Loop Diagonalization

Vedant Kumar

October 27, 2014

Overview

- Loop/matrix equivalence
- Fast exponentiation through diagonalization
- Writing the `llvm::LoopPass`

Loop/matrix equivalence

Some loops can be fully described by a matrix.

Loop/matrix equivalence

```
function FIB(n)
    a ← 1
    b ← 1
    for i ∈ [2...n] do
        tmp ← a
        a ← b
        b ← tmp + b
    end for
    return b
end function
```

Loop/matrix equivalence

function FIB(n)
  $\vec{v} \leftarrow [1, 1]^T$
  for $i \in [2...n]$ **do**
    $\vec{v} \leftarrow \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \vec{v}$
  end for
  return $\vec{v}[2]$
end function

Cost: n matrix multiplications, $O(nm^3)$

Fast exponentiation through diagonalization

$$M\vec{v} = \lambda_i \vec{v}$$

$$MP = P \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \ddots \end{bmatrix}$$

$$M = PDP^{-1}$$

Fast exponentiation through diagonalization

$$
\begin{aligned}
M^2 &= (PDP^{-1})^2 \\
&= (PDP^{-1})(PDP^{-1}) \\
&= (PD)(DP^{-1}) \\
&= PD^2P^{-1} \\
M^n &= PD^nP^{-1} \qquad \textit{(induction)}
\end{aligned}
$$

Cost: 1 diagonal matrix exponentiation, $\Theta(m \log_2 n)$
(Repeated squaring algorithm on m eigenvalues)

Fast exponentiation through diagonalization

function FIB(n)
    $[a, b]^T \leftarrow PD^{n-1}P^{-1}[1, 1]^T$
    return b
end function

Cost: $\Theta(m^3 + m \log_2 n)$

Why? The compiler diagonalized the loop!

$$M^n = \begin{bmatrix} 1 & \phi \\ \phi & -1 \end{bmatrix} \begin{bmatrix} \phi & 0 \\ 0 & 1-\phi \end{bmatrix}^n \begin{bmatrix} 1 & \phi \\ \phi & -1 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} 1 & \phi \\ \phi & -1 \end{bmatrix} \begin{bmatrix} \phi^n & 0 \\ 0 & (1-\phi)^n \end{bmatrix} \begin{bmatrix} 1 & \phi \\ \phi & -1 \end{bmatrix}^{-1}$$

Writing the `llvm::LoopPass`

- Filter away unsupported loops
- DFS on instruction graph to build coefficient matrix, M
- `EigenSolver(M).eigenvalues().asDiagonal()`
- `loop->replaceSuccessorsPhiUsesWith(...)`
- `loop->eraseFromParent()`

The end

Thank you. Any questions?

→ Full paper
→ Source code (requires 3.4, update to 3.5 still WIP)