Advances in Mathematics: Scientific Journal **9** (2020), no.9, 7363–7373
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.9.9.83

A GPU PARALLELIZATION FOR GRID GENERATION OF FUZZY TOPOGRAPHIC TOPOLOGICAL MAPPING

NORMA ALIAS[1], MUHAMMAD ZILLULLAH MUKARAM, AND TAHIR AHMAD

ABSTRACT. The fundamental in grid generation for multidimensional geometrical shape is to construct its grid form. Techniques for creating the grid forms and the smaller shapes formed is the basis of grid generation [1]. The form itself will determine the quality of the generation process and well grid-constructed to describe numerical evaluation and speed. In this paper, a structured grid for Fuzzy Topographic Topological Mapping (FTTM) is proposed to improve the quality of grid in numerical perspectives. FTTM is a mathematical model to detect the neuro-inverse magnetic region for neurological disorder [2]. The detection region is based on 4 vertices of FTTM and homeomorphic to each other [3]. A computable homeomorphism will use to define the vertices and edges components of FTTM. The edges represent their homeomorphisms. A topology on grid generation of FTTM addresses the fuzzy topographic and mapping algorithm. The mathematical modeling of FTTM performs the grid structure, design the grid-connected and synchronize the grid generation. For large and extended FTTM, mesh refinement coupled with fine granularity is used to generate the grid via multi-component and multi-version parallelization scheme. The detail of the construction and performance of the strategy is elaborated, evaluated and reported in the paper.

[1]*corresponding author*

2010 *Mathematics Subject Classification.* 65Y05, 54A40.

Key words and phrases. FTTM, Parallel algorithm, grid generation, Parallel performance evaluation.

1. Introduction

The fundamental in grid generation for multidimensional geometrical shape is to construct its grid form. The form itself will determine the quality of the generation process and well grid-constructede to describe a physical domain, in particular, numerical evaluation and speed. The grid quality of the numerical solution can be improved by well-constructed grid.

In this paper, a structured grid for Fuzzy Topographic Topological Mapping (FTTM) is proposed to improve the quality of grid in numerical perspectives. FTTM is a mathematical model to solve the neuro-inverse magnetic problem, which consists 4 vertices that are homeomorphic to each other. The edges represent their homeomorphisms. Figure 1 illustrates the basic form of FTTM to determine the location of epileptic foci of tomography brain image.

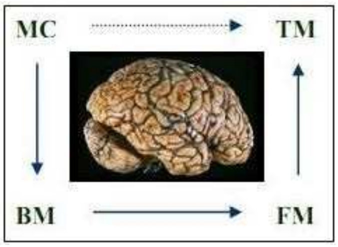

Figure 1. Components form of FTTM.

Physical plane in Figure 1 will transform to the computational plane using FTTM homeomorphism. The features of FTTM provide a structured grid generation and appropriate connectivity among the grid. Due to the topology of the FTTM domain and well-define configuration grid structure, then the segmentation process of the domain into sub-domain involves low communication cost, straightforward parallelism and automates the grid generation process.

2. Mathematical Modeling of FTTM

FTTM homeomorphism [5] stated that FTTM components are homeomorphic to each other. The property of homeomorphisms are true for finite number of components for finite decomposition. The components of FTTM (vertices) and its homeomorphisms (edges) can be presented in Figure 2.

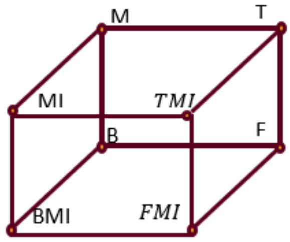

FIGURE 2. The components of FTTM (vertices) and its homeomorphisms (edges)

Generalize FTTM [7] proposed the generalize FTTM involving the topology on the grid generation of FTTM. Thus, a computable homeomorphism will use to define the vertices and edges components of FTTM . The definition of generalize FTTM in [8] can be further expanded to include n number of components and k number of $FTTM$ [6, 8].

