

## FRM: Foundation-policy Recommendation Model to Improve the Performance of NAND Flash Memory

Won Ho Lee\*, Jun-Hyeong Choi\*\*, Jong Wook Kwak\*\*\*

\*CTO, Solution Development Division, NFINITY7 Inc., Daegu, Korea

\*\*Senior Researcher, Research Development Division, Gyeongbuk Institute of IT Convergence Industry Technology(GITC), Gyeongsan, Korea

\*\*\*Professor, Dept. of Computer Engineering, Yeungnam University, Gyeongsan, Korea

### [Abstract]

Recently, NAND flash memories have replaced magnetic disks due to non-volatility, high capacity and high resistance, in various computer systems but it has disadvantages which are the limited lifespan and imbalanced operation latency. Therefore, many page replacement policies have been studied to overcome the disadvantages of NAND flash memories. Although it is clear that these policies reflect execution characteristics of various environments and applications, researches on the foundation-policy decision for disk buffer management are insufficient. Thus, in this paper, we propose a foundation-policy recommendation model, called FRM for effectively utilizing NAND flash memories. FRM proposes a suitable page replacement policy by classifying and analyzing characteristics of workloads through machine learning. As an implementation case, we introduce FRM with a disk buffer management policy and in experiment results, prediction accuracy and weighted average of FRM shows 92.85% and 88.97%, by training dataset and validation dataset for foundation disk buffer management policy, respectively.

▶ **Key words:** page replacement policy, disk buffer, NAND flash memory, workload analysis, machine learning

### [요 약]

최근, 낸드 플래시 메모리는 비휘발성, 높은 집적도, 높은 내구성으로 인하여 다양한 컴퓨터 시스템에서 자기 디스크를 대체하고 있지만 연산 처리 속도 불균형 및 수명 제한과 같은 한계를 가진다. 따라서 낸드 플래시 메모리의 단점을 극복하고자 디스크 버퍼 관리정책들이 연구되고 있다. 비록 이러한 관리정책들이 다양한 작업 환경과 응용 프로그램의 실행 특성을 반영하는 것은 명확하나, 이들을 위한 기초 관리 정책 결정 방식에 대한 연구는 그에 비하면 미흡하다. 본 논문에서는 낸드 플래시 메모리를 효율적으로 활용하기 위한 기초 관리정책 제안 모델인 FRM을 소개한다. FRM은 워크로드를 다양한 특성에 따라 분석하고 낸드 플래시 메모리가 가지는 특성들과 조합하는 모델로, 이를 통해 작업 환경에 가장 알맞은 기초 관리 정책을 제시한다. 결과적으로 제안하는 모델은 학습 데이터와 검증 데이터에 대해 Accuracy와 Weighted Average 측면에서 각각 92.85%와 88.97%의 기초 관리정책 예측 정확도를 보여주었다.

▶ **주제어:** 페이지 교체 정책, 디스크 버퍼, 낸드 플래시 메모리, 워크로드 분석, 머신러닝

- 
- First Author: Won Ho Lee, Co-author: Jun-Hyeong Choi, Corresponding Author: Jong Wook Kwak  
\*Won Ho Lee (cto@nfinity7.io), Solution Development Division, NFINITY7 Inc.
  - \*\*Jun-Hyeong Choi (jhchoi@gitc.or.kr), Research Development Division, Gyeongbuk Institute of IT Convergence Industry Technology(GITC)
  - \*\*\*Jong Wook Kwak (kwak@yu.ac.kr), Dept. of Computer Engineering, Yeungnam University
  - Received: 2023. 06. 28, Revised: 2023. 07. 24, Accepted: 2023. 07. 24.

## I. Introduction

최근 기존의 자기 디스크를 대체하기 위한 보조 기억 장치로 낸드 플래시 메모리(NAND flash memory)가 주목 받고 있다. 낸드 플래시 메모리는 전력이 공급되지 않더라도 저장된 데이터가 보존되는 비휘발성 메모리(Non-volatile memory) 중 하나이며, 물리적 장치를 내장하고 있지 않기 때문에 자기 디스크와 비교하면 내구성과 성능 측면에서 신뢰도가 높다. 또한, 낸드 플래시 메모리는 데이터를 처리하는데 오로지 전기적 신호만을 이용하기 때문에 기존의 자기 디스크에 비해 높은 집적도를 가진다. 결과적으로 임베디드 기기, 웨어러블 디바이스 그리고 DBMS 등과 같이 높은 밀도, 빠른 처리 속도 및 데이터에 대한 신뢰성을 요구하는 분야에서 낸드 플래시 메모리의 수요가 폭넓게 증가하고 있다 [1, 2].

그러나 낸드 플래시 메모리는 구조부터 기존의 저장장치들과는 다르므로 다음과 같은 특징들이 도입 이전에 고려되어야 한다. 첫째, 낸드 플래시 메모리는 자기 디스크와 달리 연산의 종류에 따른 처리 속도가 불균형하다. 단순한 읽기 연산에 비해 데이터를 기록하는 쓰기 연산이 느리며, 지우기 연산은 그보다 더 느리다. 둘째, 낸드 플래시 메모리는 각 페이지에 대한 쓰기 연산 횟수가 제한된다. 비록 낸드 플래시 메모리가 FTL(Flash translation layer)의 웨어 레벨링(Wear leveling) 및 가비지 콜렉션(Garbage collection)을 통해 쓰기 연산에 의한 문제들을 완화하고는 있지만, 잦은 쓰기 연산은 필연적으로 낸드 플래시 메모리의 수명을 단축시키며 이는 결국 시스템의 성능을 저해하게 된다 [3].

이러한 낸드 플래시 메모리의 한계들을 개선하고자 디스크 버퍼에 관한 연구가 활발히 이루어지고 있다. 만약 디스크 버퍼를 더 효율적으로 운용할 수 있는 관리정책을 고안할 수 있다면, 낸드 플래시 메모리의 한계가 개선되고 시스템의 성능 역시 잠재적으로 향상되리라 기대할 수 있다. 그러나 시스템마다 이용되는 환경 및 특징이 모두 독립적이기 때문에, 각각의 시스템에 적합한 관리정책 역시 다를 수밖에 없다. 그러므로 목표 시스템에 적합한 디스크 버퍼 관리정책을 매년 새로이 고안하는 것은 불필요한 자원 낭비를 초래할 수 있다.

