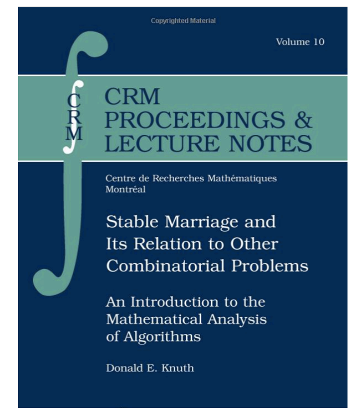
On Stable Marriages and Greedy Matchings

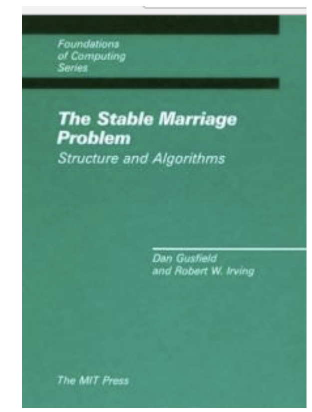
Fredrik Manne
University of Bergen, Norway

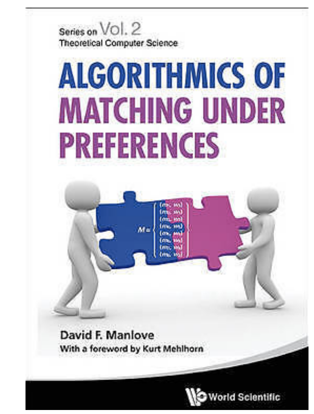
Md. Naim, Håkon Lerring, Mahantesh Halappanavar

Background

The Stable Marriage (SM) problem has a long and rigorous history.







Greedy Matchings (GM) have applications in CSC applications.

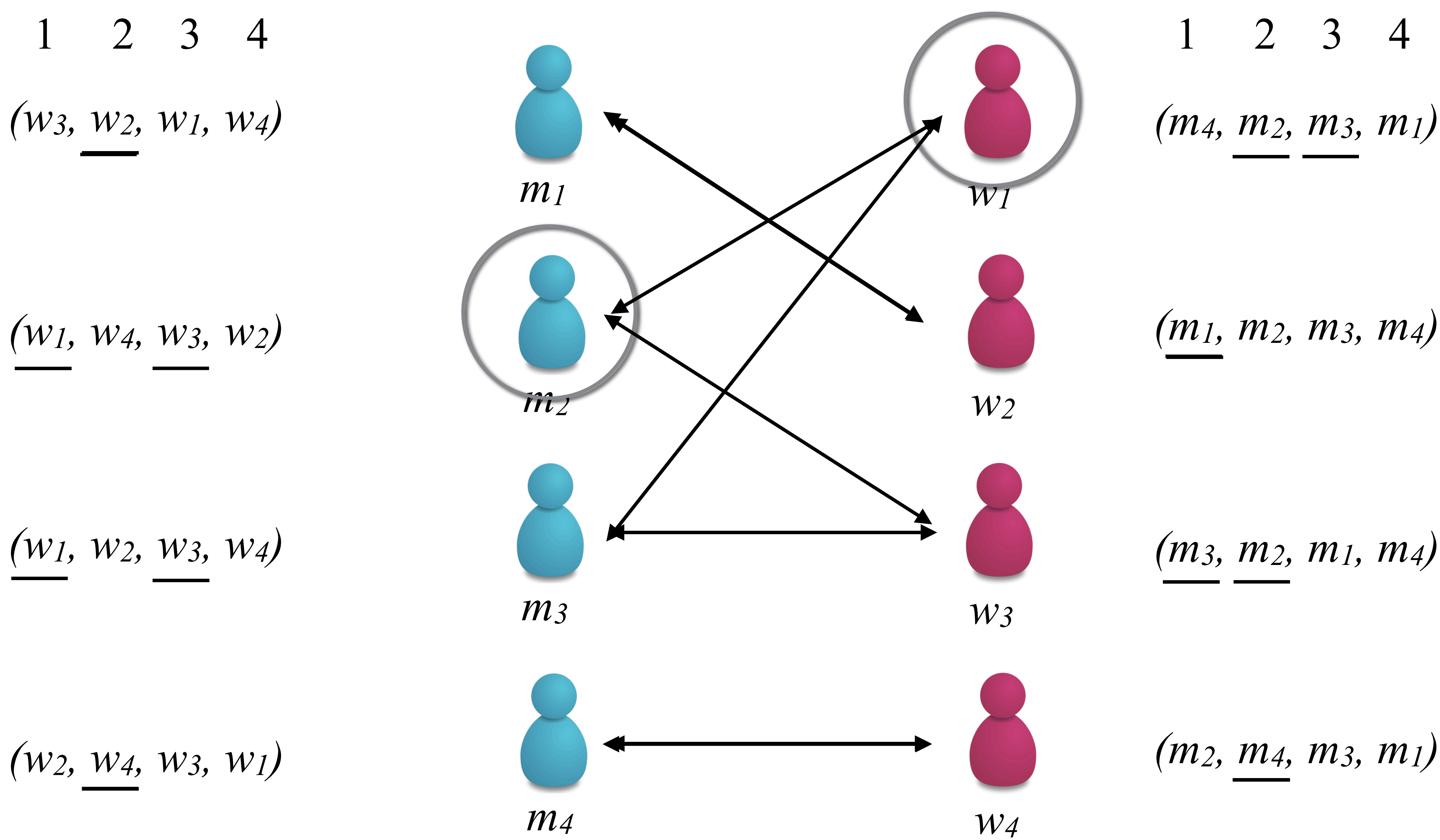
Objective:

- Formalize the connections between the Stable Marriage problem and computing Greedy Matchings.

