Exploration of Compiler Optimization Techniques

Guidance : Prof P.Viswanath

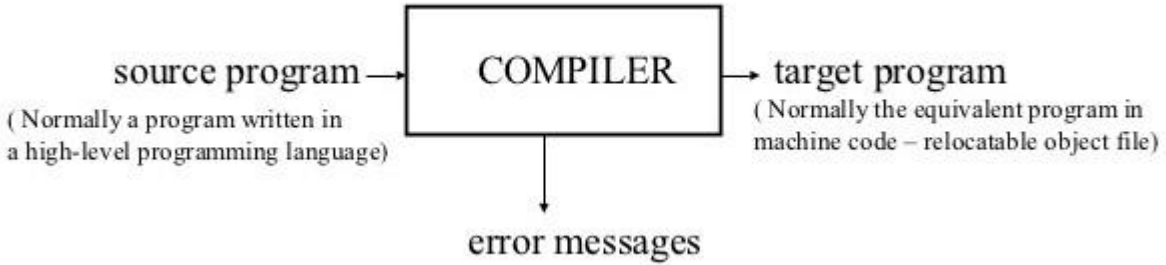
—

Team Members:
Sumanth.M(201501033)
Raviteja.G(201501022)

Contents

Introduction

Compiler :



- An Optimizing compiler is a compiler that tries to minimize some attributes of an executable computer program.
- Compiler has mainly three layers namely Front-End, Intermediate-Representation and Back-End. Currently , Research is going on IR Phase.
- Compiler converts every part of the instruction in program to provide the optimized kind of it by considering metrics.
- The metrics are Power Consumption, Execution Time, Code Size, Resource-Allocation, Scheduling, Code-generation etc.

Problems

Problems of Compiler Optimizations :

- Selection of Compiler optimizations :

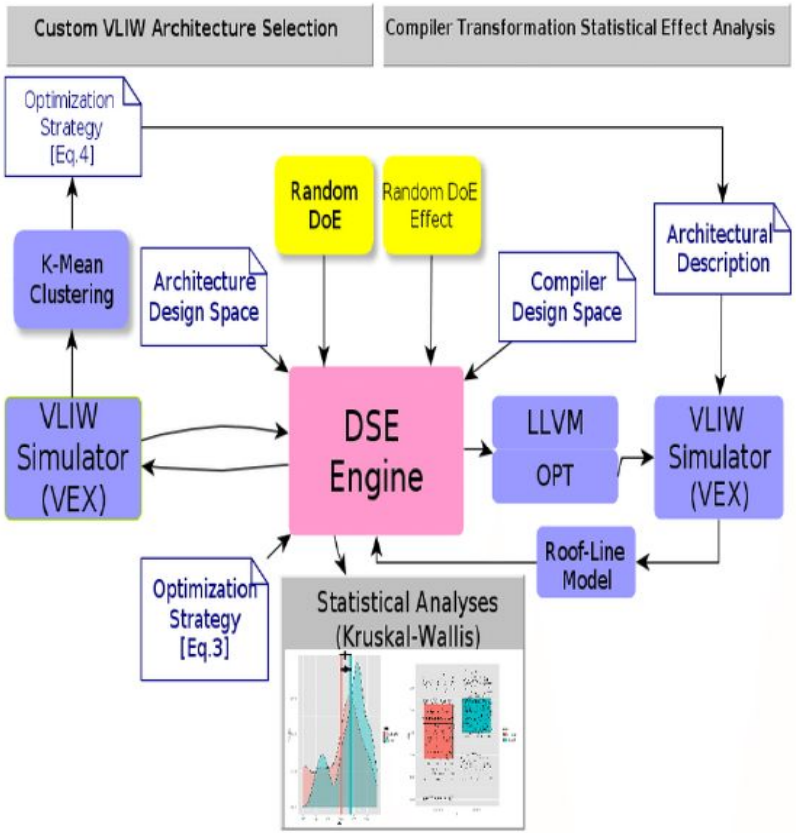
  It is about the selection of the best code optimizers.

- Phase-Ordering Problem:-

  It is about the phase-ordering of code optimizations.

