Algorithms and Data Structures

Charles A. Wuethrich

Bauhaus-University Weimar - CogVis/MMC

June 26, 2019

NP-complete problems

- NP-complete problems
- Problem reduction
- Cook's theorem

Exhaustive search

- Many problems require to examine a great number of potential solutions
- Note that this kind of problems usually requires exponential times
- Examples: exploring alternatives (chess)
- Here strategies like tree pruning are important

Exhaustive search

- Another typical problem like this is the travelling salesman problem
- Here of course we have to search exhaustively all possible paths
- The problem becomes to search a hamilton path in a graph: a no-cycle path that covers all nodes, starts and ends at the same place
- Exhaustive search over all paths=inefficient

NP-complete problems

- Not at all problems can be efficiently solved
- Often small condition changes everything
- Example:
  - find shortest path between two nodes of a weighted graph (we solved in O((E+V)logV))
  - finding longest path instead requires to explore ALL possible paths (this includes exploring ALL paths)

NP-complete problems

- Theoreticians subdivide into the following categories:
- P: All problems ST can be solved with a deterministic algorithm in polynomial time
  - deterministic: at each point the algo can do only one thing as next step
- Ex: Sorting belongs to P, since for example, insertion sorting works in N^2 and is deterministic.

NP-complete problems

- Some problems are such that we do not know if they are solvable in a deterministic polynomial time
- To solve a problem, one can expand the class of "algorithms" by adding non-determinism to them.
- This basically means, that the algorithm is such that it is capable of guessing the right solution to a problem.
- Once the solution is guessed, one would need a mechanism to prove that the solution found is correct.
- Theoreticians define

NP: All problems solvable in polynomial time with nondeterministic algorithms

NP-complete problems

- Clearly, $P \subseteq NP$, so every polynomial problem does also belong to NP
- It would make sense to conjecture that there are problems which belong to NP but not to P
  - The proof that a problem is NP is given by the fact that there is a polynomial algorithm capable of testing if a solution is correct or not
  - For example, the problem of the longest path in a graph is surely NP, because given a path one can check in polynomial time if it is the longest one or not
  - Another example is the decidibility of a logical expression: given a logical formula, with boolean variables (TRUE, FALSE) and operations (AND, OR, NOT), is there an assignment of the variables such that the expression is true? $(x_1 \vee x_3 \vee x_5) \wedge (x_1 \vee \neg x_2 \vee x_4) \wedge (\neg x_3 \vee x_4 \vee x_5) \wedge (x_2 \vee \neg x_3 \vee x_5)$

NP-complete problems

- Non determinismus seems to be a powerful tool for delivering a solution of a problem.
- However, it has not yet been proven, that it helps in any sense.
- In other words, no-one ever found a problem belonging to NP but not belonging to P
- Or conversely (negating the hypotheses), there has not been yet the proof that P=NP.
- Moreover, if one NP problem can be solved with a det. automata, then all of them can, and therefore P=NP.
- The problems of which we are sure that they are NP but not if they are P are called NP-complete problems

NP-completeness: reduction

- It is possible to change one problem into an equivalent one, and if this is possible in polynomial time the operation is called polynomial reducibility
- This allows to group problems in groups of equiv. probs. WRT computational time
- Example:
  - Travelling salesman problem (TSP): given a set of cities and of distances between city pairs, find a path through all cities that is shorter than M
  - Hamilton cycle (HC): find a simple cyclic path in a graph containing all nodes
- Reduction from HC to TSP can be done as follows: use as cities the nodes of a graph, and as distances 1 if there is an arc in the graph, and 2 if there is none. Solve then TSP with M=number of nodes in the graph

NP-completeness: reduction

- Curiously, all NP complete problems are mutually reducible.
- So, for example, the HC problem can be polynomially reduced into the decidibility of a logical expression
- Which, in turn means, that the dedicibility of a logical expression can be polynomially reduced to the TSP problem, since it can be polynomially reduced to HC which in turn can be polynomially reduced to TSP
- Concluding, polynomial reduction is used to prove the NP completeness of a problem

Cooks's Theorem

- Cook proved in 1971 that the decidibility problem is not polynomially reducible and NP complete
- He also proved that the decidibility problem is NP complete
  - This he did with aid of the def. of a computer of computers (Turing machine) capable of doing all possible computations, and by adding non-determinism to it.
  - by adding non determinism to a Turing machine, a turing machine can solve every NP problem
  - He then defined a non-deterministic turing machine able of working on the logical formulas of the decidibility problem
  - Therefore, the solution of the decidibility problem is basically "pushing a button on the ND-Turing machine" after feeding it with a formula
- Through the turing machine he basically defined Polynomial reduction

NP-complete problems

- Interesting is to know which problems are NP-complete:
  - Decidibility of a logical expression
  - Travelling salesman
  - Hamilton cycle
  - Longest path in a graph
  - Partition pb.: set of integers: are they partitionable in two sets of equal sum?
  - Integer sol. of linear programming pb.: is there an integer solution to it?
  - Given a stop time and a set of problems to solve of different complexity, can one schedule two processors, so that all problems are solved before the end before the stop time?
- ... and many others

NP-complete problems

- If a solution for one could be found, than the P=NP problem could be solved
- However, this is not the case (to date)
- All at conjecture level
- Not finding a reasonable solution leads to the suspect for a problem to be NP-complete
- Finally, partial solutions are often possible for NP problems

A Fine Video

- Credit: hackerdashery on youtube
- https://complexityzoo.uwaterloo.ca/Complexity_Zoo

Finis

- This concludes the course: I hope you enjoyed it, as I did
- Good luck and work well for the exam