



Article An Effective Selection of Memory Technologies for TCAM to Improve the Search Operations: Demonstration of Memory Efficiency in SDN Recovery

Abdulhadi Alahmadi¹ and Tae Sun Chung^{2,*}

- ¹ Department of Computer Engineering, Ajou University, Suwon 16499, Republic of Korea; hodhod@ajou.ac.kr
- ² Department of Artificial Intelligence, Ajou University, Suwon 16499, Republic of Korea
- * Correspondence: tschung@ajou.ac.kr

Abstract: Ternary Content-Addressable Memory (TCAM) is used for storing the flow tables in software-defined networking (SDN)-based OpenFlow switches. However, the TCAM can store only a certain number of flow tables (8000). Moreover, when the switch flow tables need to be updated due to the link failure in the SDN, further updates may be lost due to the flow tables limit of the TCAM space. Hence, to resolve this issue, other memories need to be used in conjunction with TCAM to enhance the memory operations of TCAM. When considering which flash memory technology is to be used in conjunction with TCAM, we need to balance several factors to ensure optimal performance, speed, endurance, reliability, integration complexity, and cost-effectiveness. Hence, it leads to a multi-criteria decision-making problem regarding the selection of other memory technologies such as 3D XPoint, Magnetoresistive RAM, Resistive RAM, and Ferroelectric RAM. In this paper, we use the analytical network process (ANP) method to select the suitable technology in conjunction with TCAM, considering the features of the memory technologies for Software-Defined Internet-of-Things (SD-IoT). We provide a comprehensive numerical model leveraging the ANP to rank the memory technologies regarding their weights. The highest weights identify the most suitable technology for TCAM. We perform simulations to show the effectiveness of the mathematical model utilizing the ANP. The results show that the suggested methodology reduces the recovery delay, improves the packets received ratio (PRR), decreases the jitter, and increases the throughput.

Keywords: TCAM; SDN; ANP; NAND; memory management

1. Introduction

TCAM (Ternary Content-Addressable Memory) is a type of memory used in digital circuits and network devices for high-speed data searches. TCAM is designed to perform content-based addressing, allowing for rapid and parallel searching of data. TCAM cells are made up of three states, often represented as "0", "1", and "X" (don't care). Unlike traditional binary memories where each cell can only store a "0" or a "1", TCAM cells can also hold the "X" value, which signifies that the corresponding bit does not matter for the search operation. Each TCAM cell typically consists of multiple storage elements, comparators, and associated control circuitry to facilitate content-based searches. When a search operation is performed, the TCAM compares the input search key to the stored values in parallel across all cells. The "X" values in TCAM cells allow for more flexible matching criteria, making it well-suited for applications where wildcard or range-based searches are common, such as in network routing and firewall rules. TCAM is commonly used in networking devices like routers and switches to quickly identify and route packets based on their content. However, TCAM is relatively expensive and power-hungry compared to traditional binary memories like SRAM (Static Random-Access Memory) or DRAM (Dynamic Random-Access Memory), which limits its use to applications where high-speed content-based searches are essential [1,2].



Citation: Alahmadi, A.; Chung, T.S. An Effective Selection of Memory Technologies for TCAM to Improve the Search Operations: Demonstration of Memory Efficiency in SDN Recovery. *Electronics* 2024, *13*, 707. https://doi.org/10.3390/ electronics13040707

Academic Editor: Stefano Scanzio

Received: 24 November 2023 Revised: 29 January 2024 Accepted: 5 February 2024 Published: 9 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). TCAM is used in software-defined networking (SDN) OpenFlow switches for the storage of flow tables. In each flow table, flow entries are stored, which determines the actions to be taken upon the incoming packets to the switches. The literature [3,4] suggests that there can be up to 8000 flow tables due to TCAM space limitations. Moreover, if the number of flow tables required increases more than the specified number, then alternative memory technologies such as NAND, Optane memory, MRAM (Magnetoresistive RAM), ReRAM (Resistive RAM), and FRAM (Ferroelectric RAM) can be combined with TCAM [5,6].