본 논문에서는 이러한 문제를 해결하고자 낸드 플래시 메모리의 수명 및 성능의 향상에 기여할 수 있는 기초 관리정책 제안 모델, FRM(Foundation-policy Recommendation Model)을 소개한다. FRM은 입력되는 워크로드를 특정 시스템의 작업 패턴으로 간주하여 고유

한 특징들로 분류하고, 낸드 플래시 메모리 연산들과의 상관관계를 분석한다. 이후 머신러닝을 통해 분석된 특성들을 학습시켜, 입력된 워크로드에 적합한 기초 관리정책을 제시한다.

이하 본 논문의 구성은 다음과 같다. 2장에서는 낸드 플래시 메모리의 특성, 디스크 버퍼에 대한 배경지식, 그리고 이전의 관련 연구들을 소개한다. 3장에서는 낸드 플래시 메모리를 위한 기초 관리정책을 제안하는 모델, 즉 FRM과 디스크 버퍼로의 구체적 적용 사례를 기술한다. 4장에서는 제안 모델의 성능을 평가하기 위한 실험 환경 설정 및 실험 평가를, 마지막 5장에서는 결론과 향후 연구 방향에 대해 서술한다.

## II. Background & Related works

### 1. Background

#### 1.1. NAND flash memory

자기 디스크의 대안인 낸드 플래시 메모리는 전기적 신호만으로 데이터를 다루기 때문에 빠른 연산 처리 속도 및 높은 집적도 등의 장점을 가지나 다음과 같은 단점들도 갖는다. 첫 번째는 낸드 플래시 메모리의 연산 처리 속도 불균형이다. 낸드 플래시 메모리는 빠른 읽기 연산 속도를 가지지만 상대적으로 느린 쓰기 및 지우기 연산 속도를 가진다. 이러한 특징은 메인 메모리와 디스크 버퍼에 많이 이용되는 DRAM과 비교하면 더욱 두드러진다. 낸드 플래시 메모리의 두 번째 단점은 제한된 수명이다. 낸드 플래시 메모리는 데이터를 저장하는 셀의 플로팅 게이트에 가해지는 전기적 신호를 통해 데이터를 저장한다. 그러나 데이터를 저장할 때 발생시키는 전압이 플로팅 게이트에 대한 손상을 유발하고, 누적되는 손상에 의해 결국 데이터의 저장이 불가능한 상황에 이르게 된다. 표 1은 앞서 언급된 낸드 플래시 메모리와 DRAM의 차이, 그리고 낸드 플래시 메모리의 특징을 나타낸다 [4-6].

Table 1. The descriptions of DRAM and NAND flash memory's operation latency

Operation	Unit	DRAM	NAND flash	
		latency	latency	P/E cycle
Read	Page	15 ns	21 $\mu$ s	< 10 <sup>5</sup>
Write	Page	15 ns	200 $\mu$ s	
Erase	Block	-	1.2 ms	

Table 2. The consideration features of each foundation-policy

Feature	LRU	CLOCK	FCFS	LFU
Temporal locality	Very high	High	Low	High
Spatial locality	High	Very high	High	Low
Order oriented	Low	Low	Very high	Low
Frequency oriented	Low	Low	Low	Very high

### 1.2. Disk buffer

디스크 버퍼는 시스템의 호스트와 보조 기억 장치의 연산 속도 차이를 보완하고자 고안된 장치이다. 디스크 버퍼는 요청이 자주 발생할만한 페이지들을 되도록 많이, 오랫동안 유지할수록 성능상 좋다. 이는 호스트에서 요청된 페이지가 상대적으로 느린 보조 기억 장치를 경유하지 않을 수 있도록 완충 역할을 해줌으로써 시스템의 효율이 증대되기 때문이다. 따라서 디스크 버퍼를 위한 효율적인 페이지 교체 정책에 관해 많은 연구가 이루어지고 있다. 이러한 기존의 디스크 버퍼 관리정책들은 그 특성 및 기반에 따라 표 2와 같이 나누어 구분할 수 있다 [7].

LRU 정책은 가장 최근까지 참조된 페이지들을 보존하기 때문에 시간 지역성(Temporal locality) 및 공간 지역성(Spatial locality)을 보존한다는 특징을 가진다. 그러나 새로운 단일 요청에 의해 리스트의 순서가 훼손될 수 있고, 참조 비트를 따로 두지 않기 때문에 참조 빈도를 고려하기는 힘들다 [8].

CLOCK 정책은 LRU와 다르게 환형 리스트를 통해 페이지들을 관리하며, 재참조 비트를 통해 시간 지역성을 고려한다. 그러나 퇴거를 제외한 다른 방식으로 인해 저장된 페이지의 위치가 변경되지 않으므로 LRU보다 공간 지역성을 고려하는데 우세하다. 또한 참조 빈도를 이용하지는 않기 때문에 빈도 지향성과는 거리가 멀다 [9].

FCFS는 큐(Queue)와 같이 선입선출 방식으로 작동하기 때문에 리스트의 순서가 훼손될 수 없으며, 이 때문에 순서 지향적이고 점진적으로 이동하는 공간 지역성을 반영한다고 볼 수 있다. 다만, 참조에 따른 고려가 포함되지 않기 때문에 다른 기초 관리정책들에 비해 빈도 지향성과 시간 지역성을 고려하지는 못한다 [10].

LFU는 각 페이지들의 참조 횟수를 통해 리스트를 관리하므로 빈도 지향성이 매우 강하며, 동일한 이유로 시간 지역성 역시 반영된다. 다만 기법상 참조 횟수만을 관리하므로 공간 지역성 및 순서 지향성에 대한 고려는 빈약하다 [11].

기존의 디스크 버퍼 관리 정책에 관한 연구들은 대부분 LRU, CLOCK, FCFS, 그리고 LFU를 기반으로 하며, 이로 인해 언급된 기초 관리정책들을 개선 및 변형하더라도 그

기본 성질은 달라지지 않는다 [8-11]. 이러한 이유로 디스크 버퍼 관리정책은 적용 시스템의 특징에 따라 기초 관리정책에 대한 이해를 동반한다. 이는 각 기초 관리정책이 가지는 특징 및 성질에 대한 이해가 곧 목표 시스템의 성능을 보장하기 위한 초석이 되기 때문이다.

