

Review on Energy Consumption of the Sorting Algorithms and Involvement of Programming Language

Shweta Tiwari¹ ShashankShekhar Tiwari² Shivendu Mishra³ ¹Assistant Professor,Rajkiya Engineering College, Ambedkar Nagar shwetatiwari08aug@gmail.com ²Assistant Professor,Rajkiya Engineering College, Ambedkar Nagar sst.010.cs@gmail.com ³Assistant Professor,Rajkiya Engineering College, Ambedkar Nagar shivendu0584@gmail.com

Article Info Volume 82 Page Number: 7033 - 7041 Publication Issue: January February 2020

Article History

Article Received: 18 May 2019 Revised: 14 July 2019 Accepted: 22 December 2019 Publication: 03 February 2020

Abstract:

In the era of the computer revolution, the computer is making us completely digital dependent on digital gadgets. Reducing energy consumption by using optimized programming techniques and energy efficiency algorithms, in order to maximize the uptime of various battery operated devices has been an interesting research topic. Many years ago, computing was performed on the basis of lesser space and quicker calculations were performed inconsiderable amount of time. Recent past energy consumption is one of the major objectives for selecting computing algorithms to achieve power efficiency and to fulfill the objective of Green Computing. This study mainly concentrates on the analysis of energy consumption of some standard sorting algorithms implementation in different programming languages, which are not only used in various devices directly but also used very frequently as a part of various other complex algorithms.

Keywords: Algorithm, Programming Language, Energy Resource, Energy Efficiency, Energy consumption, Green Computing.

I. INTRODUCTION

The main reason for the increased supply in the energy resource is continuous growth in the populations. Today, electricity is counted as the basic need for day to day survival of the people. With the increase in the time people are being dependent on computers and smart gadget. At present senario, 80% to 90% of the work/task is done on the computer. Big Data is a special example of a lot of examples. When it comes to topic Energy Consumption, all the developers of the computer world are the most focused. As the speed of the system are high and the files in the system store less space etc. Embedded system, smartphone in which battery consumption is a real problem [8]. Over the last several years software developer has found the study that the energy consumption in the system is as responsive as the hardware is the same software. Research is continuously working on energy optimization of the system. Each computer system has a different energy requirement.So what we need to do in the software sense that can reduce the energy consumption of the computer system? Algorithms also have a great effect on energy consumption. We can give two main areas of energy consumption.

• Sorting algorithms have been omnipresent used in different applications nowadays. As the size of the data increases, the energy consumption increases as well under the same system



requirement.

• Every programming languages have many advantage-disadvantage and characteristics that can affect the energy consumption and run-time performance of programming tasks implemented in such languages.

The main purposed of this paper to analyze and reduce energy consumption by implementing the sorting algorithm in better-suited programming language. There will also be a good effect on green computing. There will also be a good effect on green computing.

2 ENERGY CONSUMPTION EVALUATION SOFTWARE

2.1 JOULEMETER: This software can be used to estimate the energy usage of desktops, servers, virtual machines and also individual software. This software measures the hardware resources such as CPU, Harddisk, RAM, monitor, etc. getting used. This software changes the resource usage to actual power usage supported mechanically learned realistic power models. It focuses on following aspects of energy optimization in figure1:



Figure 1Joulemeter aspects

Modeling: Joulemeter permits modeling of the power management impact of parts like the CPU, Monitor, RAM, and Harddisk on total power use [1] (Goraczko, 2010). Joulemeter's modeling technology works in three areas which are shown in the figure2. Some of these power measurements don't seem to be doable in hardware since software package parts like VMs don't have one wire provision their power wherever a hardware electric meter could also be put in.



Figure 2 working of modeling

Optimization: we can use the transparency provided by this software modeling techniques to boost energy provisioning and consumer prices in numerous situations such as battery operated machines, High-performance computing, and knowledge centers.



