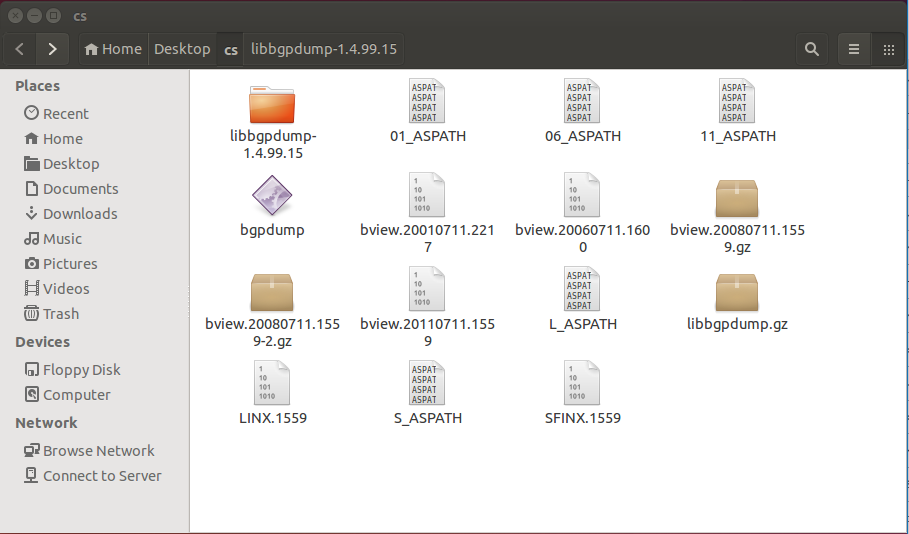
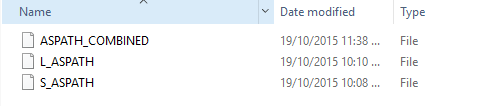
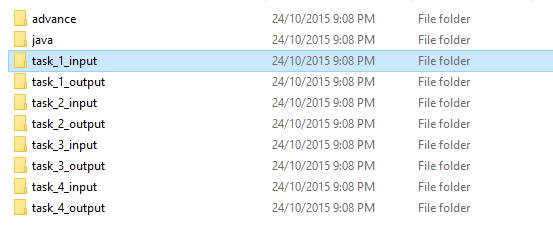
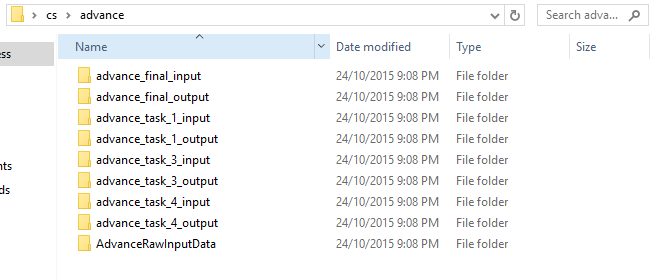
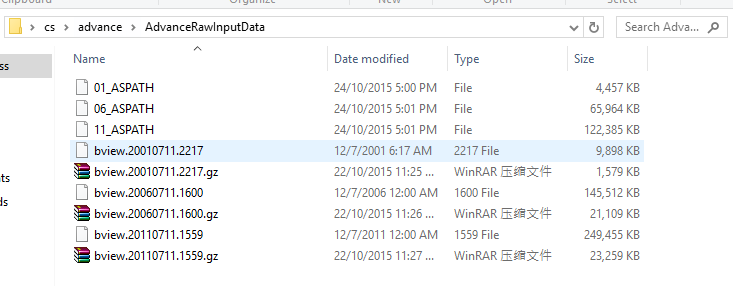
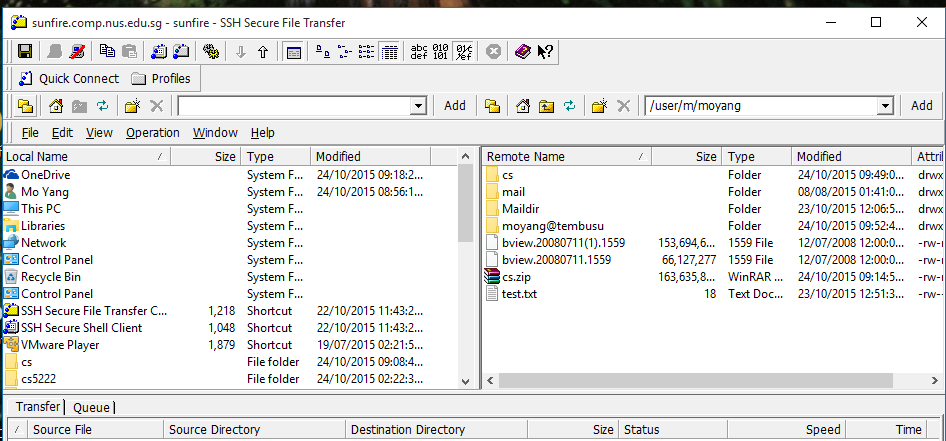
CS5229 Report

