CLA 2011

Formal Concept Analysis on Graphics Hardware

W. B. Langdon, Shin Yoo and M. Harman
Crest, Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK (Email: w.langdon@cs.ucl.ac.uk)

Non-Recursive Beam Search on GPU for Formal Concept Analysis

1 Introduction

Graphics hardware (GPUs) offers cheap parallel computation. Both now and with future improvements. Close-by-one can rapidly extract formal concepts from large sparse datasests. Its recursive depth first search is not suitable for GPU. Instead we recast it as beam search and demonstrate parallel beam search on GPU.

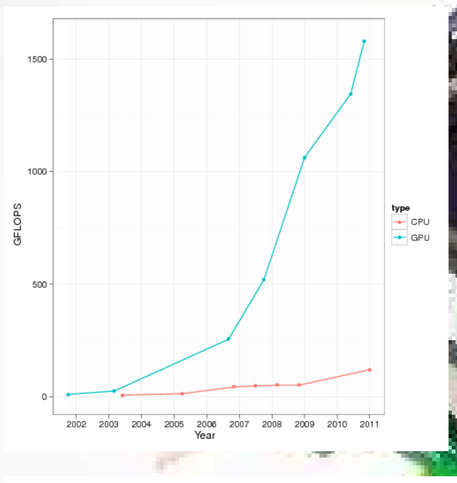

Figure 1: Comparison of increase in speed of graphics cards and CPU

2 GPU Hardware

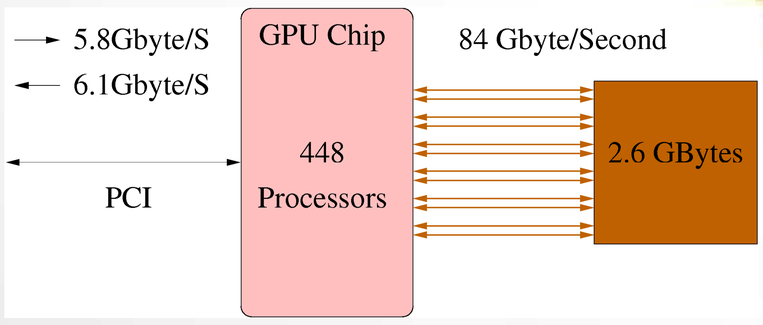

Figure 2: Links from GPU chip to host computer via PCIe bus and to memory on the GPU board.

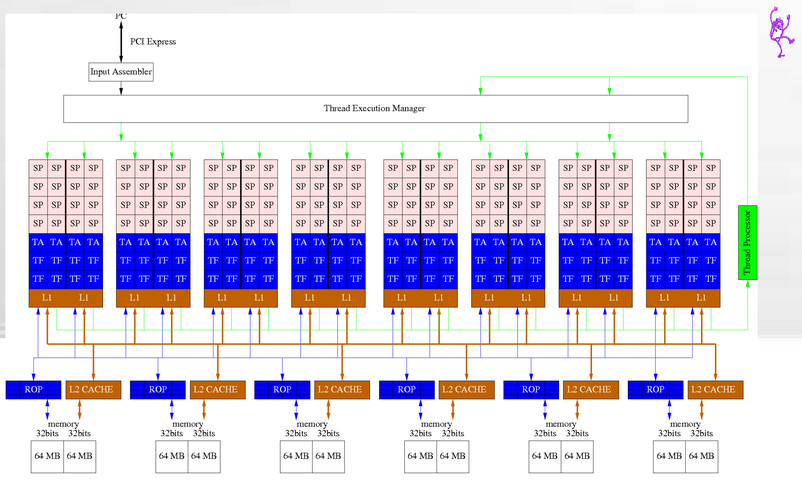

Figure 3: nVidia 8800 GTX (128 processors)

3 FCA Beam Search

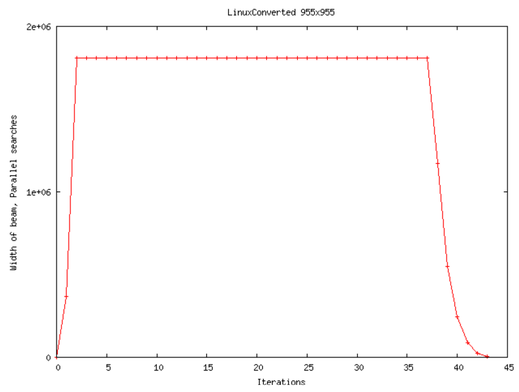

Figure 4: Number of closures processed in parallel increases rapidly with tree depth. Limit of 1.8 million means 45 iterations are needed for large tree of depth 24.

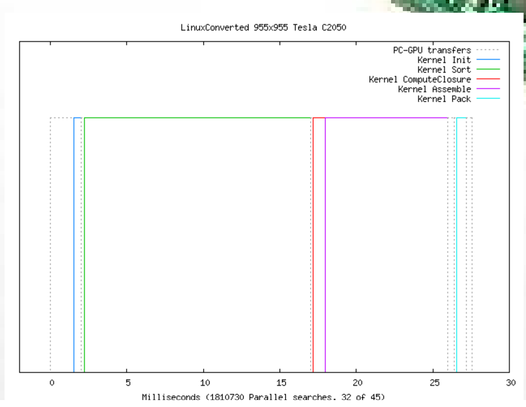

Figure 5: Elapse time for GPU kernels is dominated by Sort and Assemble.

