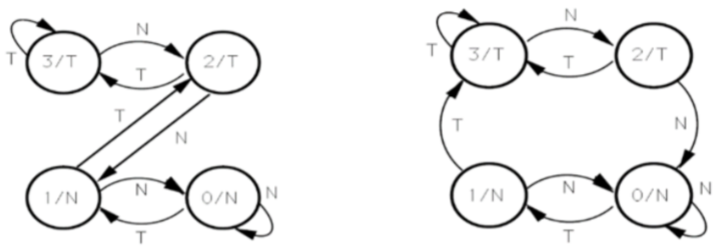
Comparison of 2 bit Branch Predictors

1 Automata used for branch prediction

A2, A3



(a) A2 (Original zsim Implementation) (b) A3 (Same as shown in the textbook)

1.1 A2

A2 is the classic **2-bit branch predictor** (bimodal 2-bit counter). Each static branch has a 2-bit finite-state machine (FSM) that resists flipping predictions too easily:

- Needs two consecutive wrong outcomes to change direction.
- Adapts slowly but is very stable for loops (often only mispredicts on loop exit).

1.2 A3

A3 is a more aggressive 2-bit FSM variant:

- Reacts to new patterns faster than A2 (fewer/shorter "weak" states).
- Performs well under phase changes or alternating behavior.
- Less stable for steady loops (can over-react).

A3 reaches *Taken* after fewer steps (e.g., $00 \rightarrow 01 \rightarrow 11$), adapting quickly but losing stability.

2 Benchmarks

Benchmark	Type of workload

blackscholes	Floating-point intensive
bodytrack	Branch-heavy, irregular
canneal	Integer + memory intensive
fluidanimate	Loop-heavy
freqmine	Data mining, conditional branches
streamcluster	Memory and loop-heavy
swaptions	Loop-heavy, predictable
x264	Branch-heavy, mixed predictability

3 Equations

CPI per Core

$$\mathrm{CPI}_{\mathrm{core}} = \frac{\mathrm{cycles} + \mathrm{cCycles}}{\mathrm{instrs}}$$

Final CPI (Arithmetic Mean)

$$\mathrm{CPI}_{\mathrm{final}} = \frac{\mathrm{CPI}_1 + \mathrm{CPI}_2 + \mathrm{CPI}_3 + \cdots + \mathrm{CPI}_8}{8}$$

Misprediction Rate per Core

$$MR(\%) = \frac{\mathrm{mispredBranches}}{\mathrm{condBranches}} \times 100$$

Final Misprediction Rate (Geometric Mean)

$$MR_{\mathrm{final}} = (MR_1 \times MR_2 \times MR_3 \times \cdots \times MR_8)^{\frac{1}{8}}$$

$$\mathrm{Accuracy}(\%) = 100 - MR_{\mathrm{final}}(\%)$$

4 Summary tables

A2

Benchmark	**CPI**	**MR(%)**	**Accuracy(%)**
Blackscholes	0.782	0.795	99.205
Bodytrack	0.603	1.533	98.467
Canneal	2.547	6.418	93.582
Fluidanimate	0.712	4.613	95.387
Freqmine	0.769	1.407	98.593
Streamcluster	0.656	0.652	99.348
Swaptions	0.640	2.758	97.242
x264	0.580	12.770	87.230

A3

Benchmark	**CPI**	**MR(%)**	**Accuracy(%)**
Blackscholes	0.783	0.823	99.180
Bodytrack	0.603	1.630	98.370
Canneal	2.977	6.830	93.170
Fluidanimate	0.739	4.770	95.230
Freqmine	0.815	1.480	98.520
Streamcluster	0.656	0.658	99.340
Swaptions	0.641	2.876	97.120
x264	0.583	13.290	86.710