Figure 2 optimization focus aspect

Table 1: Joulemeter Software details

PROPERTIES	INFORMATION
Publisher	Microsoft
Publisher Web Site	http://www.microsoft.com/
Release Date	September 29, 2011
Date Added	September 29, 2011
Version	1.2
Operating Systems	Windows/7
License Model	Free
File size	922kbs

2.2 LIGHTS-OUT: Lights-out may be an IT resolution software package. It's permitted to run your NAS and computers on demand. It monitors the backup of your computers and may wake them up and place them back to sleep [11] (Green-it-software, 2015).

Features of lights-out:



- Lights out has User-defined events such as calendar event and Wake-On-Lan.
- Lights out has an activity-based rule set such as runtimes on a calendar, Windows clients, network devices, backup time window, remote access, processes, CPU load and/or network load.

2.3 POWERWISE:

This software helps in power up, power down and power off of the PCs - mechanically - to offer you optimum energy savings, that helps prevent cash and also the environment. This software integrates with Active Directory for security, deployment, and policies. It provides a tendency to see and control in a period of time the facility state of a pc and its devices. It is hidden Powerwise notifications. Powerwise gives versatile Profile Schedules to produce profiles per the time of day and days of the week, and implement additional aggressive savings schemes throughout 'out of workplace hours'. Theme changes permit profiles to mirror the stress and usage of computers in your organization. Schedule events like wakeup and shutdown also manage by this software package [17] (Modus Interactive, 2012).Features of this software: Enterprise central control can run the console from anywhere on the network to manage the Powerwise profiles and computers, and centrally deploy the agent.

Table 2:	Powerwise	Software	Details
----------	-----------	----------	---------

PROPERTIE	INFORMATION
S	
Publisher	Ecosoftware
Publisher	http://www.PCEnergyEfficiency.co
Web Site	m
Release Date	December 10, 2009
Date Added	March 16, 2010
Version	2009
Operating	Windows
Systems	Me/NT/2000/XP/2003/Vista/Server
	2008/7
License	Free to try

Model	
File size	36.98MB

2.4 METERHERO:

This energy optimization software is generally used to provide stats that helps to save money and energy. It is very easy to configure the system settings in accordance with the software [4] (Meterhero, 2010).

Below are the major controls that are provided by the software-

- Connect –Links the utility accounts to MeterHero.
- Track Displays the energy usage on the dashboard.
- Motivate Allows creating and joining a community with other users.

2.5 VISTA BATTERY SAVER:

According to the developer vista battery saver will save up to 70% battery by disabling some of the greedy features. This software runs in the taskbar with private work set of 5.5M and 0% CPU and will control energy consumption by enabling and disabling features such as Aero Glass and Sidebar when the power source is changed.[13] (vistabattery, 2007).

Table 3: Vista Battery Saver Software Details

PROPERTIES	INFORMATION
Publisher	TamirKhason
Publisher	http://dedjo.blogspot.com/
Web Site	
Release Date	July 30, 2007
Date Added	July 30, 2007
Version	2
Operating	Windows/Vista
Systems	
License	Free
Model	
File size	910KB



2.6 GRANOLA:

This reduces up to 35% of the power consumed in PC [9] (Grano.la, 2013). Important features include:

- Granola is effective power saving software for Linux as well as Windows.
- Sophisticated workload prediction which improves upon built-in CPU performance.
- Existing power management settings can be used with this software.
- It runs in the background as lightweight userspace daemon
- No noticeable performance loss.

