Volume 2 No.7, July 2012 ISSN 2224-3577

International Journal of Science and Technology

http://www.ejournalofsciences.org

Parallel Search Using KMP Algorithm in Arabic String

Ibrahim M. Abu-Zaid, Emad Kh. El-Rayyes
Graduate Studies
Faculty of Information Technology
The Islamic University of Gaza

ABSTRACT

In our paper we will try to implement the parallel techniques in Knuth–Morris–Pratt string searching algorithm (KMP algorithm) for search about Arabic language text, Before that we will try to implement a Pre-processing processes in Arabic text (stemming Arabic text) to prepare the text to using it for KMP search algorithm in parallelism .the main goal of our paper to reduce the search time when we implement the KMP algorithm in a parallel mechanism, We describe the steps for our propose model in paper sections.

Keywords: *KMP algorithm, MPI, parallelism search, pre-process, stemming Arabic text*

I. INTRODUCTION

An algorithm or concurrent algorithm, as opposed to a traditional sequential (or serial) algorithm, is an algorithm which can be executed a piece at a time on many different processing devices, and then put back together again at the end to get the correct result.

Parallel algorithms are valuable because of substantial improvements in multiprocessing systems and the rise of multi-core processors. In general, it is easier to construct a computer with a single fast processor than one with many slow processors with the same throughput. But processor speed is increased primarily by shrinking the circuitry, and modern processors are pushing physical size and heat limits. These twin barriers have flipped the equation, making multiprocessing practical even for small systems.

Parallel Search, also known as Multithreaded Search or SMP Search Symmetric multiprocessing, is a way to increase search speed by using additional processors[4].

The string is a linear table which has a wide range of applications in computer application system, such as text editing, information retrieval, natural language translation and so on. We always need to test a specified character string whether in another string in these applications. Suppose the string s and the string t are two strings, Pattern matching algorithm is applied extensively in text editing, information retrieval, spell checking, dictionary-based language translation, WWW search engines, computer virus signature matching, data compression, DNA sequence matching and other computer application system[1].

so, in section II we will present the related work about our project and any related papers that is interested in our approach, in section III we will describe and present the stemming Arabic text concept’s and algorithm that is used in our project, in section IV we will describe and present the KMP algorithm in parallel technique, in the fifth section we will describe our model approach, finally we will discussion and conclusion the results.

II. RELATED WORKS

In [1] the researchers say the tradition pattern matching algorithm need backtrack and compare repeatedly, so that affects efficiency of algorithm. Knuth and others put forward KMP algorithm in order to promote efficiency of the pattern matching. Parallel KMP algorithm based on MPI is provided in his paper, which can get higher efficiency.

In [2] the researchers say Pattern matching is often used in intrusion detection system. On the basis of analyzing four kinds of typical pattern matching algorithms, an improved algorithm based on BMHS is presented. The algorithm uses a matching way that is from right to left. In the algorithm, the movement distance of a pattern is decided by the greater one of two movement value which is calculated by two characters. Experiments show that the improved algorithm could reduce the times of comparing and moving and it could enhance the efficiency of pattern matching.

In [3] the researchers proposed an algorithm to handle weak, eliminated-longvowel, hamzated, and geminated words since the linguistic approach does not handle such cases and a reasonably large portion of Arabic words in texts are irregular. The accuracy of the extracted roots is determined by comparing them with a predefined list of 5,405 trilateral and quadrilateral roots.

In [4] the researchers propose a new stemming technique that tries to determine the stem of a word representing the semantic core of this word according to Arabic morphology. This method is compared to a commonly used light stemming technique which truncates a word by simple rules.

III. ARABIC LIGHT STEMMER

In general, word stemming is one of the most important factors that affect the performance of information retrieval systems. The optimization issues of Arabic light stemming algorithm as a main component in natural language processing and information retrieval for Arabic language are based on root-pattern schemes. Since Arabic language is a highly inflected language and has a complex morphological structure than English, it requires superior stemming algorithms for effective information retrieval.[5].

The first thing the stemmer does is remove the longest suffix and the longest prefix. It then matches the remaining word with the verbal and noun patterns, to extract the root.

