The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual
Chapter 6, Problem 6.27

It suffices to show that Turing reductions are no more powerful within P than many-one reductions; thus, if they are more powerful within NP, it must be the case that P and NP are different.

Assume then that problem B in P Turing-reduces in polynomial time to problem A in P; we want to show that B also many-one reduces to A in polynomial time. But that is trivially easy. Let x be a "no" instance of A and y be a "yes" instance of A; given an instance z of B, our mapping f begins by solving z (using the polynomial-time algorithm for B, which must exist since B is in P), then, if the answer is yes, maps z to y and otherwise maps z to x. The mapping takes polynomial time (all it requires is the running time of the solution algorithm for B) and is a valid many-one reduction from A to B. Hence Turing reductions are no more powerful than many-one reductions within P.