Blockchain consensus technologies today sit at the two opposite ends of the scalability spectrum. PoW offers good node scalability with poor performance while BFT offers good performance with limitied scalability.

	PoW consensus	BFT consensus
Node identity management	**open, entirely decentralized**	permissioned, nodes need to know IDs of all other nodes
Consensus finality	no	**yes**
Scalability (no. of nodes)	**excellent (thousands of nodes)**	limited, not well explored (tested only up to $n \leq 20$ nodes)
Scalability (no. of clients)	**excellent (thousands of clients)**	**excellent (thousands of clients)**
Performance (throughput)	limited (due to possible of chain forks)	**excellent (tens of thousands tx/sec)**
Performance (latency)	high latency (due to multi-block confirmations)	**excellent (matches network latency)**
Power consumption	very poor (PoW wastes energy)	**good**
Tolerated power of an adversary	$\leq 25\%$ computing power	$\leq 33\%$ voting power
Network synchrony assumptions	physical clock timestamps (e.g., for block validity)	**none for consensus safety** (synchrony needed for liveness)
Correctness proofs	no	**yes**

The article goes into details of each row in the table above.
The conclusion that further work will be interesting, making the table more precise.
There is a potential of combining the best-of-both of both consensuses for adversarial and network models.

Improving blockchain scalability: GHOST (Greedy Heaviest-Observed Sub-Tree)
Scaling: blockDAG vs. linear.
Eliminating communication and resource overhead in BFT protocols: optimistic BFT. XFT, cross fault tolerance. Randomized BFT and Mixing PoW and BFT.

“New” (1999) state-machine replication algorithm that is able to tolerate Byzantine faults. Works in asynchronous environments, such as the internet. NFS, network file system.