Distributed Content Searching

Project Report

CS4262 - Distributed Systems

Group members

150535P - S.A. P.A. Rukshan
150073V - A.K.G. Chamiran
150549L - Y.J. Samarasekara
150493H - M.D.W. Rajapaksha

Design decisions we made when developing the system

Distributed Content Searching System was implemented using Java language. Following java packages were used to wire the communication among nodes.

UDP messages - java.net.DatagramSocket
TCP messages - java.net.ServerSocket

Following are the decisions we made other than the given specific details in the problem description.

JOIN message

A newly connecting node gets the details of few more nodes in the existing network with the REGOK acknowledgement reply from the Bootstrap Server (BS) Then this node joins with other nodes using JOIN messages to known nodes. Nodes that receive these JOIN messages extend their routing tables by saving the new node that send the JOIN message (see Fig. 1).

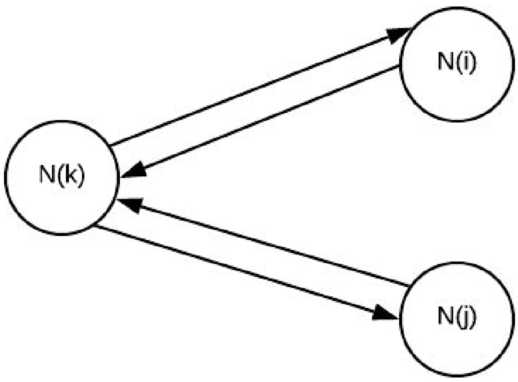

Figure 1: Node *k* nodes join with *i, j* nodes. After joining two way communication knowledge is known by both connecting node *(k)* and network nodes *(i, j)*.

SER message

When a client node need to find a file it sends the file name to BS. Then BS sends a SER message the node mentioned in the message. It first searches for the file in the own node. If it finds the file then sends a SEROK message to the Bootstrap Server. Otherwise the SER message is forwarded to the nodes in the routing table other than the node from where the SER was received. Before forwarding the SER message hop count is decremented by one. This process continues until the hop count reaches zero (see Fig. 2).

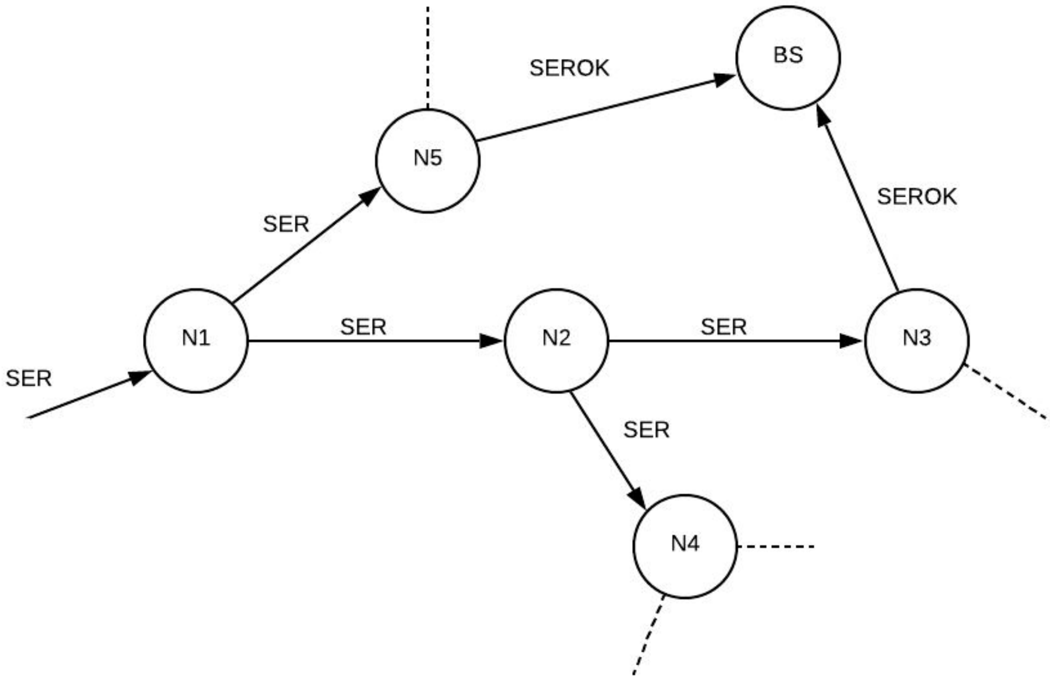

Figure 2: *N1* receives a SER messages since it does not contain the file, it forwards SER to *N2* and *N5* peer nodes then *N5* have found a file and it responds to BS. *N2* again forwads SER to its peers and *N3* peer's search is successful and it sends SEROK to BS node.

