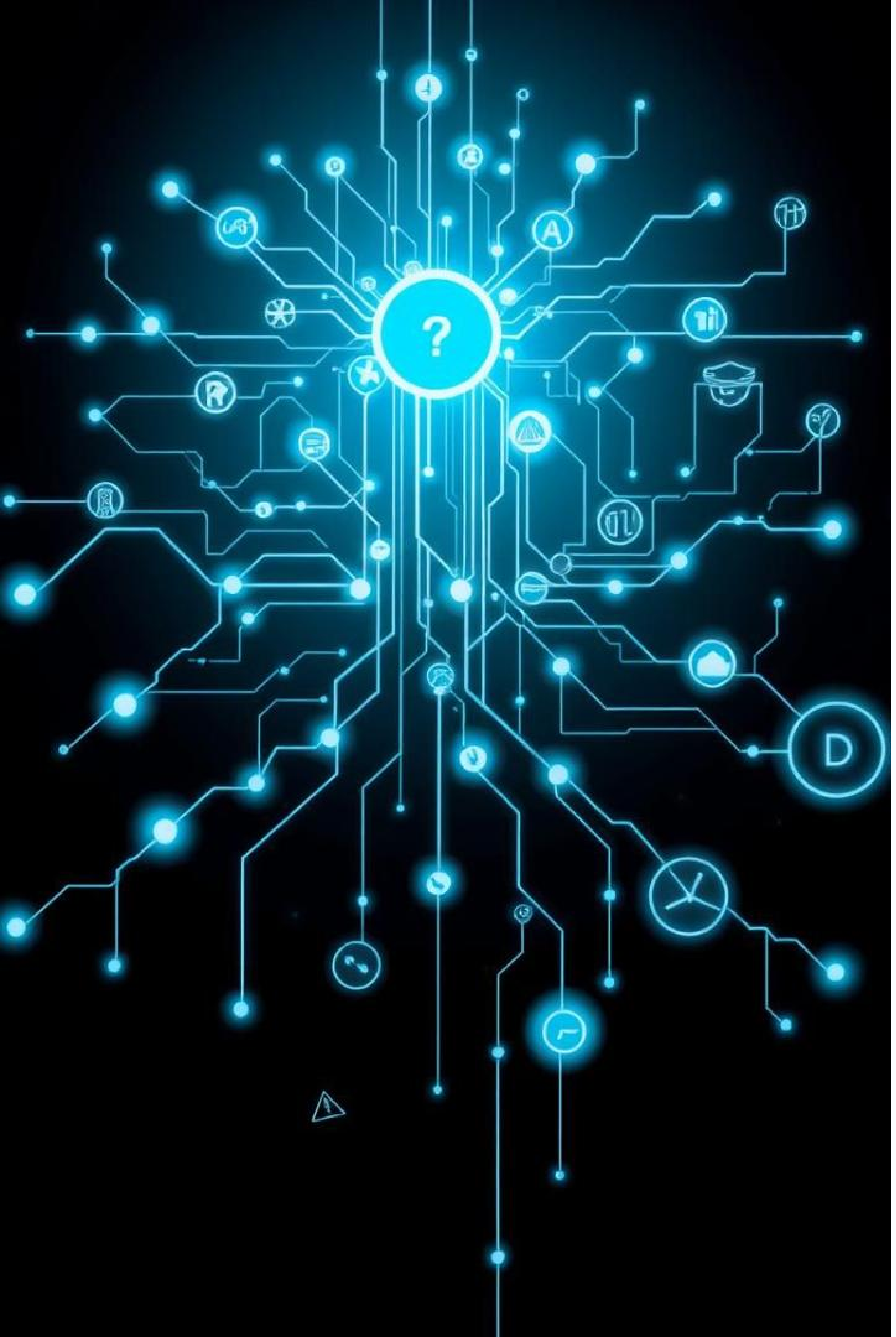


The Theoretical Foundations of Computing

This presentation explores the bedrock principles of computer science. We'll journey from abstract models to real-world applications. The aim is to understand the limits and capabilities of computation itself.

