iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: VII Month of publication: July 2026

DOI: https://doi.org/10.22214/ijraset.2026.84252

Dynamic Health Scoring and Predictive Wear Management for NAND Flash Memory Lifetime Enhancement

Anjali Pandey[1], Ritesh Sadiwala[2]
[1]*Research Scholar, Department of Electronics and Communication Engineering, RKDF University, Gandhinagar, Bhopal*
[2]*Professor, Department of Electronics and Communication Engineering, RKDF University, Gandhinagar, Bhopal*

Abstract: NAND flash-based Solid-State Drives (SSDs) are widely used in embedded systems, enterprise storage, consumer electronics, and cloud platforms because of their high speed, low power consumption, shock resistance, and compact form factor. However, NAND flash memory suffers from limited program/erase (P/E) endurance, increasing bit-error rates, latency degradation, and thermal sensitivity as blocks age. Traditional wear-management techniques, such as classical wear levelling, error-correction codes, and reactive data migration, typically address only one aspect of degradation and often rely mainly on P/E cycle count. This paper proposes a Dynamic Smart Score System (SSS) for NAND flash memory health assessment and predictive wear management. The proposed method combines four block-level degradation indicators - P/E cycle count, cumulative ECC-corrected bit errors, access latency, and block temperature - into a normalized composite health score. A workload-aware weighting mechanism dynamically adjusts the importance of each indicator under balanced, write-heavy, read-heavy, and thermal-stress workloads. The score is then used for predictive write-block selection and threshold-based proactive data migration.

The proposed framework is evaluated through a reproducible Python-based NAND flash simulator implemented in Google Colab. Experiments compare the Smart Score System against an unmanaged baseline and classical wear-levelling across controlled phased workloads, multiple random seeds, fixed versus dynamic weighting configurations, migration threshold settings, scalability sweeps, and non-uniform initial wear conditions. Results show that, compared with the unmanaged baseline, the Smart Score System extends time-to-first-failure by more than 50 times, improves wear uniformity by 95.3%, reduces cumulative ECC errors by 15.9%, and preserves all 64 blocks through the 80,000-operation simulation. Compared with classical wear-levelling, it achieves equivalent lifespan performance while providing a richer multi-metric and workload-aware health representation. Statistical testing across ten random seeds confirms that the improvements over unmanaged operation are highly significant. The results demonstrate that dynamic multi-metric health scoring can provide an interpretable, lightweight, and effective foundation for predictive NAND flash wear management.

Keywords: NAND flash memory; Solid-State Drive; SSD lifetime; wear levelling; health scoring; predictive migration; ECC errors; workload-aware weighting; data reliability; flash endurance.

I. INTRODUCTION

Solid-State Drives (SSDs) based on NAND flash memory have become a dominant storage technology in modern computing systems. They are used in consumer devices, cloud data centers, embedded systems, industrial platforms, automotive systems, and high-reliability environments because they offer high read/write performance, low energy consumption, compact physical design, and resistance to mechanical shock. Unlike magnetic hard disks, SSDs have no moving parts and therefore provide faster access latency and improved durability under vibration-prone conditions.

Despite these advantages, NAND flash memory has a fundamental limitation: each block can tolerate only a finite number of program/erase cycles. As a block is repeatedly programmed and erased, physical degradation occurs in the floating-gate or charge-trap structure. This degradation increases bit-error rates, shifts threshold voltages, raises access latency, and eventually causes block failure. The problem has become more serious as NAND flash technology has evolved from Single-Level Cell (SLC) to Multi-Level Cell (MLC), Triple-Level Cell (TLC), and Quad-Level Cell (QLC), because higher-density cells generally provide lower endurance margins.

Traditional SSD controllers use wear-management mechanisms such as wear levelling, garbage collection, bad-block management, and error-correction codes. Classical wear levelling distributes erase cycles across blocks to prevent a small subset of blocks from wearing out too early. ECC mechanisms detect and correct bit errors but usually operate after errors appear. Reactive data migration moves data after blocks become heavily worn or problematic. Although these techniques are effective, they commonly suffer from three limitations. First, many schemes rely primarily on P/E cycle count and therefore ignore other important signs of NAND degradation. Second, several methods are reactive rather than predictive, meaning they respond after degradation has already become severe. Third, static thresholds and fixed policies may not adapt well to different workloads, such as write-heavy, read-heavy, or thermal-stress conditions.

