A Test Method for Shortening Test Time of Cloud Computing Platforms

Presented at CCPQT 2023.

This presentation summarizes a novel approach to significantly reduce testing times in cloud environments. The method focuses on optimizing virtual machine (VM) test sequences. By introducing parallel testing and efficient scheduling, it addresses critical issues of cost, scalability, and efficiency.

By :Aarushi Savla - 16010123812
Harshitha Shetty -16010122339
Divya Khanolkar - 16010123818

The Problem: Long Testing Times

Cloud computing platforms suffer from extensive testing times. These are largely due to inefficient VM test sequences. Traditional sequential testing approaches are time-consuming. Long testing times impact resource utilization and operational costs. This reduces overall platform agility.

Inefficient VM Testing

Sequential test execution

Idle time during tests

Impact on Cloud Platforms

Increased operational costs

Reduced platform agility

Motivation: Why Reduce Test Time?

Reducing test time is crucial in cloud environments. Cost savings from optimized resource use is a key benefit. Scalability is improved with faster VM deployment cycles. Overall efficiency gains enhance competitiveness. These improvements allow resources to be quickly reallocated.

Cost Savings

Optimize resource usage

Scalability

Accelerate deployment

Efficiency

Enhance competitiveness

Conventional Testing Approach

Traditional VM testing involves sequential execution. Each VM is tested individually, one after another. This leads to significant idle time. This approach does not consider inter-VM dependencies. It often results in suboptimal resource utilization. Testing does not consider which tests can be run in parallel.