YANG MO A0091836X

# 1. Preparation

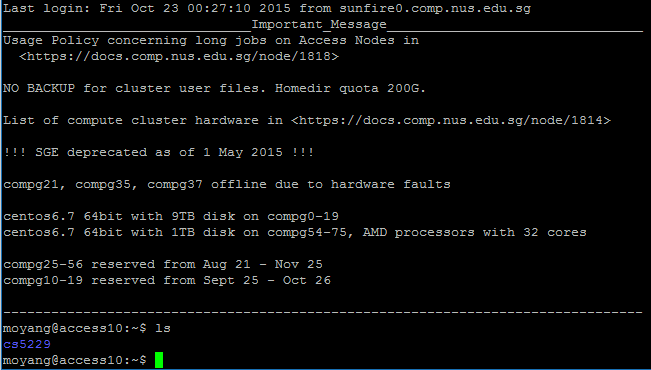
1. Unzip data set first at local Ubuntu virtual machine

Advance Task

1. Combine two files for LINX and SFINX into one combined file for Elementary and Intermediate Tasks
2. Create local folder first
3. Zip folder and upload to sunfire and unzip it
4. Copy unzipped cs folder to Tembusu Cluster and rename it to cs5229 using command:

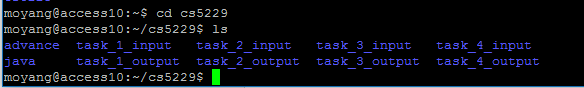
scp -r cs moyang@tembusu:cs5229

1. Log into Tembusu :



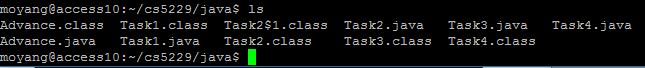
ssh –X tembusu

1. Verify all content is present in tembusu



# 2. Elementary Part --- Task 1

## 2.1 Run Task 1

Compile all the java files first

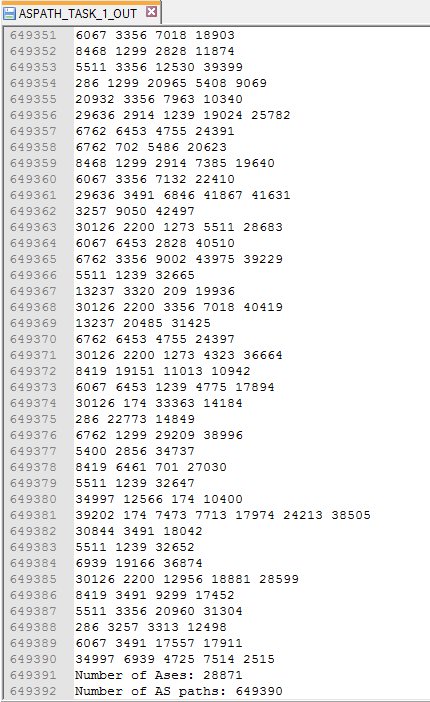
Command Pattern : java Task1 <inputfile> <outputfile>

Real Command to execute task 1 :

java Task1 ~/cs5229/task\_1\_input/ASPATH\_COMBINED ~/cs5229/task\_1\_output/ASPATH\_TASK\_1\_OUT

Task 1 will take in the combined ASPATH file and run line by line to count distinct number of AS and AS paths. The result will be written in ASPATH\_TASK\_1\_OUT

## 2.2 Result:



To be able to verify the output file locally, the output file is copied back to sunfire and then downloaded to local windows machine. We then open it using notedpad ++.