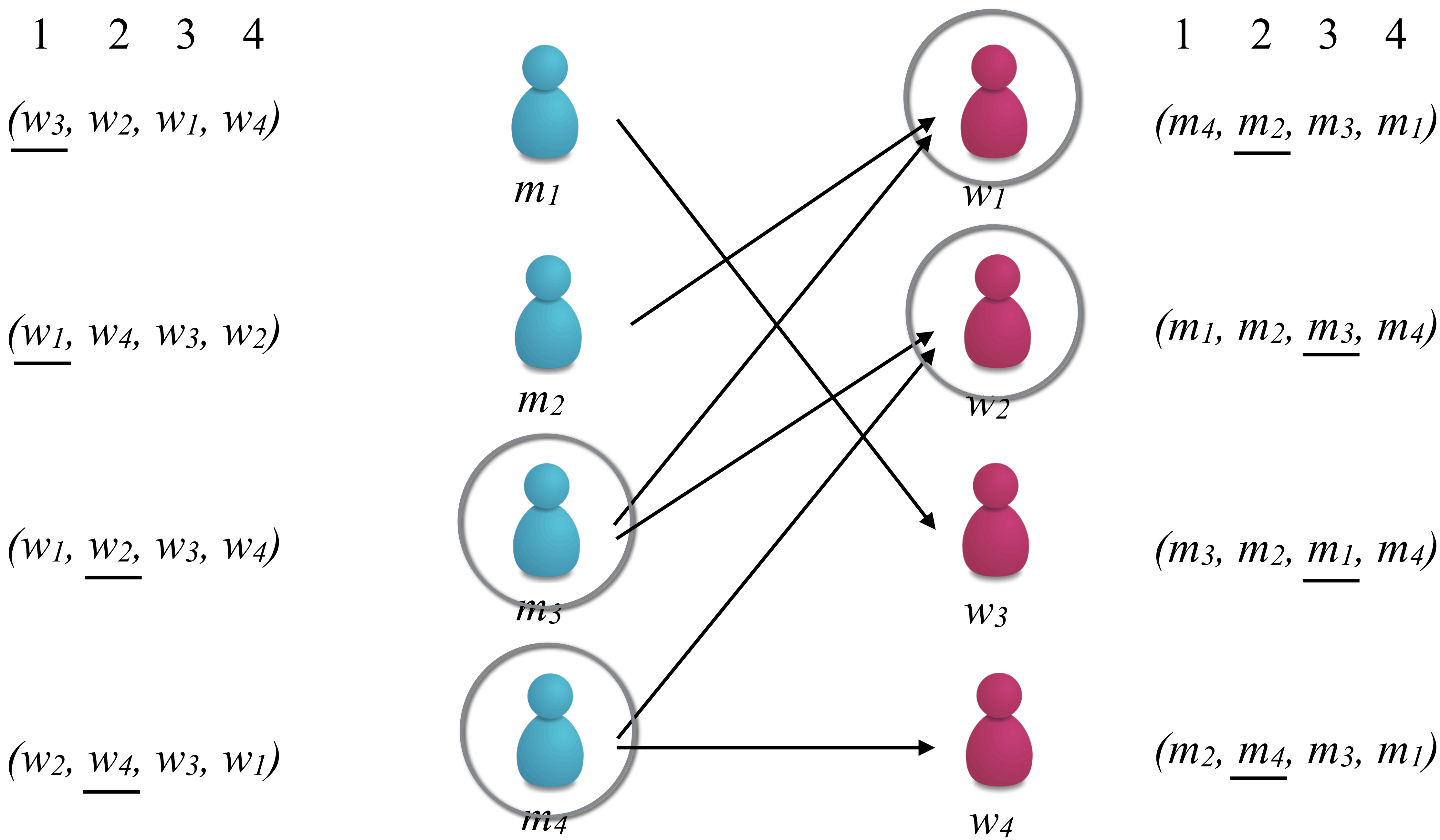
Consequences:

- Many ”new” algorithms for computing GM are variants of algorithms for SM.
- Parallel algorithms for computing GM can also be applied to SM.