VM1

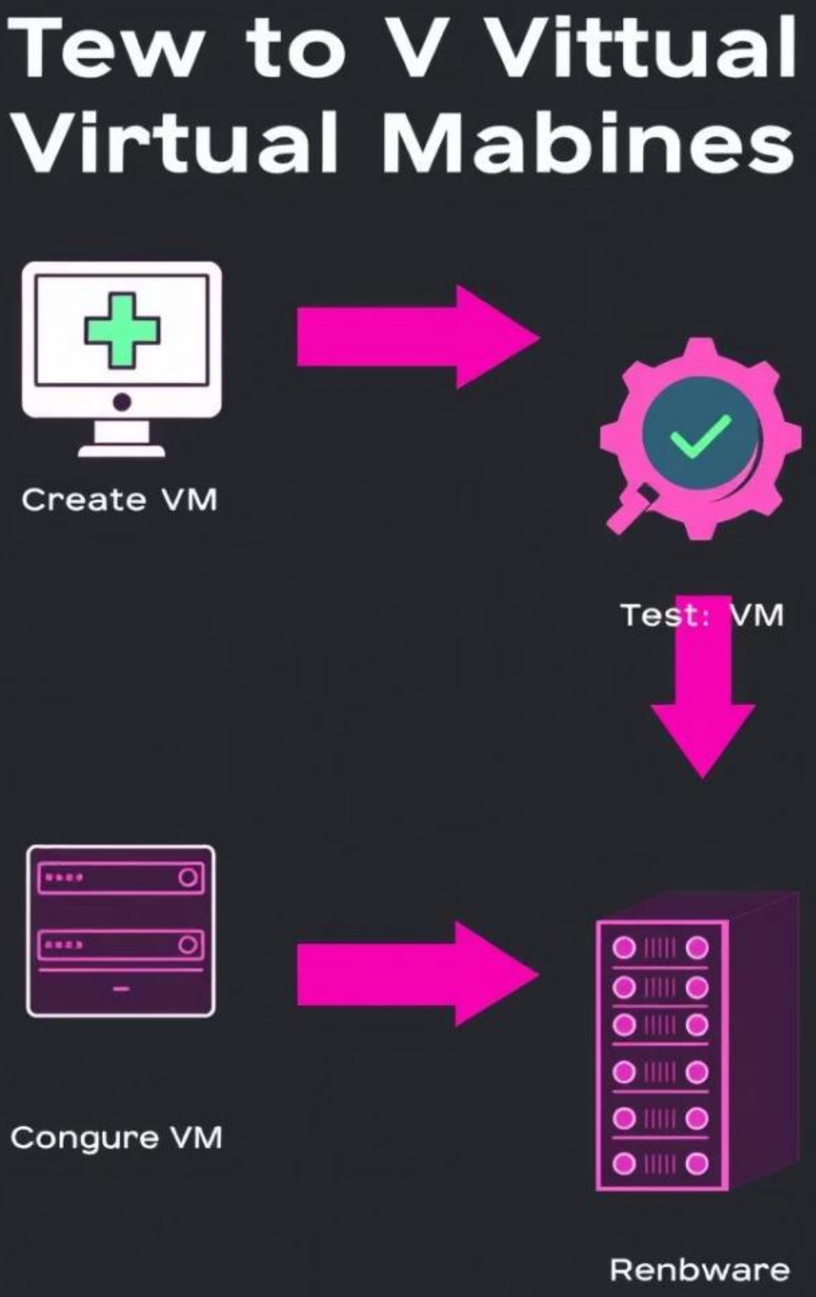
Testing VM1

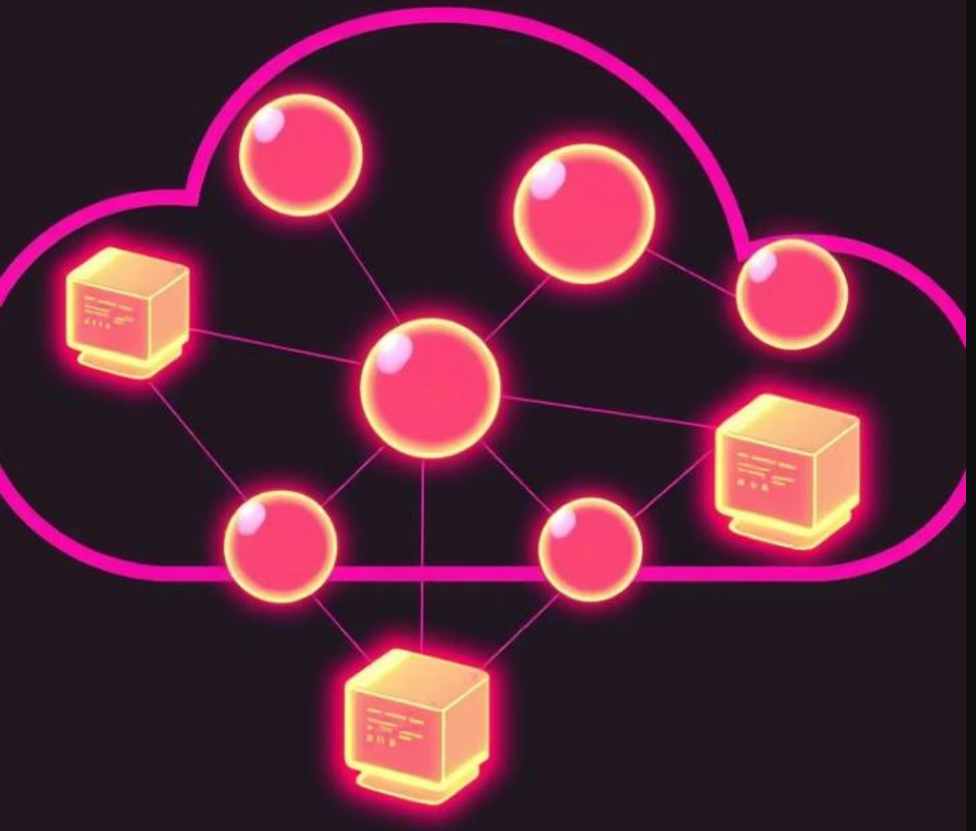
VM2

Testing VM2

VM3

Testing VM3





Proposed Method Overview

The new method tests multiple VMs in parallel. It reduces idle times by optimizing test execution order. The key is a matrix-based scheduling model. The authors create an optimized sequence. The goal is to minimize total test time. It uses dependencies and execution times.

