Paper Reading

Worst case asymptotic analysis is sometimes insufficient for evaluating the effectiveness of an algorithm. The paper by Sleator and Tarjan uses an alternative method called competitive analysis which compares the performance of an algorithm against the best possible algorithm on the particular input sequence. See http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/video-lectures/lecture-14-competitive-analysis-self-organizing-lists/ for an introduction to competitive analysis and a simpler version of Theorem 1 of Sleator and Tarjan's paper. Some background on amortized analysis from http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/video-lectures/lecture-13-amortized-algorithms-table-doubling-potential-method/ may also be useful.

Amortized Efficiency of List Update and Paging Rules

by Daniel D. Sleator and Robert E. Tarjan

1. What are the two practical problems analyzed in the paper? What are the typical applications where you can find these problems?
2. What does amortized analysis mean?
3. What are the cost of access, insert and delete in the self-organizing list?
4. What are free exchanges and paid exchanges?
5. What do each of the algorithms Move-to-front, Transpose and Frequency Count do?
6. What does worse-case analysis tell us about the performance of the algorithms?
7. What do previous average-case analysis indicate on the relative performance of the algorithms? Does it agree with experimental results?
8. Explain Theorem 1 in your own words.
9. Study the proof of Theorem 1. Note the use of potential functions in the proof.
10. Consider the result. What happens when the sequence is 'easy', in the sense that an optimal algorithm has O(m) performance? What happens when the sequence is hard, in the sense that an optimal algorithm has O(mn) performance?
11. Similar results are not possible with Transpose or Frequency Count. What are the counterexamples?
12. Consider the definition of convexity given in the paper. Is the cost function used for Theorem 1 convex?
13. Is the cost function for paging convex?
14. What are the common paging algorithms?
15. What is the optimal paging algorithm?
16. Note that you are comparing the performance of an algorithm with the optimal algorithm that has a smaller memory.
17. Interpret Theorem 5 and study the proof.
18. Give an example of a sequence that satisfy the remark following Theorem 5.
19. Interpret Theorem 6 and study the proof.
20. Are the results in the paper useful? How are they useful?
21. Are they important?
22. Are they surprising?

Pay particular attention to the abstractions used to analyse the problems. How realistic are the abstractions? Do they capture useful insights about the problems? Do they leave out important details?