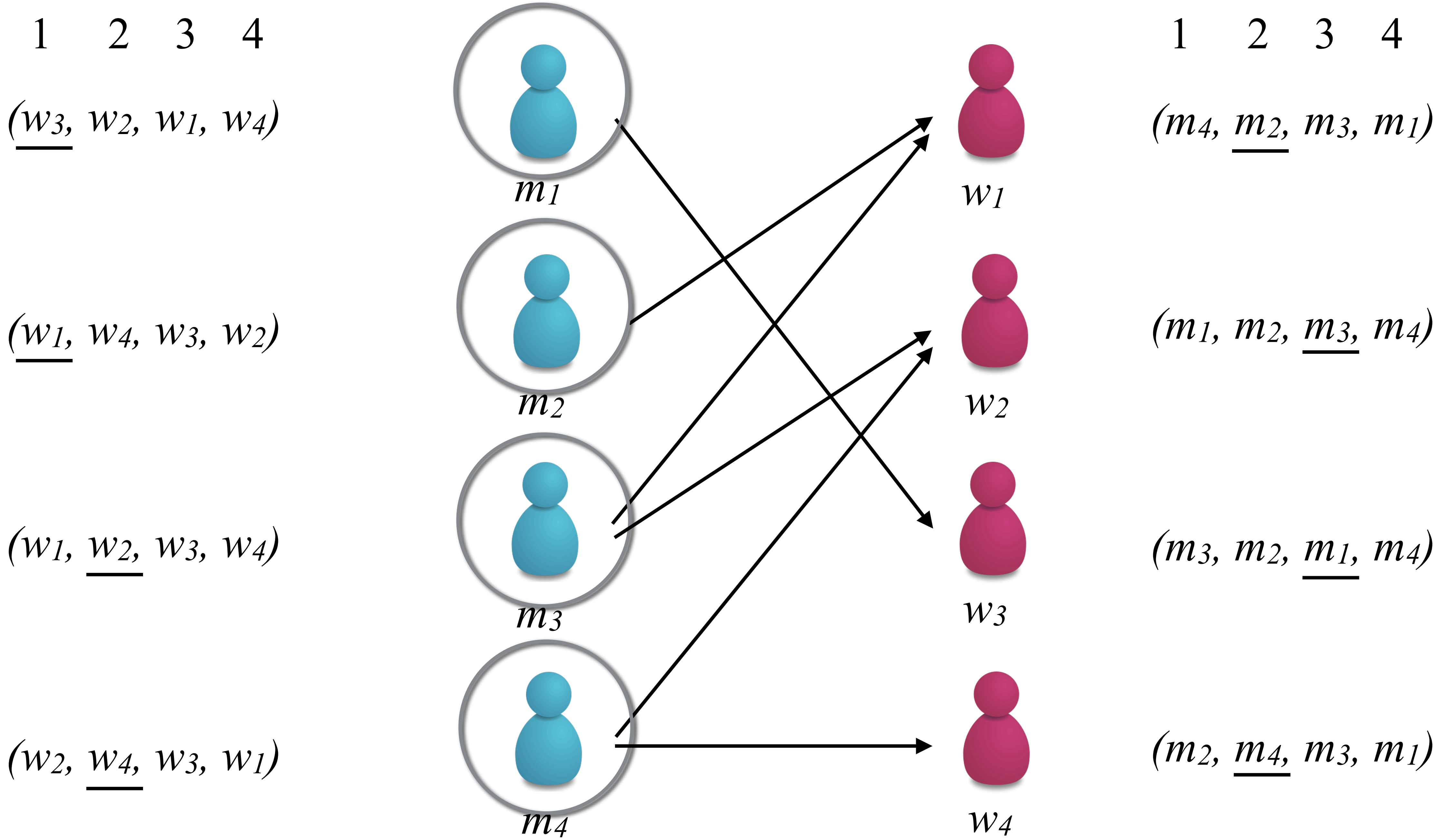
The Stable Marriage Problem



The Gale-Shapley Algorithm

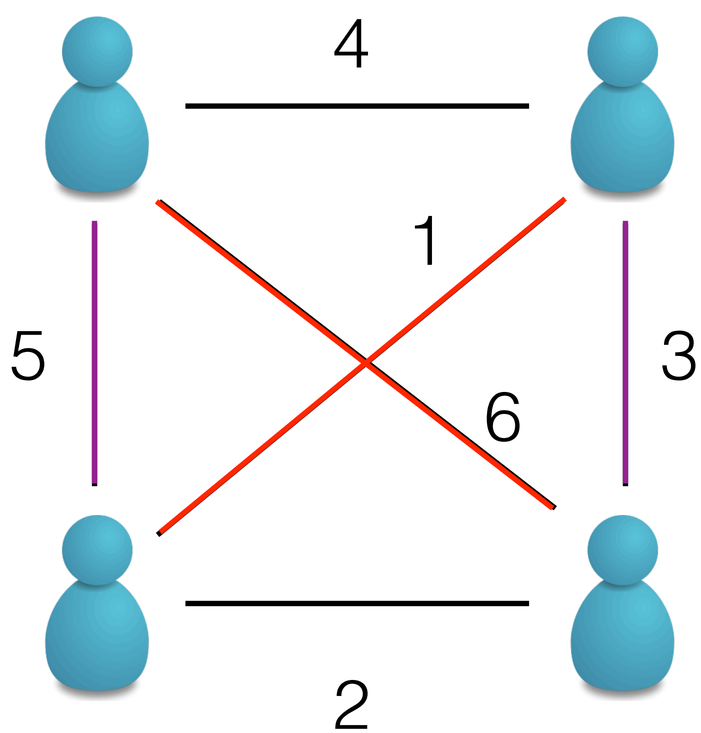


The McVitie-Wilson Algorithm



Implementation: GS uses a queue while MW uses a stack for the remaining men

Greedy Matching



M = Ø