## 2. Related works

워크로드의 실행 패턴 및 흐름을 분석하기 위해 많은 선행 연구가 이루어졌는데, 그 중 가장 최근에 행해졌으며 주목할만한 점을 가지는 연구들은 다음과 같다.

Mojtaba Tarihi et al. 가 제안한 분류 기법은 워크로드의 요청을 저장하는 큐를 이용해 각 요청을 Sequence, Stride, Overlapped, 그리고 Random 등으로 나눈다. 또한 이렇게 분류된 각 인자가 캐시를 관리하는 데 있어 지역성 및 지향성에 따른 영향을 분석하였다. 하지만 워크로드의 다양한 특성들과 낸드 플래시 메모리의 특성들이 가지는 상관관계 분석이 목표는 아니므로, 낸드 플래시 메모리의 성능 개선과는 거리가 있다 [12].

Reza Salkhordeh et al. 가 제안한 ReCA의 경우 워크로드 분석 결과를 통해 목표 시스템의 유형을 결정한다. 이렇게 분류된 워크로드의 유형은 파일 시스템 캐시의 구성을 결정하는 데 이용되며, 작업 유형에 따라 동적으로 그 크기를 변화시킬 수 있도록 한다. 그러나 ReCA는 워크로드의 유형을 결정짓는 기준을 캐시 메모리의 관점에서 다섯 가지로 한정했기 때문에 낸드 플래시 메모리의 특징을 반영하기 위한 새로운 분석 및 유형 분류가 필요하다 [13].

앞서 언급된 기존의 연구들은 워크로드의 유형을 판단하여 목표 시스템의 성능을 개선시키기 위해 수행되었다. 그러나 각각의 개선 대상이 본 논문의 목표 대상인 낸드 플래시 메모리 기반 저장장치와는 다르다. 따라서 본 논문에서는 앞선 연구들과 비교하여 낸드 플래시 메모리와 디스크 버퍼에 특화된 방식으로 워크로드의 유형을 새롭게 정의하고, 더 나아가 상관관계 분석 및 머신 러닝을 이용해 낸드 플래시 메모리 및 디스크 버퍼의 특성에 적합한 기초 관리정책 제안 모델을 제시하고자 한다.

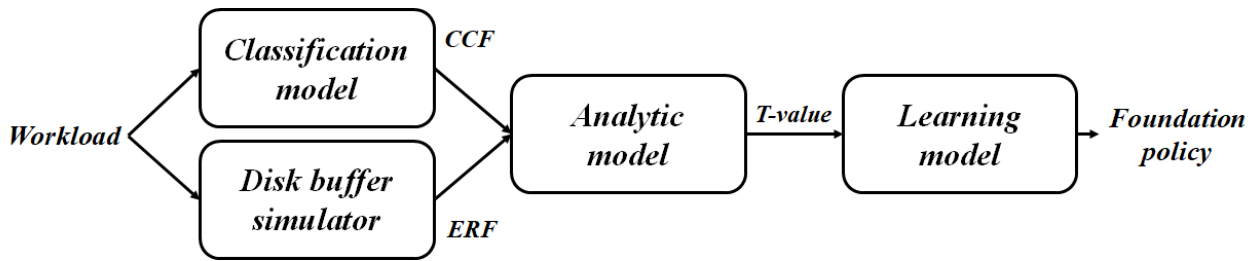


Fig. 1. The diagram and structure of FRM

### III. Foundation-policy Recommendation Model

#### 1. FRM structure

본 논문에서는 낸드 플래시를 위한 기초 정책 제안 모델인 FRM을 소개하고, 이를 디스크 버퍼 정책에 적용한다. FRM은 역할에 따라 네 가지 모듈로 구성된다. 첫 번째는 분류 모델(Classification model)로, 입력된 워크로드를 각 요청별 고유한 특성으로 분류하고 정량화한다. 두 번째는 디스크 버퍼 시뮬레이터(Disk buffer simulator)로, 워크로드의 각 요청을 시뮬레이션 결과에 따라 분류한다. 세 번째는 분석 모델(Analysis model)로, 분류된 워크로드의 특성들을 낸드 플래시 메모리의 각 연산 결과들과 대조해 상관관계를 특정한다. 특정된 상관관계는 각 워크로드가 가지는 성질을 분석한 결과로 출력되며, 이는 이후 신경망에 입력되어 학습 모델이 워크로드에 적합한 기초 관리정책을 추론하는 데 사용된다. 네 번째는 학습 모델(Learning model)로, 앞서 정의한 워크로드의 분석적 특성, 그리고 이를 통해 결정된 적합한 기초 관리정책을 머신러닝으로 학습한다. 최종적으로 각 워크로드의 특성 및 기초 관리정책에 대해 학습한 학습 모델은 완전히 새로운 유형의 워크로드가 입력되더라도 그에 알맞은 기초 관리정책을 추론하고 결과로 제시하게 된다. 제안하는 모델의 전체적인 구조는 그림 1과 같다.

#### 2. Classification model

분류 모델은 입력된 워크로드의 각 요청을 고유한 특성 중 하나로 분류한다. 비록 이전에도 워크로드를 정량화하기 위한 많은 시도가 있었으나 이들은 실험 대상으로 준비된 워크로드만을 고려했기 때문에 새로운 워크로드가 입력될 때를 고려하지는 않았다. 그러나 FRM의 분류 모델은 입력된 워크로드의 각 요청을 *Regularity*, *Range*, 그리고 *Operation*과 같은 *CCF*(Combining Classification Factor: 조합 분류 인자)들을 이용해 총 16가지의 특성 중 하나로 분류하므로 유형을 정의할 수 있다. 이때, FRM이

워크로드의 각 요청을 분류할 때 사용하는 *CCF*는 표 3과 같이 정의되어 조합된다.