LEAVE message

When a node leaves the network it does it gracefully. It first unregister itself from the BS. Therefore no new node will receive information about the leaving node. Then it notifies the nodes in its routing table to remove the entry for the leaving node using the LEAVE message.

File DOWNLOAD

To initiate the file download UDP messages are used while file transfer has been implemented using TCP messages. Lets, assume the file download happens from a peer node to a BS.

1. First BS server receives a DOWNLOAD_TRIGGER message from a client node.
2. BS sends a DOWNLOAD message to the corresponding peer node.
3. Then the peer node start a new thread to accept TCP message to send the file.
4. Peer sends a DWNLDOK to the BS.
5. Then BS gets connected to the TCP channel of the peer node and download the sent file.

Statistics of the system

Before removing 2 random nodes

Node	1	2	3	4	5	6	7	8	9	10	11	12
SER	174	50	180	251	104	92	152	72	71	87	29	92
FWD	140	176	136	174	72	0	237	31	60	94	0	84
ANS	64	6	646	92	23	18	15	23	24	5	2	15
DGR	6	4	5	7	3	4	4	2	3	2	2	2

Table 1: stats of each node after running all the given queries

SER - the number of query messages received
FWD - the number of query messages forwarded
ANS - the number of query messages answered
DGR - degree or routing table size of the node

After removing 2 random nodes

Node	1	2	3	4	5	6	7	8	9	10	11	12
SER	121	194	-	284	69	-	163	110	151	87	59	108
FWD	212	177	-	112	33	-	288	64	228	82	0	0
ANS	10	54	-	89	19	-	18	27	18	10	4	25
DGR	3	4	-	5	2	-	4	3	4	2	1	2
Cost	343	425	-	485	121	-	469	201	397	179	64	133

Table 2: stats of each node after running all the given queries

Per node cost can be stated as the total of SER, FWD and ANS messages given by the last row in Table 2.

Per query cost can be calculated as total messages / number of queries.

$$\sum_{i=1}^{12} Cost(i) \,/\, number\ of\ queries = 2817 / 50 = 56.34\ msgs\ per\ query$$

	hops	Latency (seconds)	messages per node	node degree

min	1	0	59	1
max	3	6	284	5
avg	2.222222	2.19046	134	3
std	0.736357	1.671954	67.252757	1.247219