While there are edges remaining

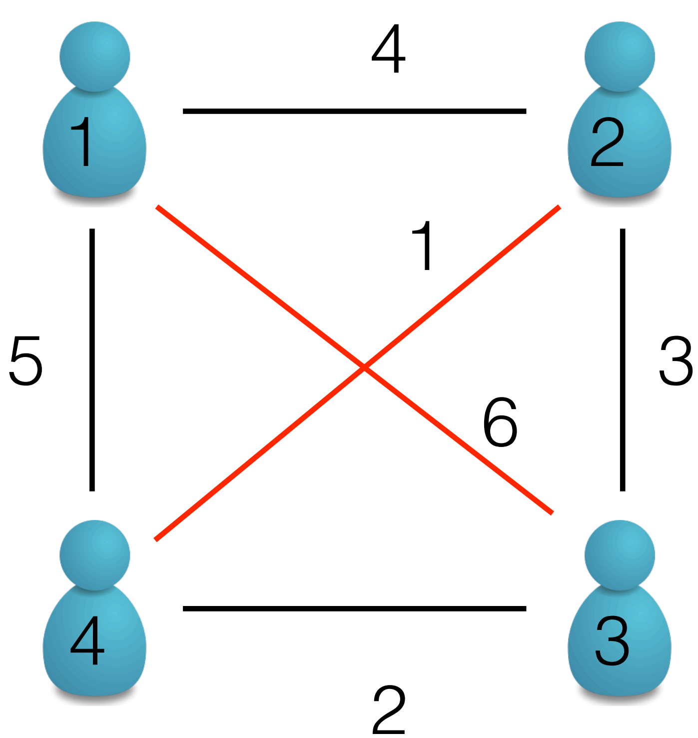
e = heaviest edge

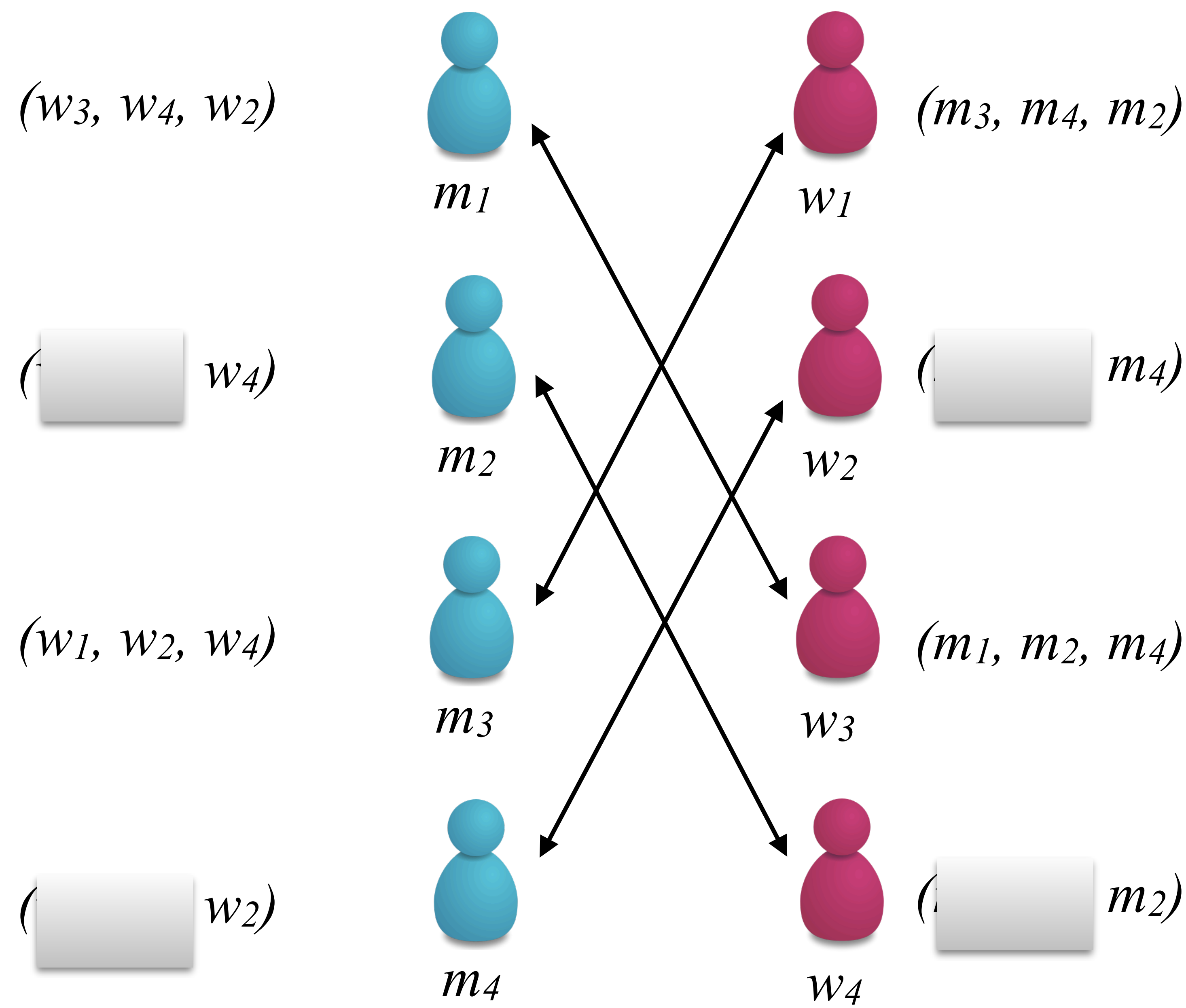
M = M U {e}

remove edges incident on e

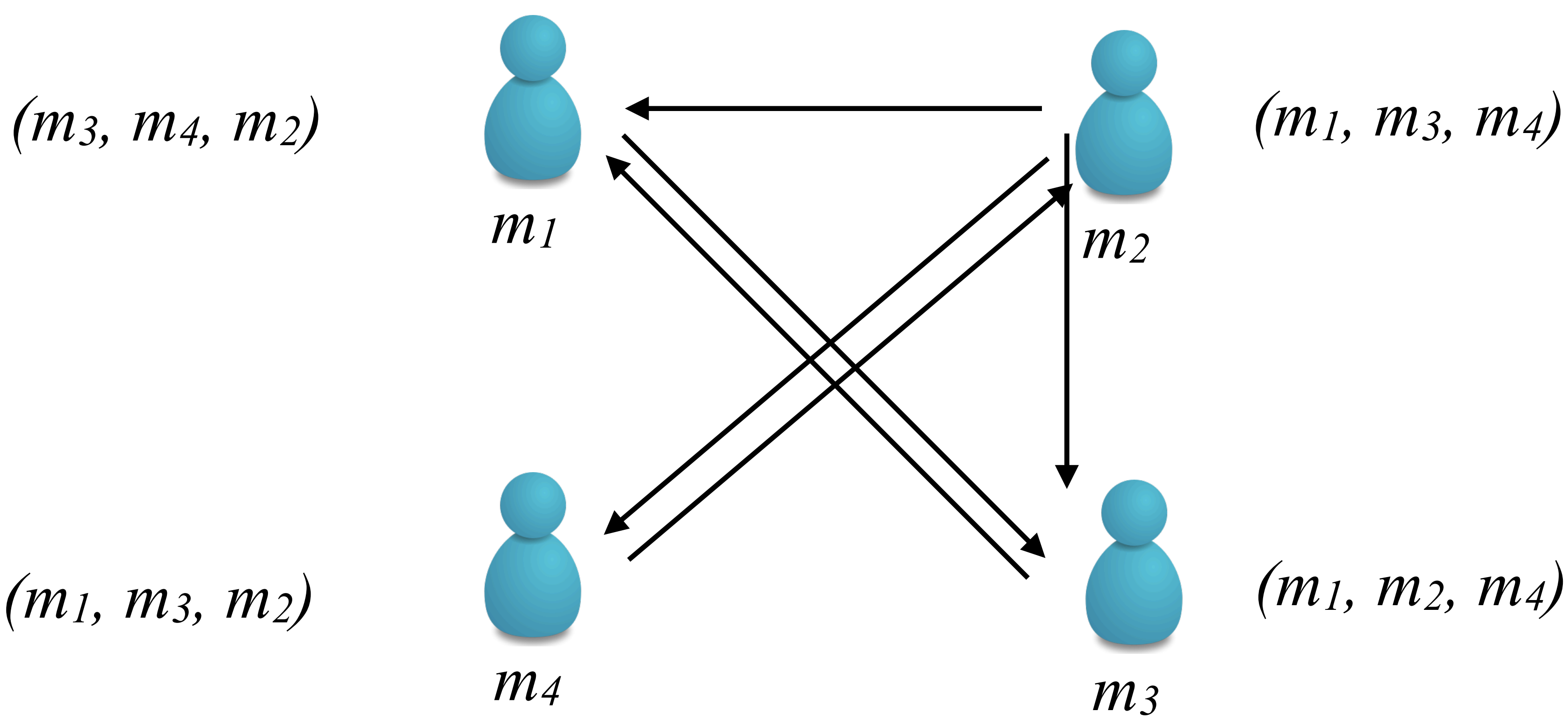
w(greedy) ≥ 0.5 w(optimal)

