# Part 2 : Simulation Report (PMO)

Daniel Haimanot

June 14, 2016

# 1 Estimating Company Income

The simulation implementation will answer parts A, and B of question 2 together. Since the answer for part A, is one of the specifications of the strategy we implement in part B. Further we assume that the player's skill set is uniformly distributed.

Firstly we notice that the exponential distribution is memory less. So the players who are already on the platform don't get "tired" of waiting. The rate of them leaving doesn't increase with time. This means if it was optimal for them to wait a millisecond ago then it is optimal for them to wait now. For that reason the system only starts to match players at "events" of change. These moments are when another players signs up, or an unmatched player leaves the platform.

When considering a pair of players $(i, j)$ for a possible match, the *"exercise value"* of the match is compared against the *"continuation value"*. The exercise value is the expected revenue from the match and is then defined as:

$$\begin{aligned} \left(\frac{2}{60}\right)\mathbf{E}\big[duration\,of\,the\,game\,in\,minutes\big] = \\ \left(\frac{2}{60}\right)\mathbf{E}\big[gamma\,dist\,, \kappa = 11 - |S_i - S_j|\,and\,rate\,\beta = 1\big] = \\ \left(\frac{2}{60}\right)\frac{\kappa}{\beta} = \frac{(11 - |S_i - S_j|)}{30} \end{aligned} \tag{1}$$

In contrast the continuation value is the value generated by players $i$ and $j$ when they are not matched between each other. Which results in the following matching conditions:

**If exercise value $\geq$ continuation value** **it is optimal to match $i$ and $j$**
**If exercise value $<$ continuation value** **it is optimal not to match $i$ and $j$**

Due to this for players $i, j$ with equal skill set levels $(S_i = S_j)$ the continuation value is always less or equal than the exercise value. So it is always optimal to match all the players for a given complexity levels if there is even number of them and match most of them if there is odd number of them. This means that shortly after every "event", there is 0 or 1 player remaining for each complexity level. These players may or may not be matched further based on the comparison of continuation values to exercise values for all pairs.

This optimization problem is complex since there are a large number of possible configurations. This makes the problem suitable for reinforcement learning techniques. These techniques would calculate the exercise value and continuation value for every pair $(i, j)$ at every state of the system, and so extract the optimal matching strategy as a function of the state of the system. However there are many possible states $(2^{10})$ to compare, because the continuation value for every pair $(i, j)$ depends on other players waiting on the platform. For instance with the scenario of another player $k$ :

$$\begin{aligned} S_k &> S_i \\ S_k\, close\, to\, S_i &\,(but\, not\, close\, enough) \end{aligned} \tag{2}$$

Then it may be difficult for player $i$ to get matched with a new player $m$; $S_m > S_i$, and $S_k$ will then be closer to $S_m$ with a higher likelihood. In other words, player $k$ will be closer to player $m$ in terms of skill set and there will be higher benefit from matching $k$ and $m$ than matching $i$ and $m$.

The strategy implemented is an approximation of the strategy described above. Since it is possible to simplify the states of the system with the assumption that the continuation value for every pair $(i, j)$ doesn't change with time (doesn't depend on the other players on the platform). This assumption reduces our calculations a lot, since if it is not optimal to match players $i$ and $j$ the first time they meet; it is not optimal to match them at any later time. Therefore:

- No matching happens when a player leaves.
- When a player signs up, only pairs containing the new player are compared for potential matches.