We can see example in table 1 below:

Table 1: Example of streaming

No.	Word before stemming	Word after stemming
1	أحمد	حمد
2	السيد	سيد
3	إمام	مام

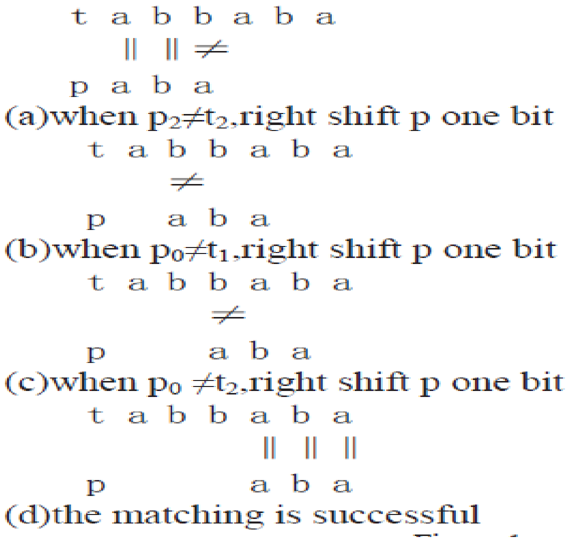
IV. KMP SEARCH ALGORITHM-PARALLEL

The goal of search is to find a particular object in this collection or to recognize that the object does not exist in the collection. Often the objects have key values on which one searches and data values which correspond to the information one wishes to retrieve once an object is found.

The algorithm was conceived in 1974 by Donald Knuth and Vaughan Pratt, and independently by James H. Morris. The three published it jointly in 1977.

The Knuth–Morris–Pratt string searching algorithm (or KMP algorithm) searches for occurrences of a "word" W within a main "text string" S by employing the observation that when a mismatch occurs, the word itself embodies sufficient information to determine where the next match could begin, thus bypassing re-examination of previously matched characters.

For example, we can see the mechanism of KMP algorithm work in the following figure:

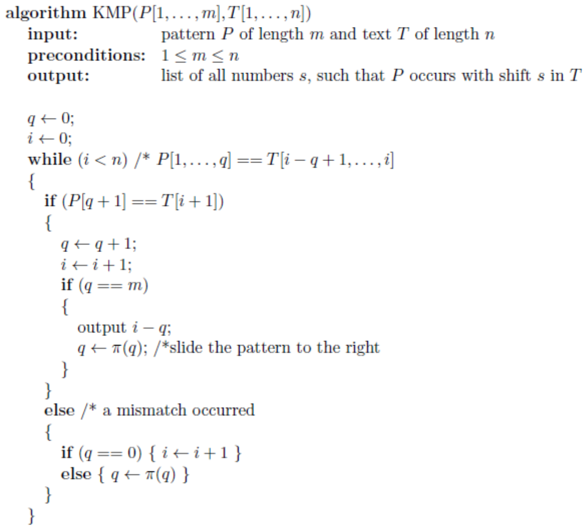


t a b b a b a
|| || ≠
p a b a
(a)when p2≠t2,right shift p one bit
t a b b a b a
≠
p a b a
(b)when p0≠t1,right shift p one bit
t a b b a b a
≠
p a b a
(c)when p0 ≠t2,right shift p one bit
t a b b a b a
|| || ||
p a b a
(d)the matching is successful

Figure 1: KMP Example

In Figure 1, we see that p0≠t0,p1=t1,p2≠t2 , and p0≠p1, and we have p0≠t1 from p0≠p1. So the comparative of one right shift character of p in (b) is also unequal; By the fact that p0=p2 , we have that p0≠t2 . Hence, the comparative of one right shift character of p in (c) is also unequal. Therefore, from (a) we can be directly right shift three character to jump to (d), start the comparison from p0 and t3. Matching is completed soon if we can find p2 i≠t2 in (a), we can directly jump to (d),and start the comparison from p0 and t3, so that we eliminate the backtracking.[1]

We can see the KMP algorithm in the following figure:



```
algorithm KMP($P[1,\dots,m]$,$T[1,\dots,n]$)
   input:            pattern $P$ of length $m$ and text $T$ of length $n$
   preconditions:    $1 \le m \le n$
   output:           list of all numbers $s$, such that $P$ occurs with shift $s$ in $T$

   $q \leftarrow 0$;
   $i \leftarrow 0$;
   while ($i < n$) /* $P[1,\dots,q] == T[i-q+1,\dots,i]$
   {
     if ($P[q+1] == T[i+1]$)
     {
       $q \leftarrow q+1$;
       $i \leftarrow i+1$;
       if ($q == m$)
       {
         output $i - q$;
         $q \leftarrow \pi(q)$; /*slide the pattern to the right
       }
     }
     else /* a mismatch occurred
     {
       if ($q == 0$) { $i \leftarrow i+1$ }
       else { $q \leftarrow \pi(q)$ }
     }
   }
```

Figure 2: KMP algorithm

So, we will try to implement this algorithm by using parallel approach, by decomposing the data in to partitions and search about the item in all partitions as a parallelism technique .each partition as a task and execution the task in one processor.