FTTM perform a well define grid structured and grid-connected systems. Grid generation is based on n number of components and k number of FTTM homeomorphism. A grid structured and geometrical shape of FTTM is a sequence of polygons $FTTM_n^k$ as shown in Figure 3.

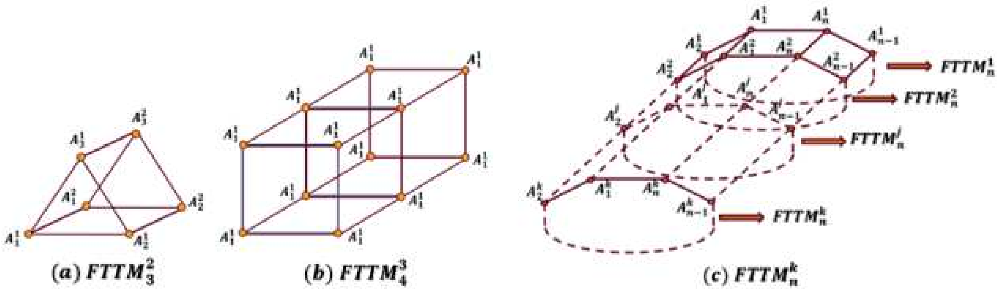

FIGURE 3. Grid generation based on degree of FTTM Homeomorphism, (a) $FTTM_3^2$, (b) $FTTM_4^3$ and (c) polygons $FTTM_n^k$

A new FTTM can be generated from the generalized $FTTM$ in [8].

$$FTTM_n^1 = \{A_1^1, A_2^1, \cdots A_n^1\}$$

$$FTTM_n^1 = \{A_1^2, A_2^2, \cdots A_n^2\}$$

$$FTTM_n^1 = \{A_1^3, A_2^3, \cdots A_n^3\}$$

$$\vdots$$

$$FTTM_n^k = \{A_1^k, A_2^k, \cdots A_n^k\}$$

where $\{A_1^{m_1}, A_2^{m_2}, \cdots, A_n^{m_n}\} \in G(*FTTM)$ and $0 \le m_1, m_2, \cdots, m_n \le k, m_i \neq m_j$ for at least one i, j.

Since a generated FTTM can be determined by sequence of polygon, FTTM graph of pseudo degree zero is root neighborhood of a degree-k FTTM. The pseudo degree of the connected components adjacent to a path included the equal generated elements. The pseudo degree of a new FTTM element is the sum of all pseudo degree of its components, $\sum_{i=1}^{n} deg_p(A_i^{m_i})$.

The extended version of the Elsafi's conjecture generate the grid can be seen in [8].

3. Sequential Algorithm

The sequential FTTM algorithm as seen in [8] consists of three main steps. The first is generated FTTM array such as dimension of the system, d and version, v . Next, determine the degree, k and lastly is check the order of FTTM, n. The *maximum convergence* of the levels with respect to the *iteration number n*, will control the synchronization of Table 1 $|G_0(*FTTM_n^k)|$ for $1 \le k \le 10$ and $1 \le n \le 10$ for the overall convergence process.

Table 1. $|G_0(*FTTM_n^k)|$ for $1 \le k \le 10$ and $1 \le n \le 10$ for the overall convergence process.

k\n	1	2	3	4	5	6	7	8	9	10
1	1	0	0	0	0	0	0	0	0	0
2	0	2	0	2	0	2	0	2	0	2
3	0	0	6	12	30	60	126	252	510	1,020
4	0	0	0	24	120	480	1,680	5,544	17,640	54,960
5	0	0	0	0	120	1,080	6,720	35,280	168,840	763,560
6	0	0	0	0	0	720	10,080	90,720	665,280	4,339,440
7	0	0	0	0	0	0	5,040	100,800	1,239,840	12,096,000
8	0	0	0	0	0	0	0	40,320	1,088,640	17,539,200

Previous studies have investigated combinations of graph with pseudo degree zero for n number of components and k number of version by using a computer algorithm [8]. The conjecture on the number of graphs with pseudo degree zero generated by a sequence of FTTM with three versions is proven analytically [8]. The combination of graph with pseudo degree zero for n number of components and k number of version performs in parallel algorithm.

