Simon Cole

12/8/2023

The Compiler Optimization Technique: Loop Fusion

Loop fusion is a compiler optimization technique wherein the bodies of two loops are combined into one. This optimization is accomplished by the compiler searching for instances in code where a safe fusion is possible. Loop fusion can only be used in very specific circumstances where variables used in the loops do not depend on intermediary operations and loops are the same number of iterations. “Fusion is safe when each use in the resulting loop has the same value that it did in the original loops” (1). See figure 1 for a simple example. In figure 1, print and someFn are executed the same number of times, but the loop control is executed half the original amount. This is where the benefits of a loop fusion optimization lie, reducing the number executions of the loop control. This in turn increases the speed and efficiency of the program.

Figure 1 (3)

A screen shot of a computer code

Description automatically generated

A common use case of loop fusion is combining loops that operate on the same data structure. This is partly because it usually inherently means the number of iterations for both loops will be the same. As Maruthamuthu et. al explain “loop fusion is particularly effective when loops operate on the same data structures, allowing for data reuse and elimination of unnecessary iterations” (2). A simple example of this can be seen in figure 2 where an array is populated in one loop then printed in another.

Figure 2 (3)

A screen shot of a computer code

Description automatically generated

Loop fusion is similar but distinct from loop unrolling. The two optimizations are sometimes done in conjunction in nested loops where the outer loop gets unrolled, and the inner loops are fused (1). In Conclusion, loop fusion is an effective optimization technique used to combine the bodies of multiple compatible loops into one. This in turn works to improve software efficiency and speed upon execution.

References

Cooper, K., & Torczon, L. (2012). *Engineering a compiler*. Elsevier.

Maruthamuthu, R., Dhabliya, D., Priyadarshini G, K., Abbas, A. H., Barno, A., & kumar, V. V. (2023). Advancements in Compiler Design and Optimization Techniques. *E3S Web of Conferences, 399*, 0407. <https://doi.org/10.1051/e3sconf/202339904047>

Cole, S. (2023). *LoopFusionCompilerOptimization* [Software]. GitHub. Retrieved from <http://github.com/simoncole/LoopFusionCompilerOptimization>