V. THE PROPOSE MODEL

In our model we have Arabic text to search about it, that's mean we have special case of words in Arabic text, So, we try to apply prepossessing technique in this text (stemming Arabic) on the words.

We can show in the figure below the text which are we find as input to the model, this text we will enter them to the stemming Arabic technique ,the output is words without "weak letter".

The second processes in using KMP search algorithm in parallel approach to search about the result from the first process (stemming) in a paragraph.

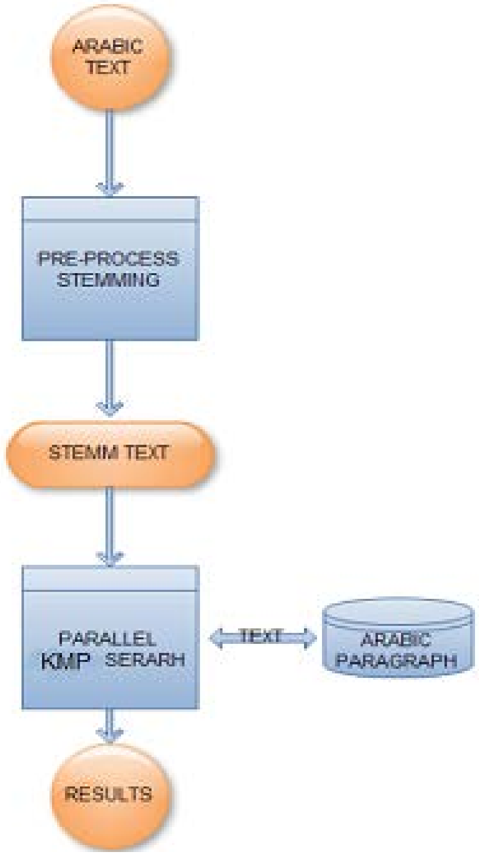


Figure 3: Proposed Model

Before solving specific problems, we need to declare some conditions:

1) The length of the string p should be less than or equal to the length of string t. That is: Length(p)≤Length(t)
2) The length of string which is distributed to each process should be larger than the length of p.

A- Collection Data

In our experimental we used data set into the Degrees of successful students the secondary school in Gaza for years from 2008 to 2011 we collected the student data from the IT department at Al-Azhar University of Gaza , the data set have about 140500 record in text file , and convert the extension type to text file.

B- Pre-process Arabic text

In Arabic Language some problem because have special case word stemming is one of the most important factors that affect the performance of information retrieval systems , so we used some tools in Java programming languages to reprocess the special case in Arabic text and help in searching to matching between the same two words, we can see the example in table 2 below :

Table 2

No.	Word before stemming	Word after stemming
1	أحمد أشرف الريس	حمد شرف ريس
2	أسعد وهيب بدوي	سعد هيب بدو
3	إبراهيم محمد أبو زيد	براهيم محمد بوزيد

So we can see in the results the preprocessing remove some string from the word because that string we can write in different cases like أشرف اشرف and that the different in mechanism writing do some conflict in searching and the preprocessing in very important in searching approach .

C- Decomposition Data

We will to apply the input decomposition data technique because we have large data set, and we will decomposition the data set into some partitions to processors (multi-clusters), and the mechanism in decomposition data set over the sum of processors in the same size to be efficient in the main processing.

D- The Experimental

We have two main steps, in our the experimental, first step preprocessing the goal Arabic string for searching in some data set, because the Arabic languages have special case in the write some words and remove that "weak letter" from the words from the pattern, after the preprocessing and the string is ready to the searching we go to the second step is used the parallel KMP searching algorithm in our experimental and want to try the parallel KMP searching algorithm in group of computers "computers clusters " with shared data set to searching in the same date and we will to try in different cluster to improve what the best case in the searching of cluster .

We applied the code of the KMP algorithm by using C++ Microsoft visual studio and MPI component tool.

