Divergence Analysis (Discussion) (v.0.0.0)

The problem that the paper discusses

What is the specific problem that the paper solves

Introduction of a static analysis that discovers data, control divergence (called, Divergence Analysis) to produce code for better register resource management in SIMD execution model.

Why is this problem important

To generate faster code for GPU's on the SIMD execution model.

Who will benefit immediately from the solution of this problem

GPU's are being used for more general purpose programming these days for their threaded execution model (parallel execution) for speed up in performance. General programming and industrial applications of software tools.

What is the theory upon which the problem is defined?

Static Analysis is the study of a program source code for compiler optimization phase of the code generation pipeline.

The context of the paper

What is the general context of the paper?

Compiler paper. Deals with providing compilers with techniques that help them understand and improve divergent code.

Since when is this context source of research?

The authors have mentioned several material and have pointed out the close links this paper mentioning the key differences in detail.

Automatic optimizations have been around for decades, the authors have mentioned various papers from the 1960's, 1990's and early 2000's.

Is there any book that provides an in-depth overview of this problem?

Chapter 10 Instruction-level parallelism from the Dragon's book.

What can we find about this context in wikipedia?

https://en.wikipedia.org/wiki/Data_dependency.

The Solution

What is the key idea used to solve the problem?

Static Analysis.

Why is this specific idea different from what had been done before?

The authors claim that their work improves on a similar technique was proposed by researches at Saarland University (DE) called vectorization analysis.
They support their claim by providing evidence in the form of raw register pressure numbers making the claim that the said technique miss catagories some cases of divergent variables.

Is there any algorithm involved in the solution?

Builds upon several other works for example converting to gated static single assignment form.
The authors have introduced a language and operational semantics, rules for the language as well as a constraint system accompanied with proofs (structural induction).

Is the solution exact, or does it approximate an optimal? In the latter case, what would be the price of finding the optimal?

Approximate. Two techniques are described in the paper, the latter being more precise.

The Organization of the Paper

How was the abstract organized?

"Context - Problem - Solution - Results".

How was the introduction organized?

Eight paragraphs:

1. Introduction
2. Context
3. Background Concepts
4. Problem definition
5. Key concept
6. Precision
7. Artifacts
8. References

What is discussed in each section of the paper?

1. Introduction
2. Background
3. Divergence Analysis
4. Divergence Aware Register Spiller
5. Experiments
6. Conclusion
7. Acknowledgement
8. References

What was left for the conclusion?

Reiteration of the problem statement, and referencing specific sections of the article where they provided solutions.
The authors pointed out that their work can be expanded upon to provided a better user experience for programmers, they have identified that their work helps compilation but not the programmer and they have left hints for further work in this regard.
Ending with why their work is important.

The Written Style

Can you give a title to each paragraph in the introduction?

1. *Introduction*: GPU programming and usage trends;
2. *Context*: GPUs: Parallel execution model;
3. *Background Concepts*: Data and control divergence and how it relates to GPU instruction generation?;
4. *Problem definition*: Purpose of this work;
5. *Key concept*: Why is Divergence analysis important and current techniques;
6. *Precision*: A more precise solution with comparison;
7. *Artifacts*: Benchmarks, statistics and availability of artifacts;
8. *References*: Impact and presentation of work.

Can you find examples of sentence topics to every paragraph in the introduction?

1. *Introduction*: and novel programming abstractions are developed for them;
2. *Context*: However, divergences may happen in less regular applications;
3. *Background Concepts*: A thread identifier, for instance, is inherently divergent;
4. *Problem definition*: The main goal of this article is to provide compilers with techniques that help them understand and improve divergent code;
5. *Key concept*: The divergence analysis is important in different ways.;
6. *Precision*: Second, in order to more precisely identify divergences;
7. *Artifacts*: our implementation of the divergence analysis runs in linear time;
8. *References*: work in divergence analysis for SIMD architectures.

Can you give examples of techniques used to connect different paragraphs?

The discussion of an aspect of the paper topic in one paragraph flows into the second paragraph where the aspect is touched on in a different light but also the issue presented in the previous paragraph is elaborated on. For example:

1. However, divergences may happen in less regular applications. (Section 1, paragraph 2)
2. Data divergence occurs if the same variable name is mapped to different . (Section 1, paragraph 3)

Can you find examples of declarative, illustrative and enumerative paragraphs?

- Declarative (Section 1, Paragraph 1, Sentence 1)
  Increasing programmability and low hardware cost are boosting the use of graphical processing units (GPU) as a tool to run general-purpose applications.
- Illustrative (Section 1, Paragraph 6, Sentence 4)
  There exists a recent number of divergence-aware code optimizations, such as Coutinho et al.'s [2011] branch fusion and Zhang et al.'s [2011]thread reallocation strategy.
- Enumerative (Section 4, Paragraph 1, Sentence 2)
  However, in the context of graphics processing units, we have different types of memory to consider. (Register..., Shared Memory..., Local Memory..., Global Memory...).

The related works

What is the purpose of the related works section in this paper?

No related work section, however there is a lot of detail in Section 2, Background about related works with references and key differences.

What are the earliest papers about this problem?

- The ILLIAC IV Computer (G H Barnes et al.) 1968

What is the most seminal paper in this field of research?

- TRANQUIL: a language for an array processing computer (Norman E. Abel et al.)
- The ILLIAC IV Computer (G H Barnes et al.)
- Barrier inference (Alex Aiken et al.)

Which good conferences have recently published papers about similar problems?

- ACM Transactions on Parallel Computing (2021)
  - Pointer-Based Divergence Analysis for OpenCL 2.0 Programs (Shao-Chung Wang et al.)
- ACM Transactions on Architecture and Code Optimization (2018)
  - On-GPU Thread-Data Remapping for Branch Divergence Reduction (Huanxin Lin et al.)

Validation

Which points do the authors try to prove with experiments?

- Reporting number of divergent variables.
- Performance improvement of generated code.
- Register pressure.

Are the experiments rigorous enough?

Since their work is used in an industry scenario they have provided more than enough experiments.

Which visual resources have the authors used to present data?

Pi Charts, bar graphs, tables and plots.

Which statistical theory have the authors used in this paper?

Not much simple benchmarks, raw performance numbers.

Resources

Do the authors use any particular type of notation?

- C source code to describe the problem
- eBNF to define a language
- Big step operational semantics to describe the Machine
- Basic block diagram with assembly style language
- Constraint system for the divergence analysis
- Annotated assembly code
- Logic rules

Which examples have the authors used to present their ideas?

One simple example and one complex used throughout the section 2, Background section and in section 3, Divergence analysis. The authors have taken the example forward making it very easy to follow with clarity.

Which visual resources have the authors used to explain their points?

- Source code: 2 programs illustrating the idea of data divergence. (Section 2: Background Fig 1)
- Basic block diagram to describe execution of 1 program from section 2. (Section 3: Divergence Analysis Fig 8)
- Conversion of the basic block diagram to GSA form. (Section 3: Divergence Analysis Fig 9)
- Example program to illustrate Higher-Degree Polynomials. (Fig 15)
- Illustrated Assembly code of Fig 1: to introduce the divergence aware register spiller. (Section 4: Divergence Analysis Register spiller Fig 16 17 19)
- Several examples of benchmarks (Pi Charts, bar graphs, tables, plots)

(Only in case there is on-line material available to support the paper) Which material is publicly available?

The code is open-source implemented in the Ocelot industry compiler. See more, (http://simdopt.wordpress.com), (https://llvm.org/doxygen/LegacyDivergenceAnalysis_8cpp_source.html) and (https://groups.google.com/g/gpuocelot).