

Employers - Workers Matching Software

Assume we have a set of employers $E = \{E1, E2, \dots, En\}$ and a set of workers $W = \{W1, W2, \dots, Wm\}$. A matching software between employers and workers is required.

Each worker is interested in a set of jobs, certain work dates, and should have jobs as close as possible to her address. These jobs should be as close as possible to each other to minimize the traveling time between these jobs.

E.g., worker W1 is interested in gardening, babysitting and would like to work on Monday from 2:00 pm - 5:00 pm and Thursday from 1:00 pm - 7:00 pm, and her address is Add1. Then we can define W1 as:

$W1 = \{\text{Gardening, Babysitting, Monday 2:00 pm - 5:00 pm, Thursday from 1:00 - 7:00 pm, Add1}\}$

Another worker could be defined as:

$W2 = \{\text{Driving, Cooking, Cleaning, Friday 8:am - 2:00 pm, Tuesday 2:00 pm - 6:00 pm, Add2}\}$ and so on.

Every employer could require one job or more and expects these job(s) to require a time interval T_i to be completed, on a certain date and has a specific address.

E.g. employers E1, E2 and E3 could be:

$E1 = \{\text{Cooking, Babysitting, } T1 = 4 \text{ hours, Monday: 3:00pm - 7:00 pm, Address}\}$

$E2 = \{\text{Cleaning, } T2 = 2 \text{ hours, Tuesday: 2:00 pm - 4:00 pm, Address}\}$

$E3 = \{\text{Driving, Friday at 5:00 pm, Address1, Address2}\}$

The ride will be from Address1 to Address2.

When a worker W_i is matched with employers, the software should match them based on jobs required, dates and find employers that closer to W_i and to each other to minimize the traveling time.

Note1

If the task required is driving, then the software will not have the time interval T and should be able to calculate the time required based on address1 and address2. Basically it will be whatever algorithm Uber uses to calculate the time and the fare.

Note 2

Please take into consideration that these jobs will be booked ahead of time up to a month. Therefore the software should be dynamic, meaning it will keep optimizing every worker's schedule every time an employer books a service.

So, every time an employer requires a job, the program will repeat the matching process between all workers and employers to reach the optimal matching. This process will stop only one day before the actual workday. A worker will know her schedule only one day before her workday.

Example

Say there are 10 workers in Toronto.

1- If an employer is within, say 20 miles from any of these workers, and books a service, then the employer should get a confirmation message on his phone, or an email based on their choice of communication method. If the employer is more than 20 miles from any worker, then the employer should get a message that the service

is not available at this time.

2- Every time an employer books a service, the software will optimize the matching between employers and workers based on the following criteria:

1- The traveling time for the worker starting from her address and back to her address is minimal

2- The worker gets as many jobs that fit within her working times as possible.

You can always minimize the traveling time by having less jobs but this would be a loss for the worker. The ideal situation is maximum number of work hours within a minimum time of traveling.

3- The software will stop optimizing the matching 24 hours before each working day and the worker will get her workday schedule on her phone or email according to their communication method choice.