Analysis of the Star Wars Social Graph

Parallel Breadth First Search Algorithm

Wiki engine is collaborative software that runs a wiki, which allows users to create and collaboratively edit "pages" or entries via a web browser. It is widely used to create thematic encyclopedia on Web such as wikix for example. Another example, Wookieepedia, contains more than 100000 pages written by the fans of the franchise. Such well-developped source can be used to explore a lot of insights on the universe (look on the Kirell Benzi's blog for the inspiration).

The goal of current project is to study the social graph of Star Wars characters based on the following rule. A character A is linked to another one B if its name appears on A's associated wiki page. More precisely, you have to develop a parallel search robot that traverses the social graph built from Wookieepedia and saves on disk the visited k-neighborhood using the *Breadth First Search* algorithm.

- The program should take three arguments from the command line: name of the character, the depth of the search k and the name of the output file, e.g., `java program Yoda 5 yoda.txt`
- You should start the parallel BFS from the page associated with given name of the character and work until it found all the characters within the distance at most k from the starting point and writing them into the output file in realtime.
- The output file is a text file containing n lines for n found characters in the neighborhood. Each line consists of the name of the character and the distance from the source character. Ordering the output by the distance first and by the alphabetic order second is an advantage. Here is the incomplete example for the `yoda.txt`.
  ```
  Anakin Skywalker, 1
  Qui-Gon Jinn, 1
  ...
  Sebulba, 2
  ...
  ```
- Robot should not visit several times the same page.
- Robot should be able to recognize the characters of the universe only and not the locations, races, gouvernements etc.)
- Try to reach the maximum rapidity of the algorithm and justify the chosen method. You are encouraged to provide some benchmarks. At least, your program should print the runtime on the screen.
- Optionally you can add any GUI/visualisation for the project.

For the questions you can contact anatolii.kostrygin@gmail.com