Optimizing TCAM using a hybrid mechanism with other flash memory technologies requires careful consideration of the specific requirements and characteristics of both TCAM and the chosen flash memory. When considering which flash memory technology to use in conjunction with TCAM, you will need to balance several factors to ensure optimal performance, speed, endurance, and cost-effectiveness. While there is not a one-size-fits-all answer, certain non-volatile memory technologies might be more suitable for hybridization with TCAM. Keep in mind that the availability and suitability of these technologies can change over time due to advancements in the field. According to the literature, here are a few options to consider: (1) 3D XPoint is a relatively new non-volatile memory technology developed by Intel and Micron. It offers higher performance and endurance compared to traditional NAND flash while maintaining lower latency. 3D XPoint memory could be used as a cache or secondary storage layer to accelerate TCAM access; (2) MRAM is a non-volatile memory technology that uses magnetic elements to store data. It has fast read and write speeds, low latency, and high endurance. MRAM could be integrated as a hybrid storage layer for TCAM, providing both speed and durability; (3) ReRAM is another emerging non-volatile memory technology that offers fast read and write speeds and relatively high endurance. It can be a candidate for use alongside TCAM to enhance performance and endurance. In addition to it, (4) FRAM is a non-volatile memory technology that combines the best attributes of RAM (fast read and write speeds) with non-volatile storage. While it may not offer the same capacity as other flash memory types, it can provide fast and reliable storage for frequently accessed TCAM data [5,6].

Furthermore, when selecting a flash memory technology to use in conjunction with TCAM, consider factors such as speed: choose a technology with low latency and fast read and write speeds to complement TCAM's high-speed search operations. In addition to speed, look for memory with high endurance, as TCAM operations involve frequent write operations. Moreover, prioritize memory technologies with strong data retention and reliability characteristics. Further, evaluate the cost-effectiveness of the chosen memory technology for your specific use case. Finally, consider the feasibility and complexity of integrating the chosen memory technology into your TCAM architecture. Hence, it is important to collaborate with experts in both TCAM design and the chosen flash memory technology to ensure compatibility, performance, and longevity in your hybrid TCAM solution. Since technology landscapes evolve, it is advisable to consult the latest industry developments and research before making a decision.

Our previous works [7,8] discuss flash memory technologies, enumerate the benefits of flash storage devices and their usage in various markets such as mobile devices, IoT, laptop computers, etc., and discuss the recovery techniques of flash memory with the flash translation layer. Moreover, the works presented in [8] illustrate an effective strategy of recovery for flash memory utilizing shadow paging. Various metrics are evaluated and compared with other schemes considering their memory type.

High-speed searching and accessing of data are primary tasks of any device to have a faster look-up operation. But it becomes critical for the operation to search more information at different instants with far-fetched throughput for various network algorithms, especially in SDN OpenFlow switches with TCAM limitations for the OpenFlow tables. The OpenFlow tables in switches are stored in TCAM. The works in [9–13] show different schemes for the efficiency and fast searching of TCAM. Moreover, they also show the cost and energy efficiency of the TCAM employing different strategies for efficiency of the TCAM.

The authors in [14,15] focused on the SDN problems regarding the location selection for controllers in SD-IoT or on the controller's selection. However, none of these works consider TCAM enhancement in combination with other memory technologies. To the best of our knowledge, these are pioneering works regarding increasing the effectiveness of TCAM through a selection of state-of-the-art memory technologies in conjunction with TCAM for SDN. The main contributions of our work are as follows.

- First, we identify the state-of-the-art and compatible memory technologies with TCAM along with their features.
- We formulate the problem for TCAM alternative technologies selection and ranking with ANP.
- Further, we make a mathematical model and pictorial diagram of TCAM technologies and their features leveraging ANP.
- Then, we apply a step-by-step method utilizing the ANP mathematical model to rank the memory technologies.
- Finally, we compare the results from the switches with memory technology and rank our proposed ANP model higher than another mathematical model, the AHP [16].
- To validate the results' effectiveness, we evaluate several metrics such as recovery delay, jitter, packets received ratio, and throughput, and compare them with the AHP model.

The rest of our paper is organized as follows. In Section 2, we formulate the problem of memory technologies selection for TCAM with ANP. In Section 3, the detailed mathematical model and step-by-step process of ANP with respect to the TCAM technologies are explained. Experiments are conducted and results are discussed in Section 4. Section 5 concludes the paper.

2. Problem Formulation

There are several memory technologies with distinct criteria and features. Hence, in this paper, we solve the problem of memory technology selection for TCAM with an analytical network process-based decision-making scheme. We represent the problem with a mathematical model and pictorial diagram, leveraging the ANP model. Then we rank the memory technologies according to their weights returned by the ANP model. The memory technologies and their characteristics as well as analysis are shown in Table 1 and the features associated with each one as well as the explanation of features regarding these memory technologies are denoted in Table 2. We have evaluated and surveyed the literature [5,6], and [9–15] the memory technologies and their features for TCAM improvement. Figure 1 shows the ANP alternatives and features.

Table 1. Alternatives (memory technologies for TCAM).

