

GPU Accelerated TOPSIS Algorithm for QoS Aware Web Service Selection



M. Sri Yogalakshmi, G. R., Karpagam, N. G. Swetha

Abstract: Web service selection is the method of selecting the best web service based on non-functional aspects. With proliferation of web services, there is a demand for efficient approaches to select the best web services from the available pool. A number of Multi criteria decision making approaches are adopted for QoS aware web service selection. The exploitation in web service deployment increases the computation power in executing web service selection. Graphics processing unit (GPU) can achieve reduction in computation power by performing multiple tasks at a time. The independent tasks in the algorithm are parallelized by utilizing GPU in order to achieve more efficiency. This works proves that GPU acceleration can be utilized in parallelizing TOPSIS algorithm for web service selection.

Keyword: High performance computing, Parallel computing on GPU, Parallel TOPSIS algorithm, web service selection, Multi-Criteria Decision Making (MCDM).

I. INTRODUCTION

Web service is software medium for communication between client and server over the network. It offers language transparency and interoperability [1] among various applications.

The service oriented architecture [2] comprise of service provider, service requestor, service registry. The service provider is the one who deploys web service and makes it accessible over the web. The service registry is a central repository of services. The service requestor is the one who requests for web service through XML request.

Web service selection process involves service consumer to invoke the best service by ranking the web services based on QoS parameters. The QoS parameters [2] play a major role in distinguishing functionally similar web services during web service selection.

The QoS parameters [3] include response time, throughput, accessibility, availability, cost etc. The QoS parameters can be divided into positive and negative parameters. The positive parameter values are expected to be high for a best web service whereas, the negative parameter values are expected to be low.

The proliferation of web services demands parallelism over web service selection. The parallelism is a technique of executing multiple instructions at the same time at multiple cores to make the implementation of an algorithm faster. This parallelism [4] is achieved using Graphics processing unit. GPU is a single chip processor, used to achieve parallel computation over multiple sets of data. GPUs are especially popular for graphics applications. The features of GPU [5] are mainly used to manage and boost the performance of video and graphics. Nowadays, GPUs are being used in PC, workstation, mobiles, gaming console and it achieves parallelization not only in graphics but also in non-graphical applications. Their highly parallel structure makes it more efficient in parallel implementation of algorithms. The GPU offers a powerful processing platform to achieve high parallelization over computer vision program.

II. RELATED WORKS

The authors [1] have proposed FUZZY TOPSIS method which is a Multi Criteria Decision making approach where a set of fuzzy numbers were considered to evaluate the weights of each criteria and ratings of each web service which able to solve consumer with fuzzy opinions. The fuzzy positive ideal solution and fuzzy negative ideal solution are determined to calculate the separation measure between ideal solution and the normalized decision matrix. Based on the separation measure ranking is performed. Although, this technique solves the problem of fuzzy opinions of the consumer, it lacks in ranking efficiently when a new web service arises. The authors [6] have proposed an approach which is a combination of Analytical Hierarchy Process and Reference Ideal Method. AHP is adopted to encode the normalized weights of each QoS parameter where each QoS parameters are compared with each other. RIM is a new MCDM approach that specifies the constraints on set of values. RIM is adopted to determine the ideal solution which is between maximum and minimum value of each criteria. Based on the reference ideal, the ranking of web services is performed. The AHP technique lacks in providing accuracy when considering too many criterion and becomes complex when encoding normalized weights. The authors[7] have proposed an approach which combines three MCDM approaches like skyline, AHP, Promethee to perform web service selection.

Manuscript published on January 30, 2020.

* Correspondence Author

M Sri Yoga Lakshmi, PG Student, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India. Email: yogasri1997@gmail.com

DrG R Karpagam, Professor and Associate Head, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India. Email: grk.cse@psgtech.ac.in

N G Swetha, Assistant Professor, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India. Email: ngs.cse@psgtech.ac.in

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

The skyline approach reduced the search space to focus only on the web services that are not dominated by any other services whereas weights for the QoS are assigned using AHP and ranking of web services based on relationships between services and the preference functions on QoS is dealt using Promethee.

The authors [8] have proposed VIKOR method a MCDM approach developed for complex problems where VIKOR method is extended to rate alternatives using fuzzy sets and criteria weights. Entropy weight model has been established to determine weights of the criteria. This technique also adopts a non-linear normalization technique for ranking. It gives an ideal and compromising solution. But this method is found to be complex for implementation.

The authors [2] have proposed TOPSIS algorithm which is a MCDM approach for web service selection to classify web services based on non-functional aspects. This method uses the shortest distance method to find the ideal solution. It considers the user preference in web service selection and provides adequate web services.