Table 3. The combining classification factors

Regularity	Range	Operation
Seq (Sequence)	H (Huge)	R (Read)
Str (Stride)		
Ovl (Overlap)	T (Tiny)	W (Write)
Rnd (Random)		

첫 번째 *CCF*는 *Regularity*로, 이는 이전에 발생한 요청들과 새로운 요청 사이의 관계를 정의한다. 만약 요청들이 서로 연결된다면 새로운 요청은 *Seq*(Sequence)로 분류되고, 주소값이 1만큼 떨어진 상태에서 연결된다면 *Str*(Stride)로 분류된다. 반면, 서로 같은 주소를 공유하는 이전의 요청이 있다면 새로운 요청은 *Ovl*(Overlap)로 분류되며, 서로 간 어떠한 관계도 없다면 *Rnd*(Random)로 분류된다. 두 번째 *CCF*인 *Range*는 새로운 요청이 요구하는 페이지의 크기를 나타낸다. 만약 새로운 요청이 워크로드의 평균 페이지 요청 크기보다 많은 페이지를 요청하면 *H*(Huge)로 분류되고 작은 페이지를 요청하면 *T*(Tiny)로 분류된다. 마지막 *CCF*인 *Operation*은 새로운 요청이 가지는 연산의 형태를 나타낸다. 새로운 요청이 읽기라면 *R*(Read)로, 쓰기라면 *W*(Write)로 분류한다.

표 4는 총 5가지의 요청을 가지는 워크로드의 예시이다. 해당 예시에서 사용된 워크로드는 모든 요청 크기의 합이 9이며 평균 1.8의 페이지 요청 크기를 갖는다. 이러한 이유로 1.8보다 큰 요청 크기를 갖는 요청 1, 3, 4의 *Range*는 *H*로 분류된다.

그림 2는 표 4에 나타난 예시 워크로드의 각 요청이 FRM에 의해 분류되는 방식을 나타낸 시나리오이다 [13]. 요청 1, 2는 이전의 요청들과 어떠한 관계도 갖지 않기 때문에 *Regularity*를 *Rnd*으로 설정한다. 반면, 요청 3의 경우 요청 2와 1만큼의 주소값 차이를 보이므로 *Str*로 분류

하고, 요청 4는 요청 3과 주소값이 겹치기 때문에 *Ov*로 분류한다. 마지막 요청 5는 최초의 요청 1과 연속되므로 *Regularity*를 *Seq*로 할당한다.

Table 4. The request list for instance workload

No.	Address	Request size	Range (H/T)	Operation (R/W)
1	9	3	H	R
2	1	1	T	W
3	3	2	H	W
4	4	2	H	R
5	8	1	T	R

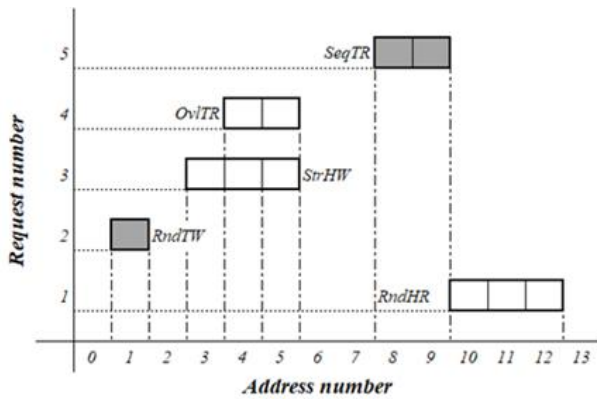


Fig. 2. The classification scenario example

### 3. Analysis model

분석 모델에서는 앞선 분류 모델에서 얻어진 워크로드의 분류 결과 및 모의실험 결과 값을 이용해 *Correl*(Correlation coefficient)을 도출한다. 이렇게 도출된 *Correl*은 입력된 워크로드 및 각 기초 관리정책이 가지는 상관관계를 의미하며, 이후 디스크 버퍼를 운용하는데 가장 적합한 기초 관리정책을 판단하는데 사용된다. 실험에 이용된 기초 관리정책들은 분석 대상인 워크로드의 지역성, 지향성 및 빈도성이 다양하게 고려될 수 있도록 LRU, CLOCK, FCFS, 그리고 LFU로 구성된다. 또한, 각 기초 관리정책을 이용한 모의실험은 본 연구를 위해 수정된 이벤트 구동형 디스크 버퍼 시뮬레이터(Event-driven Disk Buffer Simulator)를 통하여 수행하였다 [14].

워크로드의 *Correl*은 이전의 단계에서 16개의 *CCF*로 분류된 각 요청들의 개수와 동일 워크로드를 이용한 실험 결과에서 얻어지는 *ERF*(Experimental result factor; 실험 결과 인자)들을 통해 도출된다. 이때, *ERF*는 네 가지 기초 관리정책마다 각각 *RH*(Read hit), *WH*(Write hit), *FR*(Flash read), 그리고 *FW*(Flash write)로 구성되기 때문에 총 16가지의 인자로 구성될 수 있다.

$$\begin{aligned} & \text{Compatibility}(\text{Policy}) \\ &= \text{Weight}_W(\text{Correl}_{WH\text{Policy}} + \text{Correl}_{FW\text{Policy}}) \\ &+ \text{Weight}_R(\text{Correl}_{RH\text{Policy}} + \text{Correl}_{FR\text{Policy}}) \end{aligned} \quad (1)$$

*CCF* 및 *ERF*를 통해 도출된 *Correl*은 각 *CCF*에 적합한 기초 관리정책을 책정하는 데 이용된다. 제안하는 모델은 수식 (1)을 통해 워크로드와 기초 관리정책의 적합도 (*Compatibility*)를 구한다. 이때, *Weight<sub>W</sub>*(쓰기 가중치) 및 *Weight<sub>R</sub>*(읽기 가중치)는 낸드 플래시 메모리가 쓰기 연산에 불리한 것을 반영하여 각각 0.7 그리고 0.3으로 정의된다.

앞서 수식 (1)을 통해 구한 *CCF*의 *Compatibility*는 이후 학습 모델에 더 적합한 one-hot 형태의 예측 정답 값으로 재가공된다. 수식 (2)는 각 *CCF*와 정책에 대한 *Compatibility*를 비교해 순위를 매겨 경우의 수를 따지고, 해당 경우의 수에 해당하는 예측 정답 값인 *T-value*를 출력한다. 이때, *T-value*가 취할 수 있는 모든 형태의 경우의 수는 4! 즉, 24개가 된다.

$$\begin{aligned} T\text{-value} = \text{RankCase}(\{ & \text{Compatibility}(\text{LRU}), \\ & \text{Compatibility}(\text{CLOCK}), \\ & \text{Compatibility}(\text{FCFS}), \\ & \text{Compatibility}(\text{LFU})\}) \sim (0,23) \end{aligned} \quad (2)$$

결과적으로 만들어지는 *T-value*는 입력된 워크로드에 대한 *CCF*, *ERF*, 그리고 이들을 통해 도출할 수 있는 *Correl* 등을 모두 아우르는 하나의 함축적인 값이 되며, 학습 모델에서 워크로드를 나타내는 대푯값이 된다.

### 4. Learning model

학습 모델은 분석 모델에서 정의된 워크로드의 *CCF*와 *T-value*를 이용하여 신경망을 구축하며, 해당하는 인공지능 신경망의 상세는 표 5와 같다. 해당 신경망은 최초 입력으로 16개의 *CCF* 값들을 받으며, 총 세 개의 은닉층을 경유해 나간다. 활성화함수로 Sigmoid를 사용하는 최초의 은닉층을 제외한 나머지 은닉층에서는 ReLU를 사용하며, 출력층에서는 Softmax를 통해 총 24개의 값을 생성한다. 이렇게 생성된 24개의 Softmax 결과값은 모든 기초 관리정책 순위에 대한 적합도를 나타내며, 각 정책의 우선순위를 경우의 수, 즉 *T-value*로 책정한다.

이처럼 FRM의 신경망은 최초 구축을 위해 기존에 존재하는 워크로드들을 입력받아 앞선 분류 모델 및 분석 모델에서 얻어지는 각 워크로드의 *CCF* 및 *T-value*를 학습한다. 결과적으로, 이러한 과정을 통해 구축된 신경망은 임

Table 5. The details of hyper parameters for deep neural network

	Input layer	Hidden layer 1	Hidden layer 2	Hidden layer 3	Output layer
# of nodes	16	32	64	32	24
Activator	-	Sigmoid	ReLU	ReLU	Softmax
Loss	Categorical-Crossentropy				
Optimizer	Adam				

의 워크로드를 입력받아도 해당 워크로드를 운용하기에 적합한 기초 관리정책을 제시할 수 있게 된다.

## IV. Performance evaluation

### 1. Experimental environment

#### 1.1. Workload

제안하는 모델은 총 세 가지의 세부 모델로 분류되며 입력값 및 출력값이 서로 연계된다는 특징을 가진다. 따라서 우수한 학습 모델을 얻고자 한다면 분석에 적합한 최적의 워크로드들을 이용해 분류 및 분석 결과를 만들 필요가 있다. 이러한 이유로 본 논문에서 제안되는 모델의 입력은 디스크 버퍼의 실험을 수행할 때 일반적으로 이용되는 워크로드들을 수집하여 구성하였으며 그 목록 및 상세는 표 6과 같다 [15, 16].

총 32가지로 구성된 실험용 워크로드들은 FRM에서 분류 모델 및 분석 모델을 경유하며 도출되는 *CCF* 및 *Correl* 을 이용해 각 *CCF*에 알맞은 *T-value*를 책정하는데 이용되며, 실험의 마지막에 구축되는 학습 모델에 입력되어 구성된 신경망의 예측 정확도를 평가하는 데 이용된다.

### 2. Experiment results

#### 2.1. Classification model

분류 모델에서는 워크로드들에 대한 *CCF*들을 표 7에서와 같이 계산하여 각 유형을 특징한다. 제안하는 모델에서는 *CCF*를 책정하기 위해 64개의 요청을 저장하는 큐를 구성하고, 이를 통해 이전의 요청들과 새로운 요청을 구분지어 관계를 결정짓는다. 또한 평균 페이지 요청 크기를 기준으로 하여 각 요청의 *Range*를 결정하고, 연산 종류를 통해 *Operation*을 결정한다. 그림 3은 실험에 이용한 워크로드 중 특징적인 것들을 선택해 방사형 그래프로 나타낸 것이며, 그래프의 각 수치는 해당 워크로드가 가지는 *CCF*의 비율과 같다. 그래프를 통해 워크로드 *Financial2*는 무작위적이고 크기가 큰 쓰기 요청에 치중된 모습을 보이며, *online*은 연속적이고 크기가 큰 쓰기 요청에 치중되어 있음을 알 수 있다. 그리고 *home3*는 주소가 중복되며 크기가 큰 쓰기 요청이 집중적으로 발생함을 알 수 있다.

#### 2.2. Analysis model

분석 모델에서는 분류 모델에 의해 그 유형 및 특성이 결정된 표 7의 워크로드들을 활용하며, 해당 워크로드들의 각 *CCF* 및 *ERF*들을 이용해 상관관계를 도출한다. 그림 4는 그림 3에 해당하는 워크로드들의 *ERF*를 나타내며, 그

Table 6. The details of experimental workloads

Name	Footprint ratio (R/W)	Average request size	Workload size (MB)	Name	Footprint ratio (R/W)	Average request size	Workload size (MB)
Financial1	0.87/0.13	7.09	137.73	rsrch_1	1/0	24.35	0.44
Financial2	0.23/0.77	4.59	94.04	rsrch_2	0.21/0.79	8.17	6.41
hm_1	0.23/0.77	30.22	18.88	src2_0	0.79/0.21	14.41	48.95
home1	0.06/0.94	1.59	127.99	ts_0	0.58/0.42	18.01	56.56
home2	0.29/0.71	3.19	44.75	wdev_0	0.41/0.59	18.45	35.64
home3	0.05/0.95	5.94	3.51	wdev_2	1/0	16.29	5.75
home4	0/1	3.15	16.66	wdev_3	0.44/0.56	10.52	0.02
homes	0.35/0.65	9.35	394.82	web_1	0.11/0.89	58.14	5.21
mail-02	0.12/0.88	9.98	96.10	web_3	0.37/0.63	76.28	1.00
mail-03	0.11/0.89	10.85	100.10	webmail	0.34/0.66	1.00	449.37
mail-04	0.16/0.84	10.26	107.72	webresearch	0/1	8.00	64.54
mail-05	0.15/0.85	9.97	134.56	webresearch-o	0/1	3.23	21.26
mds_0	0.5/0.5	18.39	37.84	WebSearch	0/1	30.07	273.01
online	0.31/0.69	8.00	112.53	WebSearch1	0/1	30.29	28.50
online-vm-mix	0.3/0.7	4.15	834.87	webusers	0.15/0.85	8.47	152.95
prxy_0	0.93/0.07	7.32	387.26	web-vm-o	0.3/0.7	4.15	75.06