Serial#	Name	Terminology	Analysis of Memory Technology
1	NAND	MT_1	Non-volatile, a choice for portable devices, a cache layer for TCAM access
2	Optane	MT ₂	Non-volatile, higher performance and durability compared with traditional NAND flash, lower in latency, cache layer to accelerate TCAM access
3	MRAM	MT ₃	Non-volatile, fast read/write speeds, low latency, high endurance, a hybrid storage layer for TCAM
5	FRAM	MT ₅	Non-volatile that combines the finest qualities of RAM, while it may not offer the same ability as other types of flash memory, it can provide fast and consistent storage for frequently accessed data of TCAM

Serial#	Name	Explanation	Terminology
1	Speed	Analysis of the memory technologies with low latency, read and write speeds comparison to balance TCAM's high-speed operations while searching and inserting flow operations in the network (SDN) switches.	FR ₁
2	Endurance	Comparison of memory technologies with high endurance. In SDN switches, TCAM operations required frequent write operations when there was a flow modification required.	FR ₂
3	Reliability	Enlisting memory technologies with robust data retention as well as reliability features.	FR ₃
4	Cost	Evaluate the cost-effectiveness of the memory technologies for your use cases in SDN switches while there is link failure and insertion of flow operations in SDN switches.	FR ₄
5	Integration Complexity	Evaluation of feasibility as well as complexity of integrating the memory technology into TCAM architecture of SDN switches	FR ₅

	Table 2.	Criteria/	/Features	of each	memory	/ technol	ogy.
--	----------	-----------	-----------	---------	--------	-----------	------



Figure 1. ANP-based model for memory technology selection with TCAM.

3. ANP Mathematical Model for Ranking the TCAM Alternatives

Herein, we model the problem of memory technology ranking and selection with ANP. The features of memory technologies are shown in Equation (1), while the technologies to be combined with TCAM are denoted in Equation (2). The details of these are given below.

$$FR = \{FR_1, FR_2, FR_3, \dots, FR_N\}$$
(1)

$$MT = \{MT_1, MT_2, MT_3, \dots, MT_N\}$$
(2)

Alternative Memory Technologies:

NAND: NAND is the non-volatile memory, i.e., it can hold the information even without the connectivity of power. The performance of NAND is comparatively lower

than the 3D Xpoint. Hence, the ability of it to retain data without power makes it a great alternative for external as well as internal portable devices. Moreover, it can be utilized as a cache layer with TCAM access.

3D XPoint (Optane Memory):

3D XPoint is a relatively new non-volatile memory technology developed by Intel and Micron. It offers higher performance and endurance compared to traditional NAND flash while maintaining lower latency. 3D XPoint memory could be used as a cache or secondary storage layer to accelerate TCAM access.

MRAM (Magnetoresistive RAM):

MRAM is a non-volatile memory technology that uses magnetic elements to store data. It has fast read and write speeds, low latency, and high endurance. MRAM could be integrated as a hybrid storage layer for TCAM, providing both speed and durability. ReRAM (Resistive RAM):

ReRAM is another emerging non-volatile memory technology that offers fast read and write speeds and relatively high endurance. It can be a candidate for use alongside TCAM to enhance performance and endurance.

FRAM (Ferroelectric RAM):

FRAM is a non-volatile memory technology that combines the best attributes of RAM (fast read and write speeds) with non-volatile storage. While it may not offer the same capacity as other flash memory types, it can provide fast and reliable storage for frequently accessed TCAM data.

When selecting a flash memory technology to use in conjunction with TCAM, consider these factors:

Criteria for choosing a suitable memory technology:

Speed: Choose technology with low latency and fast read and write speeds to complement TCAM's high-speed search operations.

Endurance: Look for memory with high endurance, as TCAM operations involve frequent write operations.

Reliability: Prioritize memory technologies with strong data retention and reliability characteristics.

Cost: Evaluate the cost-effectiveness of the chosen memory technology for your specific use case.

Integration Complexity: Consider the feasibility and complexity of integrating the chosen memory technology into your TCAM architecture.

