California Institute of Technology
Department of Computer Science
Computer Architecture

CS184a, Winter 2003 # Final Exercise: GMI Shortest Path Monday, March 3

Due: plan – Friday, March 7, 5:00PM
Due: report – Monday, March 17, 5:00PM

The five of you are the architectural team for Graph Machines, Inc., a company specializing in computers which provide supercomputer or better performance on graph problems. Your manager has just asked you to spend the next two weeks focusing on the "Shortest Path" benchmark and make recommendations on architectures which could run this in the least amount of time (primary goal is typical/expected runtime, but please report both expected and worst case).

Shortest Path: For concreteness, please focus on a single-source shortest path algorithm as would be suitable for Leiserson-Saxe retiming as detailed on Day 17 (*e.g.* Bellman-Ford). Each edge has an associated weight. At the end, each node should know the path cost to the designated source. If the graph has negative-weight cycles, you should identify that.

Can you sometimes terminate the update rounds early? Will it be beneficial to do so in some architectures?

Graph Representation: GMI has proposed a 256b edge-weighted, small graph node representation to the IEEE. Your solutions should work on this GMI graph node representation, which is as follows:

Field	Bits
note type	4b
node ID	16b
node data 1	12b
node data 2	32b
node data 3	32b
edge type 1	4b
edge weight 1	12b
edge link 1	16b
edge type 2	4b
edge weight 2	12b
edge link 2	16b
edge type 3	4b
edge weight 3	12b
edge link 3	16b
edge type 4	4b
edge weight 4	12b
edge link 4	16b
edge type 5	4b
edge weight 5	12b
edge link 5	16b

There is ongoing debate within GMI and IEEE on how to improve the graph node representation. There will likely be other standard graph node representations (non-edge weighted, larger graph sizes, etc.). You should work with this version. You may make recommendations for modification based on your experience with this task as an appendix in your report.

For this problem, node type is unused. Edge type of 0 indicates an input and edge type of 1 indicates an output. node data 1 will hold the delay of the node (you may assume it will be zero or one for your benchmark application). You should fill node data 2 with your shortest path length result. All the nodes will be the same type (0) for this application.

Note that you can have zero delay nodes. This allows you to represent a node with 4 inputs and 4 outputs using two of these 256b graph nodes. Similarly, you represent larger fanout by building a tree of graph nodes. You may assume an abstract graph has already been converted into an appropriate collection of these graph nodes before your algorithm begins.

You may treat fields as fixed point numbers as appropriate for your application.

Architecture: You want to find the architecture which minimizes the delay in computing the result. You should consider conventional, programmable architectures and customized architectures which work on the GMI graph node data type described above. The final GMI architecture will need to run other graph problems. You are identifying which is best for the shortest path problem and noting how close or far away other architectures are. You should, at least, consider:

1. Using a conventional processor
2. Using a collection of conventional processors (How many processors? How much memory should each processor have?)
3. Designing a temporal processor which is specialized for the Graph-Machine data representation described above (specialize the datapaths, provide any special instructions necessary to support this problem efficiently)
4. Using a collection of these specialized temporal processing nodes (How many processors? How much memory should each specialized processing node have?)
5. Using a conventional FPGA
   (a) by building a generic shortest path engine on top of the FPGA
   (b) by embedding a specific graph instance into an FPGA design
6. Designing a specialized, spatial graph processing machine
7. Other Hybrids between the spatial and temporal extremes?
8. ASIC which can only do shortest path computation (but must handle any graph) – doesn't solve GMI's general problem, but will be good to know how much better one can do.

You should brainstorm additional solutions to add to this list.

Division of Work: You should divide up the architectures to consider so that each engineer is responsible for detailed elaboration of a couple of architectural and/or implementation options. You should, however, make sure to compare notes and assumptions so that the results are directly comparable to each other.

Suggested meetings:

- Wed., March 5th – Divide work, decide on strategy, brainstorm solutions, generate questions
- Wed., March 12th – Present preliminary results to the team; critique each other's analysis

Details: You should think about the impact of wiring and wire distances when computing areas, delays, and cycle times. Assume a uniform 7λ wire pitch.

Component	Area in λ^2
Memory Cell	1,000
Programmable Switch (includes memory bit)	2,500
Buffered Switch (includes memory bit)	5,000
Register	4,000
FPGA Logic Block (4-LUT, FF, interconnect)	10^6
32b MIPS processor core (1GHz, 32×32 RF, 8Kb D-mem 8Kb I-mem, incl. tags)	10^9
Chip size	2×10^{11}

Component	**Delay**
gate delay	30 ps
$3 \times 10^4\lambda$ wire+buffer	60 ps
4-LUT delay	200 ps
cascade delay	50 ps
Chip crossing	1 ns

$$T_{mem}(Size) = 2 \times 60\text{ps} \times \frac{\sqrt{(A_{mem}(Size))}}{3 \times 10^4\lambda} + 50\text{ps} \times \log_2\left(\frac{Size}{64\text{Kb}}\right) + 500\text{ps} \qquad (1)$$

Plan: The plan should describe the work assignment (who will be doing what) and contain a list of questions you would like answered – *i.e.* what assumptions should you make about technology or problems.

Report: The report should contain your final recommendations. The executive summary should be written by the team. It should be 2–3 pages and include a table rounding up the areas and speeds of the architectures considered. There should be sections (2–3 pages each) detailing each architectural alternative (please keep the primary description to 2–3 pages; additional appendices may be included at the end to support the short section in the main document as necessary); each section should be written by the individual responsible for that alternative and should clearly indicate the author. All engineers should sign off on the entire report—so please make sure you all agree with the analysis and assumptions in each individual section. The report should be a single, integrated PDF document which you deliver electronically.