4. Parallelization Strategy for Synchronizing Grid

Due to the requirement of high accuracy, good convergence, high resolution of the visualization, we proposed parallel algorithm for FTTM.

The advantages of FTTM topology are well-define configuration grid structure, appropriate grid connectivity and a structured grid generation. The properties provide a non-overlapping domain decomposition algorithm to segment the domain, S into smaller subdomains $s_1, s_2, s_3, \cdots, s_n$. The block-based grid segmentation s_i is not overlapping, independently of the neighbors, $s_{(i-1)}$ and $s_{(i+1)}$. All subdomains can be solved simultaneously. Figure 4 shows the domain decomposition is based on the blocks of $FTTM_n^k$ and performs at k-direction or n-direction.

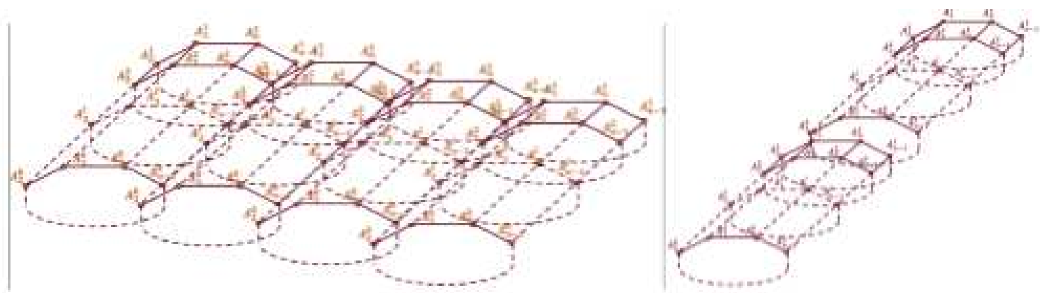

FIGURE 4. Implementation of domain decomposition, agglomerative clustering, sub-domain segmentation and mapping play.

The optimization of numerical results of parallel FTTM such as accuracy, convergence rate, error estimation, consistency can be achieved by fine granular technique. The computational cost should be greater than communication cost. Minimum the size of interval or maximum order of d, v, k and n will contribute to high computational cost, good performance of fine granularity and the best GPU grid sizes.

4.1. **Implementation parallel FTTM on CPU-GPU platform.** In a CPU parallelizm, threads are arranged as a grid of thread blocks, different kernels can be configured by different grid of blocks. Threads from the same block have access to a hybrid memory synchronizely.

For large scale FTTM, we proposed mesh refinement and fine granularity strategy to generate the grid in parallel by blocks of threads, $thread-block_1, thread-block_2, thread-block_3, ..., thread-block_b$.

High component and version of FTTM obtain large sparse simulation. Through the these parallelization scheme, multiple FTTM can be executed simultaneously and concurrently.

The 1D, 2D grid of 3D thread blocks map well to multi-core, multi-node compute resources based on single intra ction multiple data stream taxonomy (SIMD). GPU parallelization allows dynamic mapping sub-domain to several thread blocks in CUDA.

