## Example:

For 64 different time frames, we form a matrix of size 10,000x10,000. Rows and Columns of the matrix represent the 10,000 communities chosen in our template vector. The cells contain updated utilities of the web services (or communities) in the row while joining a community (or a web service) in the column. The ‘x’ means the join did not take place. Therefore, scrolling down the 10,000 row, we observe whether a web service (or a formed community) combines with others, and what is the updated utility. Each matrix is made for a time slot, so we have 64 of these tables that are sequentially made.

* For different collaboration values, different 10,000 selection of web services in template vector and different utility formulas we have different sets of this 64 (10,000x10,000) matrixes.
* Each of these matrixes is almost 61,283,845 (60MB) bytes in size.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | <348> | <1934> | <2117> | <348, 1934> | <1934, 2117> | <348, 1934, 2117> | |
| <348> | X | X | 18.027081 | 8.979266 | X | | 0.282708 |
| <1934> | X | X | 2.969072 | -3.851432 | 5.509583 | | 6.387725 |
| <2117> | 18.027081 | 2.969072 | X | 5.509583 | 2.969072 | | 6.085001 |
| <348,1934> | X | 18.027081 | -3.851432 | X | 2.969072 | | 2.969072 |
| <1934, 2117> | X | -3.851432 | 5.509583 | 2.969072 | X | | -2.043377 |
| <348, 1934, 2117> | 2.969072 | 2.969072 | 5.509583 | 5.689946 | -2.043377 | | X |

K1(<348>,1) = {<2117>}

K1(<348>,2) = {<2117>, <348, 1934>}

K1(<348>,3) = {<2117>, <348, 1934>, <348, 1934, 2117>}

K1(<348,1934>,1) = {<1934>}

K1(<348,1934>,2) = {<1934>, <348, 1934, 2117>}

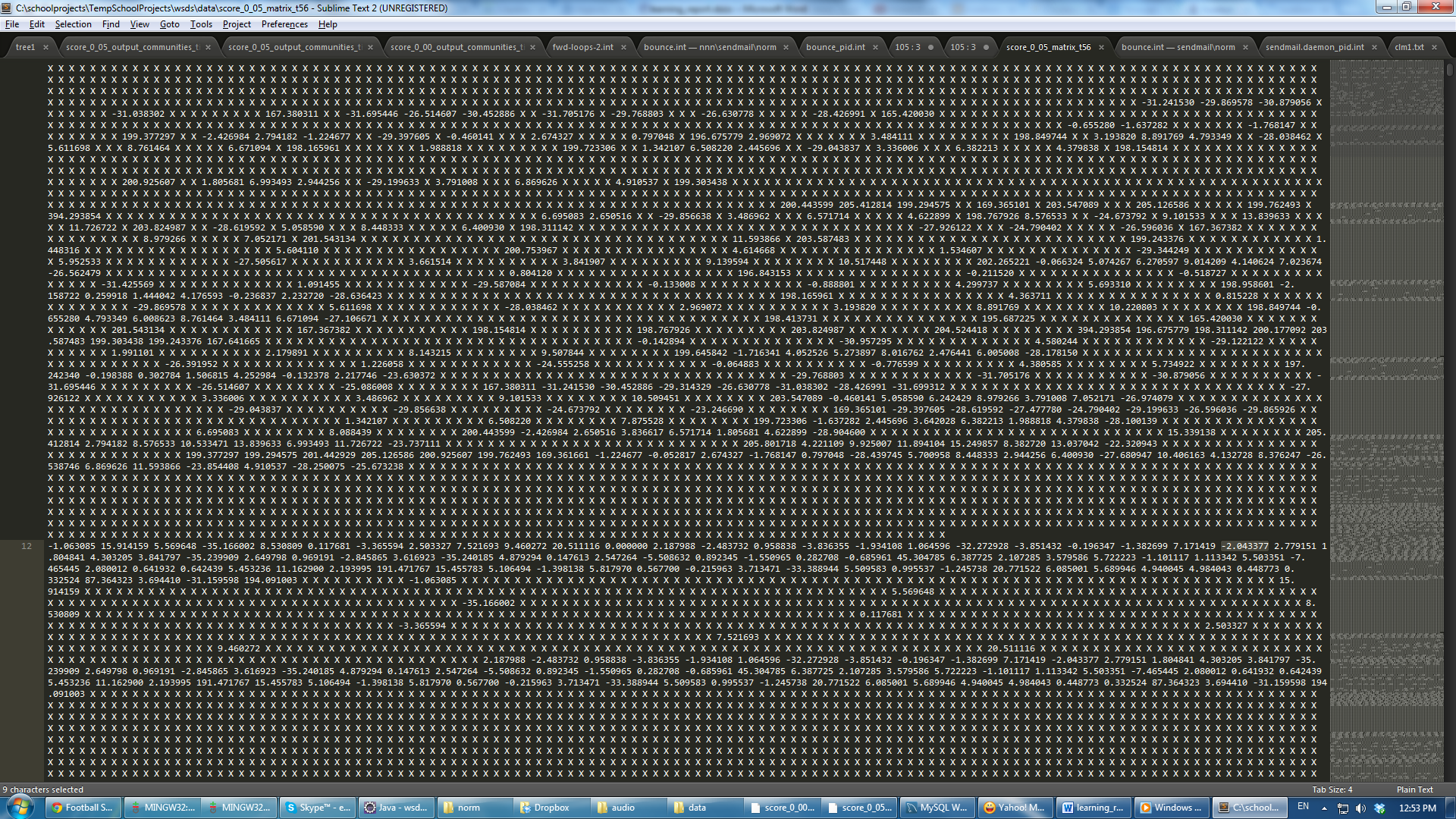
K1(<348,1934>,3) = {<1934>, <348, 1934, 2117> , <1934, 2117>}