A History: From Abacus to Turing

1 **Abacus (2700 BC)**

Early mechanical aid for arithmetic.

2 **Babbage's Engine (1837)**

Conceptual mechanical computer.

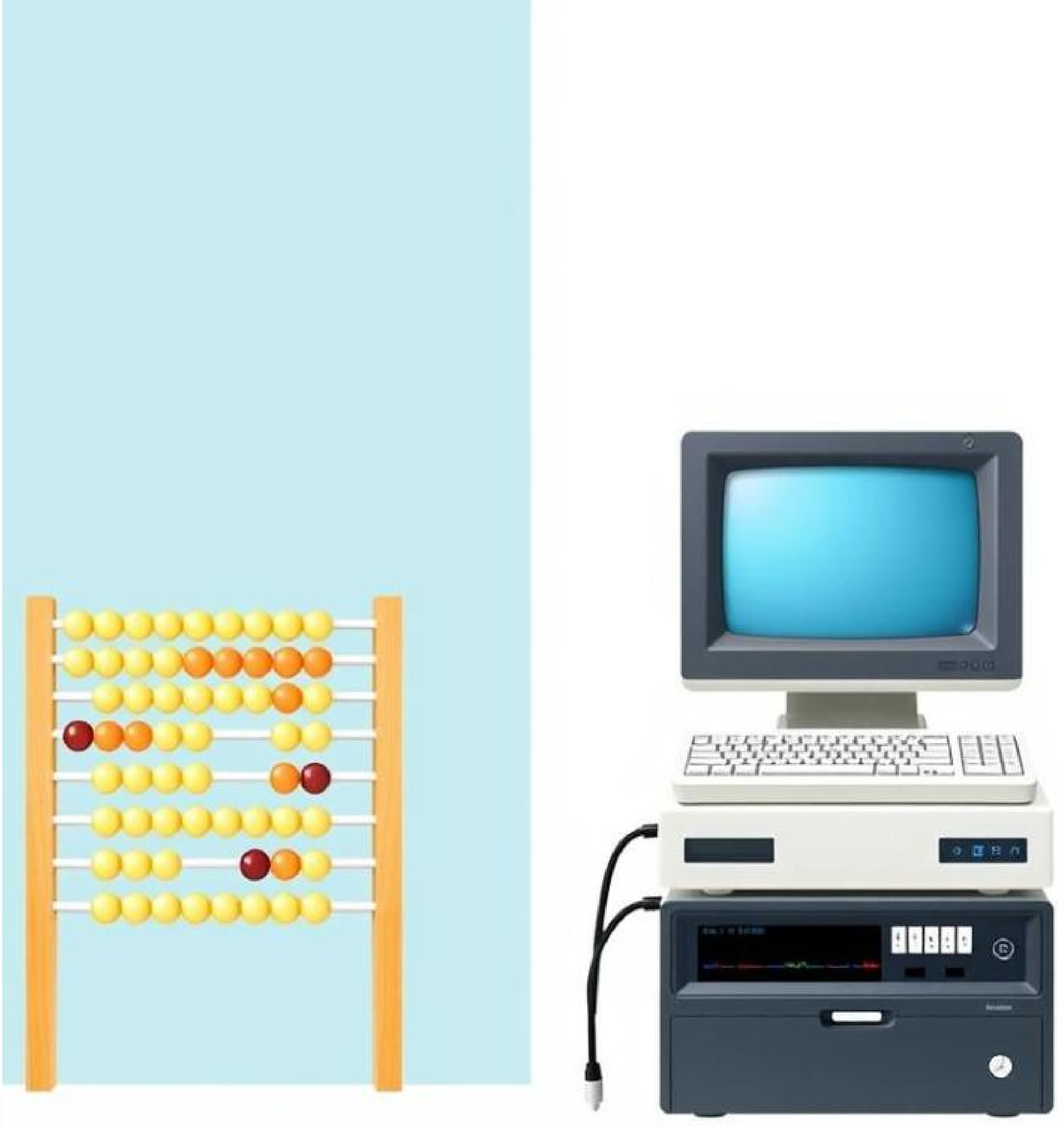
3 **Turing Machine (1936)**

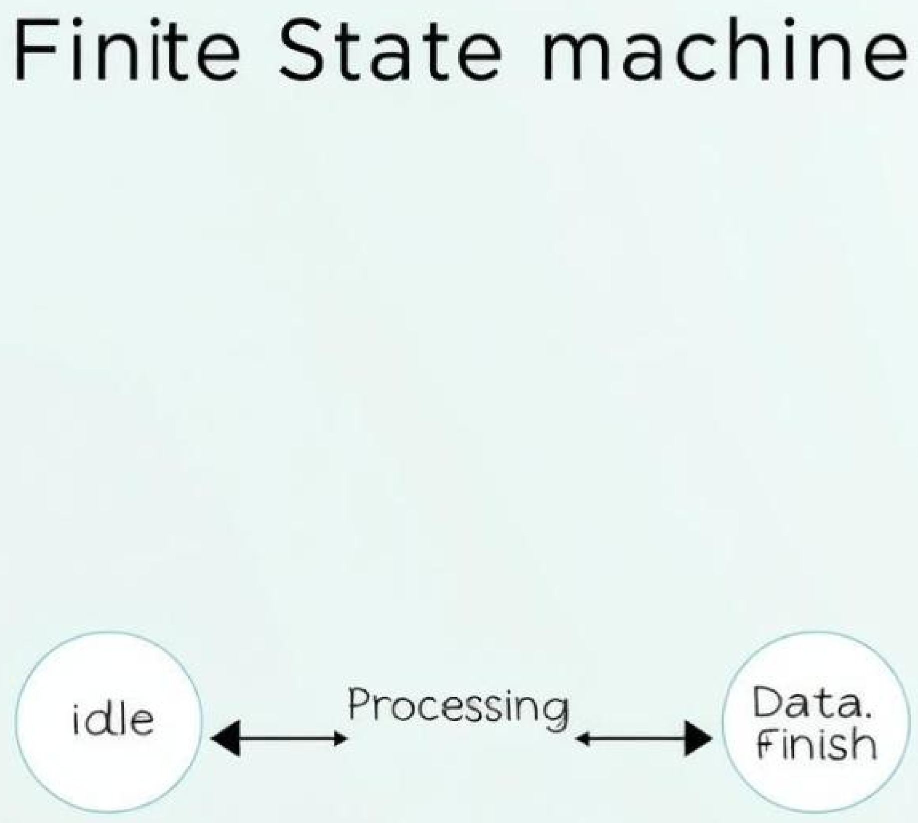
Defined computation limits.

4 **Shannon (1948)**

Mathematical theory of communication.

The abacus marked an early step. Babbage's Engine, though unbuilt, was revolutionary. Turing defined computability. Shannon defined information theory.





Automata Theory: Machines that Compute

Finite Automata (FA)

Pattern recognition and lexical analysis.

Pushdown Automata (PDA)

Parsing context-free languages.

Turing Machines (TM)

Simulate any algorithm.

Automata theory studies abstract machines. Finite automata recognize patterns. Pushdown automata parse languages. Turing machines model computation.

Computability Theory: What Can Be Computed?

