Analysis of Iterative Deepening

To talk about iterative deepening and other search algorithms, I will use the following notation and assumptions:

The search graph is a uniform search tree with the initial state as the root and with branching factor \( b, b \geq 2 \). By uniform, I mean that each non-leaf state has \( b \) children.

\( m \) is the depth of the search tree from the initial state. I.e., each state on levels 0 through \( m - 1 \) have \( b \) children. Each state on level \( m \) has 0 children.

There is a goal state within distance \( d \) of the initial state.

\( b, m, \) and \( d \) are integers. From these assumptions, the following can be demonstrated.

**Theorem 1** For each integer \( l, 0 \leq l \leq m \), there are \( b^l \) states on level \( l \) of the search tree.

**Proof:** By mathematical induction.

- **Basis:** There is one state on level 0 because level 0 consists only of the initial state. Note that \( 1 = b^0 \).
- **Assumption:** Let \( l \) be any integer between 0 and \( m - 1 \). Suppose that level \( l \) of the search tree contains \( b^l \) states.
- **Induction:** Because \( b \) is the branching factor, each state on level \( l \) has \( b \) children on level \( l + 1 \). Thus, there are \( b \cdot b^l = b^{l+1} \) states on level \( l + 1 \).

So, by mathematical induction, the theorem is proved.

End Proof.

**Theorem 2** Let \( l \) be any integer between 0 and \( m \). The number of states from level 0 to level \( l \) is:

\[
\frac{b^{l+1} - 1}{b - 1}
\]

**Proof:** The number of states on all the levels from level 0 to level \( l \) is:

\[
\sum_{i=0}^{l} b^i = \frac{(b - 1) (\sum_{i=0}^{l} b^i)}{b - 1} = \frac{\sum_{i=0}^{l} (b^{i+1} - b^i)}{b - 1} = \frac{b^{l+1} - 1}{b - 1}
\]

End Proof.

**Corollary 3** The number of states visited by depth-first search is at most:

\[
\frac{b^{m+1} - 1}{b - 1}
\]

**Corollary 4** The number of states visited by breadth-first search is at most:

\[
\frac{b^{d+1} - 1}{b - 1}
\]
**Corollary 5** *The number of states visited by depth-first search with depth limit* \(l\) *is at most:*

\[
\frac{b^{l+1} - 1}{b - 1}
\]

**Theorem 6** *The number of states visited by iterative deepening is at most:*

\[
\frac{b^{d+2}}{(b - 1)(b - 1)}
\]

**Proof:** Iterative deepening performs \(d + 1\) depth-limited depth-first searches, from depth limit 0 to depth limit \(d\), so the number of state visits is at most:

\[
\sum_{l=0}^{d} \frac{b^{l+1} - 1}{b - 1} < \sum_{l=0}^{d} \frac{b^{l+1}}{b - 1} = \frac{\sum_{l=0}^{d} b^{l+1}}{b - 1}
\]

Considering the numerator:

\[
\sum_{l=0}^{d} b^{l+1} = \sum_{l=1}^{d+1} b^l < \sum_{l=0}^{d+1} b^l = \frac{b^{d+2} - 1}{b - 1} < \frac{b^{d+2}}{b - 1}
\]

Thus,

\[
\sum_{l=0}^{d} \frac{b^{l+1} - 1}{b - 1} < \frac{\sum_{l=0}^{d} b^{l+1}}{b - 1} < \frac{b^{d+2}}{(b - 1)(b - 1)}
\]

which proves the theorem.

**End Proof.**

Note that the ratio of the iterative deepening bound over the breadth-first search bound is approximately \(b/(b - 1)\), which means that iterative deepening does not search much more than breadth-first search, while using much less memory. Iterative deepening requires memory for \(O(bd)\) states, while breadth-first search requires memory for \(O(b^d)\) states.

The weakness in the above analysis is the assumption that the search graph is a tree. What will be the result if there are significant numbers of repeated states over different branches of the search graph?