Analysis of Energy Consumption of Sorting Algorithms on Smartphones

Murlidhar Verma[1], K.R.Chowdhary[2]

[1]*Ph.D. Scholar, Department of Computer Science & Engineering, MBM Engg. College, JNV University, Jodhpur, India, ermurlisoni@gmail.com*
[2]*Prof. & Head(Retd.), Department of Computer Science & Engineering, MBM Engg. College, JNV University, Jodhpur, India, kr.chowdhary@acm.org*

Abstract: Despite the improvement in hardware and technology, Limited Battery power in Smartphone is still challenging task. Battery lifetime of mobile phones is needed to be improved. As compare to CPU power, HDD capacity, RAM and other hardware components the improvement in battery power consumption and lifetime in the past years is very low. The battery power of a Smartphone may be prolonged by optimizing hardware or software. This paper presents an approach for saving energy through software by choosing the appropriate sorting algorithm.

Energy consumption of sorting algorithms like Bubble sort, Insertion sort, Quick sort, Selection sort and Merge sort were measured. The results show that Quick sort is the most energy efficient sorting method on average cases. Bubble sort is the most energy consuming algorithm. Insertion sort is also energy efficient but as the number of input data items increase after a certain value Merge sort outperforms the Insertion sort. This work provides the general guideline to select appropriate sorting algorithm in development of energy efficient Smartphone application. A new term *Energy Complexity* is also introduced in this paper.

1. Introduction

Smartphone has become the most important part of everyone's life. Smartphone has influenced almost every part of the life. It has enhanced and advanced professional and personal life. It has become a vital part of the life. There is no limit what one can do with a Smartphone these days. There is an App for almost anything [6]. One can do online ticket booking, mobile banking, checking of the nearest coffee shop, reading the online newspaper, e-books, finding a path on maps etc.

Today's Smartphones have a higher configuration in terms of CPU power, memory capacity, and storage but their functionality is limited by low battery power capacity. The battery power of a Smartphone may be prolonged by optimizing hardware or software. Many efforts have been made by Researchers to reduce energy consumption by optimizing hardware [3][7][8]. However, energy consumption may be optimized by using software-based approach i.e. by using an appropriate algorithm. This paper presents an approach for energy saving through software by choosing the appropriate sorting algorithm. Here sorting algorithm is used because sorting algorithm efficiency is relevant to almost all applications. Sorting algorithms play a major role regarding performance and energy consumption of mobile devices.

Energy consumption of five sorting algorithms like Bubble sort, Insertion sort, Quick sort, Selection sort and Merge sort were measured. Previous work by [2] showed that Insertion sort is the most energy efficient sorting algorithm when it comes to saving energy. But our result shows that Quick sort is the most energy efficient sorting method on average cases. Bubble sort is the most energy consuming algorithm. This work provides a general guideline to select appropriate sorting algorithm in development of energy efficient Smartphone application.

The remainder of this paper is structured as follows: Next section II gives an overview of sorting algorithms, the experiment and results are presented in section III, Section IV summarizes and concludes the paper.

2. Sorting Algorithms

This section briefly describes the sorting algorithms and their time and space complexities. Almost all the applications use sorting algorithms. For the mobile system application development, there is a requirement of energy efficient sorting algorithms because these are battery-powered devices and battery power is limited. Apart from time and space complexities the energy requirement of any sorting algorithm is not properly known yet. This work tries to find out energy consumptions of popular sorting algorithms like Bubble, Selection, Insertion and Quick sort [1].

2.1 Bubble sort

Bubble sort is easy and simple to implement. In this sorting method, each pair of adjacent elements is compared and swapped if they are in the wrong order. Each element is compared with its adjacent element, If the first element is larger than the second one then the position of the elements are interchanged, otherwise, it is not changed. The same process is repeated for all the elements of the List. In the one scan of the List, one element is placed at its correct position. This sorting technique is called bubble sort because items in the List move up into the correct order like bubbles rising to the surface. For the list of *n* elements *n-1* passes are required and in each pass, there is *n-pass* number of comparisons needed to sort the List. Worst-case and average case complexity is $O(n^2)$.

2.2 Insertion sort

Insertion sort works like playing bridge game of cards. In this sorting technique, a list of elements is sorted by inserting an element into an existing sorted list [5]. For example, There is one array A with N values A[1], A[2],…A[N]. The Insertion sort algorithm scans it from A[1] to A[N], inserting each element A[J] into its proper position in previously sorted subarray A[1], A[2], A[J-1]. It has the average time complexity of $O(n^2)$. It is also known as In-place sorting algorithm because it requires a constant amount of extra memory space $O(1)$.