To address these limitations, this paper proposes a Smart Score System (SSS) for NAND flash memory health assessment and predictive wear management. The Smart Score System calculates a composite block-health score using four normalized indicators: P/E cycle count, ECC-corrected bit errors, access latency, and block temperature. These indicators represent complementary dimensions of flash degradation. The proposed score is workload-aware: its weights change dynamically according to the current workload phase, allowing the system to emphasize P/E cycles during write-heavy workloads, latency during read-heavy workloads, and temperature during thermal-stress workloads.

The score is used in two ways. First, it guides predictive write-block selection by directing writes toward healthier blocks. Second, it supports proactive data migration by identifying blocks whose health score exceeds a threshold and relocating live data before failure occurs. The method is evaluated using a Python-based discrete-event NAND flash simulator. The simulator includes physically motivated wear modelling, stochastic ECC error generation, latency drift, temperature effects, manufacturing variability, and reproducible workload generation.

The main contributions of this paper are as follows:

- A multi-metric composite health score is proposed for NAND flash blocks, combining P/E count, ECC errors, latency, and temperature into a single interpretable value.
- A workload-aware dynamic weighting strategy is introduced to adapt the score under balanced, write-heavy, read-heavy, and thermal-stress operating conditions.
- A predictive data migration protocol is designed to convert health scores into proactive relocation decisions before block failure.
- A simulation-based validation framework is developed to compare the proposed method against unmanaged operation and classical wear-levelling.
- Experimental results show that the proposed Smart Score System significantly improves lifespan, wear uniformity, and data integrity compared with unmanaged operation while matching the lifespan performance of classical wear-levelling.

The remainder of this paper is organized as follows. Section 2 presents background and motivation. describes the proposed Smart Score System and predictive migration design. Section 4 presents the simulation methodology and algorithmic implementation. reports experimental results and analysis.

II. BACKGROUND AND MOTIVATION

A. NAND Flash Wear Characteristics

NAND flash memory stores information by controlling the charge state of memory cells. During each program/erase cycle, electrons are injected into or removed from the floating-gate or charge-trap structure. Repeated cycling causes physical damage to the oxide layer, including charge trapping, threshold-voltage drift, and reduced cell margin. These effects increase the probability of bit errors and eventually cause hard block failure.

The degradation of NAND flash is not perfectly uniform across blocks. Manufacturing variability causes different blocks to have slightly different endurance limits. Workload locality can also make some blocks receive more writes than others. Thermal conditions further accelerate degradation, because higher temperature increases leakage, retention loss, and error generation. As a result, a reliable wear-management system should consider more than erase count alone.

B. Conventional Wear-Management Techniques

Wear levelling is one of the most widely used mechanisms for extending SSD lifetime. Dynamic wear levelling selects less-worn free blocks for new writes, while static wear levelling migrates cold data from lightly worn blocks to more worn blocks so that young blocks can be reused. These techniques help distribute erase cycles more evenly across blocks.

However, classical wear levelling usually focuses on the P/E count as the main wear indicator. This provides a simple and effective measure of cumulative wear, but it does not capture all degradation signals. Two blocks with similar P/E counts may have different ECC error histories, latency behavior, or temperature exposure. Similarly, ECC mechanisms improve data integrity but often act after errors occur. Data migration can reduce failure risk, but reactive migration may happen too late or may introduce unnecessary write amplification if triggered too aggressively.

C. Need for Dynamic Health Scoring

A more effective NAND flash health-management framework should satisfy four requirements:

Requirement	Explanation
Multi-metric assessment	The framework should combine P/E count, ECC errors, latency, and temperature rather than relying on a single indicator.
Predictive action	It should identify risky blocks before failure occurs and support proactive migration.
Workload awareness	It should adapt to different workload conditions because the dominant degradation signal changes across workloads.
Low overhead	It should remain simple enough for controller-level implementation.

The Smart Score System proposed in this paper is designed around these requirements. It uses a lightweight linear score for interpretability and computational efficiency while still capturing multiple degradation dimensions.

