Tremplin Recherche 2024-2025

Analysing compilation optimization levels in real-time systems

Laboratoire d'accueil: Laboratoire d'Informatique Gaspard-Monge (LIGM)

Équipe de recherche: Logiciel, Réseaux, Temps-réel (LRT).

Partenaire international si poursuite envisagée en stage à l'étranger (mai-août): Non défini

Encadrement:
Yasmina ABDEDDAÏM yasmina.abdeddaim@esiee.fr (Bureau 4254)
Mohamed Amine KHELASSI mohamedamine.khelassi@esiee.fr

Filières concernées:
Artificial Intelligence and Cybersecurity (AIC),
Systèmes Embarqués (SE),
Informatique (INF),
Data science et Intelligence Artificielle (DSIA).

Année d'étude: E4 ou E5.

Context

Real-time systems are systems that have to operate under strict time constraints. These systems are essential in critical applications such as automotive systems, medical devices and aerospace.

Compiler optimizations are used to enhance software performance by improving execution speed and reducing resource consumption. However, these optimizations can also introduce variability in execution times. Using compiler optimizations in real-time systems can enhance programs performances, but it can also negatively impact the system where the predictability of execution times is more important than their minimization.

Objectives

Our objective is to investigate the relationship between GCC compiler optimizations and the execution time variability of programs in real-time systems using statistical and machine learning techniques. The project stages are:

1. Make a bibliography [1, 2] on the use on machine learning techniques in real-time analysis.
2. Understand the different GCC optimization levels.
3. Propose some statistical dispersion parameters to measure the execution time variability.
4. Collect execution times and hardware counter information (cache misses, memory access ...) by compiling a set of benchmark programs [3] under different optimization levels (O0, O1, O2, O3) on an Arm Cortex A53 platform.

5. Analyse the relation between, the program characteristics, the compiler optimization levels and the execution time variability using statistical and machine learning techniques.

References

[1] A. H. Ashouri, W. Killian, J. Cavazos, G. Palermo, and C. Silvano, "A survey on compiler autotuning using machine learning," *ACM Computing Surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[2] J. Chang and D. Park, "Work-in-progress: Searching optimal compiler optimization passes sequence for reducing runtime memory profile using ensemble reinforcement learning," in *Proceedings of the International Conference on Embedded Software*, 2023, pp. 3–4.

[3] M. Nicolella, S. Roozkhosh, D. Hoornaert, A. Bastoni, and R. Mancuso, "Rt-bench: An extensible benchmark framework for the analysis and management of real-time applications," in *Proceedings of the 30th International Conference on Real-Time Networks and Systems*, 2022, pp. 184–195.

[4] M. A. Khelassi and Y. Abdeddaïm, "Impact of compilation optimization levels on execution time variability," in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2024.