4 Results

Dataset	Size	Density	Concepts	FCbO	Python	295 GTX	C2050
krajca	5×7	54%	16	0.00	0.11	0.01	0.01
wiki	10×5	44%	14	0.00	0.03	0.00	0.00
random	10×10	20%	16	0.00	0.04	0.00	0.00
random	100×100	2%	137	0.00	0.40	0.02	0.01
random	200×200	2%	420	0.00	4.33	0.00	0.01
random	500×500	2%	2861	0.01	162.60	0.02	0.02
bison	37×37	24%	692	0.00	0.32	0.00	0.01
compiler	33×33	6%	24	0.00	0.05	0.00	0.00
dot	42×42	28%	1302	0.00	0.71	0.00	0.01
grappa	86×86	7%	850	0.00	2.54	0.01	0.01
incl	172×172	2%	238	0.00	1.84	0.00	0.01
ispell	24×24	34%	432	0.00	0.15	0.01	0.01
linuxConverted	955×955	2%	141072	0.73	15:42:51	1.79	0.93
mtunis	20×20	29%	110	0.00	0.05	0.00	0.01
rcs	29×29	37%	1074	0.00	0.46	0.01	0.02
swing	413×413	2%	3654	0.01	208.71	0.03	0.02

Table 1: Performance on FCA benchmarks, random module dependency and software engineering datasets.

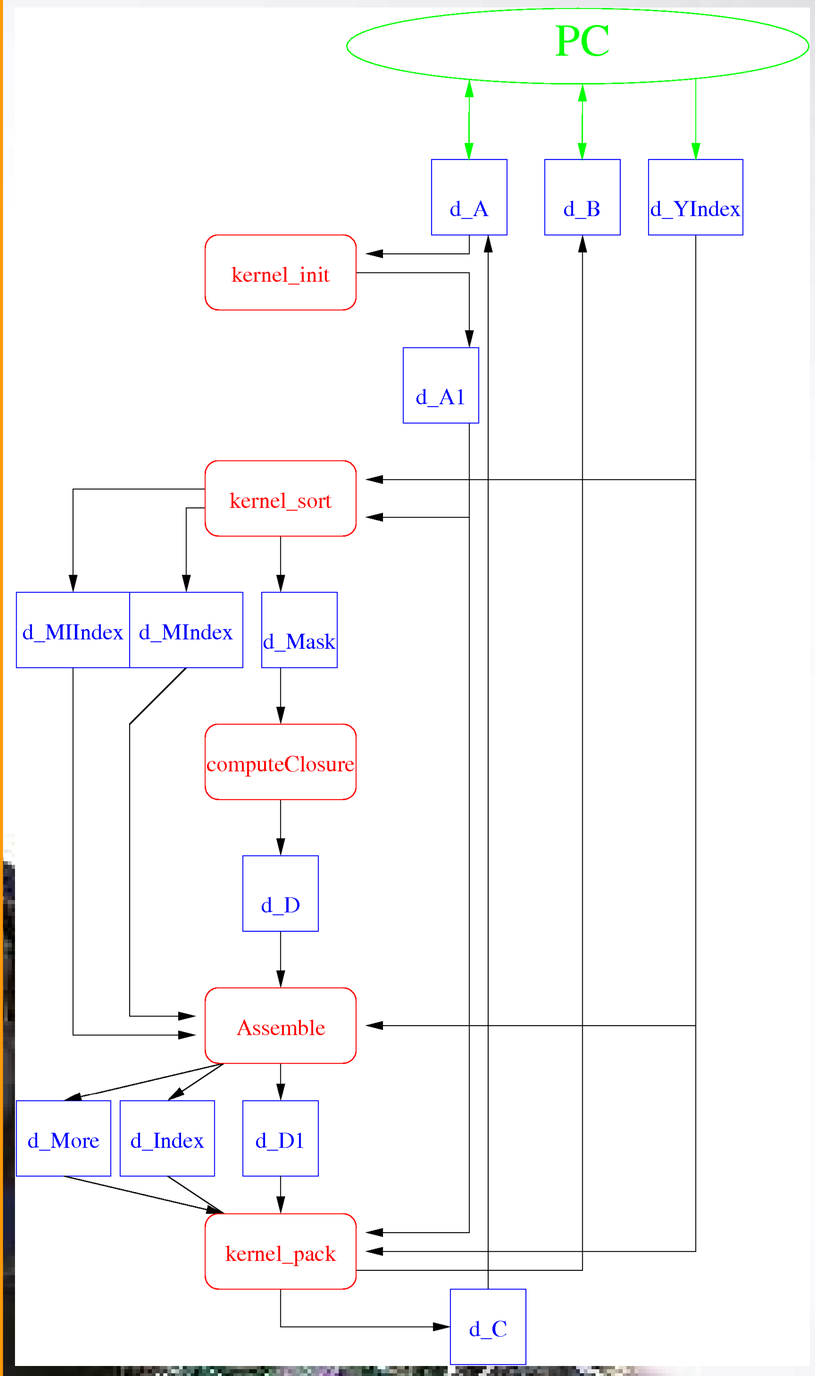

Figure 6: Main FCA global data and FCA program and connections to host PC kernels and host PC.

5 Future Work

In real world examples (e.g. clustering software engineering modules from the Linux operating system) raw CbO repeats many identical sub-calculations. The GPU code (kernel Sort) detects and removes such repeats. Would it be worth removing repeated calculations in serial FCA algorithms?

Is there some "missing step", normalisation, or pre-processing which should be done?

More details in technical report RN/11/18

6 Summary

- Beam search is a viable alternative to close-by-one (CbO)'s depth first search.
- Massively parallel beam search has been implemented for graphics hardware.
- The current CUDA implementation may suffer from FCA's low computation to data ratio. (CbO's arithmetic intensity is about 1).

Acknowledgments: C2050s given by nVidia.