III. PROPOSED SMART SCORE SYSTEM

A. Overview

The proposed Smart Score System evaluates the health of each NAND block during simulation. Each block receives a score in the unit interval, where a lower score indicates a healthier block and a score closer to one indicates a block approaching failure risk. The system uses the score for two controller decisions:

1) Predictive write-block selection: New writes are directed toward the healthiest available block.
2) Predictive data migration: If a live-data block exceeds a health threshold and a healthier destination exists, the data is migrated before failure.

The complete framework consists of four main components:

Component	Role
NAND block model	Represents block state, endurance limit, P/E count, ECC errors, latency, temperature, and validity.
Wear function	Updates block state after every program/erase cycle.
Health scorer	Computes the normalized multi-metric Smart Score.
Migration policy	Uses score thresholds and destination selection rules to relocate live data.

B. Block Representation

Each NAND block is represented with the following state variables:

Variable	Description
Block ID	Unique physical block identifier.
P/E count	Number of accumulated program/erase cycles.
ECC error count	Cumulative corrected bit errors.
Access latency	Current read latency in microseconds.
Temperature	Current block temperature in degrees Celsius.
Alive flag	Indicates whether the block is still usable.
Data tag	Logical data identifier stored in the block.
Endurance limit	Intrinsic P/E limit sampled per block.

The endurance limit is sampled from a normal distribution around a nominal value to model manufacturing variability.

C. Physics-Based Wear Model

Each write operation applies one P/E cycle to the selected block. The simulator updates the block state according to a physically motivated wear model. The wear ratio of a block is defined as:

$$r_i = \frac{PE_i}{L_i}$$

where PE_i is the accumulated P/E count of block i and L_i is its intrinsic endurance limit.

ECC error generation is modelled as a stochastic Poisson process with a mean that increases super-linearly with wear:

$$\lambda_i = 5r_i^2 A_T$$

where A_T is the thermal acceleration factor. The squared wear-ratio term reflects the rapid increase in error probability near the end of block life.

Latency is modelled as a convex function of wear:

$$Lat_i = 50 + 150r_i^{1.5}A_T + \epsilon$$

where 50 microseconds is the fresh-block baseline latency and ϵ is a small Gaussian noise term.

The thermal acceleration factor is calculated as:

$$A_T = 1 + \max\left(0, \frac{T_i - 40}{40}\right)$$

where T_i is block temperature and 40°C is the nominal operating point. When the P/E count exceeds the endurance limit, the block is marked as failed.

D. Composite Health Score

For each live block, the Smart Score is computed as a weighted linear combination of four normalized indicators:

$$S_i = w_{PE}M_{PE,i} + w_{ECC}M_{ECC,i} + w_{Lat}M_{Lat,i} + w_{Temp}M_{Temp,i}$$

where:

Symbol	Meaning
$M_{PE,i}$	Normalized P/E cycle count.
$M_{ECC,i}$	Normalized ECC error count.
$M_{Lat,i}$	Normalized access latency.
$M_{Temp,i}$	Normalized block temperature.
$w_{PE}, w_{ECC}, w_{Lat}, w_{Temp}$	Workload-specific weights.

The weights are non-negative and satisfy:

$$w_{PE} + w_{ECC} + w_{Lat} + w_{Temp} = 1$$

A failed block is assigned a score of 1, preventing it from being selected for future writes.

E. Metric Normalization

Each raw metric is normalized using min-max saturation:’

$$M_x = \min\left(\frac{x}{X_{ref}}, 1\right)$$

The reference maxima used in the simulator are shown below.

Indicator	Reference Maximum
P/E cycle count	3,000
ECC error count	500
Access latency	250 us
Temperature	85°C

Min-max normalization was selected because it produces bounded, interpretable values suitable for threshold-based migration.

F. Workload-Aware Dynamic Weighting

The Smart Score System uses four workload profiles. Each profile changes the contribution of the four indicators according to workload conditions.

Table 1. Workload-Aware Weight Profiles

Workload Profile	w_{PE}	w_{ECC}	w_{Lat}	w_{Temp}
Balanced	0.35	0.25	0.25	0.15
Read-Heavy	0.2	0.3	0.4	0.1
Write-Heavy	0.5	0.25	0.15	0.1
Thermal-Stress	0.3	0.2	0.15	0.35

The balanced profile distributes weight across all indicators. The write-heavy profile emphasizes P/E count. The read-heavy profile emphasizes latency and ECC errors. The thermal-stress profile emphasizes temperature.