Finally we implement our model in MPI program by increasing the number of clusters (multi computers) by decomposed the data in to multiple computers (2,3,4,5,6,7) and we will present the varies time of execution our experimental as the following table 3 below :

Table 3: The Results of the Experimental

Test text file :pdata.txt file size :3261 KB NO. of Records:140500			
No. of clusters	Time	Speed Up	Efficiency
Serial Algorithm (1)	0.2464		
2 Clusters	0.2322	1.06	0.53
3 Clusters	0.4240	0.58	0.19
4 Clusters	0.4304	0.57	0.1425
5 Clusters	0.4246	0.58	0.116
6 Clusters	0.4389	0.56	0.0933
7 Clusters	0.6273	0.39	0.0557

And we can present the relationship between the number of clusters and efficiency in each operation, as the below figure 4:

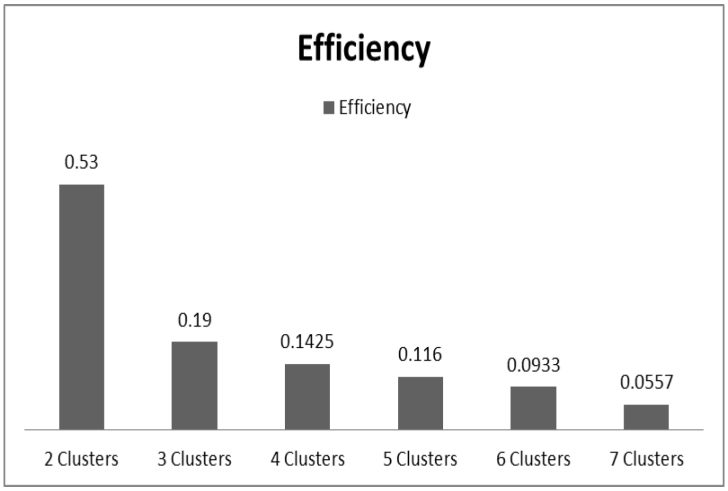


Figure 4

VI. DISCUSSION AND CONCLUSION

We implement the KMP algorithm as a serial and as a parallel mechanism, and we present the time of execution of serial and the time for execution and speed up and the efficiency of parallel implement.

We can see in the previews table, when we implement the search in multi clusters we find high efficiency when we decomposition the data in to two clusters , but when we increase the clusters we can see the efficiency is reduce because the communications is high.

Finally we can say the parallel KMP searching algorithms
is a best way to reduce the searching time when we using a big Arabic data base.

ACKNOWLEDGMENT

We would like to express our gratitude to all those who help us to complete this research work and we hope that this research work will be a good reference to other researches in future, So we are thankful to Eng.Dr.Ribhi Baraka for supervising our work, M.Sc. Manar Fayyad, M.Sc mohammed abu jaser, M.Sc Zakaria saqalla and M.Sc iyad elsami, the lab supervisor.

REFERENCES

[1] Yinchuan Ningxia," Parallel Research on KMP Algorithm",2011.

[2] Yuting Han, Guoai Xu ," Improved Algorithm of Pattern Matching based on BMHS",2010.

[3] May Y. AI-Nashashibi, D. Neagu,Ali A. Yaghi,” An improved root extraction technique for Arabic words”,2010.

[4] Youssef Kadri & Jian-Yun Nie,” Effective Stemming for Arabic Information Retrieval”,2004.

[5] Hayder K. Al Ameed, Shaikha O. Al Ketbi, Amna A. Al Kaabi,” ARABIC LIGHT STEMMER: ANEW ENHANCED APPROACH”,2004.

AUTHORS PROFILE

Ibrahim M. Abu-Zaid received his B.Sc. in Information System Computers from the Al-Quds Open University in Palestine in 2006; currently he is pursuing his M.Sc. in Information Technology from the Islamic university of Gaza. He works as administrator networks. He has over 7 years of experience in the field of administrator networks. He holds different technical certificates: MCSE, MCITP, and CCNA.

Emad KH. Elrayyes received his B.Sc. in Information System Computers from the Al-Quds Open University in Palestine in 2007; currently he is pursuing his M.Sc. in Information Technology from the Islamic university of Gaza. He works as web developer He has over 6 years of experience in the field of web developer and design. He holds different technical certificates: MCITP, MCT, MCSE, OCP, and CCNA.