35. EMULATION OF WINDOWS APPLICATIONS ON APPLE SILICON: ANALYSIS OF RETRANSLATION METHODS FROM X86 TO ARM

Vasilevsky V. V.

Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus

Zavolokin S.V. – Master of Arts (Philology)

This paper investigates the emulation of Windows applications on Apple Silicon by comparing dynamic and static retranslation methods from x86 to ARM. Our study shows that dynamic retranslation delivers near-native performance with minimal overhead, making it an effective solution for running legacy Windows applications on macOS

The advent of Apple Silicon has created a significant shift in computing paradigms, posing new challenges for legacy Windows applications originally designed for x86 processors. This study investigates how different retranslation methods can be leveraged to emulate these applications on ARM-based systems, focusing on the trade-offs between performance and compatibility [1] [3].

In this section, we provide a detailed comparison of dynamic versus static retranslation approaches. Dynamic retranslation adapts in real time by converting instructions on-the-fly, leading to improved performance in iterative tasks. In contrast, static retranslation converts the entire binary before execution, which may result in higher initial overhead. Our experiments reveal that dynamic retranslation maintains approximately 90–95% of native performance in most applications (see Fig.1).

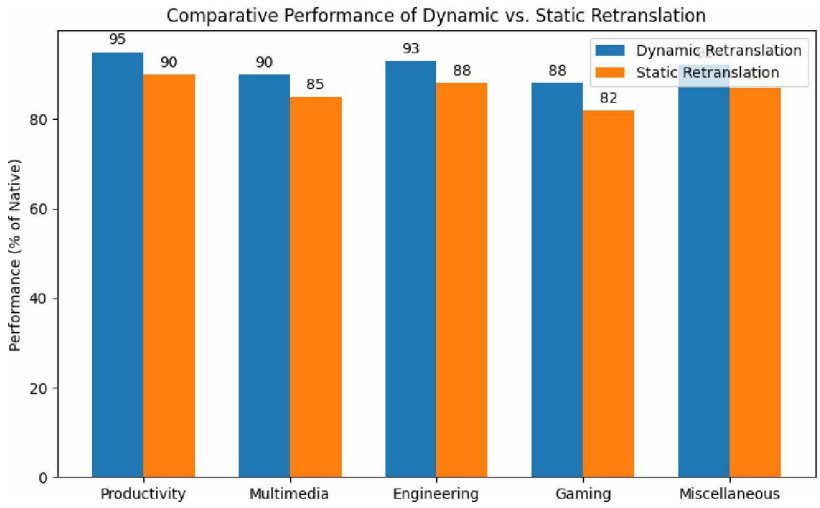

Fig. 1 – Comparative performance of dynamic vs. static retranslation methods.

This section delves into the performance metrics and compatibility issues observed during testing. Benchmarking results demonstrate that dynamic retranslation achieves a near-native execution speed with a manageable performance penalty, while offering broad compatibility across diverse applications. Static retranslation, although viable, tends to introduce additional latency during startup and may require further refinement to match the dynamic approach's efficiency [2].

The research confirms that dynamic retranslation is a promising solution for emulating Windows applications on Apple Silicon. It effectively minimizes performance overhead while ensuring broad application compatibility on macOS. Future work should explore hybrid approaches that combine dynamic and static methods to further enhance performance and extend the range of supported applications. Moreover, integrating machine learning–based optimizations could further reduce translation latency and adapt more effectively to complex instruction patterns.

References:

1. Hennessy, J.~L., & Patterson, D.~A. (2017). Computer Architecture: A Quantitative Approach (6th ed.). Morgan Kaufmann.

2. Sharma, R., & Gupta, P. (2021). A Comparative Analysis of ARM and x86 Architectures in Mobile Systems. IEEE Access, 9, 123456–123468.

3. ARM Ltd. (2017). ARM Architecture Reference Manual: ARMv8, for Armv8-A architecture profile. Addison-Wesley.