Parallel VM Testing

Multiple VMs simultaneously

Optimized Test Order

Reduced idle times

Matrix Scheduling

Dependency management

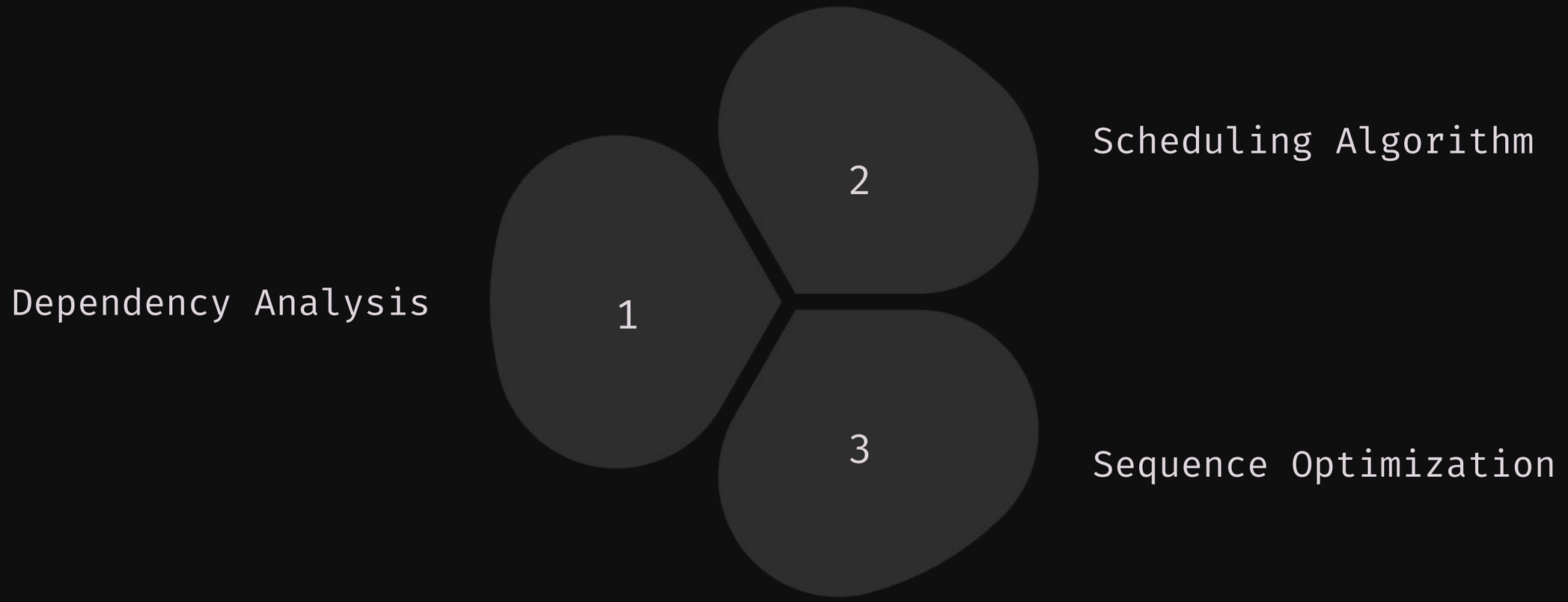
Matrix-Based Scheduling Model

A scheduling matrix models VM dependencies. It uses execution times to optimize testing. Matrix entries represent dependency relationships. The model minimizes total testing time. This is accomplished by prioritizing parallel execution. VMs are scheduled based on the matrix model.