Computing a Greedy Matching using a Stable Marriage algorithm





Using McVitie-Wilson directly

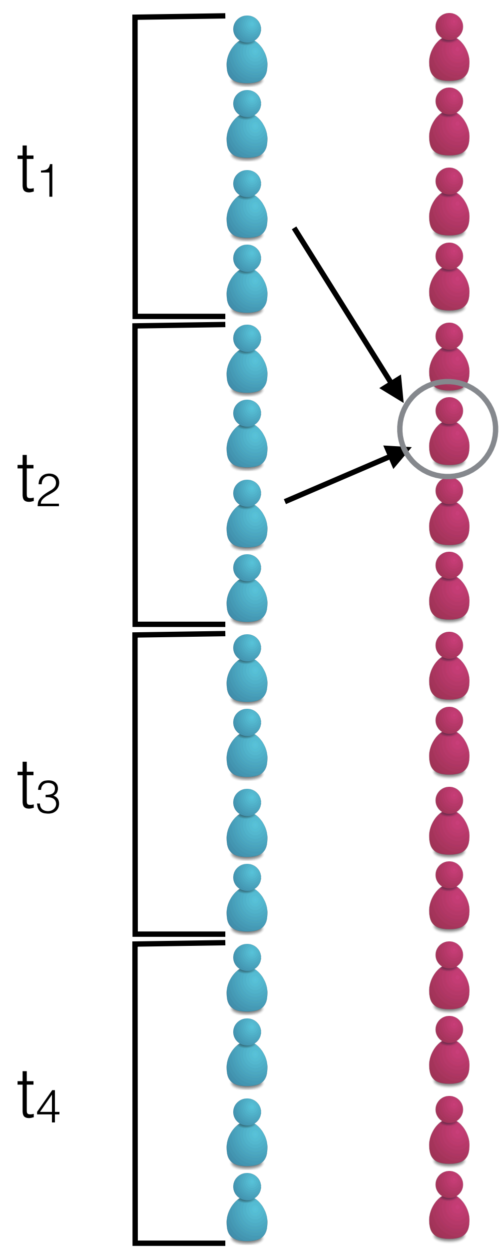


This is exactly the suitor-algorithm [Manne & Halappanavar 14]

which builds on:

- The pointer algorithm [Manne & Bisseling 08]
- The Preis algorithm [Preis 99]