III. PROPOSED WORK

A. GPU Acceleration

The parallelism is a technique to implement an algorithm faster by executing multiple instructions at the same time at multiple cores. Graphics processing unit is used to achieve parallelism. The GPU is employed with GPU acceleration [4] which facilitates in achieving parallelism. This GPU acceleration functions by executing intensive operations over GPU in parallel while the remaining parts are executed on CPU. The CUDA [9] is a software platform that interacts with GPU to enable GPU accelerated functions. The CUDA provides a virtualized structure which consists of grids, blocks and threads. The structure in fig. 1, is as follows where the host represents the CPU and the device represents the GPU.

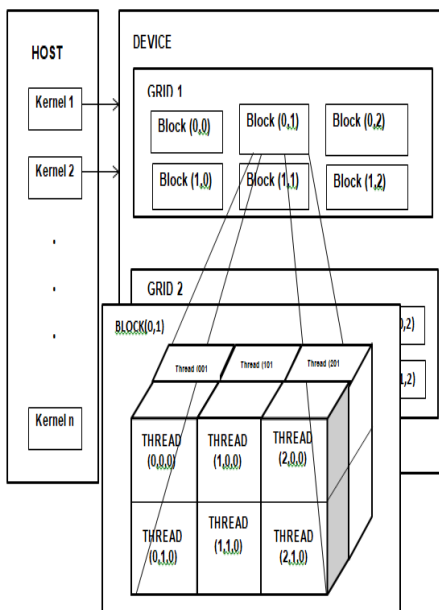


Fig. 1. CUDA architecture source:

https://www.researchgate.net/figure/Basic-CUDA-Architecture-fig1_268152

B. TOPSIS Algorithm

The TOPSIS algorithm [2] is as follows.

Step 1: Construct the input decision matrix d_{ij} which includes web services (i) in row and QoS parameters (j) in columns

Step 2: Perform the Vector Normalization over the input decision matrix d_{ij} . The vector normalization is performed to eliminate the dimensions of each criterion. The normalized value V_{ij} is as follows:

$$V_{ij} = \frac{d_{ij}}{\sqrt{\sum_{i=1}^m d_{ij}^2}} \quad i=1, 2, \dots, m \text{ and } j=1, 2, \dots, n \quad (1)$$

Step 3: Calculate the weighted normalized decision matrix. The weighted normalized value W_{ij} is as follows:

$$W_{ij} = V_{ij} * w_j \quad i=1, 2, \dots, m \text{ and } j=1, 2, \dots, n \quad (2)$$

Where w_j is the weight assigned to the j^{th} criterion and $\sum_{j=1}^n w_j = 1$

Step 4: Determine the ideal positive (ipos) and ideal negative (ineg) solution. The ideal solutions are as follows:

$$I_{pos_j} = \{ \max(W_{ij}) \mid j \in \text{positive criteria} \}, \{ \min(W_{ij}) \mid j \in \text{negative criteria} \} \quad (3)$$

$$I_{neg_j} = \{ \min(W_{ij}) \mid j \in \text{positive criteria} \}, \{ \max(W_{ij}) \mid j \in \text{negative criteria} \} \quad (4)$$

Step 5: Calculate the separation measures using Euclidean distance. The separation measures from ideal solutions are calculated as follows:

$$S_{pos_i} = \sqrt{\sum_{j=1}^n (W_{ij} - I_{pos_j})^2} \quad j = 1, 2, \dots, m \quad (5)$$

$$S_{neg_i} = \sqrt{\sum_{j=1}^n (W_{ij} - I_{neg_j})^2} \quad j = 1, 2, \dots, m \quad (6)$$

Step 6: Calculate the relative closeness of each alternative to the ideal solution. The relative closeness is as follows:

$$RC_i = \frac{S_{neg_i}}{S_{neg_i} + S_{pos_i}} \quad (7)$$

Step 6: Perform the ranking over relative closeness of each alternative.

As there is growth in web services, the functionally similar web services increases exponentially. This rapid growth influences the web service selection. As the data to be processed is huge, web service selection resembles a big data process. The big data [10] is a huge volume of data that occur in varied forms. The big data is too large and it surpasses the computation time in CPU. This big data processing demands the utilization of GPU to improve the performance.

C. Feasibility in Parallelizing TOPSIS Algorithm

The GPU acceleration is employed in achieving parallelism over TOPSIS algorithm where the CUDA virtualization structure is being used. The first step in parallelization is to identify independent and dependent tasks in an algorithm. The sub-tasks that are independent of each other are executed in parallel whereas other tasks are executed sequentially.