It is important to collaborate with experts in both TCAM design and the chosen flash memory technology to ensure compatibility, performance, and longevity in your hybrid TCAM solution. Since technology landscapes evolve, it is advisable to consult the latest industry developments and research before making a decision. The overall step-by-step illustration of the ANP model is shown in Figure 2. First of all, the memory technologies and their features are shown in the top left boxes. Then below these two boxes is a pairwise comparison box which shows that the alternatives (memory technologies as indicated in Table 1) are compared in a matrix form with respect to their features (shown in Table 2). The box with a consistency ratio (CR) is shown in Figure 2, which is for the purpose of computation of the consistency among the pairwise comparison values. In simple words, while comparing the memory technologies regarding their features, we should assign accurate values. These values are shown in upcoming paragraphs with equations (i.e., CR). From the Consistency Index, the Consistency Index (CI) values are calculated and if these values are less than or equal to 0.1, then the judgments made in the comparison matrix are regarded as precise and accurate. If the judgments are not accurate, then they are calculated. Figure 2 shows that the next two steps are the calculations of the unweighted and weighted supermatrices. These matrices show the final weights for the memory technologies. A memory technology with a high weight will be considered as the most favorable to be used

with TCAM. More details for these matrices are given in the next subsections. Herein, we provide a brief overview of the steps in the ANP. In the next subsections, more explanation is provided with the support of expressions and equations.



Figure 2. The steps in ranking the alternative memories using ANP model.

3.1. TCAM (Alternatives) and Features Pairwise Comparison

The pairwise comparison matrix, in the context of the Analytical Network Process (ANP) decision-making method, is a fundamental component used to systematically assess and quantify the relative importance or preference of one criterion or element over another within a given decision-making framework. The ANP is an extension of the Analytic Hierarchy Process (AHP) designed to handle complex decision problems with interdependencies and feedback loops among criteria and elements.

In the ANP, decision-makers or experts are asked to provide pairwise comparisons between elements, which can be criteria, sub-criteria, or alternatives, depending on the specific decision problem. These pairwise comparisons are used to construct a square matrix known as the pairwise comparison matrix. The dimensions of the matrix correspond to the number of elements being compared.

The entries in the pairwise comparison matrix represent the strength of preference or importance of one element relative to another. The scale used for these comparisons often ranges from, for example, 1 (indicating equal importance) to 9 (indicating extremely stronger importance). The elements on the diagonal of the matrix typically have a value of 1, indicating that an element is equally important or preferable to itself.

Once the pairwise comparison matrix is completed, mathematical operations such as the Eigenvector method or the Saaty scale are applied to derive the weights or priority values for each element. These weights represent the relative importance of each element within the decision hierarchy.

The pairwise comparison matrix serves as a quantification of subjective judgments and allows decision-makers to incorporate their preferences and expertise into the decision model. The ANP then extends this process to account for the complex relationships and interdependencies between elements within a network structure, providing a more comprehensive and realistic approach to multi-criteria decision-making.

$$A = \begin{bmatrix} MT_{1} & MT_{2} & MT_{3} & \cdots & MT_{n} \\ MT_{1} & 1 & A_{12} & A_{13} & \cdots & A_{1n} \\ MT_{2} & \frac{1}{A_{12}} & 1 & A_{23} & \cdots & A_{2n} \\ MT_{3} & \frac{1}{A_{13}} & \frac{1}{A_{23}} & 1 & \cdots & A_{3n} \\ \vdots & \vdots & \vdots & \vdots & 1 & \vdots \\ MT_{n} & \frac{1}{A_{1n}} & \frac{1}{A_{2n}} & \frac{1}{A_{3n}} & \cdots & 1 \end{bmatrix}$$
(3)

	Г	MT_1	MT_2	MT_3	• • •	MT _n
	MT ₁	1	9	3	• • •	6
٨	MT ₂	$\frac{1}{9}$	1	3	• • •	7
A =	MT ₃	$\frac{1}{3}$	$\frac{1}{3}$	1		1
	:	:	:	:	1	:
	MT _n	$\frac{1}{9}$	$\frac{1}{7}$	1	• • •	1 _

The Analytical Network Process (ANP), assessing the consistency of pairwise comparison matrices is essential to ensure the reliability of the decision-making process. The consistency ratio (CR) is used to measure the degree of inconsistency in a pairwise comparison matrix. To calculate the consistency ratio, we follow the following steps:

Pairwise Comparison Matrix (A):

First, you should have a pairwise comparison matrix (A), where each element A_{ij} represents the relative importance of element i compared to element j. This matrix is typically derived through expert judgment or surveys. The pairwise comparison matrix is indicated in (3). An example with filled values is shown in (4). In (4), the values are filled from a table of scale that shows the significance of one feature in comparison with other features in a memory technology. It is a nine-level scale, which is shown in Table 3.

Table 3. A scale of 9 levels indicating the importance of a memory technology feature in a memory technology.