G. Predictive Data Migration

The migration protocol is invoked periodically after a configurable number of operations. In the main experiments, the check interval is 200 operations. A live-data block becomes a migration candidate when its Smart Score exceeds the migration threshold:

$$S_i > \theta$$

The default threshold is:

$$\theta = 0.65$$

The protocol then selects a healthier destination using two tiers.
Tier 1: Empty destination selection
The system first searches for an empty block with the lowest health score. Migration occurs only if:

$$S_{source} - S_{dest} \geq 0.02$$

Tier 2: Swap destination selection
If no suitable empty block exists, the system considers occupied blocks. A swap is performed only if:

$$S_{source} - S_{dest} \geq 0.05$$

Every migration is charged as a write operation, meaning the destination block receives one additional P/E cycle. This prevents the simulation from overestimating migration benefits.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

IV. SIMULATION METHODOLOGY AND ALGORITHM DESIGN

A. *Simulation Configuration*

The Smart Score System was implemented as a self-contained Python simulator using NumPy, pandas, and Matplotlib. The simulator was executed in Google Colab to support reproducibility.

Table 2. Simulation Configuration Parameters

Parameter	Value	Purpose
Number of NAND blocks	64	Device size in main experiments
Nominal endurance per block	1,500 P/E cycles	Mean intrinsic endurance
Endurance variance	5%	Manufacturing variability
Total operations	80,000	Simulation budget
Workload phases	4 x 20,000 operations	Balanced, write-heavy, read-heavy, thermal-stress
Migration check interval	200 operations	Frequency of migration evaluation
Migration threshold	0.65	Score threshold for migration candidate
Empty-destination margin	0.02	Minimum score benefit for empty-block migration
Swap-destination margin	0.05	Minimum score benefit for swap migration
Fresh-block latency	50 us	Baseline latency
Reference max P/E	3,000	Normalization value
Reference max ECC	500	Normalization value
Reference max latency	250 us	Normalization value
Reference max temperature	85°C	Normalization value
Random seed	42	Paired comparison reproducibility

B. *Workload Phase Structure*

The workload generator divides the 80,000-operation run into four equal phases.

Table 3. Workload Phase Specification

Phase	Profile	Operation Range	Ambient Temperature	Weights	Purpose
1	Balanced	0-20,000	40°C	0.35 / 0.25 / 0.25 / 0.15	Baseline operation
2	Write-Heavy	20,001-40,000	45°C	0.50 / 0.25 / 0.15 / 0.10	P/E wear stress
3	Read-Heavy	40,001-60,000	42°C	0.20 / 0.30 / 0.40 / 0.10	Latency-sensitive stress
4	Thermal-Stress	60,001-80,000	70°C	0.30 / 0.20 / 0.15 / 0.35	Temperature stress

Each operation is treated as a write operation because writes are the primary cause of NAND wear. This creates a conservative stress scenario.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

C. *Baseline Policies*
The proposed Smart Score System is compared against two baseline policies.

Policy	Description
NoMgmt	Writes are directed to the next non-failed block by identifier. No wear management is performed.
WearLvl	Classical wear-levelling policy that writes to the block with the minimum P/E count.
SSS	Proposed Smart Score System using multi-metric scoring and predictive migration.

D. *Algorithm 1: Block Wear Function*
Input: NAND block, ambient temperature
Output: Updated block state
- If the block has failed, return without modification.
- Increment the block P/E count.
- Compute the wear ratio.
- Update block temperature using ambient temperature and Gaussian noise.
- Compute the thermal acceleration factor.
- Generate new ECC errors from a Poisson distribution.
- Update cumulative ECC error count.
- Update access latency using the wear-latency model.
- If P/E count exceeds the endurance limit, mark block as failed.

E. *Algorithm 2: Composite Smart Score*
Input: NAND block, workload profile
Output: Health score
- If the block has failed, return score 1.
- Normalize P/E count, ECC errors, latency, and temperature.
- Select the weight profile according to workload phase.
- Compute the weighted linear score.
- Cap the score at 1.
- Return the score.