VM	Test A	Test B	Test C
VM1	5s	10s	N/A
VM2	N/A	7s	12s

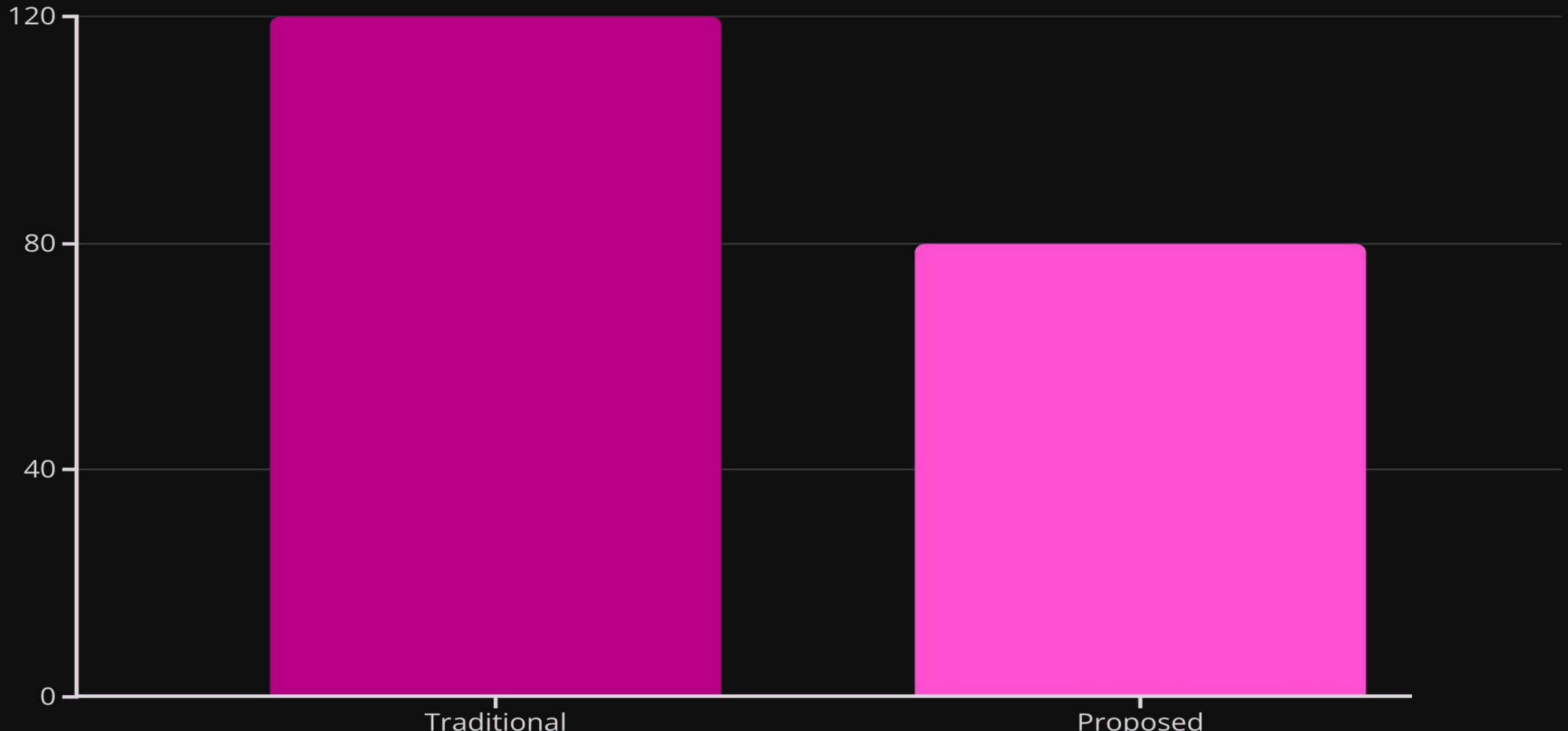
Test Sequence Optimization

Optimization reduces total test time. It takes into account key formulae. A scheduling strategy minimizes idle time. The sequence is optimized based on VM dependencies. The algorithm prioritizes VMs with fewer dependencies. This reduces the total test completion time.



Evaluation & Results

The proposed method yields reduced test time. Performance comparisons show improvements. Graphs illustrate faster testing cycles. Summarized results prove enhanced efficiency. The evaluation confirms practical benefits. These gains reduce operational costs.



Conclusion

The method enables faster, more efficient testing. It is useful for cloud platforms. Benefits include reduced test times. There is improved resource utilization. Optimizing VM test sequences saves cost. This contributes to greater scalability.

1. Faster Testing
2. Efficient Resources
3. Reduced Costs

Future Work & References

Future work may include enhanced dependency analysis. It could look at dynamic scheduling adjustments. More integration with CI/CD pipelines is another goal. This allows continuous and optimized cloud platform testing.

- Further optimize algorithms
- Integrate with CI/CD
- Dynamic scheduling

[Paper Reference]: A Test Method for Shortening Test Time of Cloud Computing Platforms, CCPQT 2023.