

Optimizing Boyer Moore Algorithm using Parallel Processing Techniques

S.Y.D. Pathirana, T.S. Ediriweera and D.R.V. L.B. Thambawita

¹*Department of Computer Science and Technology, Uva Wellassa University, Badulla, Sri Lanka*

Although computer processing power, memory and parallel processing capabilities have been significantly improved, most of the algorithms are still not improved based on novel technologies. *Boyer Moore string searching* algorithm is one of a mechanism that uses frequently. Thus, we optimized *Boyer Moore String searching* mechanism using parallel computing mechanism. Existing implementations were analyzed, compared the performance using CPU time with different pattern sizes in same condition. Python parallel version was implemented using three threads. Serial Python code uses main thread for two shifting rules calculation and searching mechanism. Thus, we used two threads for shifting rules calculation and one thread for searching mechanism. Existing implementations used one way searching mechanism to search the pattern in the text field. Then searching mechanism was changed into *bidirectional* searching process using a parallel Python implementation of the Boyer Moore Algorithm with six threads. To optimize the algorithm than bidirectional searching parallel python version, processing logic of the Algorithm was changed and used four threads. Two threads for the sifting rule calculation and after those two threads finished, string text is divided in to two parts and search begins with two threads for each part. Existing serial python version took 165.44 seconds for found correct pattern. However, our first implementation took 163.33 seconds for found correct pattern. Because of bidirectional searching way, the second implementation took 320.94 seconds,. Third implementation took 100.98 seconds for found correct pattern. Measure the time in the same conditions with the same text (T) and same pattern (P). Time is depending on text (T) size, pattern (P) size and machine type that we use to run the codes. Final implementation is bidirectional searching parallel python version with four threads. It is 38.96% effective than existing serial python version.

Keywords: Boyer Moore String searching, bidirectional