F. *Algorithm 3: Predictive Migration Protocol*
Input: Device blocks, current workload profile
Output: Updated data placement and migration count
- Compute scores for all alive blocks.
- For each live-data block:
- If score is below migration threshold, skip.
- Search for the lowest-score empty destination block.
- If the score gap is at least 0.02, migrate data and charge one P/E cycle.
- If no empty destination qualifies, search occupied blocks.
- If the score gap is at least 0.05, swap data tags and charge one P/E cycle.
- Update migration count and block states.

G. *Computational Complexity*
The wear function is constant time per operation. The scoring function is also constant time per block. For Noperations, Bblocks, and migration interval I, the number of scoring calls is approximately:

$$NB + \frac{NB}{I}$$

For the default configuration:

$$N = 80{,}000, B = 64, I = 200$$

This gives approximately 5.1 million scoring invocations per run. The unoptimized migration protocol has worst-case complexity $O(B^2)$per invocation, but this is acceptable for the 64-block simulation. For larger devices, sorted score lists or priority queues can reduce migration overhead.

V. EXPERIMENTAL RESULTS

A. Headline Comparative Performance

Table 4. Comparative Final Performance Across Policies

Policy	Ops Completed	Time-to-1st-Failure	Time-to-50%-Failure	Alive Blocks	Max P/E	Avg P/E	P/E Std Dev	Total ECC Errors	Avg Latency (us)	Migrations
NoMgmt	80,000	1,536	47,655	Nov-64	1,638	1250	541.48	1,64,873	71.2	0
WearLvl	80,000	80,000	80,000	64/64	1,250	1250	0	1,38,799	253.7	0
SSS	80,000	80,000	80,000	64/64	1,310	1250	25.53	1,38,614	256.7	0

The unmanaged baseline fails very early because writes concentrate on a small subset of blocks. Its first block fails at operation 1,536, and only 11 of 64 blocks remain alive at the end of the run. In contrast, both WearLvl and SSS preserve all 64 blocks through the full 80,000 operations.

The Smart Score System slightly reduces ECC errors compared with classical wear-levelling, showing that multi-metric scoring can provide a small data-integrity benefit. Its P/E standard deviation is higher than WearLvl because SSS sometimes chooses a block with slightly higher P/E count if it has better ECC, latency, or temperature characteristics. This reflects a deliberate trade-off between strict P/E uniformity and broader health uniformity.

B. Improvement Relative to Baselines

Table 5. Improvement of SSS Relative to Baselines

Comparison	TTFF Improvement	Time-to-50%-Failure Improvement	Wear Uniformity Improvement	ECC Error Reduction	Latency Improvement	Surviving Blocks Gain
SSS vs NoMgmt	5108.30%	67.90%	95.30%	15.90%	-260.60%	53
SSS vs WearLvl	0.00%	0.00%	0.00%	0.10%	-1.20%	0

The Smart Score System provides a very large improvement over unmanaged operation. The time-to-first-failure increases by more than 50 times. The system also saves 53 additional blocks and reduces cumulative ECC errors by 15.9%.

The apparent latency regression relative to NoMgmt requires careful interpretation. In the unmanaged case, the surviving blocks are mostly lightly used blocks that escaped wear, so their latency remains low. In SSS, all blocks remain alive but have been aged uniformly. Therefore, the latency comparison reflects survivor bias in the unmanaged baseline, not a true performance advantage.

C. *Per-Phase Block Evolution*

Table 6. Per-Phase Health-Score Evolution for Representative Blocks

Block ID	Balanced P/E	Balanced Score	Write-Heavy P/E	Write-Heavy Score	Read-Heavy P/E	Read-Heavy Score	Thermal P/E	Thermal Score
0	297	0.186	635	0.32	964	0.625	1256	0.767
12	313	0.189	637	0.32	949	0.624	1275	0.767
25	324	0.187	628	0.321	938	0.625	1238	0.767
38	316	0.187	621	0.32	898	0.627	1245	0.766
51	318	0.187	617	0.321	924	0.625	1243	0.767
63	333	0.187	606	0.32	892	0.627	1205	0.768

The representative blocks show similar P/E counts at each phase boundary, confirming that predictive selection distributes wear effectively. Scores rise from approximately 0.187 in the balanced phase to approximately 0.767 in the thermal-stress phase. Although final scores exceed the migration threshold, migrations do not occur because score differences between blocks remain too small to justify relocation.