Levels	Explanation of a Level Showing the Importance of a Feature
1	Both features have equal importance
2	Moderately important from another feature in another memory technology
3	It indicates a higher importance from the moderate significance of level 2
4	It shows a higher significance from moderate importance in level 3
5	This level reveals significant importance compared to level 4
6	Level 6 shows remarkably higher importance than level 5
7	Level 7 shows higher significance as compared with level 6
8	Level 8 shows excessive importance of a memory feature from another in a memory technology
9	Level 9 indicates excessive importance

Calculate the Weighted Sum (W): Calculate the weighted sum of each column in matrix A. To do this, multiply each element of the matrix by the corresponding weight for the column and then sum the products. Let us denote the weights as w_1, w_2, \ldots, w_n , and matrix A as A. The weighted sum for each column (S_i) is calculated as follows:

$$S_j = \Sigma (A_{ij} \times w_i)$$
 for $i = 1$ to n

where Sj is the weighted sum for column j. A_{ij} is the element in the i-th row and j-th column of matrix A. w_i is the weight assigned to the i-th criterion or alternative. n is the number of criteria or alternatives.

Calculate the Consistency Index (CI): The Consistency Index (CI) is a measure of how consistent the pairwise comparison matrix is. It is calculated using the following formula:

$$CI = (\lambda max - n)/(n - 1)$$

where λ_{max} is the maximum eigenvalue of matrix A. n is the number of criteria or alternatives. To find the maximum eigenvalue, you can use mathematical software or calculators designed for this purpose.

Calculate the Random Index (RI): The Random Index (RI) is a reference value used to determine the level of consistency expected by chance. The RI depends on the order of

the matrix (n), and you can find the corresponding RI values in pre-defined tables. For example, here are some common RI values for different matrix sizes, as shown in Table 4.

Table 4. The standard values for RI.

Number of Features with RI Value				
n = 1: RI = 0	n = 2: RI = 0			
n = 3: RI = 0.58	n = 4: RI = 0.90			
n = 5: RI = 1.12	n = 6: RI = 1.24			
n = 7: RI = 1.32	n = 8: RI = 1.41			
n = 9: RI = 1.45	n = 10: RI = 1.49			

We can use these RI values or refer to tables for larger matrices.

Calculate the consistency ratio (CR):

The consistency ratio (CR) is calculated by dividing the Consistency Index (CI) by the Random Index (RI):

$$CR = CI/RI$$

If CR is less than or equal to 0.10 (some sources may use 0.15 as the threshold), it is generally considered acceptable, indicating an acceptable level of consistency in the pairwise comparison matrix. If CR exceeds this threshold, it suggests a need for reconsidering the judgments and making adjustments to improve consistency.

In summary, the consistency ratio (CR) in the Analytical Network Process (ANP) is a measure of the consistency of a pairwise comparison matrix. It is calculated by comparing the Consistency Index (CI) to a reference value known as the Random Index (RI). A CR below a certain threshold indicates an acceptable level of consistency in the matrix. If the CR exceeds the threshold, it suggests that the judgments may be inconsistent, and revisions may be needed.

3.2. Weighted Supermatrix

The weighted supermatrix is a crucial component in the ANP that captures the relative importance or weight of each criterion and sub-criterion in a decision-making hierarchy. To illustrate this concept as a scholar, let us consider an example of a project selection process for a construction company.

Imagine you are tasked with evaluating different construction projects for your company, and you have identified various criteria such as cost, timeline, environmental impact, and safety. Each of these criteria is further divided into sub-criteria. The weighted supermatrix represents the pairwise comparisons and weights assigned to these criteria and sub-criteria.

In mathematical terms, the weighted supermatrix W is a square matrix where each element W_{ij} represents the relative importance or weight assigned to criterion i compared to criterion j. These weights are typically obtained through expert judgment or surveys. The matrix is consistent when the sum of its elements in each column is equal to 1.

3.3. Limit Supermatrix

The limit supermatrix in the ANP is used to capture the overall influence and interdependence of elements within the decision-making hierarchy. It takes into account both the direct and indirect influences among criteria, sub-criteria, and alternatives.

Continuing with the construction project example, imagine that the cost criterion directly influences the selection of materials, which in turn affects the project timeline and environmental impact. The limit supermatrix represents these complex interactions.

In mathematical terms, the limit supermatrix L is a square matrix where each element L_{ij} represents the total influence of element i on element j, considering all direct and indirect paths. It is obtained by raising the weighted supermatrix W to various powers until it

converges to a stable state. This process accounts for feedback loops and dependencies within the decision hierarchy.

In the analytical network process (ANP), the limit supermatrix gives the final weights of the alternatives or the memory technologies, as in our paper we are making a ranking of it. In our paper, this ranking is shown in Table 5 from the limit supermatrix. The weighted supermatrix was computed through raising it to a higher power till it converged into a matrix containing stable values, i.e., with no further change. The stable matrix is known as a limit supermatrix. The limit matrix shows the weights of the alternatives and the criteria, i.e., the final prioritized values. The limit matrix is the resulting matrix that comprises the final weights measured against each element in the criteria (features of memory) and alternative (memory technologies) clusters. Consequently, it was obtained from the weighted supermatrix by raising the values in the matrix to the power of 2k until we obtained the same values in each row. Herein, *k* denotes any random number.

