Learning with Analytical Models

Huda Ibeid, Siping Meng, Oliver Dobon, Luke Olson, William Gropp
University of Illinois at Urbana-Champaign
Outline

1. Motivation
2. Applications
   1. Stencil Computation
   2. Fast Multipole Method
3. Analytical Models
4. Supervised Machine Learning
5. Approach: Hybrid Model
6. Evaluation
7. Conclusions
Motivation

| Pros                                      | Cons                                           |
|-------------------------------------------|------------------------------------------------|
| No or minimal training                    | Requires domain expertise                      |
| Requires domain expertise                 | Robustness                                     |
| Rely on simplifying assumptions           | Curse of dimensionality                        |
| Increasing architecture complexity        |                                               |

Goals:
- Minimize prediction cost
- Maintain reasonable prediction accuracy
Applications

• Stencil Computation
• Fast Multipole Method
Stencil Computation

\[
\text{for } t \leftarrow 0 \text{ to } \text{timesteps do} \\
\text{for } k \leftarrow 1 \text{ to } KK - 1 \text{ do} \\
\quad \text{for } j \leftarrow 1 \text{ to } JJ - 1 \text{ do} \\
\quad \quad \text{for } i \leftarrow 1 \text{ to } II - 1 \text{ do} \\
\quad \quad \quad \chi_{i,j,k}^t = C_0 \times \chi_{i,j,k}^{t-1} + C_1 \times (\chi_{i-1,j,k}^{t-1} + \chi_{i+1,j,k}^{t-1} + \\
\chi_{i,j-1,k}^{t-1} + \chi_{i,j+1,k}^{t-1} + \chi_{i,j,k-1}^{t-1} + \chi_{i,j,k+1}^{t-1}) \\
\quad \quad \quad \text{end for} \\
\quad \text{end for} \\
\text{end for} \\
\text{end for}
\]
Stencil Computation

Assumptions

- Arithmetic and memory operations can be overlapped
- Floating point operations negligible

Given a grid size: \( N = I \times J \times K \) elements of order \( l \), total memory requirement to compute an X-Y plane

\[
S_{\text{total}} = P_{\text{read}} \times S_{\text{read}} + P_{\text{write}} \times S_{\text{write}}
\]

\[
P_{\text{read}} = 2 \times l + 1
\]

\[
S_{\text{read}} = II \times JJ
\]

\[
P_{\text{write}} = 1
\]

\[
S_{\text{write}} = I \times J
\]
Stencil Computation

On an architecture with a memory hierarchy of \( n \) cache levels, total time to compute a stencil is

\[
T = T_{L1} + T_{Li} + \cdots + T_{Ln} + T_{mem}
\]

\[
T_{Li} = T_{li}^{data} \times Hits_{Li}
\]

\[
T_{li}^{data} = data \ast \beta_{mem_{Li}}
\]

\[
Hits_{Li} = Misses_{Li-1} - Misses_{Li}
\]

\[
Misses_{Li} = \lfloor IL/W \rfloor \times JJ \times KK \times nplanes_{Li}
\]

\[
nplanes_{Li} = \begin{cases} 
1, & \text{if } R_1 \\
(1, P_{read} - 1], & \text{if } \neg R_1 \land R_2 \\
(P_{read} - 1, P_{read}], & \text{if } \neg R_2 \land R_3 \\
(P_{read}, 2 \times P_{read} - 1], & \text{if } \neg R_3 \land \neg R_4 \\
2 \times P_{read} - 1, & \text{if } R_4
\end{cases}
\]

\[
R_1 : ((size_{Li}/W) \times R_{col} \geq S_{total}), \quad R_2 : ((size_{Li}/W) > S_{total}), \\
R_3 : ((size_{Li}/W) \times R_{col} > S_{read}), \quad R_4 : ((size_{Li}/W) \times R_{col} < P_{read} \times II)
\]

\[
R_{col} = P_{read}/(2 \times P_{read} - 1)
\]
Supervised Machine Learning
Stencil Computation

- \( \mathbf{X} = (l, J, K, b_i, b_j, b_k) \) where \( l \times J \times K = \{1 \times 16 \times 16 \ldots 1 \times 128 \times 128\} \) with a 16 points stride and \( b_i \times b_j \times b_k = \{1 \times 1 \times 1 \ldots l \times J \times K\} \).

(a) Decision Trees  (b) Extra Trees  (c) Random Forests
Hybrid Model

- Analytical model
- Two ensemble methods
- Training algorithm
- Prediction algorithm
Evaluation

Stencil Computation

- First, we evaluate hybrid approach on areas that analytical models cover accurately
- $X = (I, J, K)$ where $I \times J \times K = \{128 \times 128 \times 128 \ldots 256 \times 256 \times 256\}$ with a 16 points stride

(a) Extra Trees (b) Hybrid Model
Evaluation

Stencil Computation

- Next, we add loop blocking to the analytical models
- Analytical model $MAPE = 42\%$
- $X = (I, J, K, b_i, b_j, b_k)$ where $I \times J \times K = \{1 \times 16 \times 16 \ldots 1 \times 128 \times 128\}$ with a 16 points stride and $b_i \times b_j \times b_k = \{1 \times 1 \times 1 \ldots I \times J \times K\}$

(a) Extra Trees

(b) Hybrid Model
Evaluation

Stencil Computation

• Lastly, we evaluate the hybrid model on a region that is not covered by the analytical models

• $X = (l, J, K, t)$ where $l \times J \times K = \{128 \times 128 \times 1 \ldots 176 \times 176 \times 1\}$ with a 16 points stride and the number of threads $t = \{1 \ldots 8\}$

![Box plots for MAPE (%) with different training set sizes for (a) Extra Trees and (b) Hybrid Model.](image-url)
Evaluation

Fast Multipole Method

- FMM is a highly complex algorithm with several different phases, a combination of data structures, fast transforms, and irregular data access
- We do not tune the analytical models ($MAPE = 84.5\%$)
- $X = (t, N, q, k)$ where $t = \{1 \ldots 16\}$, $N = \{4096, 8192, 16384\}$, and $k = \{2 \ldots 12\}$

(a) Extra Trees

(b) Hybrid Model
Conclusions

• The hybrid approach is effective in predicting the execution time by reducing the MAPE score of pure machine learning models.

• The hybrid model requires small training dataset to carry out predictions with reasonable accuracy, thus making it suitable for hardware and workload changes.