The Theory of Computation
Bernard M.E. Moret
Addison-Wesley-Longman, 1998

Solution Manual
Chapter 6, Problem 6.28

Intuitively, NP is closed under conjunctive polynomial-time reductions because requiring "yes" answers to a fixed pattern of calls to an oracle is not that different from requiring a "yes" answer to a single call, at least not where nondeterminism is concerned—we are basically just adding a collection of existential quantifiers, but such a collection can easily be condensed down to a single existential quantifier.

More formally, assume that problem A is in NP and that we have a conjunctive polynomial-time reduction from some (decision) problem B to A. We need to prove that B is then also in NP. Let x be some "yes" instance of B and assume that the reduction makes k calls (where k is no larger than some polynomial function of $|x|$) to the oracle for A. Each call may use a somewhat different mapping of x, so let us denote the values passed to the oracle for A as $f_1(x)$, dots, $f_k(x)$. For each $f_i(x)$, the oracle for A answers "yes," meaning that there is a certificate c_i for each such value that can be verified in polynomial time. But then x, the "yes" instance of B, has a certificate as well, the collection $c_1\#c_2\#\ldots\#c_k$, and a polynomial-time checker, basically the checker for A embedded within a loop that runs it on each of the k pieces of the certificate and verifies that all answers are "yes." This checker cannot accept a "no" instance of B, because such an instance would give rise to at least one $f_i(x)$ that has no certificate acceptable to the checker for A—and thus our checker for B will reject at least one piece of any certificate proposed for the "no" instance of B.

Note that the restriction to a conjunction of acceptances is not the only one that will keep NP closed under the reduction. Requiring a given style of majority (of "yes" answers) would be another one; many others are possible. However, note that we cannot in general define a polynomial-time truth-table reduction, for two reasons. First, specifying the truth table for a polynomial (or even linear) number of variables (the calls to the oracle) may take exponential space! Secondly, the idea used above will only work for certificates and thus for "yes" answers in the calls to the oracle; it is thus essential that the "no" answers returned by the oracle may be safely discarded.