Going parallel



Threads run either the Gayle-Shapley or the McVitie-Wilson algorithm

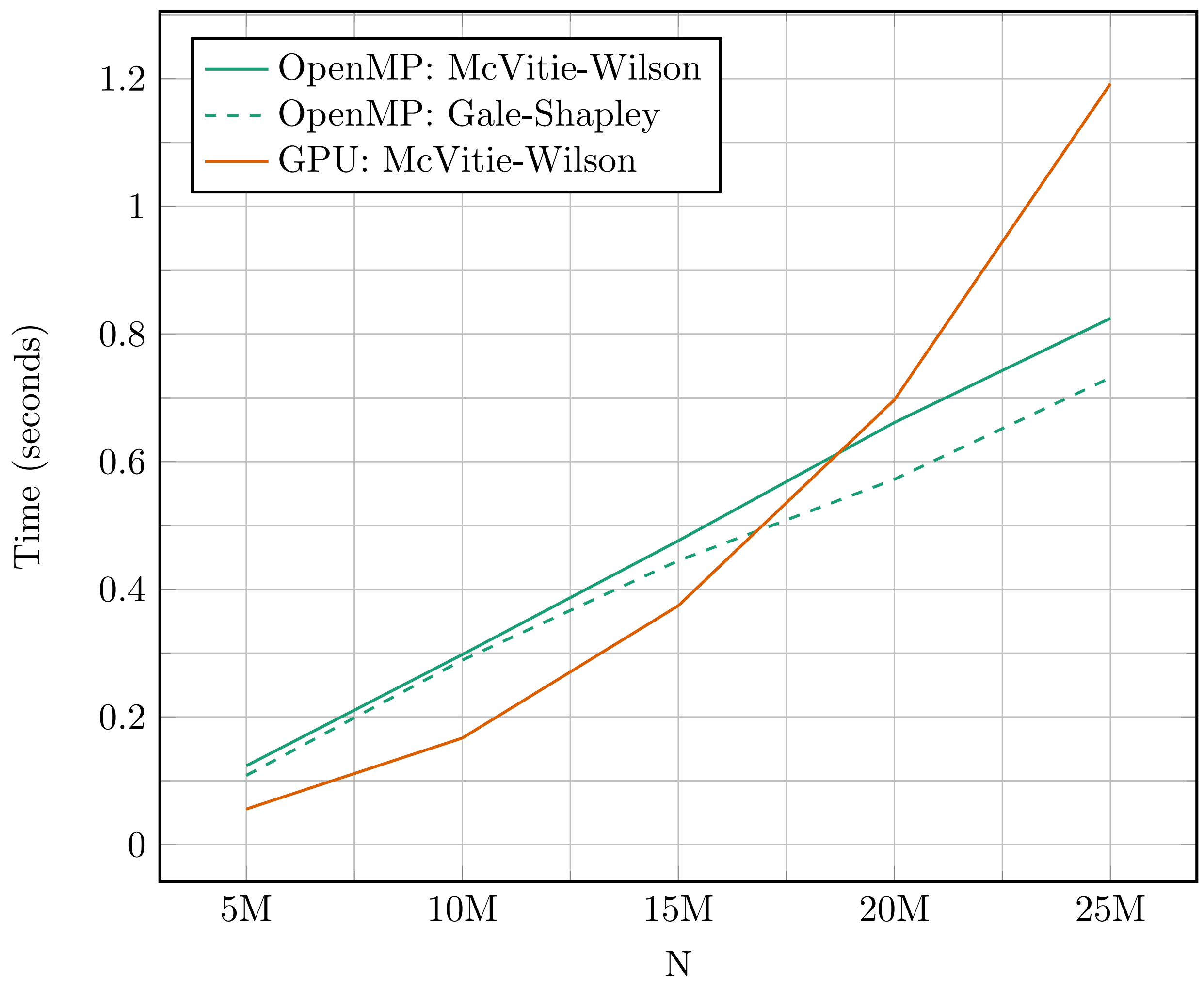
Use compare-and-swap to protect “women”

