# MAXIMUM CARDINALITY MATCHING FOR BIPARTITE GRAPHS

*Samuel Ueltschi, Conradin Roffler, Thomas Meier, Isabelle Roesch*

Department of Computer Science
ETH Zürich
Zürich, Switzerland

## ABSTRACT

Maximum cardinality matching, focus on existing algorithms and optimize the parallel versions in a highly multithreaded environment. Focus on Pothen-Fan, reason about performance.

## 1. INTRODUCTION

Bla bla

**Motivation.** Graph matching has several applications in computer science, for example the marriage problem or computing the block triangular form (BTF) of a sparse matrix [1]. As data gets bigger, the performance of algorithms that solve these problems gets more and more important.

**Related work.** [2] [3]

## 2. BACKGROUND: ALGORITHMS FOR MAXIMUM MATCHING IN BIPARTITE GRAPHS

**Maximum Matching.** What is the problem?

**State of the Art.** Best algorithms to solve this at the moment (parallel and sequential), as well as O(..) considerations

## 3. ALGORITHMS AND OPTIMIZATIONS

Focus on Pothen-Fan [2] but also report Tree Grafting [3] for completeness

### 3.1. Pothen-Fan

**Parallel Pothen-Fan.** Pseudocode for parallel ppf

**PRAM Analysis.** Show DAG, worst case O(n), best case O(1) with n processors (n nodes), but real world graph are rather O(1)

**Roofline Model.** number of operations, number of moves, what if whole graph fits into cache, etc

**Optimizations.** Test and Test and Set, Locality, Use only half of the visited array, set only half of the matching vector while setting the rest last, etc

### 3.2. Tree Grafting

Paper [3] claims that PPF is not well suited for many thin cores.

**Parallel Tree Grafting.** Pseudocode for parallel tg

**Optimizations.** Same optimizations as PPF version 3 for the augmenting paths. Non-blocking queue as a data structure.

## 4. EXPERIMENTAL RESULTS

**Experimental setup.**

Xeon Phi (5110P), GCC, -O3, 60 simplified Intel CPU cores running at 1056 MHz and supports 4 threads per core, resulting in a total of 240 threads. Each core has a 32kb L1 data cache, a 32kb L1 instruction cache and a private 512 kb L2 unified cache. [4]

**Test Data.**
List our graphs
**Benchmarks.**
Sequential Pothen-Fan
**Verification.**
Boost Edmonds
**Results.**

## 5. CONCLUSIONS

Super linear speedup because of caching effects
PPF scales well on Xeon Phi
Tree Grafting results from the paper could not be reproduced.

## 6. REFERENCES

[1] Alex Pothen and Chin-Ju Fan, "Computing the block triangular form of a sparse matrix," *ACM Trans. Math. Softw.*, vol. 16, no. 4, pp. 303–324, Dec. 1990.

[2] A. Azad, M. Halappanavar, S. Rajamanickam, E. G. Boman, A. Khan, and A. Pothen, "Multithreaded algorithms for maximum matching in bipartite graphs," in

*2012 IEEE 26th International Parallel and Distributed Processing Symposium*, May 2012, pp. 860–872.

[3] Ariful Azad, Aydin Bulu, and Alex Pothen, "A parallel tree grafting algorithm for maximum cardinality matching in bipartite graphs," in *Proceedings of the 2015 IEEE International Parallel and Distributed Processing Symposium*, Washington, DC, USA, 2015, IPDPS '15, pp. 1075–1084, IEEE Computer Society.

[4] S. Ramos and T. Hoefler, "Modeling Communication in Cache-Coherent SMP Systems - A Case-Study with Xeon Phi," in *Proceedings of the 22nd international symposium on High-performance parallel and distributed computing*. Jun. 2013, pp. 97–108, ACM.