In TOPSIS algorithm, while performing normalization of decision matrix, the normalization of each QoS parameter is independent of each other. Therefore, normalization of each criterion is assigned to different threads and computed in parallel.

Similarly in weighted normalization process, the weights are assigned to each criterion and this procedure is executed independently over each criterion. Therefore this step is also parallelizable.

The ideal solutions are determined based on maximum and minimum value of each QoS. The maximum and minimum of each QoS can be found parallel by assigning threads to each QoS which computes maximum and minimum value of each QoS. From this computed value, the ideal solutions are determined. The separation measure from each ideal solution is also computed in parallel.

Thus the above procedure justifies that TOPSIS algorithm can be parallelized to reduce the computation overhead. Following flowchart depicts the parallelization work.

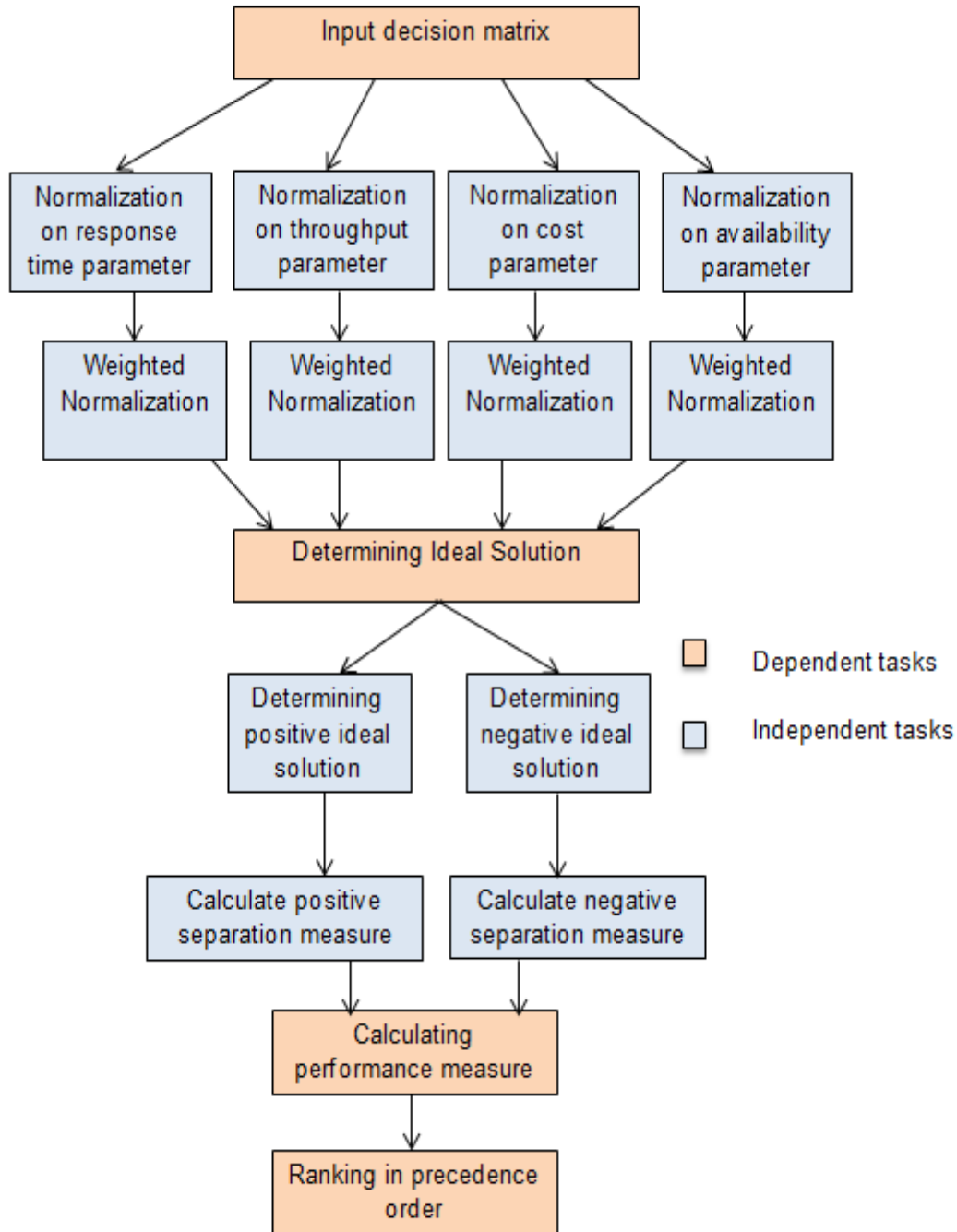


Fig.2. Parallelization workflow for TOPSIS web service selection Algorithm

D. Parallel TOPSIS Algorithm

The following is the pseudo code that depicts the parallel TOPSIS algorithm.