5 Grouped bar charts

CPI comparison

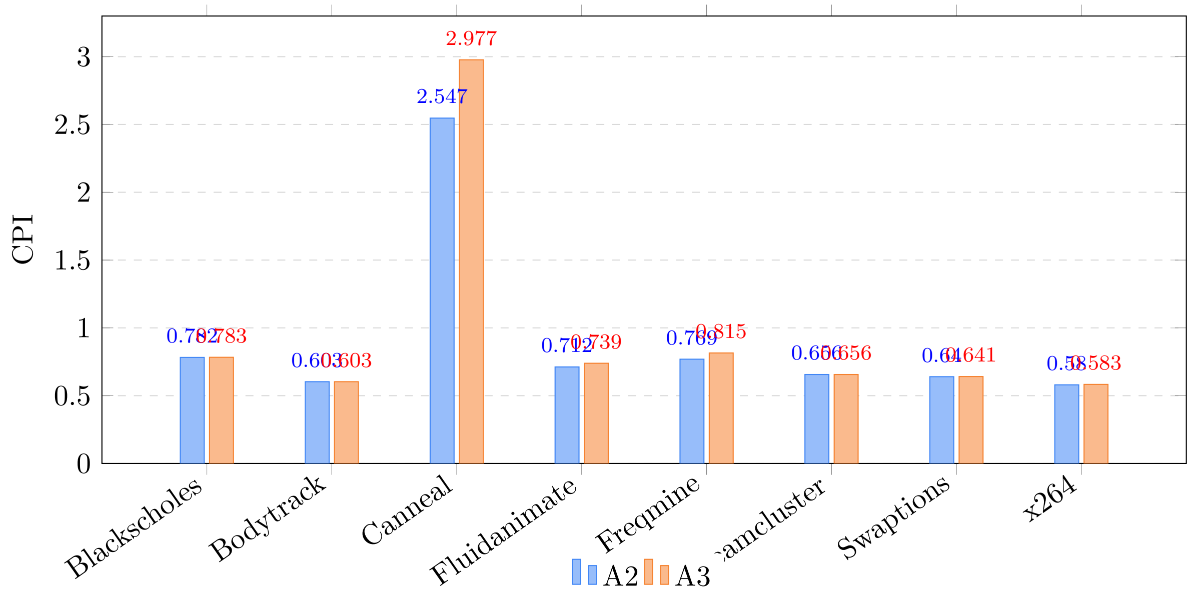


Figure 1: CPI comparison across benchmarks (lower is better).