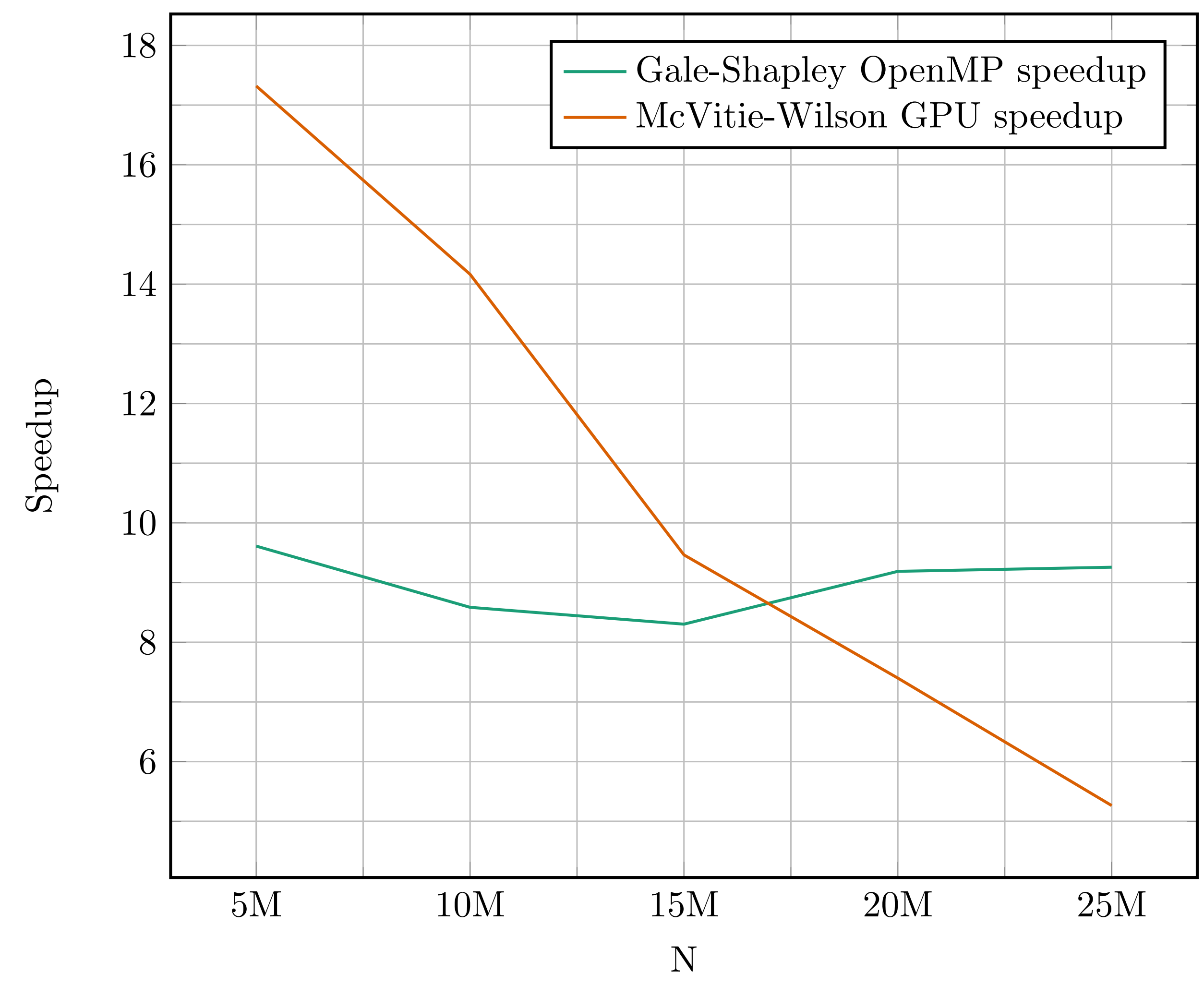
Implementations using both OpenMP and GPU

“Easy” problems

- Each man selects between *log n* and *2 log n* women and ranks randomly.
- Women rank men who rank them (randomly).
- Expected total work *n log n*

OpenMP: Using 36 threads on two Intel Xeon E5-2699 processors
GPU: Tesla K40m with 2880 cores

