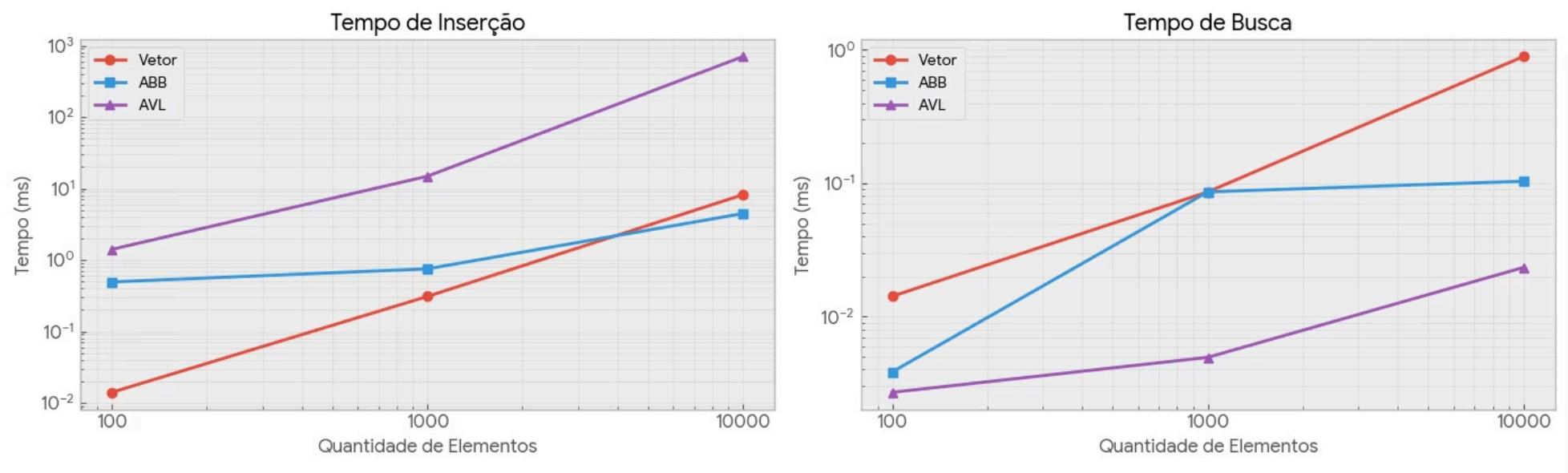




Cenário 1: Dados Aleatórios

Cenário: Aleatoria



No cenário aleatório, considerado o caso "normal" de operação, todas as estruturas apresentam desempenho aceitável. A ABB funciona eficientemente quando os dados não seguem um padrão específico, mantendo-se competitiva com o Vetor em operações de inserção. A AVL, embora mais lenta na inserção devido ao overhead de balanceamento, demonstra superioridade absoluta nas operações de busca.

Tamanho	Estrutura	Inserção (ms)	Busca (ms)
100	Vetor	0,01	0,01
	ABB	0,49	0,004
	AVL	1,41	0,003
1.000	Vetor	0,31	0,09
	ABB	0,75	0,09
	AVL	14,82	0,005
10.000	Vetor	8,17	0,89
	ABB	4,46	0,10
	AVL	700,29	0,02