2.3 Quick sort

Quick sort is also known as partition-exchange sort. It is based on divide and conquers technique. In this, a large list of numbers is partitioned into two sub-lists one of which contains values smaller than the specified value, say pivot, based on which the partition is made and another sub-list holds values greater than the pivot value. Quick sort calls itself recursively to sort the list. The average case time complexity of the Quick sort is $O(n \log n)$. In the worst case, its time complexity is $O(n^2)$. The worst case occurs when the list is already sorted.

2.4 Selection sort

Selection sort is an in-place comparison-based algorithm. This algorithm searches the minimum value in the list and exchanges it with the first element. The same process is repeated for the remaining values in the list to sort the final list. The time complexity of the selection sort is $O(n^2)$.

2.5 Merge sort

Merge sort was invented by John von Neumann and belongs to the family of comparison-based sorting. It divides input list into two halves, calls itself for the two halves and then merges the two sorted halves. Merge sort has an average and worst-case performance of $O(n \log n)$. But, Merge sort requires additional $O(n)$ memory.

3. Experimental Setup and Analysis

Equations Energy consumption of five sorting algorithms: Bubble sort, Insertion sort, Quick sort, Selection sort and Merge sort were measured. To measure the energy consumption of each algorithm five different Apps were developed for each sorting algorithm.

Power Tutor V.1.4 App was used to measure the energy consumption of each sorting algorithm. Once Sorting App was started there was no human interaction with the Smartphone. The experimental setup is as shown in Fig. 2. To find accurate result each algorithm was run 10 times for single data set and average energy consumption was noted. Samsung Galaxy J5 prime (SM-G570F) and Samsung GT-I8262 smartphones were used to run and measure the energy consumption of sorting algorithms. Almost same trends were obtained on both the smartphones. 10 different datasets of random numbers were used to find the results correctly.

Random numbers of different size as shown in TABLE-I were generated and the same numbers were input to the different sorting algorithms and energy consumption was measured. The graph in Fig.1. shows trends of experiments performed on Samsung J5 Prime.

From Graph of Fig.1 and Table I, It is clear that Quick Sort is the most energy efficient algorithm whereas Bubble Sort consumes the highest amount of energy. Insertion sort is also energy efficient but as a number of input data item increases after a certain value Merge sort outperforms Insertion sort.

3.1. Discussion

Even though there are many methods to measure energy consumption in a Smartphone and mobile devices [4][5]. In this work, App based approach was used for simplicity. *Power Tutor V.1.4* App is used to find the energy consumption of each sorting App. This gives a relatively correct result because the same App is used to measure the energy consumption of all the sorting Apps and it was repeated 10 times. The Same randomly generated data set was input to all the sorting Apps.

Time and space consumed by an algorithm are two main measures of the efficiency. As the world of computers is moving towards the world of smartphones and wireless devices. There is a need of new measure known as ***Energy or Power Complexity*** of an algorithm. It may be the energy consumed by the algorithm.

Work by [2] showed that Insertion sort is the most optimal sorting algorithm when it comes to saving energy. But our result shows that quick sort is the most energy efficient. Their work was performed on ATMega128 micro controller having external SRAM, running on a STK500/501 board. But, The experiments of our work were executed on latest smartphones having multiple cores and larger RAM. It is assumed that recursive calls will consume more energy but the task is uniformly distributed to multiple cores. These cores run on low frequency and thus consume less energy. Thus from our results, Quick sort is the most energy efficient and Bubble sort is the most energy consuming sorting algorithm. Insertion sort is also energy efficient but as the input data set size increases Merge sort performs better than Insertion sort as shown in Fig.1.

Electronic copy available at: https://ssrn.com/abstract=3168550

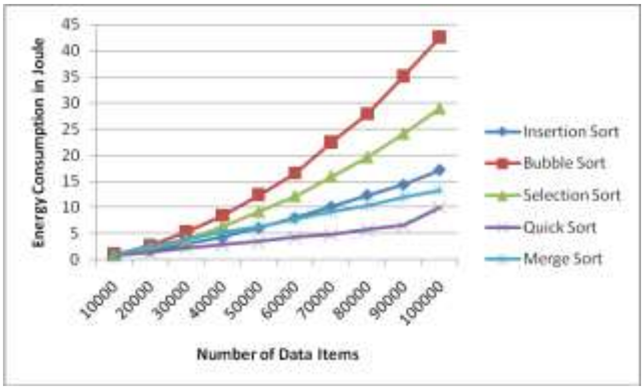

Fig. 1 - Energy Consumption of different Sorting Algorithms

Table 1 –Energy Consumption in Joule for different Data size

Data Size	Insertion Sort	Bubble Sort	Selection Sort	Quick Sort	Merge Sort
10000	0.755	0.99	0.848	0.696	0.989
20000	1.78	2.7	2.37	1.37	2.56
30000	3.03	5.28	3.86	2.1	3.83
40000	4.3	8.43	6.42	2.75	5.3
50000	6.03	12.33	9.23	3.38	6.3
60000	7.98	16.54	12.2	4.2	7.83
70000	10.1	22.45	15.98	4.8	9.17
80000	12.35	27.87	19.66	5.67	10.4
90000	14.32	35.08	24.18	6.52	12
100000	17.14	42.57	29.04	9.96	13.3