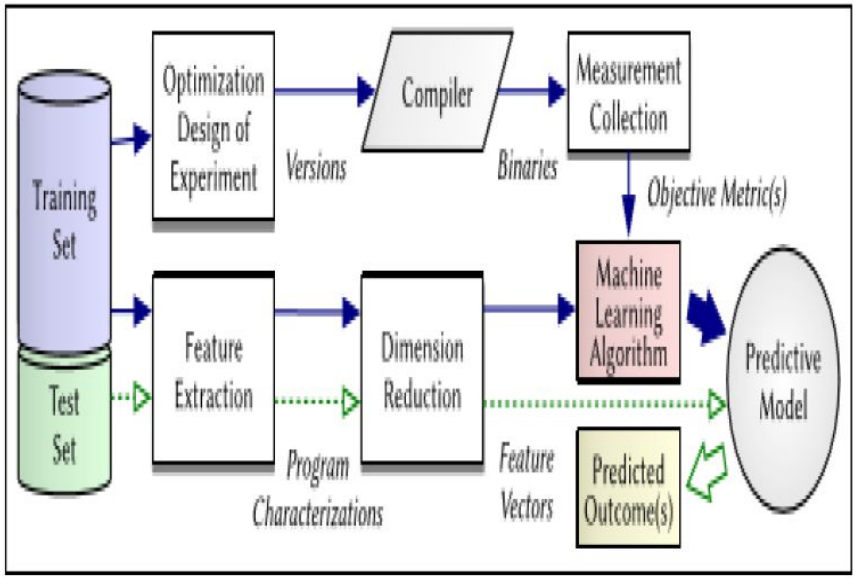
AutoTuning Overview & Techniques

Design Space Exploration



- It is a method of activity where one can explore the parameters of space before the actual design.
- It is a strategy where pruning occurs. It is the model which can utilize the design space effectively
- Platform should perform at utmost level.
- Consumption of power and non functional parameters should be minimized.

Machine Learning Approach



- The top phase is training phase flows along the way of various components.
- The bottom phase is testing phase is passed through the trained model so that we can estimate the result for the selected application.
- A Machine can explore the data by three ways
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning

Supervised Learning :

- It predicts the learning function from the training data which is labeled.
- This functions helps in predicting the testing data which are unseen.

Unsupervised Learning :

- Unsupervised Learning is special type of machine learning in which a function is inferred to draw the hidden pattern from unlabeled data.
- It is free from error signals when evaluating the output.
- Since all the data which is used for learning is unlabeled.

Reinforcement Learning :

- The most interesting part in RL is testing and training phases are inter-twined.
- Neither Supervised nor Unsupervised

Machine Learning Models :

1 . Supervised Methods :

- SVM and Linear Models
- Random Forests and Decision trees
- Graph Kernels
- Bayesian Networks

2 . Unsupervised Methods :

- Clustering Methods

3 . Reinforcement Methods :