The result is clear that:

Number of distinct Ases is 28871

Number of distinct AS Paths is 649390

The ASPATH\_TASK\_1\_OUT file does not contain any duplicate AS Path.

# 3. Elementary Part --- Task 2

## 3.1 Prepare input data

Since task 2 need to use the output from task 1. ASPATH\_TASK\_1\_OUT is copied from task\_1\_output folder to task\_2\_input folder:



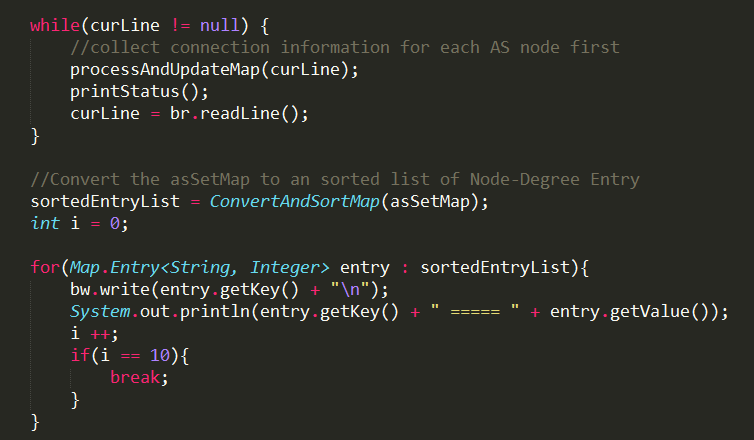


## 3.2 Run Task 2

Command Pattern : java Task2 <inputfile> <outputfile>

Real Command to execute task 2 :

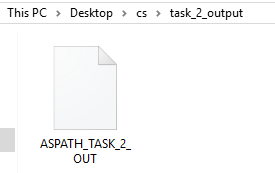
java Task2 ~/cs5229/task\_2\_input/ASPATH\_TASK\_1\_OUT ~/cs5229/task\_2\_output/ASPATH\_TASK\_2\_OUT

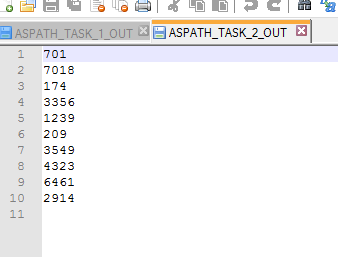
Task 2 will first collect connection information for each AS node first. Then it will compute the degree for each AS node. Before it write to the file, it will sort based on the degree and list on Top 10 AS nodes.

## 3.3 Result

To be able to verify the output file locally, the output file is copied back to sunfire and then downloaded to local windows machine. We then open it using notedpad ++.







The results is clear that only top ten AS nodes are ouput in the output file.

And 701 is the AS of highest degree.

# 4. Intermediate Part --- Task 3

## 4.1 Prepare input data

Since task 3 need to use the output from task 1. ASPATH\_TASK\_1\_OUT is copied from task\_1\_output folder to task\_3\_input folder:



## 

## 4.2 Run Task 3

Command Pattern : java Task3 <L> <R> <inputfile> <outputfile>

Real Command to execute task 3 :

java Task3 1 60 ~/cs5229/task\_3\_input/ASPATH\_TASK\_1\_OUT ~/cs5229/task\_3\_output/ASPATH\_TASK\_3\_OUT

Here L is set to 1 and R is set to 60. These values are chosen due the original paper.

Following the algorithm given, the program contains five parts:

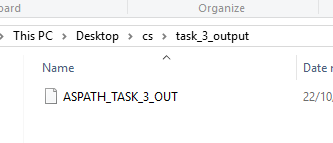
* Non p2p Phase 1
* Non p2p Phase 2
* Non p2p Phase 3
* p2p Phase 2
* p2p Phase 3