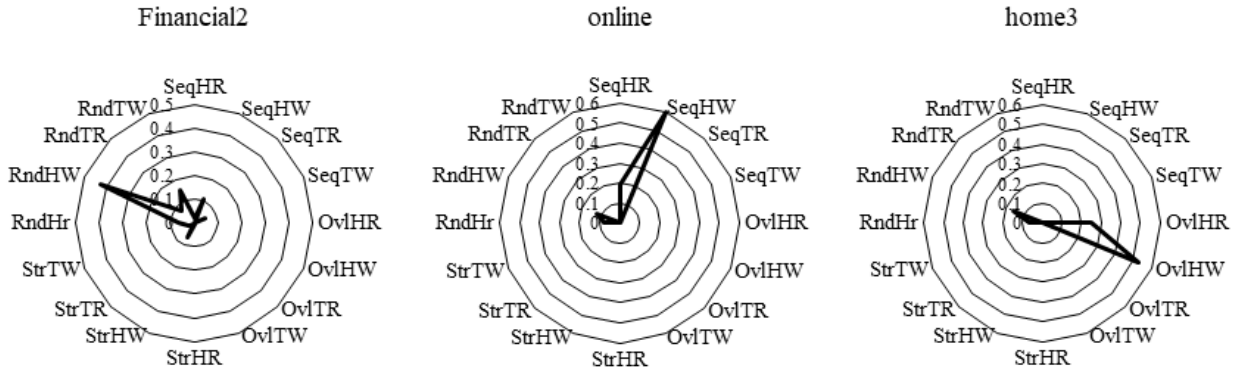


Fig. 3. The CCF of Financial2, online, and home3

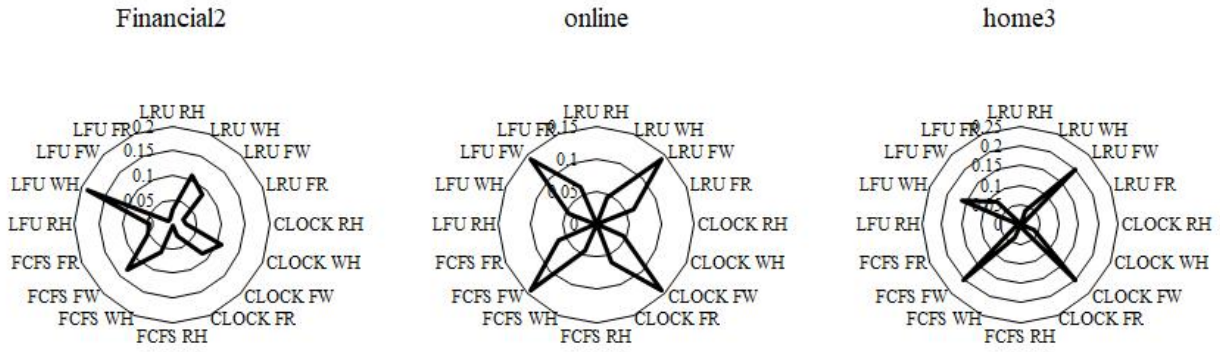


Fig. 4. The ERF of Financial2, online, and home3

래프의 각 수치는 디스크 버퍼 및 플래시 메모리에서 발생한 연산과 같다. 그래프를 통해 Financial2는 LFU를 기초 관리정책으로 삼을 때 쓰기 요청을 가장 잘 처리한 반면, online은 어떤 정책을 써도 유사한 실험 결과를 보여주며, 마지막 home3는 LRU, CLOCK, FCFS에서 유사한 결과를 나타낸다. 결과적으로 그림 3과 그림 4의 결과를 취합하면 RndHW는 LFU WH와, SeqHW는 모든 기초 관리정책의 FW와, 그리고 OvlHW는 LFU를 제외한 모든 기초 관리정책의 FW와 밀접한 관계에 있음이 밝혀진다. 이러한 분석을 모든 워크로드에 적용하여 정책별 상관 관계를 도출한 Correl이 그림 5와 같으며, 이는 실험에 이용된 전체

32개의 워크로드에 대한 CCF와 ERF의 관계를 포함하게 된다. 각각의 CCF는 양의 상관 관계를 맺을수록 ERF에 대해 연한색을 띄고, 그 반대의 경우라면 진한색을 띄게 되는데, 이는 각 요청에 적합한 기초 관리 정책을 판단하기 위한 기저가 된다.

이처럼 도출된 Correl 수치는 각 워크로드에 적합한 기초 관리정책을 판단하기 위해 이용되는데, 이를 위해 각 CCF에 적합한 기초 관리정책을 설정해야 한다. 기초 관리정책은 수식 (1)의 CCF에 대한 ERF 수치를 대입하여 CCF와의 적합도를 계산할 수 있으며, 그 결과가 앞서 소개한 표 7이다. 이렇게 계산된 적합도를 수식 (2)에 대입하

Table 7. The compatibility of each CCF

CCF	LRU	CLOCK	FCFS	LFU	CCF	LRU	CLOCK	FCFS	LFU
RndHR	0.6034	0.6033	0.5992	0.5849	RndTW	0.5662	0.5662	0.5446	0.5368
StrTR	0.5709	0.5711	0.5486	0.5459	SeqHR	0.5625	0.5626	0.543	0.5322
SeqTR	0.3815	0.3837	0.3782	0.3686	OvlHW	0.5349	0.5352	0.5281	0.5019
SeqTW	0.5797	0.58	0.5623	0.5623	OvlHR	0.5041	0.5046	0.5253	0.491
StrTW	0.5831	0.583	0.5504	0.5702	SeqHW	0.5598	0.5604	0.5947	0.507
StrHR	0.5763	0.5762	0.5434	0.5753	OvlTW	0.5746	0.5744	0.5421	0.5752
StrHW	0.5746	0.5744	0.5421	0.5752	OvlTR	0.5519	0.5641	0.5898	0.5022
RndTR	0.5738	0.5739	0.5484	0.5477	RndHW	0.3996	0.4009	0.3962	0.4603

면 각 CCF에 적합도 순위를 결정할 수 있게 되는데, 그 결과는 표 8과 같다. 결과적으로 각 CCF는 적합한 기초 관리정책에 분배되며, 해당 분배를 통해 각 워크로드의 유형 즉, T-value를 결정할 수 있게 된다.

