1.

Embedded systems are targetted because they have very well defined speed, code size, power, and cost requirements

Function level searches may have better performance becefits, bu may need slower searches and more evaluations and simulations

Paper evaluates performance benefits and search time increases of function level compared to program level

Also, present a search algorithm that requires only one program simulation, and conducts function searches simultaneously to decrease time

Phase selection – whether or not certain optimizations should be applied

Phase order – the order we conduct all phases

Why embedded systems – Embedded system developers construct systems with just enough compute power and memory as necessary for a particular task, they are also limited on power, so reducing the speed, code size, and power requirements could reduce the cost of products greatly

Because we either have to simulate or use the slower, embedded processor for testing, search times can be exaggerated

Reducing search times is critical for evolutionary searches on embedded systems

Iterative searches can be performed at program-level or Function-level

Function level are more expensive as they may require several times more simulations, depending on the number of functions

A program with n functions needs n times more executions at the function level

However, with even more customization, they can lead to more efficient executables

Final product has function-level customization, with less then program-level cost

MAIN CONTRIBUTIONS:

Compare function-level vs program-level

Introduce File-level searches

Introduce the Hybrid search strategy

3.

Finer levels of search granularities enable greater flexibility in selecting distinct best sequences for different code segments in a program

Search algorithms can be implemented in the compiler or out

Inside has more flexibility and are faster, but is more difficult to implement

Also needs to be ported to different compilers (lots of work)

Outside more portable, bu a ot of compilers don’t allow different phase orders

All other implementations of function-level conduct their searches for each function individually,, and in isolation of the searches performed on the remaining functions in the program

Search strategies that employ wall-time or execution cycles, for evaluating the merit of each phase sequence, program execution/simulation time is typically the dominant factor in overall search time(instad of compilation time)

Victor’s on searching