GPU Accelerated Parallel TOPSIS Algorithm:

```

m- number of web services.
n- number of QoS parameters.
w_matrix- weights of each criteria
Input: decision matrix.
Output: rank of web services

device execution:
assign thread dimension
assign block dimension
assign grid dimension
call kernel (device) function from host
device execution:
__global__ Normalization(in_matrix,norm_matrix,m,n)
{
tid=threadIdx.y+blockDim.y*threadIdx.y
//to address the thread that were allotted
column wise

if(tid<n) //each thread perform this operation
simultaneously
{
fori =0 to m
perform vector normalization over norm_matrix
}
}
__global__ weighted_norm(norm_matrix,w_matrix,wnor
m_matrix,m,n)
{
tid=threadIdx.y+blockDim.y*threadIdx.y
//to address the thread that were allotted
column wise

```

```

initialize j=0
if(tid<n) //each take weight of each criteria and multiply with
normalized matrix
{
for i=0 to m
wmorm_matrix<-wmatrix[tid*j]*norm_matrix[tid+n*j]
}
}

determine ideal positive and ideal negative solution
calculate distance from ideal solution

host execution:
calculate closeness of each alternatie to ideal solution
rank web services based on the closeness value

```

E. GPU Server Configuration

The implementation was realized in a GPU enabled server. With the following configuration:

- ✓ Processor- Intel(R) Xeon(R) W-2104 CPU @ 3.20 GHz.
- ✓ GPU - ZOTAC NVIDIA GeForce RTX 2080 Ti.
- ✓ Anaconda + CUDA Toolkit

IV. RESULTS AND DISCUSSIONS

TOPSIS algorithm is implemented with varying web services in CPU using python program. Four QoS parameters are used. The computed results are depicted in table: 1.

The parallel TOPSIS is implemented in GPU using Pycuda program and the computed results are depicted in table: 2.

Table-I: Performance of TOPSIS algorithm in CPU

Number of web service	5	10	50	100	500	1000	3000	5000	10000
CPU time (seconds)	0.026	0.035	0.05	0.1	0.35	0.6	0.85	1.3	1.8

Table-II: Performance of TOPSIS algorithm in GPU

Number of web service	5	10	50	100	500	1000	3000	5000	10000
GPU time (seconds)	0.01	0.015	0.02	0.031	0.06	0.084	0.091	0.12	0.21

Performance of TOPSIS in CPU and GPU is depicted in fig:3.

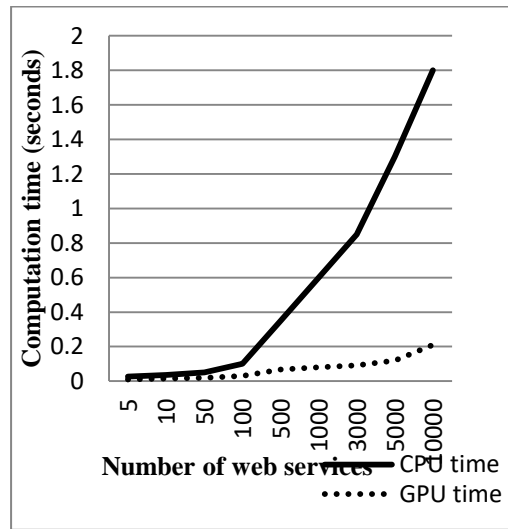


Fig. 3. Performance comparisons of TOPSIS algorithm in CPU and GPU

inferred that the computation time of implementing TOPSIS algorithm in CPU is 0.05 for 50 web services and in GPU, it is 0.02. But as the number of web services increases to 100, 500, 1000, 3000....., 10000, there is a huge difference in computation time of CPU and GPU in implementing TOPSIS algorithm can be seen.



V. CONCLUSION AND FUTURE WORK

As the number of web services increases enormously, the computation time to implement web service selection using TOPSIS algorithm in CPU increases. Therefore, GPU accelerated computing functions are utilized to address the issue of increase in computation time. This work demonstrates the fact that GPU implementation of TOPSIS algorithm is much faster than CPU implementation for web service selection.

As for now only 4 QoS parameter were considered for web service selection using parallel TOPSIS algorithm. In future, this work can be extended by considering more number of QoS parameters.

