HW 4 - Efficiency & Reliability: PlagiarismDetector & ReliablePathFinder

Joseph Haymaker

11/10/2017

Retry block implementation in `ReliablePathFinder`:

After finishing the specifications for the `checkPath()` and `findPath()` methods in the `ReliablePathFinder` class I originally had the following implementations:

Parallel Recovery Block

thread1 - runs DFS, checks acceptance test and if it fails then runs BFS (like in-class example).

thread2 - runs BFS and checks acceptance test then waits for thread 1 to finish.

Retry block

Ran DFS with (`dest`, `src`) as parameters then runs acceptance test on the results.

After coming up with this intial implementation I thought the final decision of using BFS or DFS in the retry block would come based off of a tradeoff of time *efficiency* and *accuracy*. The question then became how to measure these two things. I then decided to run some trials changing the implementation to see how the time taken (measured similarly to `PlagiarismDetector`) and outcome (successes and failures out of the 1000 trials) varied. The results are summarized in the below table.

Trial	Specifications	Time (secs.)	Outcome
trial1	Full method (*thread1* -DFS & BFS, *thread2* -BFS & DFS, retry block - DFS)	287.343	1000 successes
trial2	Full method (same as above)	293.772	1000 successes
trial3	Full method (same as above)	295.529	1000 successes
trial4	*thread1*- DFS & BFS, *thread2*- just BFS, retry block - DFS	251.335	1000 successes (*** **winner for sake of due diligence & accuracy** ***)
trial5	*thread1* -just DFS, *thread2* -just BFS, retry block - DFS	265.265	1000 successes
trial6	(same as above)	261.433	1000 successes

Trial	Specifications	Time (secs.)	Outcome
trial7	running just *thread1* DFS then DFS retry block	32.575	1000 successes
trial8	running just *thread2* BFS then DFS retry block	51.65	1000 successes
trial9	running just *thread2* BFS then BFS retry block	59.253	**Success: 997; Fail: 3
trial10	running just *thread1* DFS then BFS retry block	32.917	**Success: 999; Fail: 1
trial11	*thread1* - just DFS, *thread2* - just BFS, retry block BFS	257.778	successes: 1000

As you can see the original implementation when 1 or both of the threads ran both DFS and BFS the running time was close to 5 minutes. The most interesting results came in trials 7-10. We can see that when we just run *thread 1* with DFS and BFS retry block (trials 7 & 10, respectively) there is little difference in run time. Similarly, when we run just *thread 2* with DFS and then BFS retry block (trials 8 & 9) the BFS retry block takes only slightly longer. The more salient feature is the increased number of failures when using BFS in the retry block (trials 9 & 10). Due to the increase in failures more than anything else I opted to use DFS in the retry block.