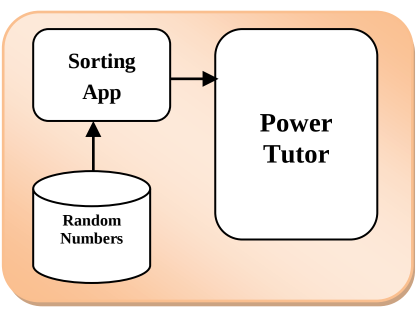

Fig. 2 - Experimental Setup

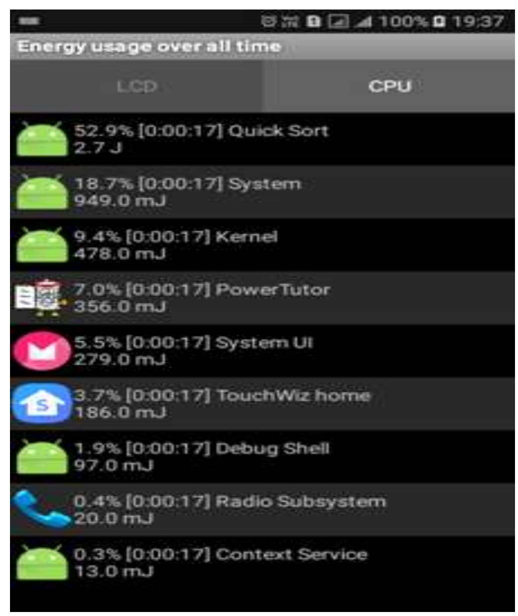

Fig. 3 – Power Tutor running on Galaxy J5 Prime

4. Conclusion and Summary

Reducing the energy consumption is directly related to the more operating time of a Smartphone. Developing energy efficient applications is really a challenging task. By using appropriate algorithm energy consumption

Electronic copy available at: https://ssrn.com/abstract=3168550

may be reduced.

Previous work shows Insertion sort to be the best. But, this work shows that quick sort is the most energy efficient algorithm on average case. Bubble sort is the most energy consuming. Even for the larger input data size Merge sort outperforms Insertion sort.

This work provides general guild line for selecting appropriate sorting algorithm in designing of energy efficient applications. Sorting algorithms were used because these are used in almost all the applications. A new term *Energy Complexity* was pointed out in this work. In future, the work can be further extended for other algorithms and to find concrete mathematical notation for *Energy Complexity*.

Acknowledgements

I would like to express my deep sense of gratitude and wholehearted thanks to my guide Dr. K.R. Chowdhary Sir, for giving his valuable guidance, inspiration and encouragement to complete this paper. Without his support, guidance, and assistance this paper would not have been completed.

References

Aaron Tenenbaum, Yedidyah Langsam and Moshe Augenstein (2012). Data Stucture using 'C'. Pearson Education India.

Christian Bunse, Hagen Höpfner, Essam Mansour,Suman Roychoudhary (2009). Exploring the Energy Consumption of Data Sorting Algorithms in Embedded and Mobile Environments. Tenth International Conference on Mobile Data Management Systems, Services and Middleware (pp. 600–607).

J. Chen and L.Thiele (2008). Expected system energy consumption minimisation in leakage-aware DVS systems. In Proc. the of the 13th ACM ISLPED (pp. 315–320).

Lide Zhang, et al.(2010). Accurate Online Power Estimation and Automatic Battery Behaviour based Power Model Generation for Smartphones. CODES/ISSS'10 Proceedingd of the Eighth IEEE/ACM/IFIO International Conference on Hardware and Software Sodesign and Sytem Synthesis (pp. 105-114).

M. Dong and L. Zhong (2010). Sesame: A self-constructive virtual power meter for battery-powered mobile systems. Tech. Rep.

Murlidhar Verma, and K.R. Chowdhary (2015). Smartphone as a Tool for Different Applications", International Journal of Innovative Research in Science & Technology(IJIRST), vol. 2, Issue 7(pp. 89–92).

N. Liveris, H. Zhou and P.Banerjee (2008). A dynamic programming algorithm for reducing the energy consumption of pipelined system-level streaming applications. In Proc. of the IEEE conference on Asia and South Pacific design automation (pp. 42–48).

T. Tuan, S. Kao, A. Rahman, S. Das, and S. Trimberger. A 90 nm low-power FPGA for battery-powered applications. In Proc. of the ACM/SIGDA 14th intern. symposium on field programmable gate arrays (pp. 3–11).

Electronic copy available at: https://ssrn.com/abstract=3168550