Table 5. The effect of high rank memory technology with TCAM.

Memory Technology (MT)	Memory Technology with TCAM Weight (Ranking)	Weights from Limit Supermatrix
MT_1	W3	2
MT ₂	W_1	0.5
MT ₃	W_4	3.5
MT_4	W2	1.8
MT ₅	W5	4

The weighted supermatrix and limit supermatrix are fundamental components of the Analytical Network Process (ANP). They play a key role in structuring and evaluating complex decision-making problems, allowing decision-makers to consider both the importance of criteria and the interrelationships among them in a rigorous and systematic manner. Table 5 indicates the final rank values from the limit supermatrix. It shows that the higher weight is for memory technology 5 (MT₅, W₅ with 4) and the lesser weight is indicated for the memory technology 2 (W₁), i.e., MT₅ and MT₂.

4. Simulations and Results

In this section, we provide the effectiveness of combining the suitable memory technology with TCAM. In this regard, we assume that the selected technology (W_5 associated with TM₅) has less delay in writing the flow operations when the link recovery process is performed, and the additional flows are to be written in the flow table in TCAM and its combined memory. To prove our hypothesis, we make an experimental evaluation. Moreover, we compared it with the AHP. Further, we make a network with Abilene, RedIris, USNet, DFN, and Interoute topologies in Mininet [17]. Then we make a link failure in the network. Then we recover that link with the alternate shortest path recovery algorithm (Dijkstra) [18].

Herein, we calculate the recovery delay for these five topologies in Mininet. The equation for recovery delay is given below. The recovery delay is a cumulative sum of the path finding delay, the delay for insertion of flow rules (modified). Figure 3 shows the recovery delay in five topologies. To calculate the recovery delay, we made a link fail in each topology, then we used the Dijkstra shortest path algorithm to find an alternative route for the failed route and calculated this time. Moreover, we also calculated the delay in inserting the flow entries in the flow table of the switches in the data plan. The total delay for each topology is calculated using the two kinds of switches that we rank through the AHP and our proposed methodology. The graphs show that the delay for our proposed ANP strategy is smaller than the AHP scheme. The effectiveness of the proposed method is due to the effective selection of suitable memory technology which reduces the flow insertion time, resulting in the reduction in the total delay. Moreover, the graph also shows

the delay in Abilene is smaller than other layouts. Abilene has the smallest number of nodes and links in the topology. Hence, the complexity of the Abilene network is lower than other topologies, resulting in less delay. In addition, Figure 3 shows the highest recovery delay for the Interoute Network. The Interoute Network has a greater number of nodes and links than other topologies such as RedIris, DFN, and USNet. Hence, the recovery delay is also higher than other networks due to the increased complexity of the Interoute. Equation (5) in this paper shows the recovery delay in milliseconds (ms). The recovery delay is a sum of two parameters, D_p and D_F . D_P shows the path calculation delay, which is for finding the shortest path for the failed link (the alternative shortest path) and D_F shows the flow insertion delay (which is for pushing the flow rules in the TCAM of the SDN switches).



Figure 3. Comparison of recovery delay using proposed ANP scheme and AHP method.

Herein, we want to explain further, to clarify the recovery delay more for the reviewer. Here is a stepwise explanation.

Step 1: In the network topologies (which are used for experiments), we deliberately made a link fail with the link failure command in the Mininet Emulator.

Step 2: The SDN controller periodically checks the link failures with the link layer discovery protocol (LLDP). Consequently, the controller in SDN discovers the failed link through the LLDP.

Step 3: Then, the controller recovers the link, and we calculate the delay which is the recovery delay indicated with D_R . The D_R consists of two parameters, i.e., D_P and D_F . D_P shows an alternative path for the failed link (original path). Moreover, D_F shows the flow insertion delay, i.e., the flow rules for the modified alternative path. The flow insertion for the ANP selected memory takes less time, as indicated in Figure 3. We have cross-calculated the recovery delay in five network topologies to verify that the recovery delay is less than the AHP selected memory technology.

$$D_{\rm R} = D_{\rm P} + D_{\rm F} \tag{5}$$

Figure 4 shows the packets received ratio (PRR). The PRR is calculated using D-ITG utility [19] in the Mininet Emulator. To calculate the PRR, we opened two graphical interfaces utilizing Xming server utility between the source and destination hosts. Then,

we started sending traffic from the sender to the receiver host. Moreover, we made a link fail, and the Dijkstra algorithm recovered it during this activity. However, some packets were lost during the activity due to the link failure. Hence, we recorded the total number of received packets over the total number of sent packets, which we call the PRR. The PRR is calculated using two kinds of switch memory technologies obtained with the AHP and the proposed ANP approach. Our proposed approach shows a high PRR as compared to the AHP. The effectiveness of the proposed method is verified through a high PRR.