Table 3: stats of the content searching system

CDFs of parameters

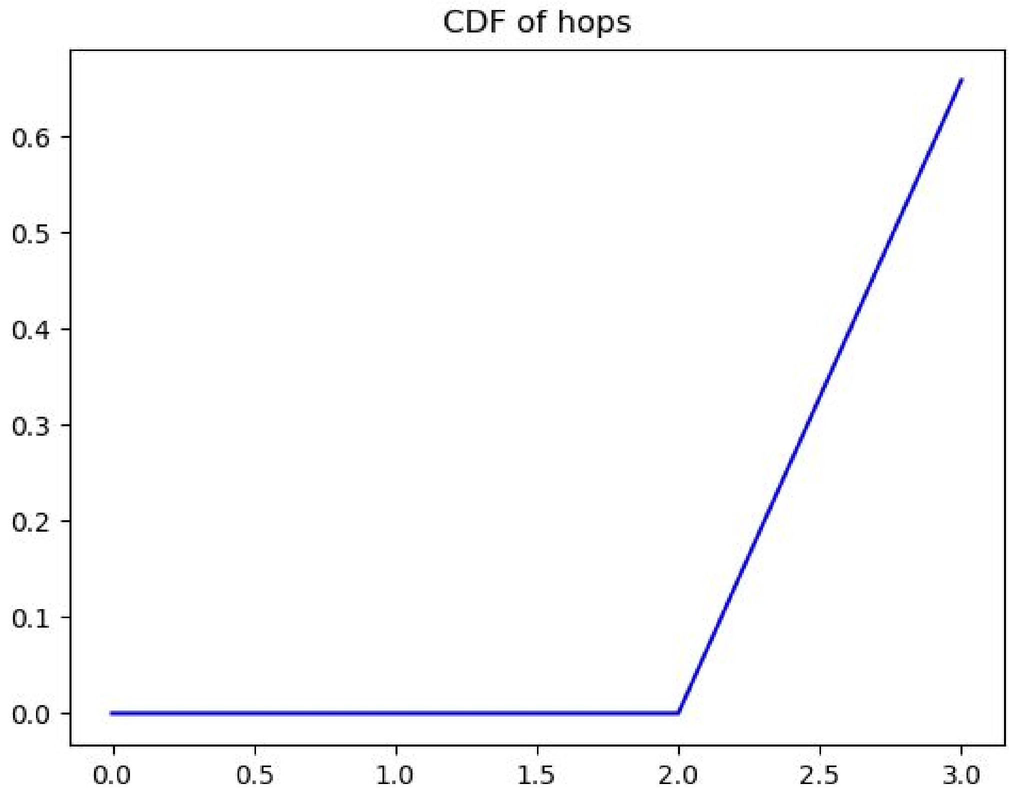

Figure 3: CDF of hops

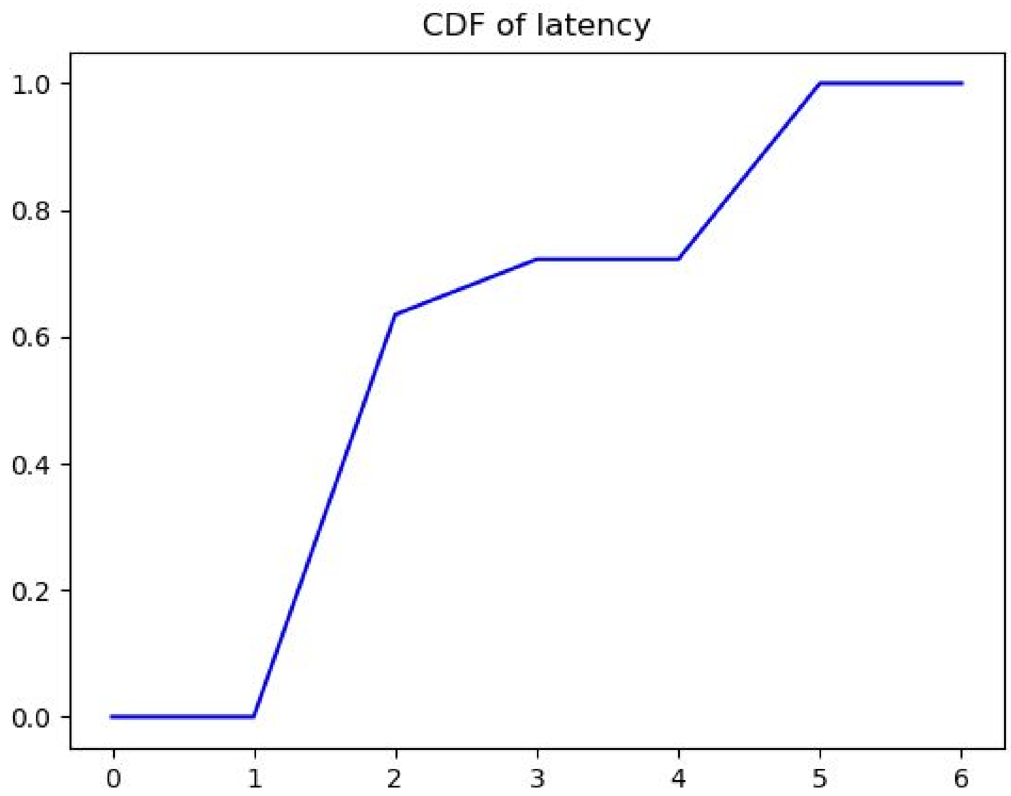

Figure 4: CDF of latency

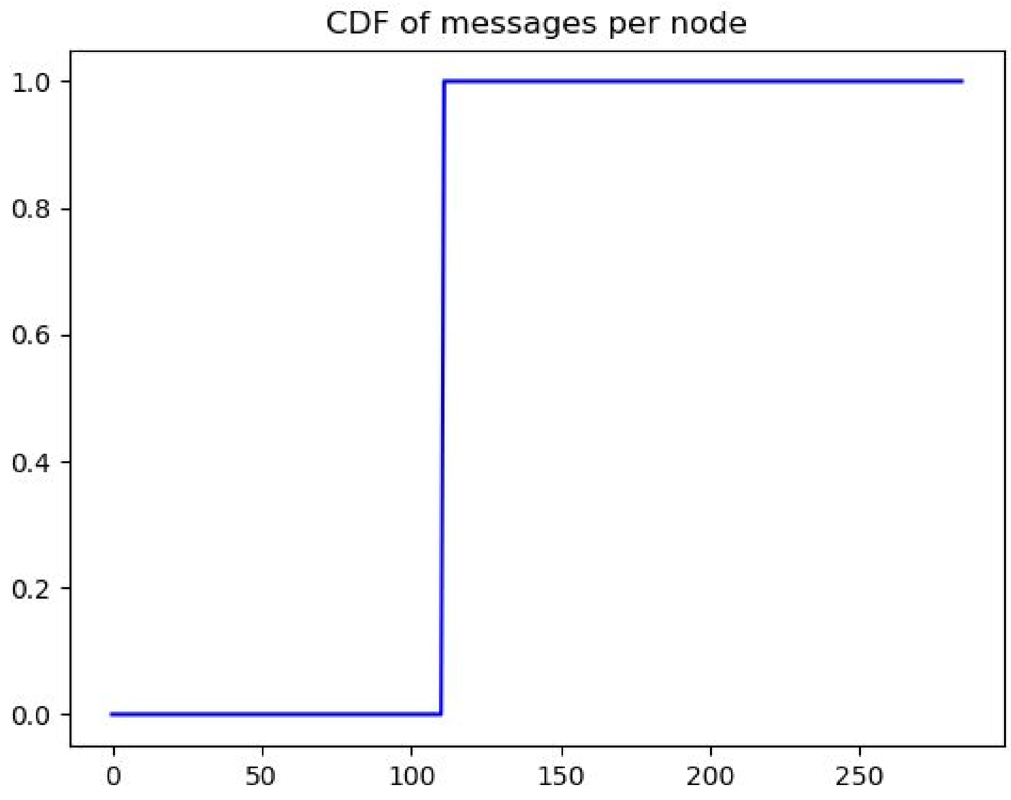

Figure 5: CDF of messages per node

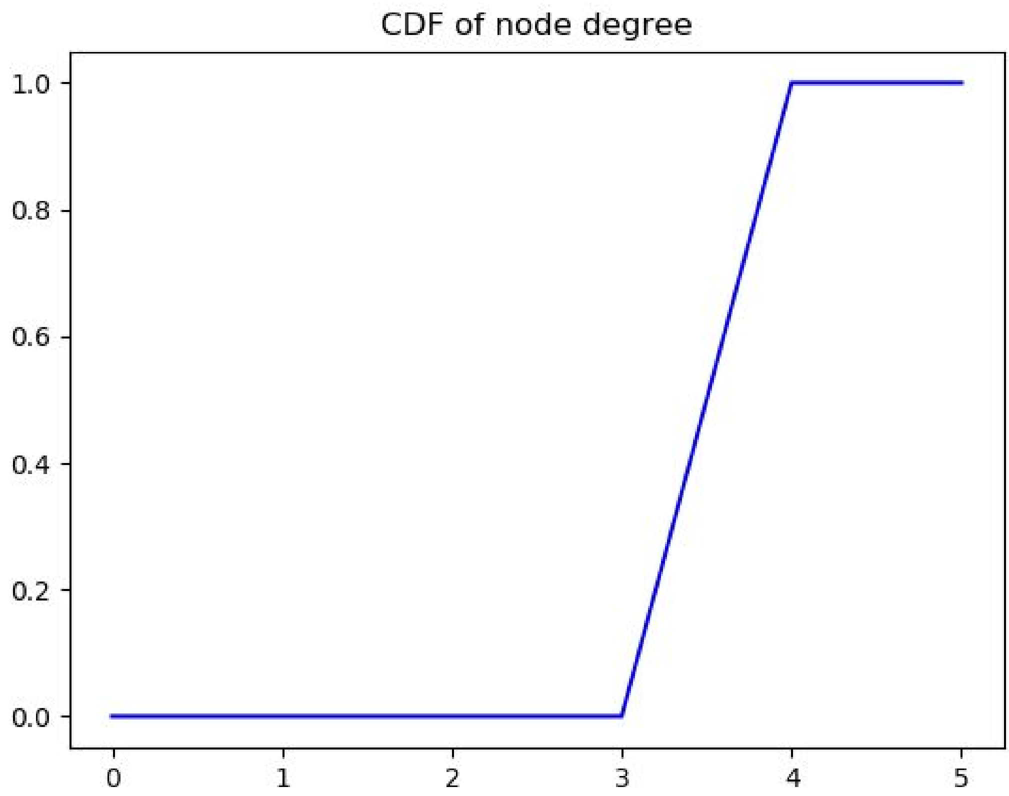

Figure 6: CDF of node degree

Q. Discuss how your solution will behave, if number of queries (Q) are much larger than the number of nodes (N) (Q >> N) and vice versa (N >> Q). Comment on how to improve the query resolution while reducing messages, hops, and latency.

When Q >> N, lots of queries will happen among nodes and the network congestion will be high. If N >> Q then the communication will spread among other nodes and congestion will be relatively low.

One obvious solution to improve the query resolution is to not to forward the SER where it receives. Optimal hops count can be calculated practically by plotting the latency against hops count.

Reflections on the assignment

It was really hard to test cause setting up the network takes lots of time. Therefore setting up the nodes and querying were automated using shell scripts. Then querying and resolving bugs in code was easier.