D. *Migration Threshold Sensitivity*

Table 7. Sensitivity to Migration Threshold

Threshold	Migrations Triggered	P/E Std Dev	Total ECC Errors	Time-to-1st-Failure	Alive Blocks	Avg Latency (us)
0.45	0	25.53	1,38,614	80,000	64/64	256.7
0.55	0	25.53	1,38,614	80,000	64/64	256.7
0.65	0	25.53	1,38,614	80,000	64/64	256.7
0.75	0	25.53	1,38,614	80,000	64/64	256.7
0.85	0	25.53	1,38,614	80,000	64/64	256.7

Threshold variation has no effect under the controlled phased workload because the predictive write-selection mechanism keeps the score spread small. Migration requires both threshold violation and a meaningful score gap between source and destination. Since blocks remain similarly healthy, no migration is beneficial.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

E. Fixed Versus Dynamic Weighting

Table 8. Fixed and Dynamic Weighting Comparison

Configuration	Migrations	P/E Std Dev	Total ECC Errors	Time-to-1st-Failure	Avg Latency (us)	Alive Blocks
Fixed Balanced	0	21.03	1,38,470	80,000	256.5	64/64
Fixed Write-Heavy	0	13.34	1,39,025	80,000	256.1	64/64
Fixed Read-Heavy	0	29.95	1,38,462	80,000	256.6	64/64
Fixed Thermal	0	30.91	1,38,690	80,000	257.5	64/64
Dynamic Proposed	0	25.53	1,38,614	80,000	256.7	64/64

All weighting strategies preserve all blocks for the full run. Fixed write-heavy weighting produces the lowest P/E standard deviation because it emphasizes P/E count most strongly. Dynamic weighting provides a balanced outcome across phases and supports adaptability to changing workload conditions.

F. Multi-Seed Reproducibility

Table 9. Reproducibility Across Three Random Seeds

Policy	Seeds Tested	Mean TTFF	Std TTFF	Mean ECC Errors	Std ECC Errors	Mean Alive Blocks	Mean P/E Std Dev
NoMgmt	3	1,514	19	1,64,083	789	11.0/64	553.27
WearLvl	3	80,000	0	1,36,486	1,842	64.0/64	0
SSS	3	80,000	0	1,36,309	1,811	64.0/64	29.24

The qualitative ranking is stable across seeds. The unmanaged baseline fails early, while WearLvl and SSS preserve all blocks. This confirms that the main findings are not artifacts of a single random seed.

G. Per-Phase Operational Dynamics

Table 10. Per-Phase Operational Dynamics Under SSS

Phase	Operations	Migrations Fired	Peak Mean Score	Avg Mean Score	ECC Errors Added	P/E Std Dev at End
Balanced	20,000	0	0.191	0.151	1,541	13.54
Write-Heavy	20,000	0	0.322	0.228	11,838	18.49
Read-Heavy	20,000	0	0.634	0.515	29,531	34.52
Thermal-Stress	20,000	0	0.77	0.73	95,704	25.53

ECC errors grow sharply across phases. The thermal-stress phase contributes the majority of errors because blocks are already worn and temperature accelerates degradation.

H. Scalability Across Device Sizes

Table 11. Scalability of SSS Across Device Sizes

Device Size	Operations	NoMgmt Alive	WearLvl Alive	SSS Alive	SSS TTFF Improvement	SSS ECC Reduction	SSS Migrations
16 blocks	20,000	Mar-16	16/16	16/16	1202%	17%	0
32 blocks	40,000	Jun-32	32/32	32/32	2504%	16%	0
64 blocks	80,000	Nov-64	64/64	64/64	5108%	16%	0
128 blocks	1,60,000	21/128	128/128	128/128	10317%	17%	0

The qualitative behavior remains consistent across device sizes. The unmanaged baseline loses most blocks, while both managed policies preserve all blocks. The TTFF improvement grows with device size because more blocks are available for distributing wear.

VI. DISCUSSION

A. Composite Score Decomposition

The Smart Score System's behavior can be better understood by decomposing the total score into its indicator contributions.