Figure 4. Comparison of packets received ratio using proposed ANP scheme and AHP method.

Figure 5 shows the throughput (packets received in 1000 s from sender to receiver hosts) comparison computer using two kinds of switches, i.e., the one with our proposed scheme and one with the AHP strategy. The throughput is greater than the AHP approach. The throughput is calculated using IPERF [20] over a period of 1000 s. The figure shows the average number of packets received at the receiver for a period of 1000 s. The graph shows the results for throughput calculated in the five topologies. The throughput through IPERF is computed by selecting a random pair of sending and receiving hosts attached to the switches in each topology. Then, we started Xming interfaces, sending and receiving hosts, and utilized IPERF utility for computing the throughput.

Figure 6 indicates the jitter calculation of the proposed approach, leveraging ANP with the AHP method. Figure 6 represents that the jitter is less for the proposed approach than the AHP-based method. The jitter is computed during the recovery of networks (Abilene, RedIris, USNet, DFN, and Interoute). During the recovery of these networks, the TCAM was consulted for inserting the flow rules in the switches. Hence, the recovery jitter (time in milliseconds) for the ANP-based method is less because of the effective memory used with TCAM as compared with the AHP. It is also observed from Figure 6 that the jitter is high in layouts or topologies with a higher number of nodes and edges, such as RedIris, USNet, DFN, and Interoute, as compared to Abilene. From Abilene to Interoute, the number of nodes and edges increases towards the Interoute topology, which has the higher jitter compared to Abilene.



Figure 5. Comparison of throughput using proposed ANP scheme and AHP method.



Figure 6. Comparison of recovery jitter using proposed ANP scheme and AHP method.

5. Conclusions

In this paper, we developed a scheme for enhancing the capacity of TCAM with stateof-the-art memory technologies. TCAM is used in OpenFlow switches for storing the flow tables. Hence, when flows are inserted in OpenFlow switches, the appropriate memory technology helps in reducing the flow insertion delay, and jitter. Moreover, the proper selection also increases the throughput and PRR. First, we proposed a method leveraging the ANP mathematical decision-making model to rank the memory technologies according to their features. Then, we used two kinds of switches in the experimental evaluation to validate the effectiveness of the mathematical model, i.e., the one we selected through the ANP and the other one through the AHP. Furthermore, we evaluated the effectiveness of suggested methodology through various experiments in five different topologies emulated in the SDN emulator, i.e., Mininet, where we made a failure in the network (link failure). Then, the network recovered and inserted the updated flow entries in the underlying switches with memory technologies selected using the ANP and AHP. More and more, we have evaluated the recovery delay, jitter, throughput, and PRR through these two kinds of switches, with memory technologies selected with the ANP as well as the AHP. The experiments showed that the memory technology selected through the ANP for TCAM in SDN switches has higher throughput, less delay and jitter, and higher PRR. In a nutshell, the mathematical model experiments were verified by experimental evaluations.