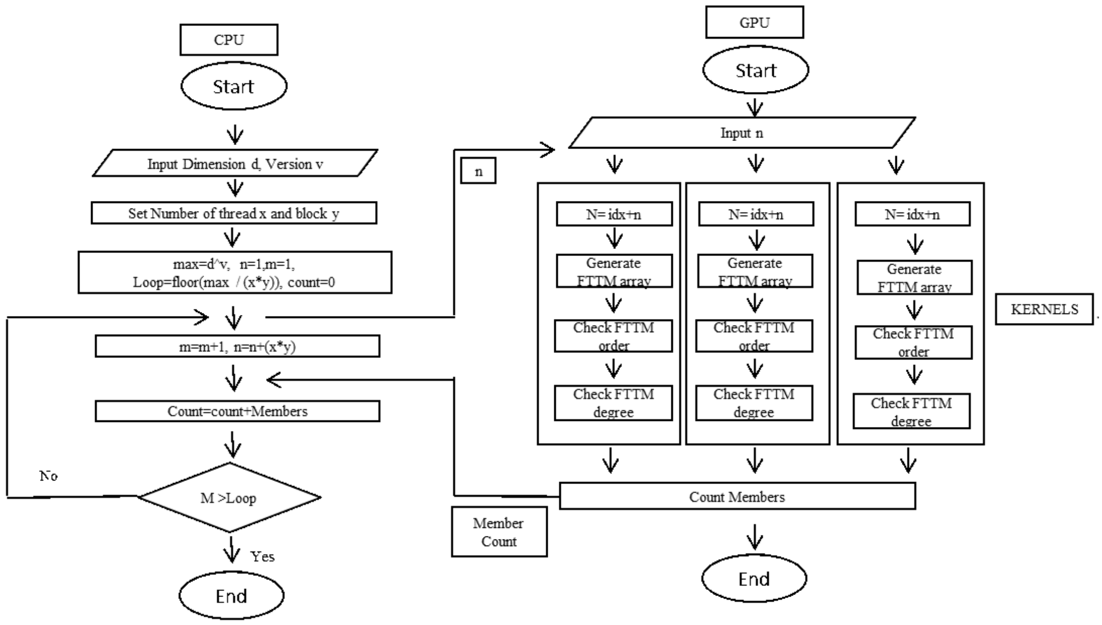

FIGURE 5. The parallel algorithm of FTTM, the communication activities, kernel and threads.

4.2. **Result and discussion.** The growth in the application of FTTM generators for off-grid and grid-connected systems especially in microgrid scale cluster.

[8] shows that the increasing run time of FTTM is based on the increasing of components, A and versions, k . Almost 1 million seconds is needed to execute $FTTM_{13}^5$ and half million seconds is needed to execute $FTTM_{13}^4$. The low numbers of A and, k are well suited for CPU implementation and not suited for GPU parallelization.

Thus, lets focus the PPE analysis on high numbers of A and, k as captured in Table 2.

TABLE 2. Run Time for the Parallel Algorithm $k = 3$.

Run time							
Components	Threads						CPU
	8	16	32	64	128	256	
12	2,698	1,791	1,205	866	776	652	489
13	6,941	4,115	2,473	1,616	1,129	988	1,465
14	21,291	11,412	6,677	4,002	2,853	1,769	4,469
15	70,284	37,957	22,630	11,820	7,106	4,766	13,744
16	189,184	105,340	60,371	34,251	20,149	13,907	45,406
17	597,535	375,649	164,250	103,807	62,178	38,552	140,313

Table 2 shows the comparison of run time on GPU platform with multi- thread and CPU platform for $FTTM_n^3$, $n = 12, 13, 14, 15, 16$ and 17. It shows that more that 64 threads are significant to support huge simulation of $FTTM_n^3$, $n = 14, 15, 16$ and 17. A small number of components, $FTTM_{13}^3$ is well suited for 128 threads. However, CPU platform is well suited to support $FTTM_{12}^3$, because of the minimum the computational cost and maximum communication cost. In details, the investigation of the PPE can be explained in Figure 6.

Figure 6 shows the PPE comparison and evaluation for component $FTTM_{12}$, $FTTM_{13}$ and $FTTM_{14}$ to solve the neuro-inverse magnetic problem based on (a) run time, (b) speedup, (c) efficiency and (d) effectiveness.