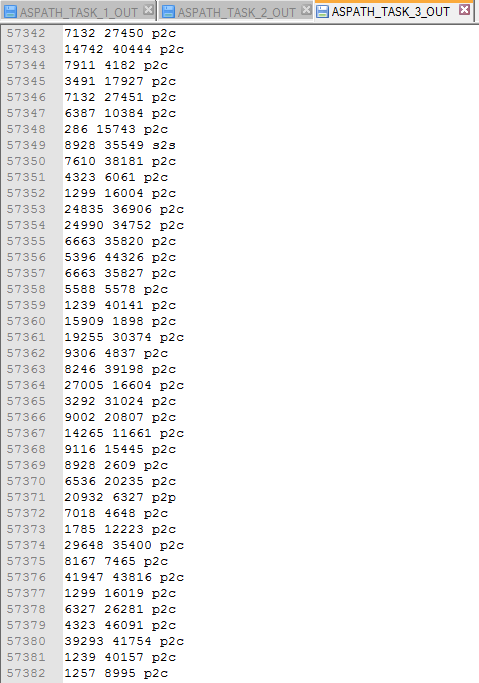
Where Non p2p phases annotate only s2s, c2p, p2c relationships.

P2p phases will annotate those node pairs who has p2p relationship.

## 4.3 Result:

To be able to verify the output file locally, the output file is copied back to sunfire and then downloaded to local windows machine. We then open it using notedpad ++.





It is clear that there are totally 57382 node pairs which have been annotated.

# 5. Intermediate Part --- Task 4

## 5.1 Prepare input data

Since task 4 need to use the output from task 3. ASPATH\_TASK\_3\_OUT is copied from task\_3\_output folder to task\_4\_input folder:



## 

## 5.2 Run Task 4

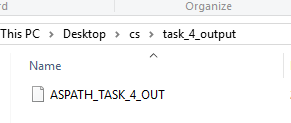
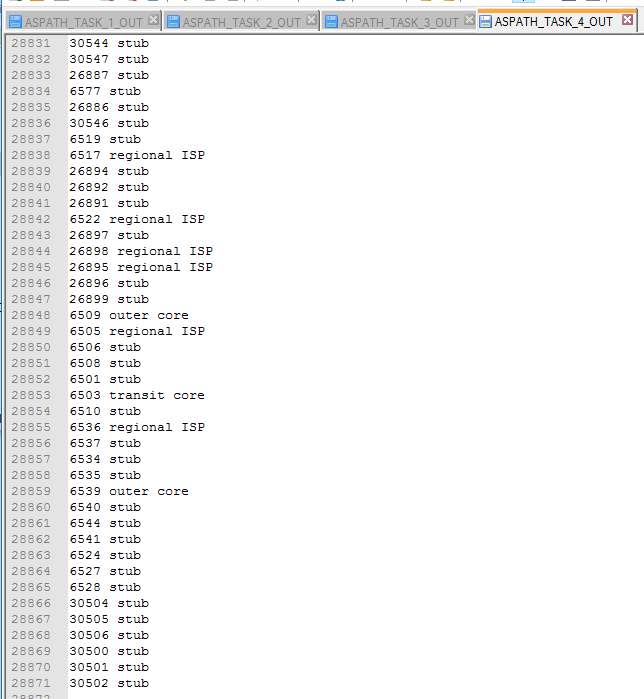
Command Pattern : java Task4 <inputfile> <outputfile>

Real Command to execute task 4 :

java Task4 ~/cs5229/task\_4\_input/ASPATH\_TASK\_3\_OUT ~/cs5229/task\_4\_output/ASPATH\_TASK\_4\_OUT

## 5.3 Result

To be able to verify the output file locally, the output file is copied back to sunfire and then downloaded to local windows machine. We then open it using notedpad ++.

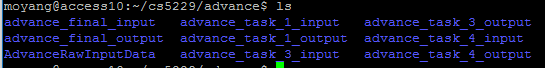


It can be seen that there are total 28871 AS nodes which have been classified.

According to output pf task 1, Number of distinct Ases is 28871, which means that all the AS nodes have been classified successfully.

# 6. Advance Part

## 6.1 Prepare input data



To get the edge count as well as distribution of degree, all the data for year 01, 06 and 11 need to go through task 1, task 3 and task 4 to get the output files from each task. These three kinds of output files will be copied into advance\_final\_input folder to enable the Advance task.