3 PROGRAMMING LANGUAGES

The oldest computers were programmed bv manually changing 1, 0 and alternating the circuit. It was a difficult task to write programs as most were used for only specific applications. They were large in size so they were quite limited. So programming language invention was a revolutionary idea. The keywords are limited in programming languages and developers create different types of programs by combining these keywords. There are special software that convert the programs into machine language. We can give instructions to a computer in a language the computer understands with the help of programming language. We can implement algorithms using Programming languages. Translate programming language into binary is known as "compiling". Computer scientists tend to form a new programming language. Since the oldest languages are COBOL programming and FORTRAN. Many different programming languages have been formed every year. Some of the programming languages have imperative form (sequence of operations to perform). Other programming languages have the declarative form (the desired result is specified, not how to achieve it). The programming language is commonly divided into two elements: syntax and semantics [30]. Therefore, "programming language is а representation for writing programs which are specification of a computation or algorithm"[27] (SanghpriyaGautam, 2019). Each language has its

advantages and disadvantages. Now, we will discuss some couple of popular programming languages(C language, java, python, go) how they differ from each other on the basis of properties, paradigms, and features criteria.

3.1 C LANGUAGE:

Firstly, the source code is sent to the preprocessor. The preprocessor converts directives into respective values and also generates expanded source code. Expanded source code is passed to the compiler which converts it into assembly code [23]. And then assembly code is passed to the assembler converts it into object code. Now, the generated file is simple.obj . The object code is passed to the linker which links to the library. Now finally it generates the executable code. The executable code is passed to loader which loads into memory then it is executed [25].

3.2 JAVA LANGUAGE:

It has two-step processes. In the first step, java compiler compiles the source code and generates byte code. And byte code saved as a .class file. In the second step, .class file passed as input to java interpreter and it executes the byte code and generates output. Java interpreter is also known as JVM (Java virtual machine) [22]. Java is an object-oriented programming language. [23].

3.3 PYTHON LANGUAGE:

Firstly, Python compiles your source code in the form of .py file. And generate byte code. In other words, we can say compilation is a translation step code platform-independent and byte is а representation of source code. The .pyc file usually stores compiled code or byte code and can be regenerated after the source code update or whenever necessary. The byte code is loaded by python runtime and then interpreted by a Python Virtual Machine. The compilation of byte code is automatic and the PVM is a part of the installed Python system on the machine [24]. Technically, this is the last step of the Python interpreter. Whenever



Python interpreter runs the program, firstly it must convert source code into machine code and also link the runtime libraries. The programs executed by the interpreter are slower than a comparable program written in a compiled language. Python is clever to boost its performance. Python program compiles to the byte code for the first time when it executes a file. This process helps in improvement of the execution of the code next time the module is imported or executed.

3.4 GO LANGUAGE:

This language goal is to approach the performance of the C language. The performance of many programs depends on libraries. The compiler is good and the garbage collector is not fast enough yet to over-perform the C [29]. The current garbage collector implementation is mark and sweep collector. The collector runs in a separate CPU core in parallel with the main program. Major work on the collector in recent years has reduced pause times often to the sub-millisecond range, even for large heaps, all but eliminating one of the major objections to garbage collection in networked servers. But this is very better than many high-level languages. The go compiler gccgo implements go routines using a technique called segmented stacks, supported by recent modifications to the gold linker [26, 29].Comparison of programming language in different aspects in table 4.

LANGUAGE	C	JAVA	PYTHON	GOLANG
INTENDED USE	Application,	Application,	Application,	Application, web,
	system, general	business, client-	general, web,	server-side
	purpose, low-	side, general,	scripting,	
	level operations	mobile	artificial	
		development,	intelligence,	
		server-side, web	scientific	
			computing	
IMPERATIVE	Yes	Yes	Yes	Yes
OBJECT-ORIENTED	No	No	Yes	Yes
FUNCTIONAL	No	No	Yes	Yes
PROCEDURAL	Yes	Yes	Yes	Yes
SGENERIC	Yes	No	Yes	Yes
REFLECTIVE	No	Yes	Yes	Yes
EVENT-DRIVEN	No	Yes	Yes	Yes
FILE-EXTENSION	.c, .h	.go	.java, .class, .jar	.ph, .pyc, .pyd,
				.pyo, .pyw, .pyz
LEVEL	Low	High	High	High
DEVELOPERS	Dennis Ritchie	Google Brendan Eich		Guido Van
				Rossum
DATE	1972	2007	1995	1991
IMPLEMENTATION	Compiled	Compiled	Interpreted	Interpreted
METHOD				
PURPOSE	General	General	Web	General
FREE	Yes	Yes	Yes	Yes
SCRIPT	No	No	Yes	Yes

Table 4:	Comparison	of Program	mming L	anguage
1 4010 1.	Comparison	or i rogiui	inning r	Junguuge



SPEED	10	6-9	10	4-6
OTHER PARADIGM	-	Compiled,	Concurrent	Aspect-oriented
		concurrent,		
		imperative,		
		structured		

4 SORTING

One of the most basic and important problems in computer science is sorting. Sorting is both a practically important and theoretically interesting problem. Since a significant portion of commercial data processing involves sorting large quantities of data, efficient sorting algorithms are of considerable economic importance. List of elements is rearranged in a certain order. This technique is called sorting. This is also for the benefit of searching and locating the information in an easy and efficient way. Numerical order and lexicographical order are the most commonly applied orders [20]. There are number of various good reasons for studying sorting computer algorithms. "Remember that many scientists have much worked on sorting algorithms."Efficient sorting is important for the efficiency of other algorithms such as search and merges algorithms [21]. Comparison of sorting algorithm is given by different aspects in table 5. Types of Sorting Techniques: Sorting techniques are classified into two sorting.First one is internal (where the data resides in the random access memory) [15]. And the second one is external (where the data is predominantly outside the random access memory).

- Internal sorting: Internal sorting is important in algorithm design as well as commercial applications. In those cases where sorting arises as part of another algorithm, the number of elements to be sorted is usually small enough to fit in random access memory [14, 15]. Internal sorts are three types.
- Selection sort
- Insertion Sort
- Exchange Sort
- External sorting: External sorting is an internal

part of such application as account processing, which usually involves far more elements that can be stored in random access memory at one time. Thus external sorting methods for data which are on secondary storage devices (such as a disk memory or a magnetic tape) have great commercial importance [20]. All types of external sort are based on the process of merging. Firstly, different parts of the data are sorted separately and then merged together. Eg-Merge Sort.

However there are numerous of sorting algorithms, sorting problem has shelled a lot of researches. By the use, we found many applications including realtime systems, operating systems, and discrete event simulations [21]. Generally, the efficiency of an application also depends on the sorting algorithm.

4.1 INSERTION SORT: It is a comparison based sorting algorithm. The element is inserted at a proper place alike to card insertion [5]. During this, the list is split into two parts unsorted and sorted sub-lists [7]. This is an insertion sort.

4.2 SELECTION SORT:

Selection sort is a comparison based sorting algorithm. The array is split into two sub-array sorted and unsorted [19]. These two arrays have split by imaginary wall [5]. This algorithm searchs the smallest element from unsorted sub-array and swap it with the initial. And then shift the wall one element for forward, while the sorted array is increasing and the unsorted array is decreasing [7, 12]. This is a selection sort.

4.3 QUICK SORT:

It is a comparison based sorting algorithm. It is based on partition. Quick sort is also known as partition exchange sorting. The concept of this sorting algorithm is to select one element from an



array and rearranges the remaining elements around it. This element splits the main array into two subarray [5]. We can call this element, Pivot. After choosing pivot, shift all the elements less than pivot to the left and elements greater than the pivot are shifted to the right side [3, 7]. Pivot selection and partition the array is applied recursively until subarray consists of only one element. This is the divide and conquer paradigm. This is an exchange sort.

4.4 MERGESORT:

It is a comparison based sorting algorithm. The basic concept of merge sort to splits the array into two smaller sub-array of almost equal size. And repeat this strategy recursively until only one element is left in the sub-array [5]. And then various sorted sub-array are merged to form sorted the parent list. Repeat this process recursively to find the original sorted array. This is the divide and conquer paradigm.

4.5 HEAP SORT:

It is a comparison based sorting algorithm and based on a Binary Heap data structure. This is one of the best sorting methods being in-place and with no quadratic worst-case running time [5]. It is alike to selection sort where firstly we find the maximum element and place the maximum element at the end [7]. The same process is repeated for the remaining element. But it is not a stable sort. This is a selection sort.

4.6 BUCKET SORT:

Bucket sort is a non-comparison sorting algorithm. Everyone called sometimes BIN SORT. It is commonly useful when the unsorted input is uniformly distributed over some range. It runs in linear time. Each bucket can be sorted using a different sorting algorithm or by applying the bucket sorting algorithm recursively [16]. This is a distribution sort.

4.7 COUNTING SORT:

It is a non-comparison sort. This algorithm is for sorting a collection of elements according to keys that are small integers. It has another name, integer sorting algorithm. This counts the number of elements that have each distinct key value and uses arithmetic on those counts to determine the positions of eachkey value in the output sequence [16]. It has linear running time in the number of items and the difference between the maximum and minimum key values. So this algorithm is only suitable for the cases when the variation in keys is not significantly greater than the number of items [2]. This is a distribution sort.

4.8 RADIX SORT:

Radix sort is a non-comparison sorting algorithm. It sorts data with integer keys. It groups the keys by individual digits that share the same significant position and value (place value) [3, 5, 16]. This is a distribution sort.

Sorting	Best	Average	Worst	Space	Stability	In	Method
	Case	Case	Case			Place	
Insertion Sort	O(n)	$O(n^2)$	$O(n^2)$	1	Yes	Yes	Comparison sorting
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$	1	No	Yes	Comparison sorting
Quick Sort	O(nlogn)	O(nlogn)	$O(n^2)$	O(logn)	Depends	Yes	Comparison sorting
Merge Sort	O(nlogn)	O(nlogn)	O(nlogn)	Depends	Yes	No	Comparison sorting
Heap Sort	O(nlogn)	O(nlogn)	O(nlogn)	1	No	Yes	Comparison sorting
Bucket Sort	-	O(n+r)	O(n+r)	O(n+r)	Yes	No	Non comparison sorting
Counting Sort	-	O(n+r)	O(n+r)	O(n+r)	Yes	No	Non comparison sorting
Radix Sort	-	O(n.n/d)	O(n.n/d)	n	Yes	No	Non comparison sorting

Table 5 Comparison of Sorting Algorithm



5 CONCLUSION

In this, we use a different programming language(C, Java, Python, Go) and different algorithm(Selection sort, Insertion sort, Quick Sort, Merge sort, Heap sort, Count sort, Bucket sort, Radix sort). By given discussion showed that software is also responsive to the system's energy consumption. Due to fewer energy resources and more computational energy, the area of green computing is increasing rapidly. It will be helpful to reduce the time spent in searching the data in the search engine. And also save the energy of the system for processing. Green computing will also help you, which will not only be responsible for saving energy but will also manage operational costs.

6 REFERENCES

- Goraczko, M. (2010, February 23). Joulemeter: Computational Energy Measurement and Optimization. november 19, 2018 tarihinde Microsoft: https://www.microsoft.com/enus/research/project/joulemeter-computationalenergy-measurement-andoptimization/?from=http%3A%2F%2Fresearch. microsoft.com%2Fenus%2Fprojects%2Fjoulemeter%2Fdefault.aspx adresinden alındı
- Kaul, A. Space Optimization of Counting Sort. International Journal of Computer Applications, 975, 8887.
- 3. Aliyu, A. M., & Zirra, P. B. Evaluation of Power Consumption of Modified Bubble, Quick and Radix Sort, Algorithm on the Dual Processor.
- 4. Meterhero. (2010). Meterhero. January 6, 2019 tarihinde stemhero: https://stemhero.com/meterhero-contact adresinden alındı
- Kulalvaimozhi, V. P., Muthulakshmi, M., Mariselvi, R., Devi, G. S., Rajalakshmi, C., & Durai, C. (2015). Performance analysis of sorting algorithm. International Journal of Computer Science and Mobile Computing, 4(1), 291-306.
- Bunse, C., Höpfner, H., Roychoudhury, S., & Mansour, E. (2009, July). Choosing the" Best" Sorting Algorithm for Optimal Energy Consumption. In ICSOFT (2) (pp. 199-206).