ACKNOWLEDGEMENT

This work was supported by Centre for Artificial Intelligence and Research Laboratory, PSG College of Technology.

REFERENCES

1. Chi-Chun Lo, Ding-Yuan Chen, Chen-Fang Tsai, Kuo-Ming Chao "Service selection based on fuzzy TOPSIS method", IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, 2010.
2. Houcine BELOUAAR, Okba KAZAR, Khaled REZEG, "Web service selection based on TOPSIS algorithm", International Conference on Mathematics and Information Technology, 2017.
3. E. Cables, M. Lamata, J. Verdegay, "Rim-reference ideal method in multicriteria decision making", Information Sciences, 2016.
4. Hazem A. Abdelhafez, Mohamed M. Rehan, Hossam A. H. Fahmy, "Efficient GPU utilization in heterogeneous big data cluster using token-based scheduler", IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), 2017.
5. Ian Buck, "GPU Computing: Programming a Massively Parallel Processor", International Symposium on Code Generation and Optimization, 2007.
6. WalidSerrai, AbdelkrimAbdelli, Lynda Mokdad, AshrefSerrai, "Dealing with User Constraints in MCDM Based Web Service Selection", 9th IEEE International Workshop on Performance Evaluation of Communications in Distributed Systems and Web Based Service Architectures (PEDISWESA), 2017.
7. AbdelazizOuahdah, KarimBenouaret, AllelHadjali, Fahima Nader, "Combining Skyline and Multi-Criteria Decision Methods to Enhance Web Services Selection", 12th International Symposium on Programming and Systems (ISPS), 2015.
8. Wu Ying-yu, Yu De-jian, "Extended VIKOR for multi-criteria decision making problems under intuitionistic environment", International Conference on Management Science & Engineering 18th Annual Conference Proceedings, 2011.
9. Marko J. Mišić, Đorđe M. Đurđević, Milo V. Tomašević, "Evolution and trends in GPU computing", Proceedings of the 35th International Convention MIPRO, 2012.
10. <https://www.guru99.com/what-is-big-data.html>
11. AbdallahMissaoui, KamelBarkaoui, "A Neuro-Fuzzy Model for QoS Based Selection of Web Service", 3rd International Conference on Information Technology and e-Services (ICITeS), 2013.
12. Cen Chen, Kenli Li, AijiaOuyang, ZengZeng, Keqin Li, "GfLink: An In-Memory Computing Architecture on Heterogeneous CPU-GPU Clusters for Big Data", IEEE Transactions on Parallel and Distributed Systems, June 2018.
13. Chen Wu, Elizabeth Chang, "Intelligent Web Services Selection based on AHP and Wiki", IEEE/WIC/ACM International Conference on Web Intelligence, 2007.
14. Ling Zhang, De-QunZhouLing Zhang, De-Qun Zhou, "A Study on R-TOPSIS Method and Its Application to Web Site Building Alternatives Selection", ICCIC, 2011.
15. NeerjaNegi, Satish Chandra, "Web service selection on the basis of QoS parameter", Seventh International Conference on Contemporary Computing (IC3), august 2014.
16. Ronak R Kayastha, JwalantBaria, "A survey on Web Service Selection and Ranking Methods", International Journal of Computer Applications", May 2015.

17. ZibinZheng, Yilei Zhang, Michael R. Lyu, "Distributed QoS Evaluation for Real-World Web Services", IEEE International Conference on Web Services, July 2010.

AUTHORS PROFILE



M. Sri Yogalakshmi pursuing Master degree in PSG College of Technology, Coimbatore. She completed her Bachelor degree in Panimalar Institute of Technology, Chennai. Her areas of interest includes Semantic Web, Web service Composition, Parallel Computation.



Dr. G. R. Karpagam Professor and Associate Head with 23 years of experience in Computer Science and Engineering at PSG College of Technology, Coimbatore. She obtained her B.E., M.E., Ph.D in Computer Science and Engineering. Her areas of specialization Database Management System, Data Structures and Algorithms, Service Oriented Architecture, Cloud Computing, Artificial Intelligence and Security.



N. G. Swetha currently working as Assistant Professor in the Department of Computer Science and Engineering, PSG College of Technology. She has completed her Master degree at PSG College of Technology, Coimbatore in 2017. She completed her Bachelor degree at Sri Krishna College of Technology, Coimbatore in 2015. She is currently pursuing Ph.

D under Anna University. Her areas of research interest include Web Service Composition, Artificial Intelligence and Multi Criteria Decision Making Problems.