## 6.2 Run step:

1. Run task 1 for data from year 01, 06, 11

2. Run task 3 based on the results from step 1 for year 01, 06 , 11

3. Run task 4 based on the results from step 1 for year 01, 06, 11

4. Run advance task based on the results from step 1, 2 and 3

## 6.3 Run Command

Step 1:

java Task1 ~/cs5229/advance/advance\_task\_1\_input/01\_ASPATH ~/cs5229/advance/advance\_task\_1\_output/01\_TASK\_1\_OUT

java Task1 ~/cs5229/advance/advance\_task\_1\_input/06\_ASPATH ~/cs5229/advance/advance\_task\_1\_output/06\_TASK\_1\_OUT

java Task1 ~/cs5229/advance/advance\_task\_1\_input/11\_ASPATH ~/cs5229/advance/advance\_task\_1\_output/11\_TASK\_1\_OUT

Step 2:

java Task3 1 60 ~/cs5229/advance/advance\_task\_3\_input/01\_TASK\_1\_OUT ~/cs5229/advance/advance\_task\_3\_output/01\_TASK\_3\_OUT

java Task3 1 60 ~/cs5229/advance/advance\_task\_3\_input/06\_TASK\_1\_OUT ~/cs5229/advance/advance\_task\_3\_output/06\_TASK\_3\_OUT

java Task3 1 60 ~/cs5229/advance/advance\_task\_3\_input/11\_TASK\_1\_OUT ~/cs5229/advance/advance\_task\_3\_output/11\_TASK\_3\_OUT

Step 3:

java Task4 ~/cs5229/advance/advance\_task\_4\_input/01\_TASK\_3\_OUT ~/cs5229/advance/advance\_task\_4\_output/01\_TASK\_4\_OUT

java Task4 ~/cs5229/advance/advance\_task\_4\_input/06\_TASK\_3\_OUT ~/cs5229/advance/advance\_task\_4\_output/06\_TASK\_4\_OUT

java Task4 ~/cs5229/advance/advance\_task\_4\_input/11\_TASK\_3\_OUT ~/cs5229/advance/advance\_task\_4\_output/11\_TASK\_4\_OUT

Step 4 (final Step):

Command Pattern : java Advance <asPathFile> <edgeFile> <classFile> <outputFile>

Real Command to run advance task:

java Advance ~/cs5229/advance/advance\_final\_input/01\_TASK\_1\_OUT ~/cs5229/advance/advance\_final\_input/01\_TASK\_3\_OUT ~/cs5229/advance/advance\_final\_input/01\_TASK\_4\_OUT ~/cs5229/advance/advance\_final\_output/01\_FINAL\_OUT

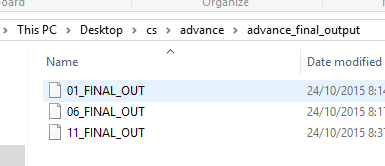
java Advance ~/cs5229/advance/advance\_final\_input/06\_TASK\_1\_OUT ~/cs5229/advance/advance\_final\_input/06\_TASK\_3\_OUT ~/cs5229/advance/advance\_final\_input/06\_TASK\_4\_OUT ~/cs5229/advance/advance\_final\_output/06\_FINAL\_OUT

java Advance ~/cs5229/advance/advance\_final\_input/11\_TASK\_1\_OUT ~/cs5229/advance/advance\_final\_input/11\_TASK\_3\_OUT ~/cs5229/advance/advance\_final\_input/11\_TASK\_4\_OUT ~/cs5229/advance/advance\_final\_output/11\_FINAL\_OUT

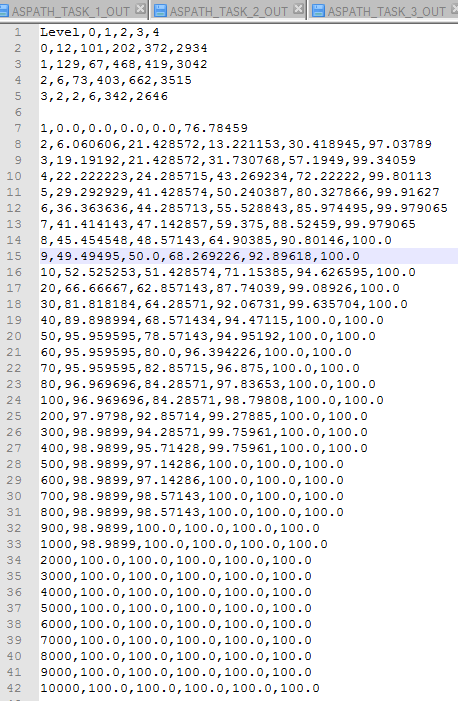
## 6.4 Result

To be able to verify the output file locally, the output file is copied back to sunfire and then downloaded to local windows machine. We then open it using notedpad ++.





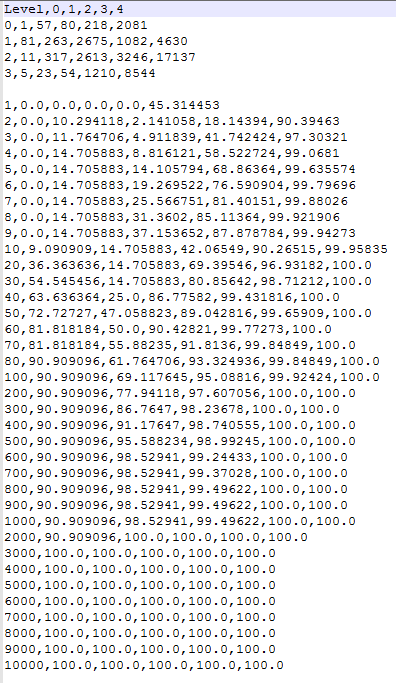
### 6.4.1 Output file for Year 2001



Cumulative distribution of AS degree by level.

Inter-connectivity of ASes across levels.

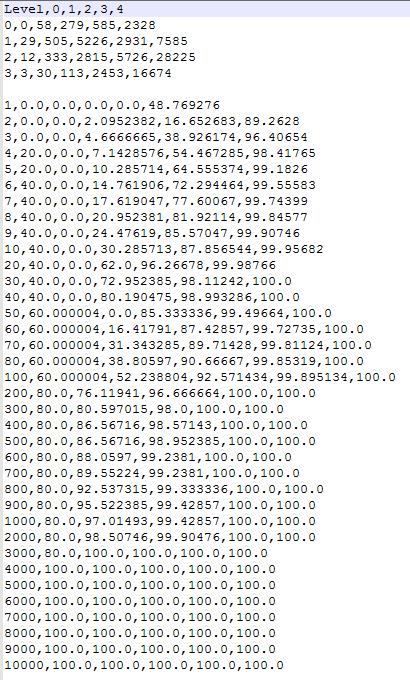
### 6.4.1 Output file for Year 2006



Inter-connectivity of ASes across levels.

Cumulative distribution of AS degree by level.

### 6.4.1 Output file for Year 2011



Inter-connectivity of ASes across levels.

Cumulative distribution of AS degree by level.

## 6.5 Analysis of Results:

### 6.5.1 Inter-connectivity tables for year 2001, 2006 and 2011

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Level | 0 | 1 | 2 | 3 | 4 |
| 0 | 12 | 101 | 202 | 372 | 2934 |
| 1 | 129 | 67 | 468 | 419 | 3042 |
| 2 | 6 | 73 | 403 | 662 | 3515 |
| 3 | 2 | 2  2001 | 6 | 342 | 2646 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Level | 0 | 1 | 2 | 3 | 4 |
| 0 | 1 | 57 | 80 | 218 | 2081 |
| 1 | 81 | 263 | 2675 | 1082 | 4630 |
| 2 | 11 | 317 | 2613 | 3246 | 17137 |
| 3 | 5 | 23 | 54 | 1210 | 8544 |

2006

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Level | 0 | 1 | 2 | 3 | 4 |
| 0 | 0 | 58 | 279 | 585 | 2328 |
| 1 | 29 | 505 | 5226 | 2931 | 7585 |
| 2 | 12 | 333 | 2815 | 5726 | 28225 |
| 3 | 3 | 30 | 113 | 2453 | 16674 |

2011

It can be seen that from 2001 to 2011

* Inter connectivity between level 0 AS is decreasing, which means dense cores are less connected with each other than before.
* Level 3 to level 4 connections have increased hugely. This means that regional ISP are more closely connected to normal users.

### 6.5.1 Cumulative distribution tables for year 2001, 2006 and 2011

From above, we can see that:

* Level 1 transit cores distribution are shifting from low degree to high degree
* Level 4 Customer distribution have also seen an huge shift from low degree to high degree, which could mean that there are more and more c2p relations from level 4 to upper levels.

We can predict that level 4 and level 1 distribution will keep shift to higher degree while others may see a much slower change.