L1(<348>,1) = {<2117>}

L1(<348,1934>,1) = EMPTY SET

L1(<348,1934>,2) = {<1934, 2117>}

A real one:



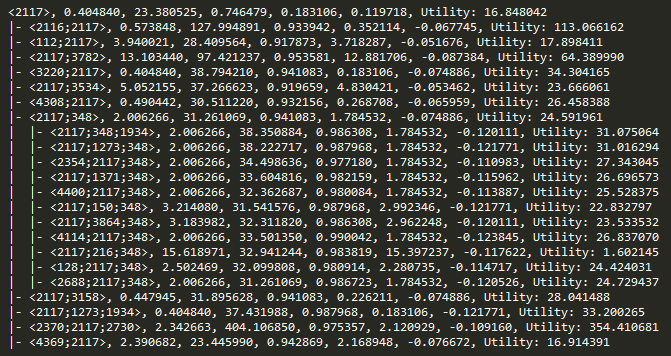
On early steps of simulation web services are more strict and as time goes on we let them choose, second, third best options too. The rate this increases is parameterized and by changing that we get more number or less number of communities.

At end we list all the communities formed, this is one example:

<http://users.encs.concordia.ca/~e_khosr/score_0_05_output_communities_time_10.txt>

And also another result is a tree, in which we see how each web service individually played the game and joined other services in our experiment, this link and image is one example:

<http://users.encs.concordia.ca/~e_khosr/score_0_05_output_communities_time_10_tree.txt>



The above figure shows how web service <2117> formed communities and the 5 abovementioned metrics and overall utility of the web service and corresponding communities.

Now we can follow a path a web service has taken, here is an example:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| WS | RT | TP | AV | Ex1 | Ex2 | **Utility** |
| 2117 | 0.4 | 23.3 | 0.7 | 0.1 | 0.1 | 16.8 |
| 2117,348 | 0.4 | 31.3 | 0.9 | 0.1 | -0.08 | 24.5 |
| 2117,348,1934 | 0.4 | 38.3 | 0.9 | 0.1 | -0.12 | 31.07 |

Is it obvious in this case, TP, AV, Ex2 and utility are increasing and Ex1 and RT are non-decreasing. Now having a distance function, we can estimate with a distance what will happen to other web services and their communities. We can claim with 95% confidence that this path will take place for web service <2117> and these are the utilities it will gain in different collaborative groups. Also using regression based on whether communities formed or not, on this 10,000 vector template sample, we can expand and predict how likely other collaborative groups would form and how likely possible options for different web services are beneficial options over time to join and collaborative with them or not.