**Optimizing Relational Computing Performance on Heterogeneous Processors**

H. Wu, I. Saeed, J. Young, C. Kersey, and S. Yalamanchili
School of Electrical and Computer Engineering
Georgia Institute of Technology

**High Performance Relational Computing**

- Current applications process 1 to 50 TBs of data [1]
- Not a traditional domain for GPU acceleration
- Goal: **10X-100X** application speedup over multicore processors

**Multi-Predicate Join for SIMD Accelerators**

- Implementation of Leapfrog Triejoin (LFTJ) on GPUs
- A worst-case optimal multi-predicate join algorithm
- CPU-version - T. L. Veldhuizen, ICDT 2014

**Key idea**: Leap over large segments to seek a specific (e.g., join) value

Primitives `next()` and `seek()`

- **Benefits**
  - Smaller memory footprint for temporary results
  - No data reconstruction, e.g., sorting or hash table construction

**Near Memory Data Intensive Computing**

- Parametric C++ processor synthesis environment
- HARP family of data parallel processors
- RISC processor core
- Assembler/emulator tools
- OpenCL compiler (in progress)

**References**