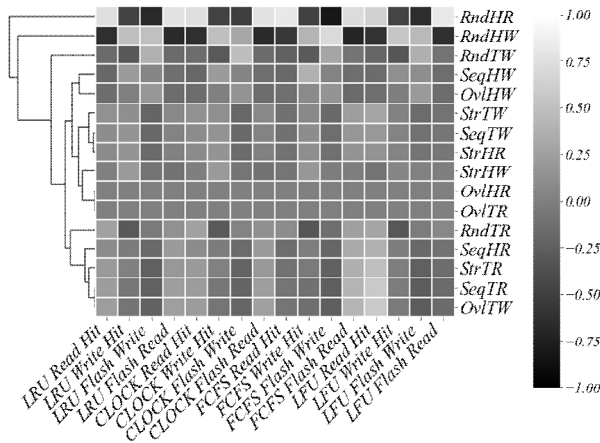


Fig. 5. Correlation coefficient of CCF and ERF

Table 8. The distribution of CCFs for each policy

	LRU	CLOCK	FCFS	LFU
CCF	RndHR	StrTW	OvIHW	OvITW
	StrTR	StrHR	OvIHR	OvITR
	SeqTR	StrHW	SeqHW	RndHW
	SeqTW	RndTR	-	-
	-	RndTW	-	-
	-	SeqHR	-	-

2.3. Learning model

학습 모델에서는 신경망을 구축하고, 해당 신경망을 통해 입력되는 워크로드들에 대한 분류 성능을 평가한다. 그러나 표 6에 제시된 워크로드들을 이용하여 신경망을 구축하기엔 CCF가 불균형하므로 분류 모델 및 분석 모델에서 정의된 분배표를 이용해 합성 데이터를 새로이 구성하였다. 합성 데이터는 신경망이 총 24개의 T-value를 균등하

게 학습할 수 있도록 하나의 경우마다 42,000개의 데이터를 가지도록 만들어졌으며, 총 1,008,000개의 합성 데이터로 구성된다. 이들은 각 기초 관리정책에 대한 비율, 그리고 각 정책별 CCF 비율을 반영해 무작위로 생성된다. 이렇게 생성된 합성 데이터는 CCF 및 기초 관리정책에 따른 균형성을 보장하며 신경망을 구축하기에도 충분한 양으로 구성된다. 결과적으로 구축된 신경망은 합성 데이터 기준, 75%의 학습 데이터와 25%의 검증 데이터로 나뉘어 학습되었다.

본 논문에서 제안하는 모델을 통해 표 6과 같은 기존의 워크로드들에 대한 예측 정확성을 판단하기 위하여, 신경망에 모든 워크로드를 검증용 데이터로 사용하였다. 표 9는 표 6의 모든 워크로드를 앞서 구축된 신경망에 입력시켰을 때의 예측 정확도를 나타낸다. 표 9에서 보이는 것과 같이 신경망을 학습시키기 위해 사용된 워크로드의 유형 (Support)이 불균형하므로 일반적인 정확도를 나타내는 Accuracy보다 가중 평균치인 Weighted Average가 더욱 정확한 예측 결과를 나타낸다. 또한 인공지능 모델의 정밀도 및 재현율에 대한 조화 평균을 의미하는 F1-Score를 도출해 워크로드에 대한 편향성을 한 번 더 검증하였다. 결과적으로, 본 논문에서 제안하는 FRM 모델은 가중 평균치(Weighted average)에 대한 F1-Score가 88.97%에 달함을 알 수 있다.

V. Conclusions

본 논문에서는 낸드 플래시 메모리를 효율적으로 활용하기 위한 기초 관리정책 제안 모델인 FRM을 소개하였고, 이에 대한 적용 사례로 FRM을 이용해 워크로드별로 적합한 기초 관리 정책 결정 과정 및 모델을 제시하였다. 해당 모델은 낸드 플래시 메모리의 성능을 개선하기 위해 낸드

Table 9. The classification report of neural network for experimental workloads

T-value	Rank of policy				F1-score	Support
	1	2	3	4		
1	LRU	CLOCK	LFU	FCFS	0.6667	1
3	LRU	FCFS	CLOCK	LFU	0	1
7	CLOCK	LRU	FCFS	LFU	1	1
8	CLOCK	LRU	LFU	FCFS	0.8024	6
10	CLOCK	LFU	LRU	FCFS	1	2
12	FCFS	LRU	CLOCK	LFU	1	1
14	FCFS	CLOCK	LRU	LFU	0.9583	11
16	FCFS	LFU	LRU	CLOCK	1	1
21	LFU	CLOCK	LRU	FCFS	0.8571	4
22	LFU	FCFS	CLOCK	LRU	1	4
Accuracy					0.9285	32
Weighted Average					0.8897	32

플래시 메모리의 특징과 워크로드의 특성을 연계해 분류 및 분석하며, 이를 바탕으로 인공지능 신경망을 구축해 새로운 워크로드가 입력되더라도 적합한 기초 관리정책을 판단할 수 있도록 하였다. 실험 결과 제안된 모델은 입력된 워크로드들을 대상으로 할 때, 정확도와 가중 평균치를 기준으로 각각 92.85%와 88.97%의 기초 관리 정책 예측 정확도를 보여주었다. 향후 연구에서는 구축된 신경망을 기존의 분류 기법들로 평가하고 모델의 분류 정확도를 추가로 확인해 볼 것이다. 또한 Ablation 기법을 도입하여 각 인자의 상관관계를 더 심도 있게 판단해 신경망의 분류 성능을 증가시킬 방법을 모색한다.

