Contents

• Introduction - CBD
• Optimizing compiler.
• PHYSBE – Physiological simulation benchmark experiment.
Causal Block Diagram (CBD)

- A graph of connected operational blocks.
- Connections denote signals.
- Blocks may be algebraic (Adder, Product) or Time-delay based (Delay, Integrator, Derivatives). Order of blocks is important – cause & effect.

\[ z = x + y; \quad x = 2, \ y = 3; \quad z = 5; \]
\[ x, \ y, \ z \in \mathbb{R} \]
\[ f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \]
(semantic mapping)
Causal Block Diagram (CBD)

• Concrete syntax (block-diagram) is transformed into abstract syntax.
• Abstract syntax transformed to semantic syntax and semantic mapping – the meaning of the graph (CBD).
Issues

• Non-varying computations are often recomputed adding to performance overheads.
• Non-trivial scientific problems usually involve intensive and large number of computations.
• Optimization must not change the meaning of a model. Still, optimization possible.
Opportunities for optimization: leveraging the dependency graph.

- Constant propagation may improve performance.
- Redundant computations may be eliminated - loop invariants; expressions may be separated into parts: constant, invariant and variable. Only variables need to be recomputed.
- Compiled code typically runs faster than interpreted code.
Examples:

• x, y are constants; z = x + y
  for iteration = 1 to 60000
    z.signal[iteration] = x + y

• Optimized execution:
  z.signal[0] = x + y
  for iteration = 2 to 60000
    z.signal[iteration] = z.signal[0]
Examples: Re-association

• We can exploit algebraic properties; e.g., addition is commutative ...

• $x, y, z$ are constants; $a, b$ are variables:
  
  ```
  for iteration = 1 to 100000
      s.signal[iteration] = x + y + z + a + b
  ```

• Optimized execution:
  
  ```
  temp = x + y + z
  
  for iteration = 1 to 100000
      s.signal[iteration] = temp + a + b
  ```

• Savings: $2 \times 100000 - 2$ additions
Optimizing compiler for CBD

- Transforms CBDs into some optimized C programs.
- CBDs may have cycles: uses optimized LAPACK library for solving systems of linear equations, where necessary.
- Redundant computations are eliminated without loss of accuracy.
Model with 2 linear loops
PHYSBE: A Physiological simulation benchmark

- A classical model of the human circulatory system (John McLeod, 1966).
- An example of a non-linear continuous system. Linearization through approximation.
- The mathworks: Simulink
PHYSBE: Equivalent model in AToM$^3$. 

- A classical model of the human circulatory system (John McLeod, 1966).
- An example of a non-linear continuous system. Linearization achieved.
Aorta:
Right Heart:
Generated C – code for Aorta

```c
for (i = 1; i < N; ++i)
{
    result[0][i] = 1.0;
    result[1][i] = 0.0;
    result[2][i] = 3.0;
    result[3][i] = 4.0;
    result[4][i] = result[4][0];
    result[5][i] = 0.0;
    result[6][i] = 0.0;
    result[7][i] = result[7][i-1] + result[22][i-1] * 0.01;
    result[8][i] = result[7][i] + 0.0;
    result[9][i] = 0.0;
    result[10][i] = result[10][i-1] + result[24][i-1] * 0.01;
    result[11][i] = 100.0;
    result[12][i] = result[10][i] + 100.0;
    result[13][i] = 1 / result[12][i];
    result[14][i] = result[6][i] * result[13][i];
    result[15][i] = result[4][0] * result[14][i];
    result[16][i] = -result[15][i];
    result[17][i] = 0.0;
    result[18][i] = 5.0;
    result[19][i] = result[19][0];
    result[20][i] = 6.0;
    result[21][i] = result[21][0];
    result[22][i] = result[16][i] + result[21][0] + 0.0;
    result[23][i] = result[23][0];
    result[24][i] = result[24][0];
    result[25][i] = result[14][i];
    result[26][i] = 1.25;
    result[27][i] = result[27][0];
    result[28][i] = result[12][i] * result[27][0];
    result[29][i] = result[28][i];
}
```
More ... Circle test

• The Next slide shows the result for circle test using 60,000 iterations.
Circle test result: 60000 iterations.
Conclusion

• Hierarchical ordering in CBD provides opportunities for optimizations.
• Elimination of redundant computations might result in performance gains.
Questions?