We can also see that players with skill sets at the boundaries ($S = 1\, or\, S = 10$) have more difficulties finding a match than players in the middle ($S = 5\, or\, S = 6$). Therefore their continuation value is lower. We approximate the continuation value of a pair $(i, j)$ as a linear function of the average *uniqueness* of players $i$, and $j$, where the *uniqueness* of player $i$ is defined as :

$$U(i) = \min\left(|S_i - 5|,\, |S_i - 6|\right) \tag{3}$$

Note that $U(5) = U(6) = 0$, and $U(1) = U(10) = 4$. The linear approximation implies that we match players $i$ and $j$ if and only if :

$$|S_i - S_j| = 11 - E_V \times 30 \leq 11 - C_V \times 30 = A + B(U(i) + U(j)) \tag{4}$$

$E_V = Exercise\, Value$
$C_V = Continuation\, Value$

Since we should always match $S_i = 5$ to $S_j = 5$, and $S_i = 6$ to $S_j = 6$, we must have $A \geq 0$. Since the continuation value decreases with uniqueness we must have $B \geq 0$. In addition since $|S_i - S_j| \leq 9$ for any players $i$ and $j$ we must have :

$$A \leq 9 \qquad (looking\, at\, any\, pair\, (i, j)) \tag{5}$$

Finally, it follows from (4) that beyond $MAX_{\{i,j<>5,6\}} \frac{|S_i - S_j| - A}{U(i)+U(j)}$, the exercise decisions do not change (we match everybody). So we keep $B$ as:

$$B \le MAX_{\{i,j<>5,6\}} \frac{|S_i - S_j| - A}{U(i) + U(j)} \tag{6}$$

In particular we need : $B \le 9 - A$

The matching strategy for a given specification is implemented in $Revenue\_from\_Matching.R$ method. The simulation runs until the estimate revenue rate does not change more than 0.01% between periods $T$, and $T+1000$ hours. The $Revenue\_for\_Specifications.R$ method looks at different specifications, and produces a table with the simulation results. From the calculations above and experimentation an appropriate step size of $B$ was found to be 0.125. The total simulation time for these parameters is around 40 minutes.

The simulation output produces a table in csv format with revenue rates in $(\frac{pounds}{minutes})$ for the different $A$, $B$ specifications. The output table is shown below. Corresponding to part ($B$) of the question, we find from *Table 1* that the optimal revenue rate is 0.25790 $(\frac{pounds}{min})$ (marked in purple) for optimal specification ($A = 3, and\, B = 0.375$). For part ($A$) of the question, we find the revenue rate of 0.24119 $(\frac{pounds}{min})$, marked red in the table for specifications that allow all matches.

The following R commands were run to generate the table :

```
> source('Revenue_for_Specifications.R')
> source('Revenue_from_Matching.R')
> Revenue_for_Specifications(A_Vals = c(0:9), B_Vals=seq(0,3,0.125))
```

Table 1: Simulation Results

| **B \ A** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.12087 | 0.21197 | 0.24452 | 0.25569 | 0.25787 | 0.25570 | 0.25135 | 0.24670 | 0.24328 | 0.24119 |
| **0.125** | 0.12087 | 0.21182 | 0.24456 | 0.25571 | 0.25785 | 0.25564 | 0.25135 | 0.24685 | 0.24119 | NA |
| **0.25** | 0.16249 | 0.23195 | 0.24959 | 0.25781 | 0.25564 | 0.25135 | 0.24685 | 0.24119 | NA | NA |
| **0.375** | 0.19288 | 0.23160 | 0.25333 | 0.25790 | 0.25564 | 0.24685 | 0.24119 | NA | NA | NA |
| **0.5** | 0.20421 | 0.24478 | 0.25779 | 0.25569 | 0.24685 | 0.24119 | NA | NA | NA | NA |
| **0.625** | 0.21299 | 0.24452 | 0.25775 | 0.25137 | 0.24119 | NA | NA | NA | NA | NA |
| **0.75** | 0.21299 | 0.25268 | 0.25561 | 0.24119 | NA | NA | NA | NA | NA | NA |
| **0.875** | 0.21299 | 0.25254 | 0.24119 | NA | NA | NA | NA | NA | NA | NA |
| **1** | 0.24993 | 0.24119 | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.125** | 0.24762 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.25** | 0.23386 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.375** | 0.23442 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.5** | 0.23659 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.625** | 0.23659 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.75** | 0.23659 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **1.875** | 0.23659 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2** | 0.23937 | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.125** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.25** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.375** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.5** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.625** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.75** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **2.875** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| **3** | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |