jh260017

Takahiro Katagiri (Nagoya University)

LLM-Driven Auto-Tuning for High-Performance Code Generation on Large-Scale HPC

● Purpose and Significance of the Proposed Research Project

Generative AI is rapidly advancing and is expected to support automatic program code generation. In HPC, high performance requires code optimized for each architecture and system. This proposal aims to develop AI-based code generation technologies that automatically produce high-performance code tailored to specific architectures, enabling sustained performance improvements in scientific and engineering applications.

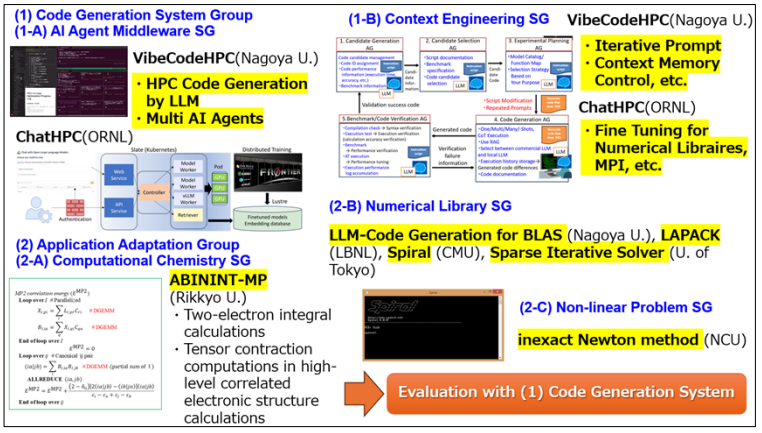
While supercomputing technologies such as “Fugaku” have improved software performance through **reliable Fortran code and MPI-based parallelization**, the recent AI boom has increased demand for GPU-based numerical computing. However, the number of skilled HPC and GPU programmers is limited. To address this, the project will develop an AI platform that improves HPC software development efficiency **using LLMs** and RAG. It will also integrate **mixed-precision computation** and **automatic performance tuning** with LLM-based code generation to build a sustainable HPC software infrastructure.

● Main Contributors

1. **Code Generation System Group: (1-A) AI Agent Middleware SG** (Leader: Takahiro Katagiri (Nagoya U.); Hoshino, Mukunoki, Graduate Students on Nagoya U. (Natsume, Akiba, Hayashi, Morita, Kotama, Isobe, Sakaguchi, Uchida), Ohshima (Kyusyu U.), Valero-Lara, Teranishi (ORNL) **(1-B) Context Engineering SG** (Leader: Tetsuya Hoshino (Nagoya U.); Katagiri, Mukunoki, Graduate Students on Nagoya U. (as same as 1-A), Ohshima (Kyusyu U.))
2. **Application Adaptation Group: (2-A) Computational Chemistry SG** (Leader: Yuji Mochizuki (Rikkyo U.); Katagiri, Hoshino, Mukunoki (Nagoya University)) **(2-B) Numerical Library SG** (Leader: Daichi Mukunoki (Nagoya U.); Katagiri, Hoshino, Morisaki (Nagoya U.), Nakajima (The U. of Tokyo), Valero-Lara, Teranishi (ORNL), Franchetti (CMU), Marques (LBNL)) **(2-C) Non-linear Problem SG** (Leader: Feng-Nan Hwang (NCU); Members: Graduate Students on NCU (Planned), Katagiri (Nagoya U.), Wang (NTU))

● Research Plan

The project consists of two groups. The **Code Generation System** Group will extend **VibeCodeHPC** to study AI agent middleware and context engineering, including agent assignment, tool use, iterative prompts, memory compaction, code search, local LLMs, and fine-tuning. In collaboration with ORNL, systems such as **ChatHPC** will be transformed into agent-based frameworks. The **Application Adaptation Group** will evaluate code-generation AI for computational chemistry, numerical libraries, and nonlinear solvers, including GPU acceleration, BLAS/FFT/LAPACK-related optimization, sparse solvers, and inexact Newton methods. Through these studies, the project will clarify the effectiveness and challenges of integrating code-generation AI with auto-tuning.

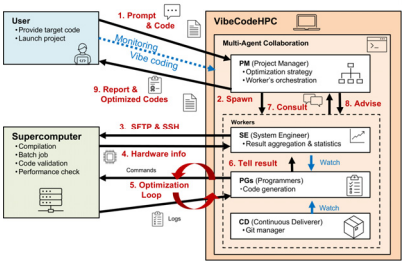


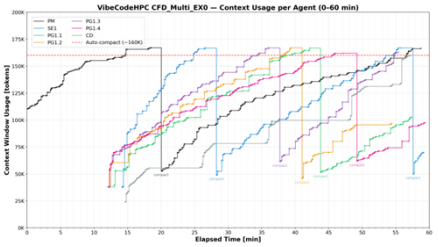
● VibeCodeHPC

https://github.com/Katagiri-Hoshino-Lab/VibeCodeHPC

● Multi-CLI Multi-Agent Auto-Tuning Framework

Multiple AI coding CLIs coordinate via tmux — no external orchestration framework required. Pluggable strategies adapt it to various tasks.





● Key Features

1. **Hierarchical Multi-Agents**: PM → SE ↔ PG × N → CD
2. **Pluggable Strategies**: HPC parallelization, local LLM deployment, GPU optimization — add your own
3. **Evolutionary Exploration**: Flat directory structure for parallel search
4. **tmux IPC**: Inter-agent communication with no special runtime

● Preliminary Result: CFD (Himeno Bench) Optimization

- ✓ Blocking/SIMD
- ✓ OpenMP
- ✓ OpenACC
- ✓ CUDA