- Hammad, J. (2015). A comparative study between various sorting algorithms. International Journal of Computer Science and Network Security (IJCSNS), 15(3), 11.
- Bunse, C., Höpfner, H., Mansour, E., & Roychoudhury, S. (2009, May). Exploring the energy consumption of data sorting algorithms in embedded and mobile environments. In 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware (pp. 600-607). IEEE.
- 9. Grano.la. (2013). Grano.la. November 20, 2018 tarihinde Betternames: https://www.betternames.com/offers.php?domain =grano.la adresinden alındı
- Abdulsalam, S., Lakomski, D., Gu, Q., Jin, T., & Zong, Z. (2014, November). Program energy efficiency: The impact of language, compiler and implementation choices. In International Green Computing Conference (pp. 1-6). IEEE.
- Green-it-software. (2015). Lights-Out. November 10, 2018 tarihinde green-it-software: https://www.green-it-software.com/ adresinden alındı
- Jadoon, S., Solehria, S. F., Rehman, S., & Jan, H. (2011). Design and analysis of optimized selection sort algorithm. International Journal of Electric &Computer Sciences (IJECS-IJENS), 11(01), 16-22.
- 13. vistabattery. (2007, july 30). vistabattery. january
 1, 2019 tarihinde Archive.codeplex: https://archive.codeplex.com/?p=vistabattery
 adresinden alındı
- Mishra, A. D., & Garg, D. (2008). Selection of best sorting algorithm. International Journal of Intelligent Information Processing, 2(2), 363-368.
- Flores, I. (1961). Analysis of internal computer sorting. Journal of the ACM (JACM), 8(1), 41-80.
- 16. Kazim, A. (2017). A comparative study of well known sorting algorithms. International Journal
- Jnr, B. A., Majid, M. A., & Romli, A. (2018). A descriptive study towards green computing practice application for data centers in IT based industries. In MATEC Web of Conferences (Vol. 150, p. 05048). EDP Sciences.
- Gill, S. K., Singh, V. P., Sharma, P., & Kumar, D. (2019). A Comparative Study of Various



Sorting Algorithms. International Journal of Advanced Studies of Scientific Research, 4(1).

- 19. Dave, P. H. (2009). Design and analysis of algorithms. Pearson Education India.
- 20. Aho, A. V., & Hopcroft, J. E. (1974). The design and analysis of computer algorithms. Pearson Education India.
- Arnold, K., Gosling, J., & Holmes, D. (2005). The Java programming language. Addison Wesley Professional..
- Prechelt, L. (2000). An empirical comparison of seven programming languages. Computer, 33(10), 23-29.
- 23. Sanner, M. F. (1999). Python: a programming language for software integration and development. J Mol Graph Model, 17(1), 57-61.
- Kernighan, B. W., Ritchie, D. M., Tondo, C. L., & Gimpel, S. E. (1988). The C programming language (Vol. 2). Englewood Cliffs, NJ: prentice-Hall.
- 25. Pike, R. (2009). The go programming language. Talk given at Google's Tech Talks.
- 26. Sestoft, P. (2017). Programming language concepts. Springer.
- Donovan, A. A., & Kernighan, B. W. (2015). The Go programming language. Addison-Wesley Professional.
- Gosling, J., Joy, B., Steele, G., & Bracha, G. (2000). The Java language specification. Addison-Wesley Professional.