CS 440 WA 3 Answers
Ryan Petrat

1. a. If, $\alpha \leq v \leq \beta$, then $v' = v$:

   Consider the statement, $\alpha \leq v \leq \beta$, This means that the value v falls within the bounds set by α and β. In the context of the minimax algorithm, v represents the value of the current node being evaluated, which is the best value found so far for the player at that level. Thus, it implies that the value v has not caused any pruning to occur. This is because if v were less than α or greater than β, pruning would have occurred. Since no pruning has occurred and alpha-beta pruning with no pruning is just the minimax algorithm then both v and v' are both the same. Hence, the statement is true.

   b. If, v $\leq \alpha$, then $v' \leq \alpha$:

   Given that, $v' \leq \alpha$, this indicates that the subtree rooted at the current node has been pruned as anything less than the most optimally maximized value (for the maximizing player) will be pruned therefore the subtree rooted at the current node cannot produce a higher result than alpha and is pruned. Thus, if $v \leq \alpha$, then the same must also be true as it would be comprised of the same subtree as v', with the exception of the branches that are pruned as they were less than α for the previous layers, and through the minimax algorithm would generate the same value for v as for v'. Since alpha-beta pruning will only remove branches that are less than the already determined maximum value for α, for the purpose of improving runtime, there is no way that there is some value $v' \geq alpha$. Hence, the statement is true.

   c. If, $v \geq \beta$, then $v' \geq \beta$:

   Given that, $v' \geq beta$, this indicates that the subtree rooted at the current node has been pruned as anything greater than the most optimally minimized value (for the mimimizing player) will be pruned therefore the subtree rooted at the current node cannot produce a lower result than beta and is pruned. Thus, if $v \geq \beta$, then the same must also be true as it would be comprised of the same subtree as v', with the exception of the branches that are pruned as they were more than β for the previous layers, and through the minimax algorithm would generate the same value for v as for v'. Since alpha-beta pruning will only remove branches that are more than the already determined minimum value for β, for the purpose of improving runtime, there is no way that there is some value $v' \leq \beta$. Hence, the statement is true.

2. To prove that any n-ary constraint can be converted into a set of binary constraints we can consider a constraint involving n variables $\{X_1, X_2, ..., X_n\}$ and a function f which takes in the random variables $f\{X_1, X_2, ..., X_n\}$, we can convert this n-ary constraint into a set of binary constraints by introducing $n-1$ synthetic variables, with each synthetic variable Y_i having a domain of the Cartesian product of the domains of the original variables in the constraint. for example if we have a function constraint of $f\{X_1, X_2, X_3\}$ we can replace the 3 constraint with a set of binary constraints $Y_1 = \{X_1, X_2\}$, $Y_2 = \{Y_1, X_3\}$. Thus, these binary constraints ensure that the values of the synthetic variables Y_1 and Y_2 correspond to the values that satisfy the original constraint. Therefor any n-ary constraint can be converted into a set of binary constraints by introducing synthetic variables, and as a result, all CSPs can be converted into binary CSPs.