700ms

Inserção AVL

Tempo máximo para 10.000
elementos

0,02ms

Busca AVL

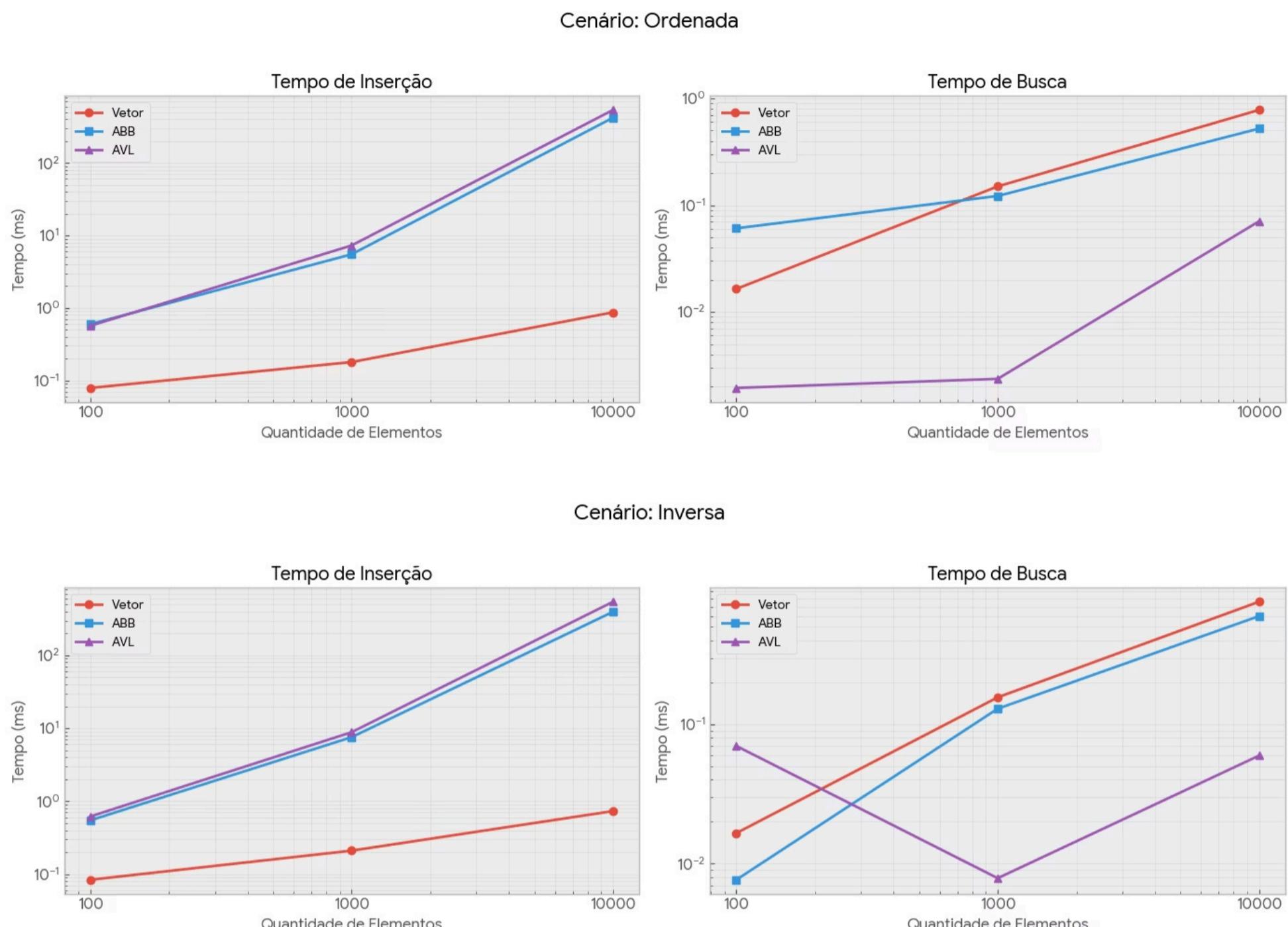
Velocidade superior em grandes
volumes

4,46ms

Inserção ABB

Desempenho equilibrado no cenário
ideal

Cenários 2 e 3: Dados Ordenados e Inversos



Os cenários ordenado e inversamente ordenado revelam o pior caso para a ABB, onde ela degenera em uma estrutura linear semelhante a uma lista encadeada. Com 10.000 elementos ordenados, a ABB apresenta tempo de inserção de 421,04 ms - quase 100 vezes mais lento que no cenário aleatório. No cenário inverso, o comportamento é igualmente catastrófico com 396,03 ms. A AVL, por outro lado, mantém desempenho consistente através do balanceamento automático, provando seu valor em aplicações críticas.