Figure 6 shows the run time decrease significantly for high number of threads. Higher number of FTTM components, $FTTM_{14}$ perform better PPE compared to the lower number of FTTM: $FTTM_{13}$ and $FTTM_{12}$. Effectiveness of $FTTM_{12}$ is

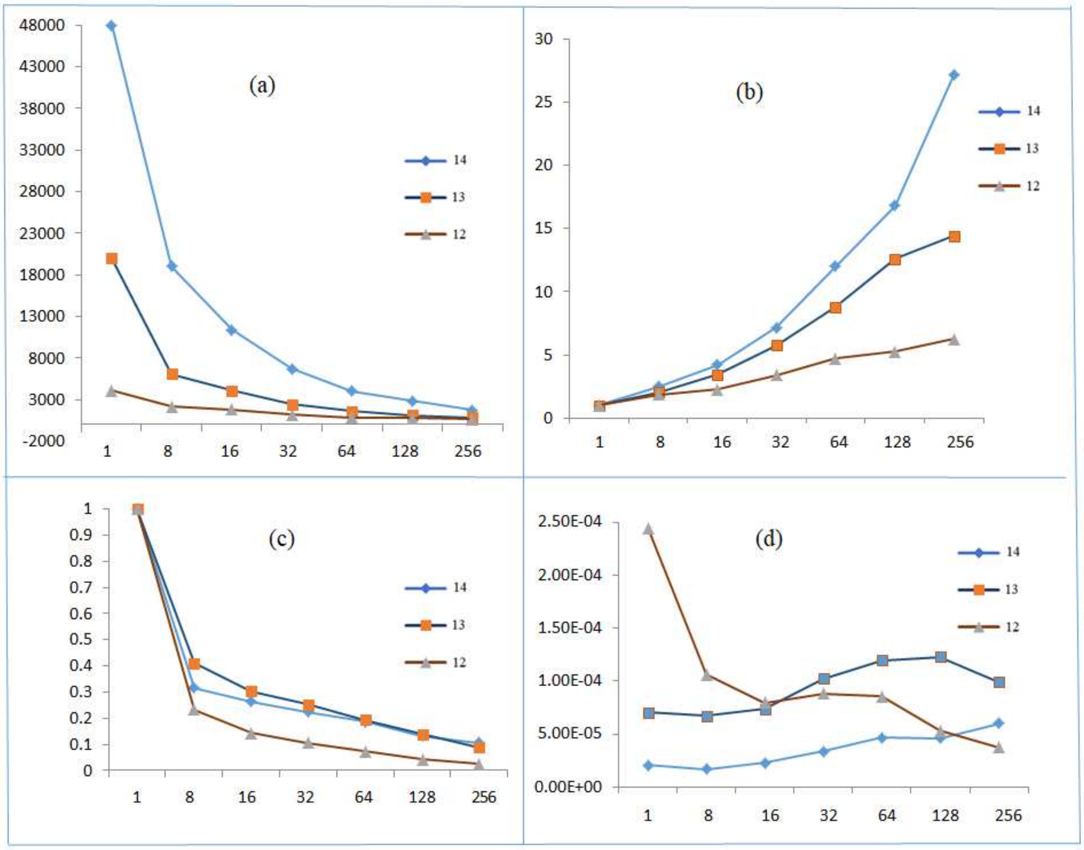

FIGURE 6. The PPE comparison for component $FTTM_{12}$, $FTTM_{13}$ and $FTTM_{14}$ based on (a) run time, (b) speedup, (c) efficiency and (d) effectiveness.

weak for implementing on more than 16 threads obviously. The optimum number of threads for the effectiveness of $FTTM_{13}$ and $FTTM_{12}$ is 64. The computational cost, the granularity of $FTTM_{14}$ were highest compared to lower component of FTTM. Thus the speedup and the efficiency of $FTTM_{14}$ are highest than $FTTM_{13}$ and $FTTM_{12}$. The graph of temporal performance is approximate similar to (d) effectiveness.

A concise comparison of the components is presented in Figure 7.

