Assign the High (H) and Low (L) threshold values of the five-branches comb for the five DVFI groups of P packets:

- \( th_H[1,2,3,4,5] = \{25, 30, 35, 40, 45\} \)
- \( th_L[1,2,3,4,5] = \{10, 17, 24, 31, 38\} \)

while (a video packet arrives)

```c
if (packet_type == 'I')
/* Merged thresholds of the Step Function for I packets. */
    threshold_high = maximum_AC[2].buffer_size - 1
    threshold_low = maximum_AC[2].buffer_size - 1
else if (packet_type == 'P')
/* high and low thresholds according to DVFI grouping. */
    switch DVFI {
        case(DVFI belongs to group_1):
            threshold_high = th_H[1]
            threshold_low = th_L[1]
        case(DVFI belongs to group_2):
            threshold_high = th_H[2]
            threshold_low = th_L[2]
        case(DVFI belongs to group_3):
            threshold_high = th_H[3]
            threshold_low = th_L[3]
        case(DVFI belongs to group_4):
            threshold_high = th_H[4]
            threshold_low = th_L[4]
        case(DVFI belongs to group_5):
            threshold_high = th_H[5]
            threshold_low = th_L[5]
    }
end
DVFI_CQM(packet_type, threshold_low, threshold_high)
}
void DVFI_CQM(packet_type, threshold_low, threshold_high) {
    if (length(AC[2]) < threshold_low)
        video packet \rightarrow AC[2]
    elseif (length(AC[2]) < threshold_high)
        if (packet_type = 'I')
            video packet \rightarrow AC[2]
        else
            Prob_transition = \left(\frac{\text{length(AC[2]) - threshold_low}}{\text{threshold_high - threshold_low}}\right)^2
            rn = uniform(0,1)
            if (rn > \text{prob_transition})
                video packet \rightarrow AC[2]
            else
                Min_Delay()
        end
    else
        Min_Delay()
    end
else
    Min_Delay()
end
}
void Min_Delay() {
    if (AC[0] has a shorter queue delay than AC[1])
        video packet \rightarrow AC[0]
    else
        video packet \rightarrow AC[1]
end
```