

# Recent Optimizations in the Open64 Compiler

Michael Lai

AMD

# Types of Optimizations

- Aggressive loop optimizations
- Redundancy elimination
- Vectorization
- Structure optimizations
- IPA improvements
- High level optimizations
- Other improvements
- ~~Code generation~~ (next presentation)

# Aggressive Loop Optimizations

- Unrolling, unroll and jam
- Distribution
- Peeling
- Triangular loop optimization
- Proactive loop fusion, proactive loop interchange (if merging, code duplication, alias analysis, partial inlining)
- Multi-versioning

# Redundancy Elimination

- Redundant clearing of memory after calloc()
- Useless stores in inner loop
- Advanced dead store elimination

# Vectorization

- Recognized more operators: max, min, xor, shift, ...
- Outer loop vectorization
- Loops with conditional statements
- Cost function tuning, loops with small iteration count
- Reduced vectorized loop overhead and optimized generation of remainder loop

# Structure Optimizations

- Instance interleaving
- Array remapping
- Structure array copy optimization
- Enhanced structure splitting

# IPA Improvements

- Virtual function inlining
- Function devirtualization
- Indirect function call promotion/inlining
- Functions with “noreturn” attribute
- Improved function inlining heuristics and enhanced inline expansion of library functions

# High Level Optimizations

- Prefetching of arrays with inductive base addresses
- Reassociative CSE
- Factorization
- Improved inductive expression simplification
- If conversion

# Other Improvements

- OpenMP compilation
- Auto-parallelization
- OpenMP/auto-par library
- Improved processor affinity mapping in the OpenMP and parallel runtime library
- New scalability optimization flag (“-mso”)

# Other Improvements

- Optimized exception handling
- Optimized processing of Fortran 90 temporary arrays
- C++ STL tree optimization
- Structure copy return optimization
- Improvement to compile time and space
- Made `-O3` more aggressive

# Leverage Open64 Community

- In sync with Open64.net
- Merged in major infrastructure/optimization framework