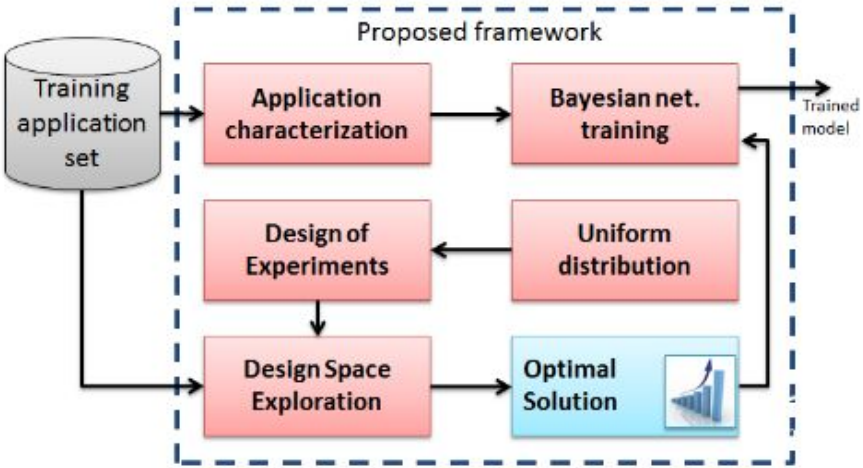
- Markov Decision Process

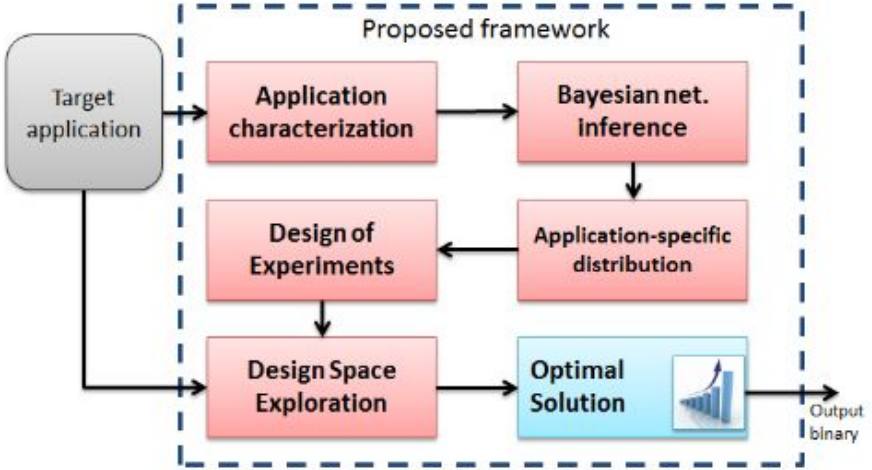
Supervised Learning :

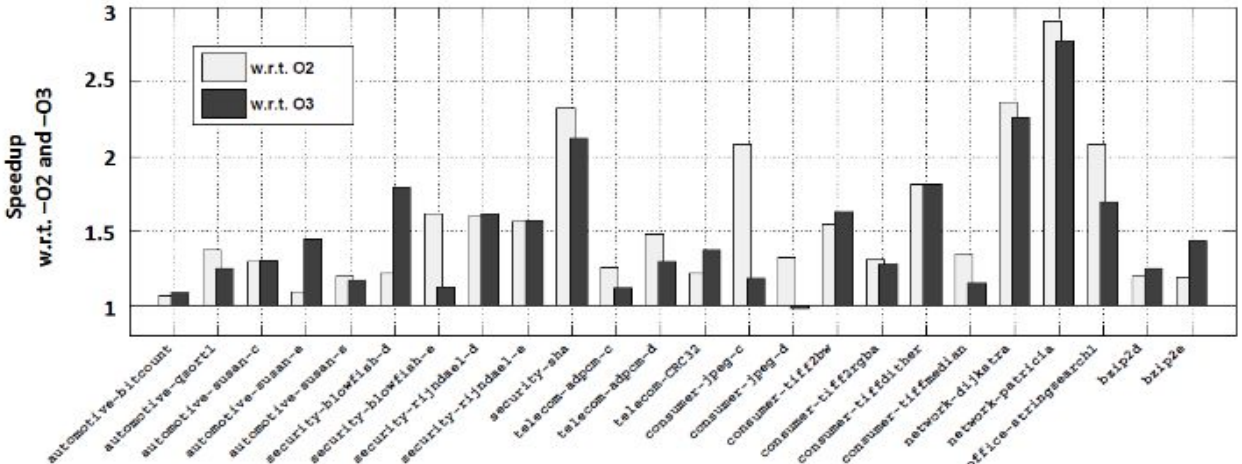
- Bayesian Network is a direct acyclic graph in which every node indicates variables and every edge is inter-dependent of another.

- Probabilities can be feed to the model as input which is called as Evidence. Select top 15% speedup compiler sequences and feed to input

- Training Phase & Testing Phase along with improvement in performance.


Proposed framework
Training application set
Application characterization
Bayesian net. training
Trained model
Design of Experiments
Uniform distribution
Design Space Exploration
Optimal Solution


Proposed framework
Target application
Application characterization
Bayesian net. inference
Design of Experiments
Application-specific distribution
Design Space Exploration
Optimal Solution
Output binary


Speedup w.r.t. –O2 and –O3
w.r.t. O2
w.r.t. O3
3
2.5
2
1.5
1
automotive-bitcount
automotive-qsort1
automotive-susan-c
automotive-susan-e
automotive-susan-s
security-blowfish-d
security-blowfish-e
security-rijndael-d
security-rijndael-e
security-sha
telecom-adpcm-c
telecom-adpcm-d
telecom-CRC32
consumer-jpeg-c
consumer-jpeg-d
consumer-tiff2bw
consumer-tiff2rgba
consumer-tiffdither
consumer-tiffmedian
network-dijkstra
network-patricia
office-stringsearch1
bzip2d
bzip2e

Predictions

Predictions

- **Speedup Prediction :** Construction, Examination and Validation of a model to obtain the outcome of unobserved timing.

- **Compiler Sequence Prediction :** Model which gives the output of set of best compiler sequences or passes for applying on a given application.

- **Tournament and Intermediate Prediction :**
  - Prediction of two input sequences of optimizations which are from two different classes is Tournament Prediction.
  - Intermediate Prediction is a model in which we repeatedly estimate the speedup of the application which is optimized.

- **Feature Prediction :** Feature Prediction Model uses the most important features which can affect performance of a model.

THANK YOU