Deterministic Parallel Routing for FPGAs based on Galois Parallel Execution Model

Yehdhih Moctar*, Mirjana Stojilovic†, Philip Brisk*

† IBM Research, Austin  
‡ École Polytechnique Fédérale de Lausanne (EPFL)  
* University of California, Riverside

FPGA Routing Problem

Routing resource graph RRG = (V, E)
- Nodes (vertices) are pins and wires
- Edges are programmable connections between two vertices

Net characteristics:
- Signal i to route through RRG forms a net N_i
- Every net emanates from a single source and connects to a set of sinks
- Two paths emanating from the same source may overlap
- Two paths emanating from two different sources must be disjoint
- The paths from the source to all of the sinks form a routing tree RT(N_i)

RRGs of modern FPGAs are huge. Finding non-overlapping paths is very time-consuming.

Galois

Galois takes data-centric view of algorithms:
- An algorithm is the repeated application of an operator to an element of a graph. Vertices or edges on which a computation is centered are called active elements; the computation performed on active elements is called an activity; the activity results from the application of an operator. Set of nodes and edges that are read or written in performing the activity are the neighborhood of that activity.
- Deterministic Galois constructs interference graph (the vertices are tasks, the edges represent conflicts) and assigns unique IDs to the vertices.

Galois implementation of an algorithm:
- A C++ program that uses specialized foreach constructs to iterate over unordered or ordered data sets.
- All data structures need to implement an existing abstract data type (graph, tree, ... ) and all operations need to be atomic.
- Galois compiler transforms the code into a program that executes on top of the Galois runtime system, which orchestrates the parallel execution.


PathFinder (Negotiated Congestion) Algorithm

Inputs: RRG, nets to route
Outputs: disjoint routing trees

All-net router

while iter_cnt < MAX_ITER do {
    Signal router
    for all i = 1 to n_nets (rip up the existing RT(N_i))
    Maze expansion
    route net N_i
    update present congestion
    iter_cnt++
    update historical congestion
}

Deterministic Parallel Routing for FPGAs

Both Signal Router and Maze Expansion are irregular algorithms with parallelism that is best discovered at runtime.

Experimental Setup and Results

Experiments using VPR ported into Galois:
- VPR 5.0 and IWLS benchmarks, for direct comparison with previous work
- Channel width set to 1.4 x Wmin
- Max number of Pathfinder iterations (MAX_ITER) set to 100
- Server with Intel Xeon E5540 (2.53 GHz, 40 GB shared memory)
- 1, 2, 4, and 8 threads, repeated for both the non-deterministic and deterministic Galois schedulers

Issues that impede performance of the parallel Signal Router include:
- Multiple threads concurrently accessing lock-based shared data structures
- Rolling back partially routed threads when the Galois runtime system detects conflicts
- Load balancing among threads

The first two issues are the primary performance bottlenecks

Deterministic router: 1.84x and 3.67x for 8 threads  
Non-deterministic router: 2.67x and 5.46x for 8 threads