Lab Assignment 2: Distributed Graphs

Ricardo Rojas Ruíz
Gabriela Martínez
Kaoutar Chenaff

Date: March 27th, 2019

Exercise 1

Superstep 0
In this first stage, all the vertices have their initial states set to 9, 1, 6 and 8 respectively (following their order). Later, all these nodes will receive a message containing the MAX_VALUE integer and they will compare themselves against this number and as a result, the nodes will be initialized with their original values: 9, 1, 6, 8.

Also, note that vertex 1 will send a message to vertex 2 because the value within vertex 2 is lower than the value of the vertex 1 (1 < 9). The remaining nodes will not send more messages.

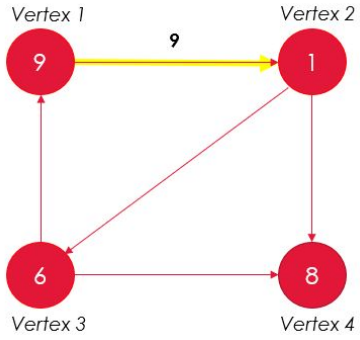

Figure 1. Illustration of superstep 0

Superstep 1
At this step, vertex 2, which received a message from vertex 1 in the previous superstep, will be updated with the value of 9 that was contained in that message, since 9>1. Subsequently, vertex 2 will send a message containing the value of 9 to the vertices 3 and 4. The remaining vertices are not emitting more messages and the final state of them is as follows: 9, 9, 6, 8.

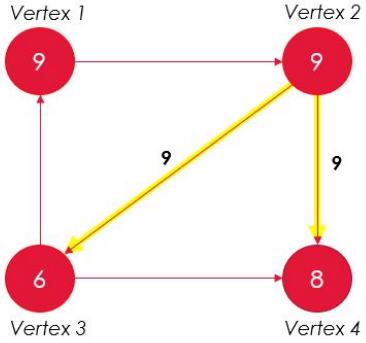

Figure 2. Illustration of superstep 2

Superstep 2

As a consequence of superstep 1, vertices 3 and 4 receive the message with the value of 9 and update themselves with this new value, as 9 is greater than 6 and 8, the oldest values of these vertices. After this point, no more messages are to be sent.

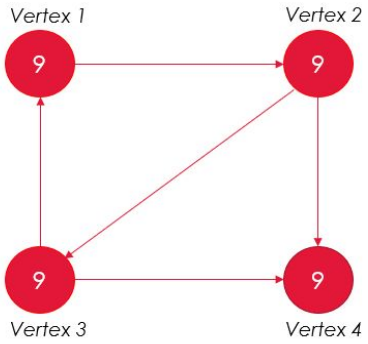

Figure 3. Illustration of superstep 2 and final state of the graph

The following tables illustrate how these supersteps were performed:

Superstep 0

Vertex	*Initial state*	*Message received*	*Calculations*	*Resulting state*	*Message to send*
1	9	Integer.MAX_VALUE	message == Integer.MAX_VALUE	9	9
2	1	Integer.MAX_VALUE	message == Integer.MAX_VALUE	1	-
3	6	Integer.MAX_VALUE	message == Integer.MAX_VALUE	6	-
4	8	Integer.MAX_VALUE	message == Integer.MAX_VALUE	8	-

Table 1. Changes performed in superstep 0.

Superstep 1

Vertex	*Initial state*	*Message received*	*Calculations*	*Resulting state*	*Message to send*
1	9	-	-	9	-
2	1	9	Math.max(vertexValue, message)	9	9, 9
3	6	-	-	6	-
4	8	-	-	8	-

Table 2. Changes performed in superstep 1.

Superstep 2

Vertex	*Initial state*	*Message received*	*Calculations*	*Resulting state*	*Message to send*
1	9	-	-	9	-
2	9	-	-	9	-
3	6	9	Math.max(vertexValue, message)	9	-
4	8	9	Math.max(vertexValue, message)	9	-

Table 3. Changes performed in superstep 2 and final state of the graph.

Exercise 4: analyzing Wikipedia articles relevance

The following is the list of the top 10 Wikipedia articles that are more relevant according to the Page Rank algorithm we implemented in Java 8:

ID	*Article*
8830299306937918434	University of California, Berkeley
1746517089350976281	Berkeley, California
8262690695090170653	Uc Berkeley
1735121673437871410	George Berkeley
7097126743572404313	Berkeley Software Distribution
8494280508059481751	Lawrence Berkeley National Laboratory
6990487747244935452	Busby Berkeley
5820259228361337957	Xander Berkeley
1630393703040852365	Berkeley, CA
1899425950029509269	Berkeley County, West Virginia

Table 4. Top 10 Wikipedia articles found via PageRank with a dampingFactor = 0.5 and 10 iterations.

To obtain these results consistently, the damping factor could have been set within the range [0.05, 0.95]. However, this parameter together with the number of iterations may result in different scenarios. For instance, a damping factor of 0.85 could give us the right result but only if the number of iterations is equal to or greater than 10. In the end, we found that for smaller damping factors the number of repetitions of the algorithm is not relevant to obtain the same consistent top 10. Therefore, we recommend to "optimally" set the damping factor at any probability that fits the domain of study (in this case would be ideal to have it around 0.5) and take into account the behavior of this parameter together with the number of iterations, which may result in a very expensive algorithm for a very large graph. For this particular scenario, we would manage 10 repetitions.

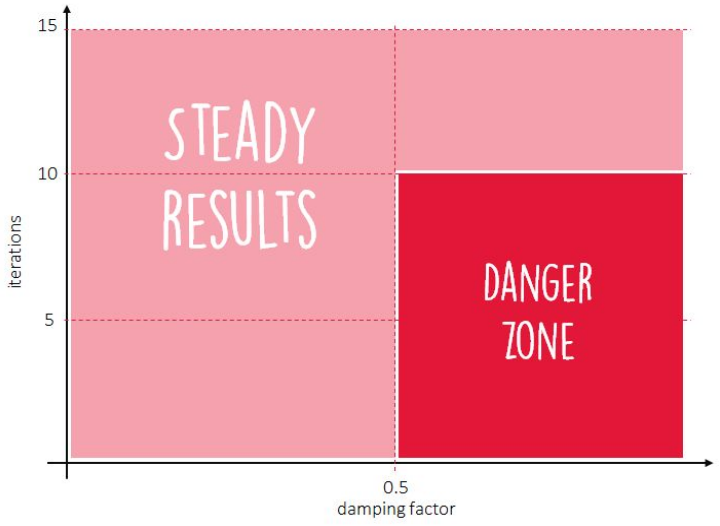

Figure 4. The relationship between the number of iterations and the damping factor for the PageRank algorithm.