High computational cost increases the run time (a). $FTTM_{17}$ needs more run time compared to $FTTM_{16}$ and $FTTM_{15}$.

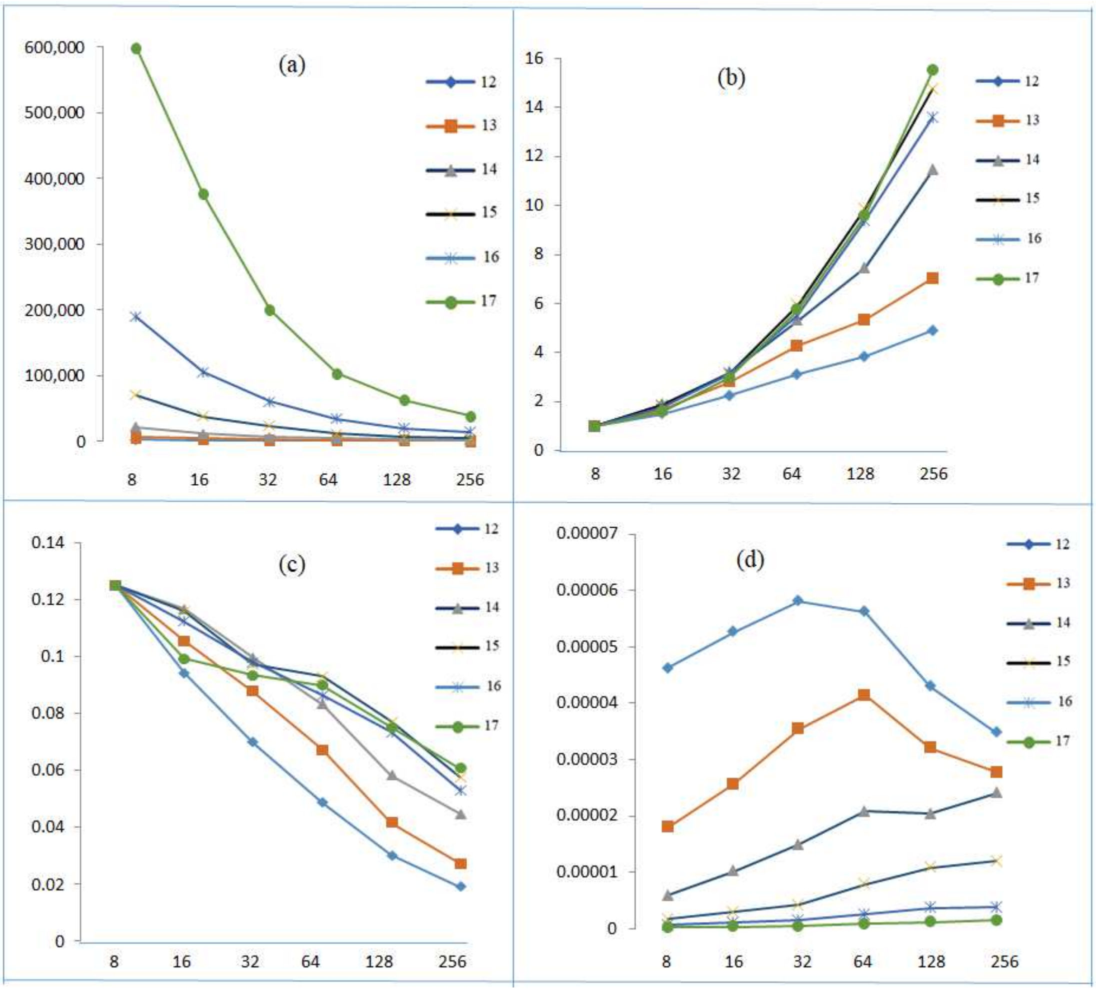

FIGURE 7. The PPE comparison for all component based on (a) run time, (b) speedup, (c) efficiency and (d) effectiveness.