Table 12. Composite Score Decomposition by Phase

Workload Phase	P/E Contribution	ECC Contribution	Latency Contribution	Temperature Contribution	Total Score	Dominant Indicator
Balanced	0.036	0.012	0.067	0.071	0.187	Temperature
Write-Heavy	0.104	0.105	0.059	0.053	0.321	ECC
Read-Heavy	0.062	0.3	0.213	0.05	0.626	ECC
Thermal-Stress	0.125	0.2	0.149	0.293	0.767	Temperature

The dominant indicator changes across phases. Temperature is dominant early because other indicators have not yet accumulated. ECC dominates in write-heavy and read-heavy phases due to super-linear error growth. Temperature again dominates in the thermal-stress phase due to elevated ambient temperature and higher temperature weight.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

B. ECC Error Burden

Table 13. Per-Phase ECC Error Burden

Phase	Operations	P/E Cycles Added	ECC Errors Added	Percent of Total ECC	Errors per 1000 P/E	Mean Block Temp
Balanced	20,000	20,000	1,541	1.10%	77	40.6°C
Write-Heavy	20,000	20,000	11,838	8.50%	591.9	45.2°C
Read-Heavy	20,000	20,000	29,531	21.30%	1476.5	42.6°C
Thermal-Stress	20,000	20,000	95,704	69.00%	4785.2	70.3°C

Although each phase contributes the same number of operations, the thermal-stress phase generates 69% of total ECC errors. This confirms that temperature is a major driver of NAND flash degradation and should be included in modern wear-management policies.

C. Statistical Significance

Table 14. Paired t-Test Results Across Ten Random Seeds

Comparison	Metric	Mean SSS	Mean Baseline	Mean Difference	t-statistic	p-value	Significant
SSS vs NoMgmt	Time-to-First-Failure	80,000.00	1,533.80	78,466.20	5010.57	0	Yes
SSS vs NoMgmt	Cumulative ECC Errors	1,36,311.40	1,63,861.70	-27,550.30	-74.58	0	Yes
SSS vs NoMgmt	Alive Blocks	64	11	53	n/a	n/a	Identical
SSS vs NoMgmt	P/E Std Dev	29.1	554.4	-525.3	-205.65	0	Yes
SSS vs WearLvl	Time-to-First-Failure	80,000.00	80,000.00	0	n/a	n/a	Identical
SSS vs WearLvl	Cumulative ECC Errors	1,36,311.40	1,36,477.70	-166.3	-2.29	0.0477	Yes
SSS vs WearLvl	Alive Blocks	64	64	0	n/a	n/a	Identical
SSS vs WearLvl	P/E Std Dev	29.1	0	29.1	29.68	0	Yes

The differences between SSS and unmanaged operation are statistically significant across all major metrics. Compared with WearLvl, SSS provides a small but statistically significant ECC reduction while allowing slightly higher P/E variation.

D. *Cost-Benefit Analysis*

Table 15. Cost-Benefit Analysis

Policy	Score-Evaluation Overhead	Migration Cost	Blocks Saved	ECC Errors Avoided	Lifespan Gain	Benefit/Cost Ratio
SSS vs NoMgmt	~5.1M scoring calls	0 P/E cycles	53	26,259	78,464 ops	Effectively infinite
WearLvl vs NoMgmt	~5.1M P/E lookups	0 P/E cycles	53	26,074	78,464 ops	Effectively infinite

The Smart Score System achieves its main benefits with no migration-induced write overhead under the tested workloads. Its computational overhead is also low because scoring uses only basic arithmetic operations.

E. *Non-Uniform Initial Wear Stress Test*

Table 16. Performance Under Non-Uniform Initial Wear

Policy	Ops Completed	Time-to-1st-Failure	Time-to-50%-Failure	Alive Blocks	P/E Std Dev	Total ECC Errors	Migrations Fired
NoMgmt	80,000	1,536	44,137	Jun-64	410.15	1,75,945	0
WearLvl	80,000	79,605	80,000	62/64	0.87	1,73,688	0
SSS	80,000	78,481	80,000	61/64	21.51	1,73,633	0

Under non-uniform initial wear, the unmanaged baseline performs poorly, retaining only 6 of 64 blocks. WearLvl and SSS remain robust, preserving most blocks through the full operation budget. SSS shows slightly lower ECC errors but slightly earlier first failure than WearLvl, reflecting its multi-metric trade-off.

