Advanced Topics in Compilers

SCCManager

Simone Campanoni
simone.campanoni@northwestern.edu
Outline

• What is it and why NOELLE provides it

• SCC and its metadata
SCCManager

• It provides information/knowledge/semantics about a given SCC of a loop dependence graph
• Such semantics can be used to create degrees of freedom about satisfying the constraints dictated by the dependences included in an SCC

```cpp
/*
 * Dependences.
 */
auto sccManager = loop->getSCCManager();
```

Instance of the class `arcana::noelle::LoopContent`

Instance of the class `arcana::noelle::SCCDAGAttrs`
SCC and its metadata

For every SCC in an SCCDAG, there is an instance of the class `arcana::noelle::GenericSCC` that describes the semantics of such SCC.

```cpp
/*
  * Dependences.
  */
auto sccManager = loop->getSCCManager();
auto SCCDAG = sccManager->getSCCDAG();

/*
  * Check all SCCs.
  */
for (auto scc : SCCDAG->getSCCs()){
  // 8 lines: Print the instructions that...
}

/*
  * Fetch the SCC information.
  */
auto sccInfo = sccManager->getSCCAtrrs(scc);
```
Outline

• What is it and why NOELLE provides it

• SCC and its metadata
Instance of the class `arcana::noelle::GenericSCC`

```c++
/*
 * Fetch the SCC information.
 */
auto sccInfo = sccManager->getSCCAtrrs(scc);

/*
 * Check the SCC type
 */
if (isa<LoopCarriedSCC>(sccInfo)) {
    if (isa<ReductionSCC>(sccInfo)) {
        errs() << " It can be reduced\n";
    } else if (isa<LoopCarriedUnknownSCC>(sccInfo)) {
        errs() << " We don't know how to avoid executing this SCC sequentially\n";
    }
} else {
    auto liSCC = cast<LoopIterationSCC>(sccInfo);
    errs() << " It doesn't have loop-carried dependences\n";
}
```
Instance of the class `arcana::noelle::GenericSCC`

```c++
/*
 * Fetch the SCC information.
 */
auto sccInfo = sccManager->getSCCAtrrs(scc);
```

Instance of the class `arcana::noelle::ReductionSCC`

```c++
if (auto redSCC = dyn_cast<ReductionSCC>(sccInfo)){
/*
 * Use the APIs of this specific SCC.
 */
errs() << " Identity value = " << redSCC->getIdentityValue() << "\n";
errs() << " PHI accumulator = " << redSCC->getPhiThatAccumulatesValuesBetweenLoopIterations() << "\n";
}
```

This code can be found here: `noelle/examples/passes/scc_manager`
Always have faith in your ability

Success will come your way eventually

Best of luck!