'MATH+ECON+CODE' MASTERCLASS ON COMPETITIVE EQUILIBRIUM: WALRASIAN EQUILIBRIUM WITH SUBSTITUTES

Alfred Galichon (NYU)

Spring 2018
Day 2, May 22 2018: lattices and order
Block 6. Code: matching algorithms without transfers

Learning objectives: block 6

- Parallelizing the coordinate update algorithm
- Algorithms for matching without transfers: Gale-Shapley vs Adachi

- Reminder to upload only two files: one pdf file, one file for code.

- **Q1**. Take the coordinate update algorithm seen yesterday (Jacobi version), and run it in parallel on your laptop's cores. Optional: run in on NYU's (or your university's) HPC cluster. For each number of core used, report the time taken.

- Consider the same grid as yesterday:

  $$\mathcal{X} = \mathcal{Y} = \{1/10, 2/10, 3/10, ..., 9/10, 1\} \times \{1/10, 2/10, 3/10, ..., 9/10, 1\}$$

  and as before assume that:
  - there is a mass one $n_x = 1$ of passengers at each $x \in \mathcal{X}$.
  - if a car $y \in \mathcal{Y}$ has coordinates $\left(y^1, y^2\right)$ there is a mass $m_y = y^1 \times y^2$ of passengers at y.
- Assume that the preferences of both sides of the market are now given by $\alpha_{xy} = -1\left\{d\left(x, y\right) \geq 0.5\right\}$ and $\gamma_{xy} = -d\left(x, y\right)$, where $d\left(x, z\right)$ is the Euclidian distance between x and z, given by $d\left(x, z\right) = \sqrt{\left(x^1 - z^1\right)^2 + \left(x^2 - z^2\right)^2}$.
- The reservation utilities are set to -2, so everyone prefers to be matched with anyone rather than remaining unassigned.

- **Q2**. Code Gale and Shapley's algorithm and Adachi's algorithm, and run them on the setting above. What do you notice?
- In your answer, you should report for each algorithm:

(i) the total welfare on the driver's side
(ii) the total welfare on the passenger
(iii) the number of iteration
(iv) the time taken on your machine

- In order to break ties, we will create two matrices of tie-breakers

```
set.seed(0)
tiebreakersalpha = 0.01 * matrix(runif(10000),100,100)
tiebreakersgamma = 0.01 * matrix(runif(10000),100,100)
```

- You should verify that the seed is right by finding exactly the same numerical values as below:

```
> tiebreakersalpha[50,50]
[1] 0.005664511
> tiebreakersgamma[50,50]
[1] 0.009242289
```

- **Q3**. Add the tiebreakers matrices to the preference matrices α and γ, and repeat Q3. Comment.