Lecture 20:

How to Give a Bad Talk

Professor David A. Patterson
Computer Science 152
Fall 1997

CS 152: Remaining Schedule

- **What's Left:**
- **Thursday 12/4: Oral Presentations, Noon to 7PM, 6thfloor**
- **Friday 12/5: Last lecture**
- **Monday 12/8: Writtenrepoert turned into 634 Soda**
- **Monday 12/15: grades posted**

7 Talk Commandments for a Bad Career

I. Thou shalt not illustrate.
II. Thou shalt not covet brevity.
III. Thou shalt not print large.
IV. Thou shalt not use color.
V. Thou shalt not skip slides in a long talk.
VI. Thou shalt cover thy naked slides.
VII. Thou shalt not practice.

Following all the commandments

- We describe the philosophy and design of the control flow machine, and present the results of detailed simulations of the performance of a single processing element. Each factor is compared with the measured performance of an advanced von Neumann computer running equivalent code. It is shown that the control flow processor compares favorably , given a reasonable degree of parallelism in the program.
- We present a denotational semantics for a logic program to construct a control flow for the logic program. The control flow is defined as an algebraic manipulator of idempotent substitutions and it virtually reflects the resolution deductions. We also present a bottom-up compilation of medium grain clusters from a fine grain control flow graph. We compare the basic block and the dependence sets algorithms that partition control flow graphs into clusters.
- Our compiling strategy is to exploit coarse-grain parallelism at function application level: and the function application level parallelism is implemented by fork-join mechanism. The compiler translates source programs into control flow graphs based on analyzing flow of control, and then serializes instructions within graphs according to flow arcs such that function applications, which have no control dependency, are executed in parallel.
- A hierarchical macro-control-flow computation allows them to exploit the coarse grain parallelism inside a macrotask, such as a subroutine or a loop, hierarchically. We use a hierarchical definition of macrotasks, a parallelism extraction scheme among macrotasks defined inside an upper layer macrotask, and a scheduling scheme which assigns hierarchical macrotasks on hierarchical clusters.
- We apply a parallel simulation scheme to a real problem: the simulation of a control flow architecture, and we compare the performance of this simulator with that of a sequential one. Moreover, we investigate the effect of modelling the application on the performance of the simulator. Our study indicates that parallel simulation can reduce the execution time significantly if appropriate modelling is used.
- We have demonstrated that to achieve the best execution time for a control flow program, the number of nodes within the system and the type of mapping scheme used are particularly important. In addition, we observe that a large number of subsystem nodes allows more actors to be fired concurrently, but the communication overhead in passing control tokens to their destination nodes causes the overall execution time to increase substantially.
- The relationship between the mapping scheme employed and locality effect in a program are discussed. The mapping scheme employed has to exhibit a strong locality effect in order to allow efficient execution. We assess the average number of instructions in a cluster and the reduction in matching operations compared with fine grain control flow execution.
- Medium grain execution can benefit from a higher output bandwidth of a processor and finally, a simple superscalar processor with an issue rate of ten is sufficient to exploit the internal parallelism of a cluster. Although the technique does not exhaustively detect all possible errors, it detects nontrivial errors with a worst-case complexity quadratic to the system size. It can be automated and applied to systems with arbitrary loops and nondeterminism.

Conclusion: Alternatives to a Bad Talk

- **Practice, Practice, Practice!**
  - **Use casette tape recorder to listen, practice**
  - **Try videotaping**
  - **Seek feedback from friends**
- **Use phrases, not sentences**
  - **Notes separate from slides (don't read slide)**
- **Pick appropriate font, size (≥ 24 point to 32 point)**
- **Estimate talk length**
  - **≈ 2 minutes per slide**
  - **Use extras as backup slides (Question and Answer)**
- **Use color tastefully (graphs, emphasis)**
- **Don't cover slides**
  - **Use overlays or builds in powerpoint**
- **Go to room early to find out what is WRONG with setup**
  - **Beware: PC projection + dark rooms after meal!**