The speedup (b) increases for high number of components and threads. The largest speedup is 16. Speedup increases when more components of FTTM and more computational cost are needed to maximize speedup.

The communication cost, idle time and waiting time contribute to the lowest efficiency (c) of $FTTM_{12}$. The computational cost is cheap for the lower number of components significantly.

Figure 7 created a great baseline for effectiveness(d) comparisons with difference component of FTTM. 64 threads are the optimum performance for large sparse simulation of FTTM on CPU-GPU platform.

$FTTM_{17}$ is the alternative component to be used for detecting the region of neurological disorder in brain.

CONCLUSION

Inverse localization of brain activity based on neuro-inverse electromagnetic equipment can be determined by $FTTM$ topology on GPU platform. For accurate prediction, multi-component and multi-version parallelization scheme contribute to large sparse simulation. GPU Parallelization of $FTTM$ able to be implemented by using sub-domain and grid generation strategies. $FTTM_17^3$ is the alternative topology to detect the region of neurological disorder with good PPE indicator and optimum iteration. GPU parallelization of $FTTM$ is a pioneer strategy to be applied on distributed computing platform [4]. GPU parallelization of $FTTM$ is ready to be upgrade on the latest HPC generation of GPU architecture such as Tesla T4, Tesla P100 and Tesla V100 GPUs devices.

REFERENCES

[1] D. SIEGER, M. STEFAN, B. MARIO: *On Shape Deformation Techniques for Simulation-Based Design Optimization*, In New challenges in grid generation and adaptivity for scientific computing, (2015), Springer, Cham, 281–303.

[2] E. COMBRISSON, K. JERBI: *Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy*, Journal of Neuroscience Methods, **250**(2015), 126–136.

[3] Z.M. MUKARAM, T. AHMAD, N. ALIAS: *Graph of pseudo degree zero generated by FTTM*, AIP Conference Proceedings, AIP Publishing LLC, **2184**(1) (2019), 020007.

[4] N. ALIAS, R. DARWIS, N.S. YUSNIMAN, N. HASHIM: *Distributed parallel performance of compression molding model for some characteristics identification of tire tread block*, International Journal of Engineering and Advanced Technology, **9**(1)(2019), 4886–4892.

[5] T. AHMAD, R.S. AHMAD, L.Y. LIAU, F. ZAKARIA, W.E.Z. ZARINA, W.A. RAHMAN: *Homeomorphisms of fuzzy topographic topological mapping (FTTM)*, MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics, **21**(2005), 35–42.

[6] M. SAYED, T. AHMAD: *Elements for Fuzzy Topographic Topological Mapping*, Life Science Journal, **11**(10) (2014), 316–319.

[7] M. SAYED, T. AHMAD: *On properties of the graph of fuzzy topographic topological mapping*, Malaysian Journal of Fundamental and Applied Sciences, **9**(3) (2013), 139–142.

[8] Z. M. MUKARAM, N. ALIAS, T. AHMAD: *High performance analysis in generating FTTM graph of pseudo degree zero using CPU-GPU*, JIP Conference Proceedings, **2013**(1) (2018), 020005.

IBNU SINA INSTITUTE FOR SCIENTIFIC AND INDUSTRIAL RESEARCH (ISI-SIR),
UNIVERSITI TEKNOLOGI MALAYSIA,
SKUDAI, JOHOR
Email address: norma@ibnusina.utm.my

MATHEMATICAL SCIENCES, FACULTY OF SCIENCE,
UNIVERSITI TEKNOLOGI MALAYSIA,
SKUDAI, JOHOR
Email address: azil.utmskudai@gmail.com

IBNU SINA INSTITUTE FOR SCIENTIFIC AND INDUSTRIAL RESEARCH (ISI-SIR),
UNIVERSITI TEKNOLOGI MALAYSIA,
SKUDAI, JOHOR
Email address: tahir@ibnusina.utm.my