The language consists of a^n with $n \geq 3$ and $a^k baa$ with $k \geq 0$. This gives an independent semantic description. Four states count a pure-a prefix up to three; three states record that a b has occurred and that zero, one or two required final a symbols have followed; one dead state handles any forbidden continuation. These eight states correspond to the eight DFA rows. Distinct pure-prefix states are separated by the number of a symbols needed to reach length three, except that appending a b and then aa distinguishes a pure prefix from a post-b state. Among post-b states, suffixes aa, a, ε distinguish the three remaining lengths. The dead state accepts no suffix. This supplies a distinguishing-suffix proof of minimality.

Comparative cost and preferred method

The fragment NFA has nine states; the total subset DFA has eight. Simulation of the DFA requires one transition per input symbol, hence $\Theta(m)$ time and constant state memory for a length-m string. A generic subset construction can take exponential preprocessing, so its advantage is not free. Counting words through a matrix resolvent is an optional different output; it does not replace the required automata.