Summary
Data dependence profiling for speculative optimizations
Venkatesan Packirisamy
February 23, 2005

1 To be completed before class

What are the problems solved by this paper? (50 words)
Compiler optimizations have been used to improve the performance of many programs. The compiler has the advantage of being able to see the entire program and decide on the optimizations, but the main disadvantage of the compiler is that it doesn’t have run-time information. Because of this the compiler is forced to take conservative decisions resulting in less optimized code. For example, if there is lot of pointer accesses, the compiler cannot tell which pointer access which data. With profiling, the compiler can get some runtime information and hence can do more aggressive optimizations. Data dependence information is essential to do many optimizations in the compiler. Data dependence profiling can be used to get dynamic dependence information and that would help the compiler to do more aggressive optimizations with the help of some hardware like the ALAT of Itanium. The data dependence profiling can be very costly if done in software. So there is a need for doing efficient data dependence profiling.

What are the approaches attempted by this paper? (50 words)
In this paper, the code is first instrumented at each memory reference (register dependences can be easily got from static analysis). The instrumentation is done after the global optimization phase of the ORC, to reduce the number of instrumented memory references. Each memory reference is assigned a static reference id. When the instrumentation library call is executed, the memory reference is mapped to a location in the shadow space depending on its address. By examining the locations in the shadow space we will be able to get the dependence information. This is much more efficient than pair-wise comparisons. Also the paper tries to reduce the granularity of profiling and also used sampling to improve the efficiency.

What are the main conclusions of this paper? (50 words)
The data dependence profiling is used to improve speculative Partial redundancy elimination (PRE). This gives an average performance improvement of 8.22% for Spec2000 benchmarks. This is also found to be much better than alias profile based optimization. This shows that if compiler can get the data dependence information, it can do much better optimizations. The paper also shows that having a granularity of 32 bytes will lead to just about 15% false dependences. So if we use optimizations to improve our data dependence profiling, we won’t lose much information and still we will be able to get good speedup.

2 To be completed after class

Did this paper address an important issue? Explain. (100 words)
Data dependencies analysis has been used to parallelize scientific programs. But such analysis has only limited scope in C or C++ programs. In C programs the use of pointers leads to additional complexity. Even after detailed alias analysis, it is very difficult to know statically in the compiler which location a pointer reference is pointing to. This is because the compiler lacks dynamic information. This problem can be solved using data dependence profiling. Though the need for data dependence is well known it
was difficult to use the profile information. This is because profiling cannot tell in absolute certainty about the pointer references. So to ensure the correctness of the code, the compiler cannot rely on that data. But current architectures like Itanium have support for data speculation. Now the compiler can use data dependence profiling information and exploit such hardware features to optimize the code. The paper shows the performance of one such optimization - speculative PRE using the data dependence profile information. A major difficulty in collecting data dependence profile information is its huge volume. The paper addresses this issue by proposing an efficient way of collecting profiles. Also the paper studies techniques to optimize the profile collection process by using sampling or reducing granularity.

**Are the proposed approaches valid? Describe its strength and weakness. (100 words)**

For scientific programs written FORTRAN language, there is not much need for data dependence profile information. But as there is a trend towards using object oriented languages, the use of pointers is getting very frequent (java, C++, etc). The absence of alias information makes compiler to take conservative decisions. This makes the compiler almost un-useable for such programs. So there is definitely a need to do data dependence profiling. This is the only way to make the compiler effective. The paper also demonstrated the capability of data dependence profiling technique by applying it to do speculative PRE for Itanium processor. This shows the usefulness of doing data dependence analysis. Also it is shown that data dependence analysis is much better than alias profile information.

- **Strengths:** The paper proposes an efficient way to do profiling. Also the profiling technique is not limited by any structures of the program like loops, functions, etc.
- **Weakness:** The weakness of any profile driven optimization is that some dependences can be input sensitive. There should be a way to characterize the dependences such that we use the profile information only for input independent data dependences.

**Do the results support the conclusions? Explain. (100 words)**

The paper studied the usefulness of the data profiling information by applying it to do speculative partial redundancy elimination in Itanium processor. The performance improvement is 8.22% in the average. This shows that data dependence profiling is very important to do speculative optimizations. Also the paper compared it with using alias profile information and found that the data dependence profile is much more accurate than the alias profile information. The paper also shows how the efficiency of the profiling can be improved. By reducing the granularity of the profiling, we cause only 15% of fake dependences. Also it is shown that fairly accurate profile can be collected using sampling. Thus the technique proposed is found to be very efficient and the profile is also shown to be very useful in improving speculative optimizations.

**Describe the potential future works? (100 words)**

The main disadvantage of the profiling is that some data dependence can be input dependence. So if we run the optimized program that was optimized using the profile collected due to one input, with another input, then the performance sometimes can be very less. This can even reduce the performance. So there must be someway of characterizing which information is input dependent and which is not. Here the data dependence is used only to do speculative PRE. We could use the data dependence information to do further optimizations like automatic parallelization of loops. Many compiler optimization techniques could benefit from dependence information. This is very important in in-order processors where many optimizations are done in the compiler.