Author Contributions: Conceptualization, A.A. and T.S.C.; data curation, A.A. and T.S.C.; formal analysis, A.A. and T.S.C.; funding acquisition, A.A.; investigation, T.S.C.; methodology, T.S.C.; project administration, T.S.C.; resources, A.A.; software, T.S.C.; supervision, T.S.C.; validation, T.S.C.; writing—original draft, A.A.; writing—review and editing, T.S.C. and A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) under the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2023-RS-2023-00255968) grant and the ITRC (Information Technology Research Center) support program (IITP-2021-0-02051) funded by the Korea government (MSIT). College of Computer Science and Engineering, Yanbu, Taibah University.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- 1. Sethi, D.; Kaur, M.; Singh, G. Design and performance analysis of a CNFET-based TCAM cell with dual-chirality selection. *J. Comput. Electron.* **2017**, *16*, 106–114. [CrossRef]
- Kumar, S.; Noor, A.; Kaushik, B.K.; Kumar, B. Design of Ternary Content Addressable Memory (TCAM) with 180 nm. In Proceedings of the 2011 International Conference on Devices and Communications (ICDeCom), Mesra, India, 24–25 February 2011; pp. 1–5.
- Kreutz, D.; Ramos, F.M.V.; Verissimo, P.E.; Rothenberg, C.E.; Azodolmolky, S.; Uhlig, S. Software-Defined Networking: A Comprehensive Survey. Proc. IEEE 2015, 103, 14–76. [CrossRef]
- Malik, A.; Aziz, B.; Adda, M.; Ke, C.-H. Optimisation Methods for Fast Restoration of Software-Defined Networks. *IEEE Access* 2017, 5, 16111–16123. [CrossRef]
- Zhong, H.; Cao, S.; Yang, H.; Li, X. Dynamic Ternary Content-Addressable Memory Is Indeed Promising: Design and Benchmarking Using Nanoelectromechanical Relays. In Proceedings of the 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), Virtual, 1–5 February 2021.
- 6. Han, R.; Shen, W.; Huang, P.; Zhou, Z.; Liu, L.; Liu, X.; Kang, J. A novel ternary content addressable memory design based on resistive random access memory with high intensity and low search energy. *Jpn. J. Appl. Phys.* **2018**, *57*, 04FE02. [CrossRef]
- Alahmadi, A.; Chung, T.S. Crash Recovery Techniques for Flash Storage Devices Leveraging Flash Translation Layer: A Review. *Electronics* 2023, 12, 1422. [CrossRef]
- 8. Alahmadi, A.; Chung, T.S. RSLSP: An Effective Recovery Scheme for Flash Memory Leveraging Shadow Paging. *Electronics* **2022**, *11*, 4126. [CrossRef]
- Mahendra, T.V.; Hussain, S.W.; Mishra, S.; Dandapat, A. Energy-Efficient Precharge-Free Ternary Content Addressable Memory (TCAM) for High Search Rate Applications. *IEEE Trans. Circuits Syst. I Regul. Pap.* 2020, 67, 2345–2357. [CrossRef]

- Alrushood, H.; Abed, S.; Alabdulhadi, A. Improving Energy Consumption in Content-Addressable Memory Through Precomputation. In Proceedings of the 2023 International Conference on Information Technology (ICIT), Amman, Jordan, 9–10 August 2023; pp. 68–72.
- Zhan, H.; Wang, C.; Cui, H.; Liu, X.; Liu, F.; Cheng, X. High-Speed and Energy-Efficient Single-Port Content Addressable Memory to Achieve Dual-Port Operation. In Proceedings of the 2023 Design, Automation & Test in Europe Conference & Exhibition (DATE), Antwerp, Belgium, 17–19 April 2023; pp. 1–6.
- 12. Ali, J.; Shan, G.; Gul, N.; Roh, B.-H. An Intelligent Blockchain-based Secure Link Failure Recovery Framework for Software-defined Internet-of-Things. J. Grid Comput. 2023, 21, 57. [CrossRef]
- Shim, M.; Jeong, J.; Song, J.; Kim, Y.; Choi, W. Segmented Match-Line and Charge-Sharing Based Low-Cost TCAM. *IEEE Trans. Circuits Syst. II Express Briefs* 2022, 69, 5104–5108. [CrossRef]
- Ali, J.; Roh, B.H. An Effective Approach for Controller Placement in Software-Defined Internet-of-Things (SD-IoT). Sensors 2022, 22, 2992. [CrossRef] [PubMed]
- Ali, J.; Roh, B.-H.; Lee, S. QoS improvement with an optimum controller selection for software-defined networks. *PLoS ONE* 2019, 14, e0217631. [CrossRef] [PubMed]
- 16. Manik, M.H. Addressing the supplier selection problem by using the analytical hierarchy process. *Heliyon* **2023**, *9*, e17997. [CrossRef] [PubMed]
- De Oliveira, R.L.S.; Shinoda, A.A.; Schweitzer, C.M.; Prete, L.R. Using Mininet for Emulation and Prototyping Software-Defined networks. In Proceedings of the IEEE Colombian Conference on Communication and Computing (COLCOM), Bogota, Colombia, 4–6 June 2014; pp. 1–6.
- 18. Ali, J.; Lee, G.-M.; Roh, B.-H.; Ryu, D.K.; Park, G. Software-Defined Networking Approaches for Link Failure Recovery: A Survey. *Sustainability* 2020, *12*, 4255. [CrossRef]
- Botta, A.; Dainotti, A.; Pescape, A. A Tool for the Generation of Realistic Network Workload for Emerging Networking Scenarios. In *Computer Networks*; Elsevier: Amsterdam, The Netherlands, 2012; Volume 56, pp. 3531–3547.
- 20. Iperf. Available online: https://iperf.fr/ (accessed on 20 September 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.