their efficiency against the KMC[8] engine.

Specifically we wish to solve the following problems:

- Is it possible to programatically convert patterns used by the scan-for-matches program into regular expressions for the KMC engine? If not all patterns used by scan-for-matches then which ones?
- Is it possible to achieve speeds matching or exceeding scan-for-matches with the generated regular expressions and the KMC engine?
- Are there features missing from the KMC engine (such as backtracking), which if they were present would yield better performance in the case of these specific patterns?

1.1 Limits

- We will not attempt to modify the KMC engine.

2 Motivation

Institute for Bioinformatics have a allocated, and are still allocating, a lot of DNA sequencing data. Currently they have around a total of 2 petabytes of data. These sequences of DNA contain a lot of information, but searching through the data currently uses the scan-for-matches program, which while performing very well, is not very user friendly and have some unfortunate limitations when running many consecutive scans, since it performs I/O operations for every run.

Recently Fritz' group have developed a regular expression engine called KMC, which so far have performed five times better than current industry standard engines. Since scan-for-matches outperforms NR-grep[2], we are hoping that optimized regular expressions running on KMC will be able to outperform NR-grep and subsequently scan-for-matches.

If we could achieve a performance improvement over scan-for-matches, it would greatly benefit the bioinformatics team. As such we see this as a chance to make a unique contribution to ongoing and future research projects, while at the same time providing a chance for the KMC team to have their engine tested in a new scenario.

3 Tasks and Schedule

- Develop a standalone Ruby and C application as a solution to the problem.
  - **Product:** A fully functional Ruby/C application, that can translate scan-for-matches patterns into regular expressions, understood by the KMC engine.
  - **Resource demands:** Our contact persons with insight in the KMC engine.
  - **Dependencies:** The KMC engine itself. Test data in the fasta format.
  - **Time demands:**
- Test and analyse the efficiency of our application compared to scan-for-matches.
  - **Product:** An extensive analysis of our application, with possible suggestions for improvements.
  - **Resource demands:**
  - **Dependencies:**
  - **Time demands:**

References

[1] 2008.

[2] Gonzalo Navarro. Nr-grep: A fast and flexible pattern matching tool. `http://www.dcc.uchile.cl/~gnavarro/ps/spe01.pdf`. Visited 18th February 2015.

[3] Dexter Kozen Niels Bjørn Bugge Grathwohl, Fritz Henglein. Infinitary axiomatization of the equational theory of context-free languages. *Proc. 9th Workshop Fixed Points in Computer Science (FICS 2013)*, page 44–55, 2013.

[4] Konstantinos Mamouras Niels Bjørn Bugge Grathwohl, Dexter Kozen. Kat + b! *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, page 44:1–44:10, 2014.

[5] Ulrik Terp Rasmussen Niels Bjørn Bugge Grathwohl, Fritz Henglein. Two-pass greedy regular expression parsing. *Implementation and Application of Automata, volume 7982 of Lecture Notes in Computer Science*, pages 60–71, 2013.

[6] Ulrik Terp Rasmussen Niels Bjørn Bugge Grathwohl, Fritz Henglein. A crash-course in regular expression parsing and regular expressions as types. *Department of Computer Science (DIKU), University of Copenhagen*, pages 1–37, 2014.

[7] Ulrik Terp Rasmussen Niels Bjørn Bugge Grathwohl, Fritz Henglein. Optimally streaming greedy regular expression parsing. *Theoretical Aspects of Computing — ICTAC 2014*, pages 224–240, 2014.

[8] KMC Team. Kleene meets church: Regular expressions and types. `http://www.diku.dk/kmc/`. Visited 18th February 2015.

[9] The SEED Team. Scan for matches. `http://blog.theseed.org/servers/2010/07/scan-for-matches.html`. Visited 18th February 2015.