VII. CONCLUSION

This paper proposed a Dynamic Smart Score System for NAND flash memory health assessment and predictive wear management. The system combines four key degradation indicators - P/E cycle count, ECC error count, access latency, and temperature - into a normalized composite score. A workload-aware weighting mechanism adapts the score across balanced, write-heavy, read-heavy, and thermal-stress workloads. The score is then used for predictive write-block selection and proactive migration.

Simulation results show that the Smart Score System significantly improves device reliability compared with unmanaged operation. It extends time-to-first-failure by more than 50 times, improves wear uniformity by 95.3%, reduces ECC errors by 15.9%, and preserves all blocks through the main 80,000-operation run. Compared with classical wear-levelling, it achieves equivalent lifespan performance while providing additional multi-metric health awareness and a small ECC-error reduction.

The results also show that predictive write-block selection is the main contributor to lifespan improvement. Under controlled phased workloads, the migration protocol rarely fires because block scores remain uniformly distributed. This suggests that migration should be viewed as a fallback mechanism for non-uniform initial wear, hot-spot workloads, locality constraints, or externally induced imbalance.

Overall, the Smart Score System provides a lightweight, interpretable, and effective framework for NAND flash wear management. It extends classical wear-levelling by integrating multiple degradation indicators and adapting to workload conditions without introducing significant operational overhead.

A. *Declaration of Interests*

The author declares no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

REFERENCES

[1] L.-P. Chang, "On efficient wear leveling for large-scale flash-memory storage systems," Proceedings of the ACM Symposium on Applied Computing, 2007.
[2] Y.-H. Chang, J.-W. Hsieh, and T.-W. Kuo, "Improving flash wear-leveling by proactively moving static data," IEEE Transactions on Computers, 2010.
[3] M. Murugan and D. H. C. Du, "Rejuvenator: A static wear leveling algorithm for NAND flash memory with minimized overhead," IEEE Symposium on Mass Storage Systems and Technologies, 2011.
[4] J. Liao, F. Zhang, L. Li, and G. Xiao, "Adaptive wear-leveling in flash-based memory," IEEE Computer Architecture Letters, 2015.
[5] M.-C. Yang, Y.-H. Chang, T.-W. Kuo, and F.-H. Chen, "Reducing data migration overheads of flash wear leveling in a progressive way," IEEE Transactions on Very Large Scale Integration Systems, 2016.
[6] B. Van Houdt, "On the cost of near-perfect wear leveling in flash-based SSDs," ACM Transactions on Modeling and Performance Evaluation of Computing Systems, 2023.
[7] J. Wu, J. Li, Z. Sha, Z. Cai, and J. Liao, "Adaptive switch on wear leveling for enhancing I/O latency and lifetime of high-density SSDs," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022.
[8] S. Maneas, K. Mahdaviani, T. Emami, and B. Schroeder, "Operational characteristics of SSDs in enterprise storage systems: A large-scale field study," USENIX FAST, 2022.
[9] Z. Jiao and B. S. Kim, "Wear leveling in SSDs considered harmful," ACM Workshop on Hot Topics in Storage and File Systems, 2022.
[10] T. Wang, M. Yang, J. Jiang, H. Liu, S. Li, and Z. Ma, "Adaptive multi-phased wear leveling method for SSD lifetime enhancement," Journal of Information and Intelligence, 2025.
[11] Y. Hu, H. Jiang, D. Feng, L. Tian, H. Luo, and C. Ren, "Exploring and exploiting the multilevel parallelism inside SSDs for improved performance and endurance," IEEE Transactions on Computers, 2013.
[12] Q. Luo, X. Fang, Y. Sun, J. Ai, and C. Yang, "Self-learning hot data prediction: Where echo state network meets NAND flash memories," IEEE Transactions on Circuits and Systems I, 2020.
[13] Y. Pan, H. Zhang, R. Yu, Z. Lu, H. Zhang, and Z. Liu, "LightWarner: Predicting failure of 3D NAND flash memory using reinforcement learning," IEEE Transactions on Computers, 2023.