If m keep hash to a 2^{nd}-level table:

$E[\#\text{collisions}] = \sum_{x \neq y} \Pr[x \& y \text{ collide}] = \binom{m}{2} \cdot \frac{1}{m^2} < \frac{1}{2}$.

$\Pr[\#\text{collisions} \geq 1] < 1/2$.

So after $\Theta(\lg m)$ attempts, find 2^{nd}-level hash avoiding collisions.

$E[\text{size}] = E\left[\sum_{i=0}^{m-1} n_i^2\right] = n + 2E\left[\sum_{i=0}^{m-1} \binom{n_i}{2}\right] = n + 2E[\#\text{collisions @ } 1^{ST} \text{ level}]$

$= n + 2\binom{n}{2} \cdot \frac{1}{n} < 2n$.

$\Pr[\text{size} \geq 4n] < 1/2$

Dynamic: (doubling trick) build structure for $2n^*$ elements

if collision: rebuild entirely (ditto if too much space) during build/inserts

if n doubles: rebuild for $4n^*$.

~~if n halves ($n^*/2$)~~

if $n = n^*/4$: rebuild for n^*. ← keep space down

$E[\text{cost of rebuild}] = O(n)$

amortized cost $= O(1)$.

Search takes $O(1)$ worst-case.