Cenário Ordenado

Tamanho	ABB Inserção	AVL Inserção
100	0,60 ms	0,57 ms
1.000	5,50 ms	7,28 ms
10.000	421,04 ms	538,75 ms

Cenário Inverso

Tamanho	ABB Inserção	AVL Inserção
100	0,55 ms	0,62 ms
1.000	7,56 ms	8,85 ms
10.000	396,03 ms	543,92 ms

Degradação da ABB

Em dados ordenados, a ABB perde até 94% de eficiência, transformando-se em uma estrutura linear com complexidade $O(n)$ ao invés de $O(\log n)$.

Estabilidade da AVL

O balanceamento automático garante que a AVL mantenha altura logarítmica, preservando desempenho previsível independente da ordem de inserção.

Busca Consistente

Mesmo nos piores cenários, a AVL mantém tempos de busca inferiores a 0,07 ms para 10.000 elementos, demonstrando superioridade absoluta.

Conclusões e Recomendações

Vetor: Simplicidade

Ideal para conjuntos pequenos (≤ 100 elementos) onde a simplicidade de implementação supera a necessidade de otimização. Inserção rápida, mas busca linear.

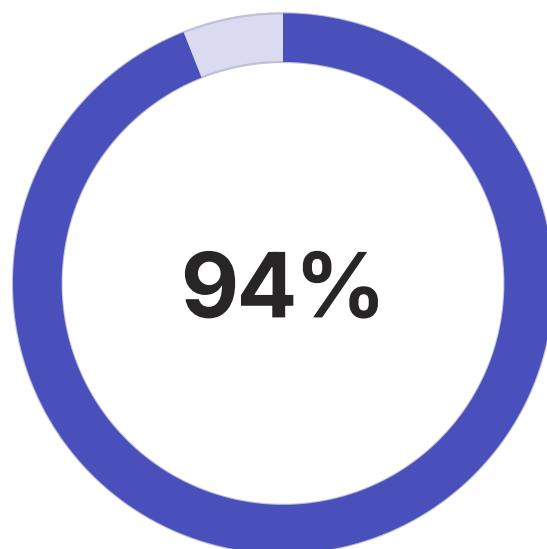
ABB: Uso Limitado

Adequada apenas quando há garantia de dados aleatórios. Vulnerável a degradação severa em dados ordenados, tornando-se impraticável para aplicações críticas.

AVL: Confiabilidade

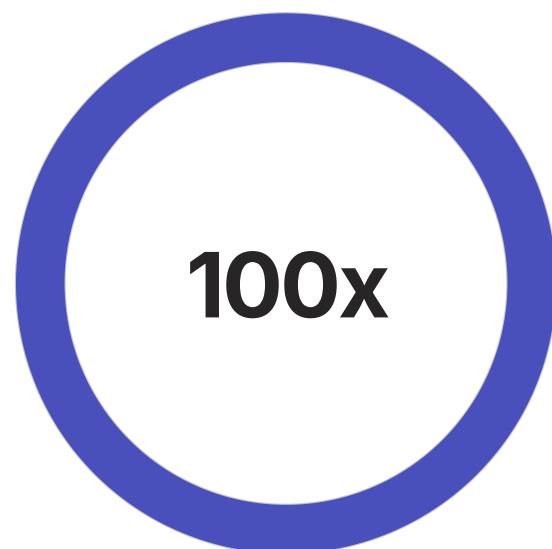
Recomendada para aplicações que exigem desempenho previsível e buscas frequentes. O overhead de balanceamento é compensado pela estabilidade garantida.

A análise comparativa demonstra que a escolha da estrutura de dados deve considerar não apenas o desempenho médio, mas também o comportamento no pior caso. Para sistemas de produção onde a ordem dos dados não pode ser controlada, a AVL oferece garantias de desempenho que justificam sua complexidade adicional. Os gráficos em escala logarítmica revelam diferenças de até 100x no desempenho, evidenciando que decisões arquiteturais aparentemente simples podem ter impacto dramático na escalabilidade do sistema.



Perda de Eficiência

Degradação máxima da ABB em dados ordenados



Diferença de Velocidade

AVL vs ABB em buscas com dados ordenados