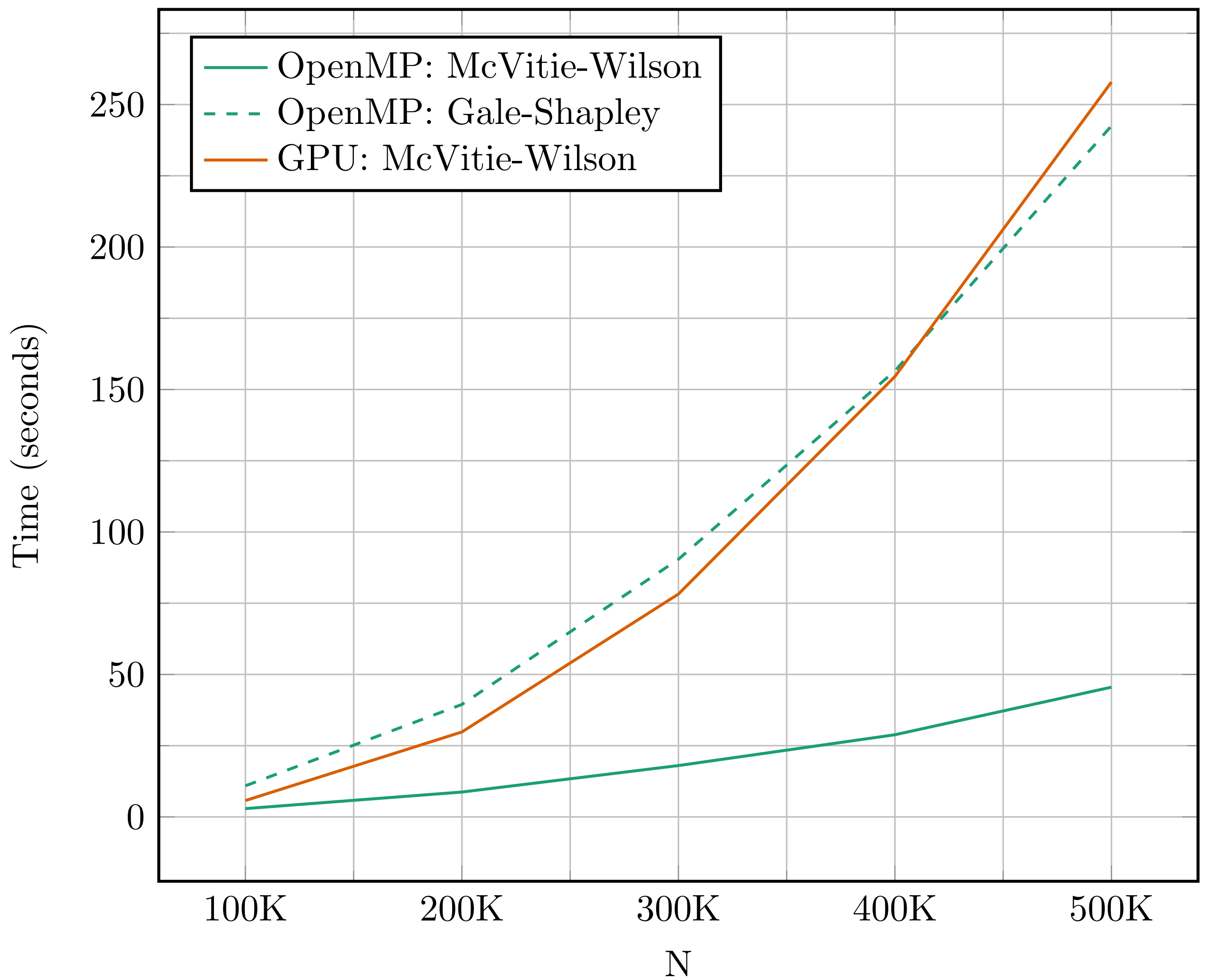


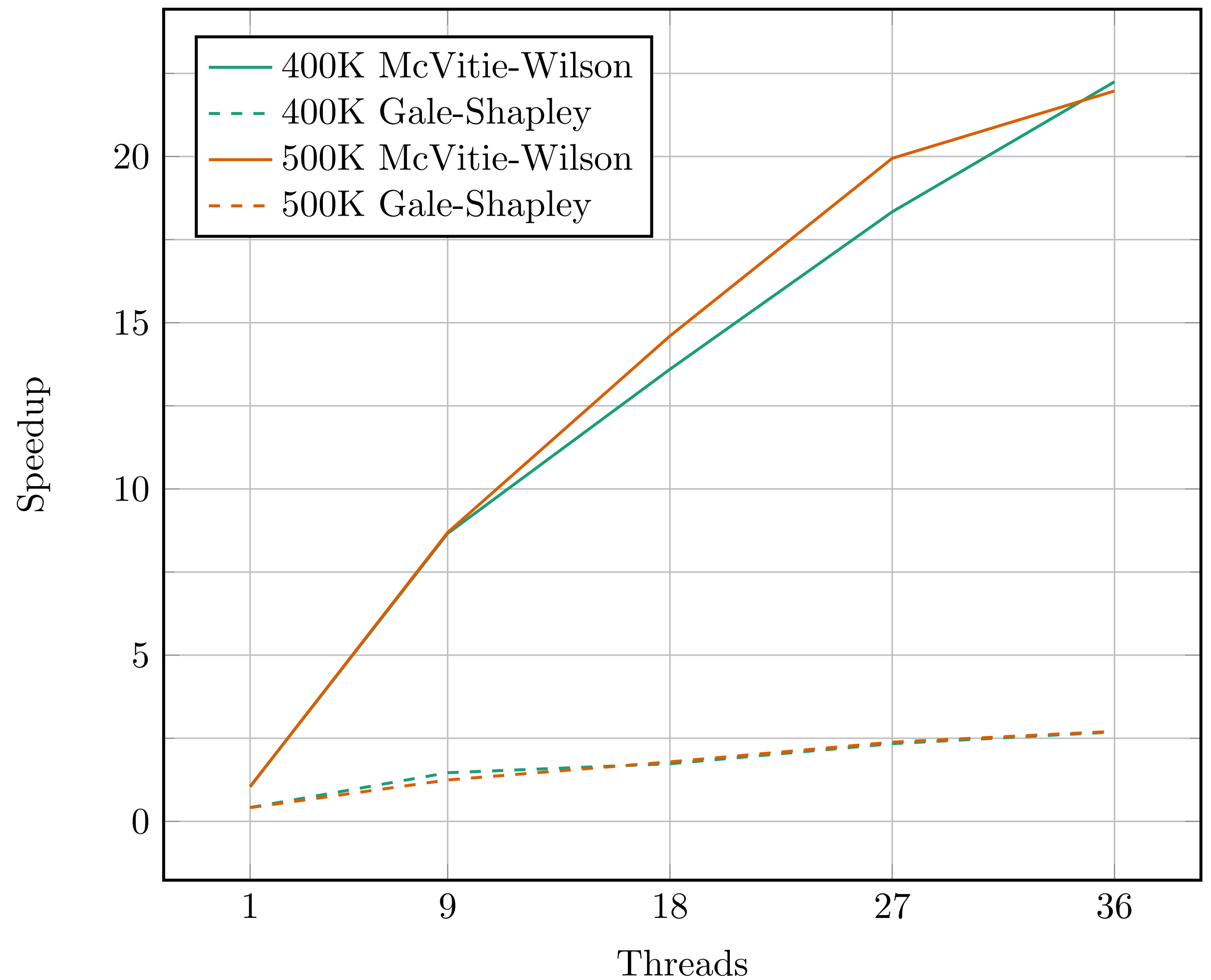


“Hard” problems

- Each man uses the same total ranking of all the women
- Each woman uses the same total ranking of all the men.

- Expect high contention for the same “women”
- Total work will be (n+1)n/2





Concluding Remarks

- Recent greedy b-matching algorithms also follows directly from algorithms for the many-to-many stable assignment problem.

- Major open question [Manlove 13]: Is Stable Marriage in NC?
  - Maybe not so relevant…

- Stable Marriage assumes sorted priority lists, whereas Greedy Matching makes no such assumption.

  - Preis solved Greedy Matching in O(m) time.
  - Can Stable Matching with unsorted weighted priority lists also be solved in O(m) time?

Gale-Shapley implementation

```
Place all vertices in queue Q
while Q ≠ Ø
    u = Q.first()
    p = nextCandidate(u)
    while r_p(u) > r_p(suitor(p))
        p = nextCandidate(u)
    if suitor(p) ≠ null
      Q.add(suitor(p))
    suitor(p) = u
```