## REFERENCES

- [1] R. Caceres, F. Douglass, Kai Li and B. Marsh, "Operating system implications of solid-state mobile computers", Proceedings of IEEE 4th Workshop on Workstation Operating Systems. WWOS-III, pp. 21-27, October 1993. DOI:10.1109/wwos.1993.348178
- [2] Lawton, George, "Improved flash memory grows in popularity", Computer, Vol. 39, No. 1, pp. 16-18, 2006. DOI:10.1109/mc.2006.22
- [3] Kim, Han-joon, and Sang-goo Lee, "A new flash memory management for flash storage system", Proceedings Twenty-Third Annual International Computer Software and Applications Conference (Cat. No. 99CB37032), pp. 248-289, 1999. DOI:10.1109/compasac.1999.812717
- [4] Dong, Xiangyu and Xu, Cong and Xie, Yuan and Jouppi, Norman P, "Nvsim: A circuit-level performance, energy, and area model for emerging nonvolatile memory", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 31, No. 7, pp. 994-1007, July 2012. DOI:10.1109/tcad.2012.2185930
- [5] Saxena, Mohit and Swift, Michael M, "FlashVM: Virtual Memory Management on Flash.", USENIX Annual Technical Conference, June 2010.
- [6] Xie, Yuan, "Emerging Memory Technologies: Design, Architecture, and Applications", pp. 15-50, 2013. DOI:10.1007/978-1-4419-9551-3
- [7] Paulson, Hitha and Ramachandran, Dr Rajesh, "Page Replacement Algorithms-Challenges and Trends", International Journal of Computer & Mathematical Sciences IJCMS, Vol. 6, No. 9, pp.112-116, September 2017.
- [8] Yuan, Youwei and Zhang, Jintao and Han, Guangjie and Jia, Gangyong and Yan, Lamei and Li, Wanqing, "DPW-LRU: An efficient buffer management policy based on dynamic page weight for flash memory in cyber-physical systems", IEEE Access, Vol. 7, pp. 58810-58821, April 2019. DOI:10.1109/access.2019.2914231
- [9] Li, Cong, "CLOCK-pro+ improving CLOCK-pro cache replacement with utility-driven adaptation", Proceedings of the 12th ACM International Conference on Systems and Storage, pp. 1-7, May 2019. DOI:10.1145/3319647.3325838
- [10] Das, Sayak and Das, Nirvik Ranjan and Basu, Soumik Kr and Mondal, Hrididhi and Bose, Avijit, "Page Replacement Technique on the Basis of Frequency of Occurrence of Pages", Proceedings of International Conference on Frontiers in Computing and Systems, pp. 823-831, 2021. [https://doi.org/10.1007/978-981-15-7834-2\\_77](https://doi.org/10.1007/978-981-15-7834-2_77)
- [11] Geetha, Krishnan and Gounden, N Ammasai, "Dynamic semantic lfu policy with victim tracer (dslv): a customizing technique for client cache", Arabian Journal for Science and Engineering, Vol. 42, No. 2, pp. 725-737, 2017. DOI:10.1007/s13369-016-2287-z
- [12] Tarihi, Mojtaba and Asadi, Hossein and Haghdoost, Alireza and Arjomand, Mohammad and Sarbazi-Azad, Hamid, "A hybrid non-volatile cache design for solid-state drives using comprehensive I/O characterization", IEEE Transactions on Computers, Vol. 65, No. 6, pp. 1678-1691, 2015. <https://doi.org/10.1109/tc.2015.2455978>
- [13] Salkhordeh, Reza and Ebrahimi, Shahriar and Asadi, Hossein, "ReCA: An efficient reconfigurable cache architecture for storage systems with online workload characterization", IEEE Transactions on Parallel and Distributed Systems, Vol. 29, No. 7, pp. 1605-1620, 2018. DOI:10.1109/tpds.2018.2796100
- [14] Choi, Jun Hyeong, Kyung Min Kim, and Jong Wook Kwak. "TA-CLOCK: Tendency-aware page replacement policy for hybrid main memory in high-performance embedded systems." Electronics, Vol. 10, No. 9, pp. 1111, 2021. DOI:10.3390/electronics10091111
- [15] OLTP trace from UMass trace repository, <http://traces.cs.umass.edu/index.php/Storage/Stroage>
- [16] Block traces from SNIA, [http://iota.snia.org/historical\\_section?tracetype\\_id=3](http://iota.snia.org/historical_section?tracetype_id=3)

## Authors



Won Ho Lee received a B.S and a MS degree in Computer Engineering from Yeungnam University, Gyeongsan, Korea, in 2014 and 2021, respectively. From 2021, he worked as a CTO in the NFINITY7 Inc.

His current research interests include classification model based machine learning, low-power mobile embedded systems and non-volatile memory.



Jun-Hyeong Choi received a B.S degree in Department of Computer Engineering from Yeungnam University College, Daegu, Korea in 2013, and the M.S. and Ph.D. degree in computer engineering from Yeungnam

University, Gyeongsan, Korea, in 2016 and 2021, respectively. From 2021, he worked as a Senior Engineer in the research development division, at Gyeongbuk Industry Institute of IT Convergence Industry Technology(GITC). His current research interests low-power mobile embedded systems, cloud systems and non-volatile memory.



Jong Wook Kwak received a B.S. degree in Computer Engineering from Kyungpook National University, Daegu, Korea in 1998, a M.S. degree in Computer Engineering from Seoul National University, Seoul, Korea in

2001, and a Ph.D. degree in Electrical Engineering and Computer Science from Seoul National University, Seoul, Korea in 2006. From 2006 to 2007, he worked as a Senior Engineer in the SoC R&D Center, at Samsung Electronics Co., Ltd. During 2011~2012, he was a Guest Researcher at the Research Institute of Advanced Computer Technology, Seoul National University. During 2012~2013, he was a Visiting Scholar at the Georgia Institute of Technology, Atlanta, GA, USA. As a Head Director, he led DREAM Software Human Resource Training Center from 2014~2015. During 2018~2019, he was a Visiting Scholar at Arizona State University, Tempe, AZ, USA. He is currently a professor in the Department of Computer Engineering, Yeungnam University. His research interests include advanced processor architecture, low-power mobile embedded systems, and high performance parallel and distributed computing.