Misprediction rate (MR%) comparison

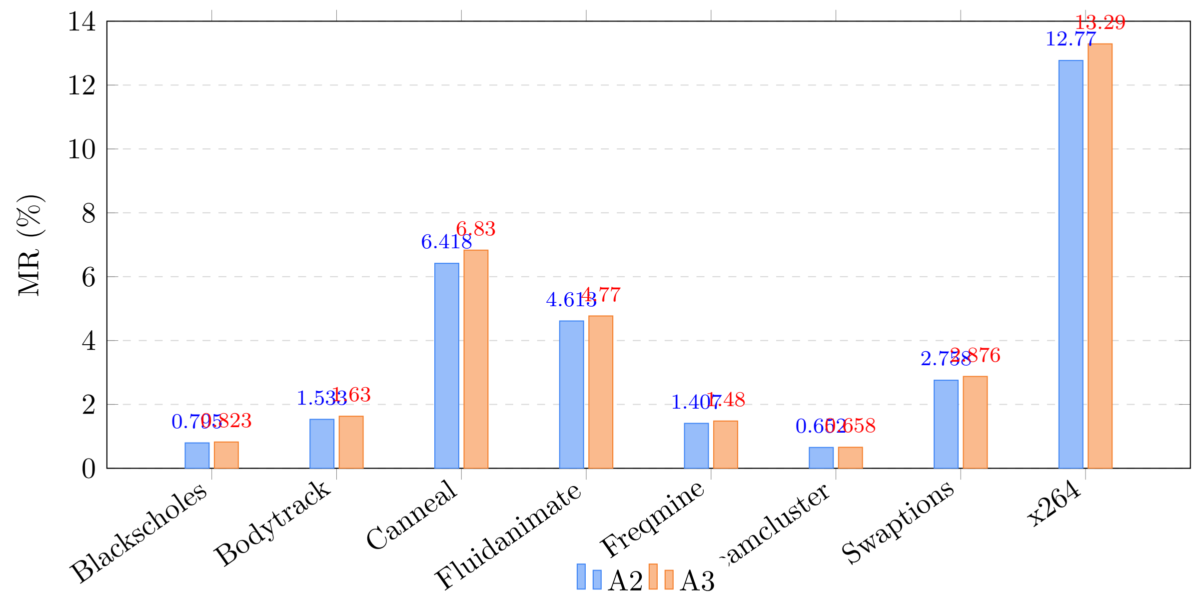


Figure 2: Misprediction rate comparison (lower is better).

Accuracy (%) comparison

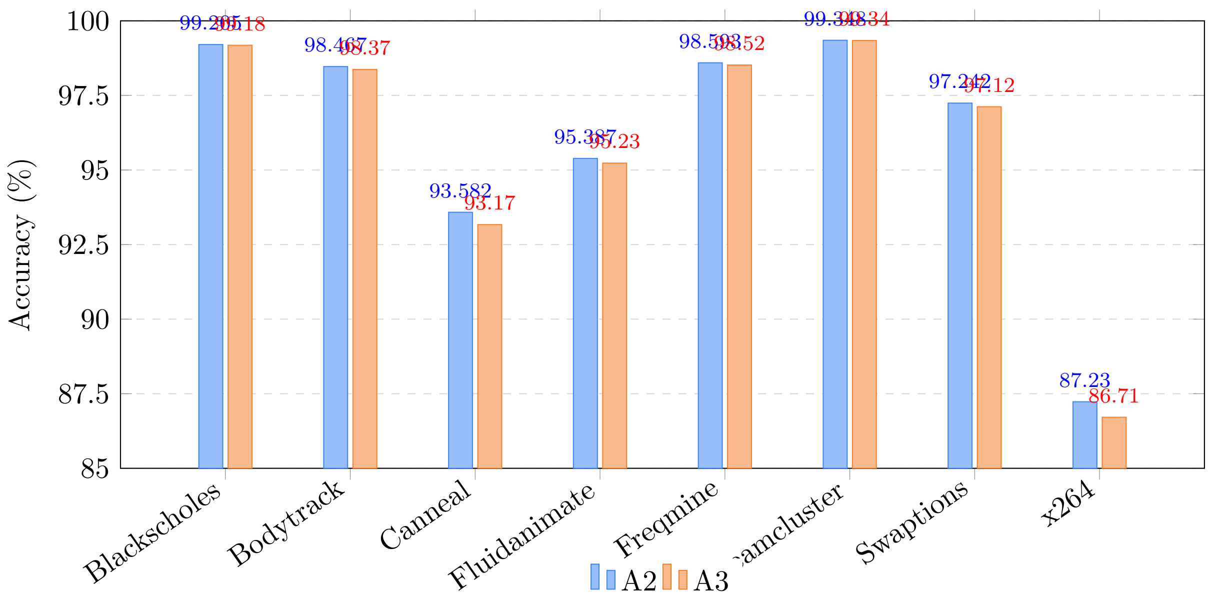


Figure 3: Accuracy comparison (higher is better).

6 Conclusion

Overall, **A2** demonstrates greater stability and performs better on *loop-heavy or predictable workloads* such as *swaptions* and *fluidanimate*, achieving consistently lower misprediction rates (MR%) and slightly lower CPI.

In contrast, **A3** adapts more quickly to *phase-changing or irregular branch patterns*, but tends to overreact on steady behavior, resulting in higher MR% and CPI in benchmarks like *canneal*, *fluidanimate*, and *x264*.

Thus, **A2 is preferable for predictable control flows and stable program behavior**, whereas **A3** offers advantages in workloads with frequent phase changes or irregular branching behavior.