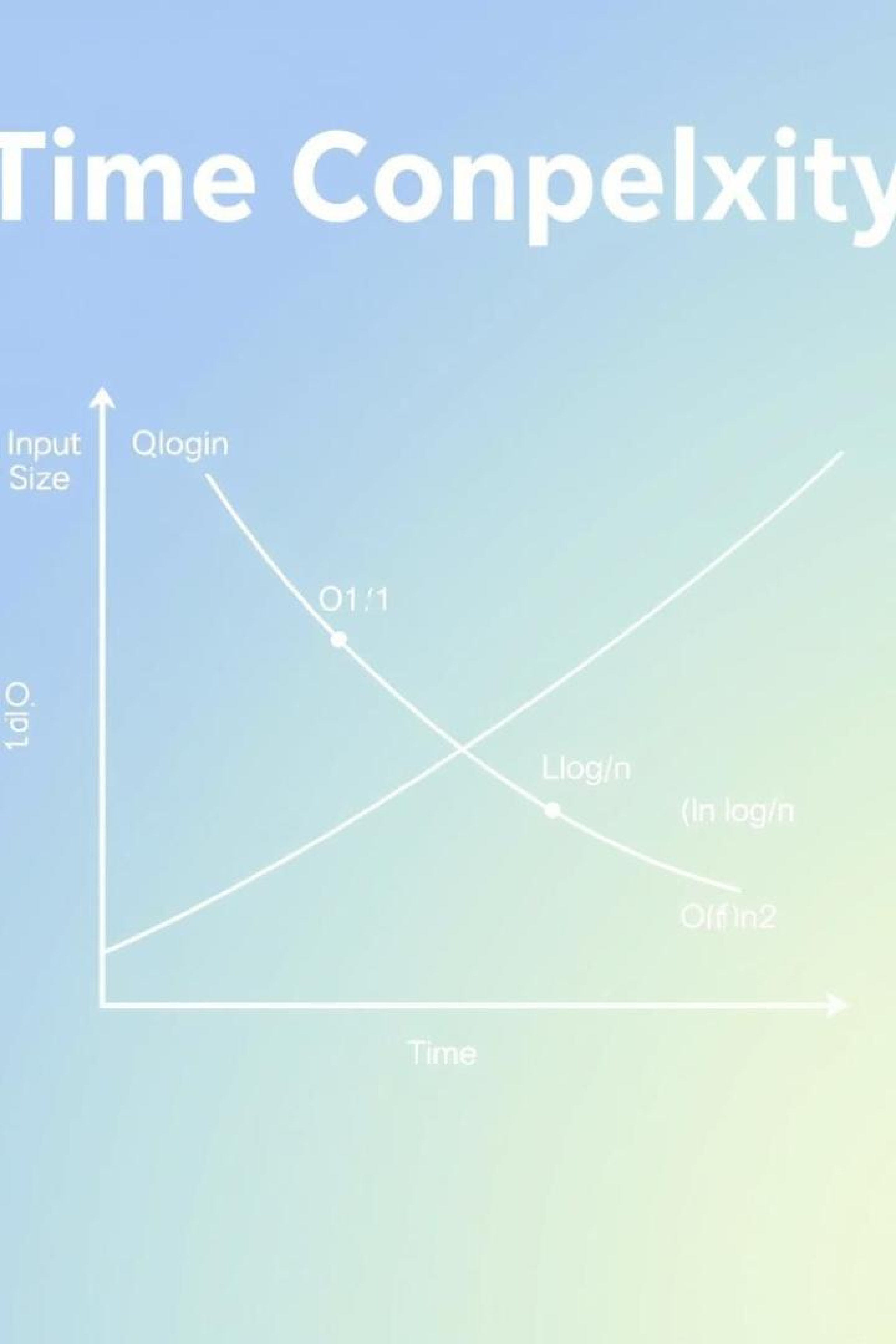
Halting Problem

No algorithm can decide if a program halts.

Reducibility

Proving problem hardness.

This theory explores the limits of computation. The Halting Problem is a classic undecidable case. Reducibility shows problem relationships.



Complexity Theory: Efficiency of Computation

Time Complexity

Algorithm time as a function of input size.

Space Complexity

Algorithm memory as a function of input size.

P vs. NP

Can quickly verified problems be quickly solved?

It classifies problems by inherent difficulty. Time and space complexity are key measures. The P vs. NP question remains a major unsolved puzzle.

Key Complexity Classes

P is polynomial time solvable. NP is polynomial time verifiable. NP-Complete are the hardest in NP. NP-Hard are at least as hard as NP.

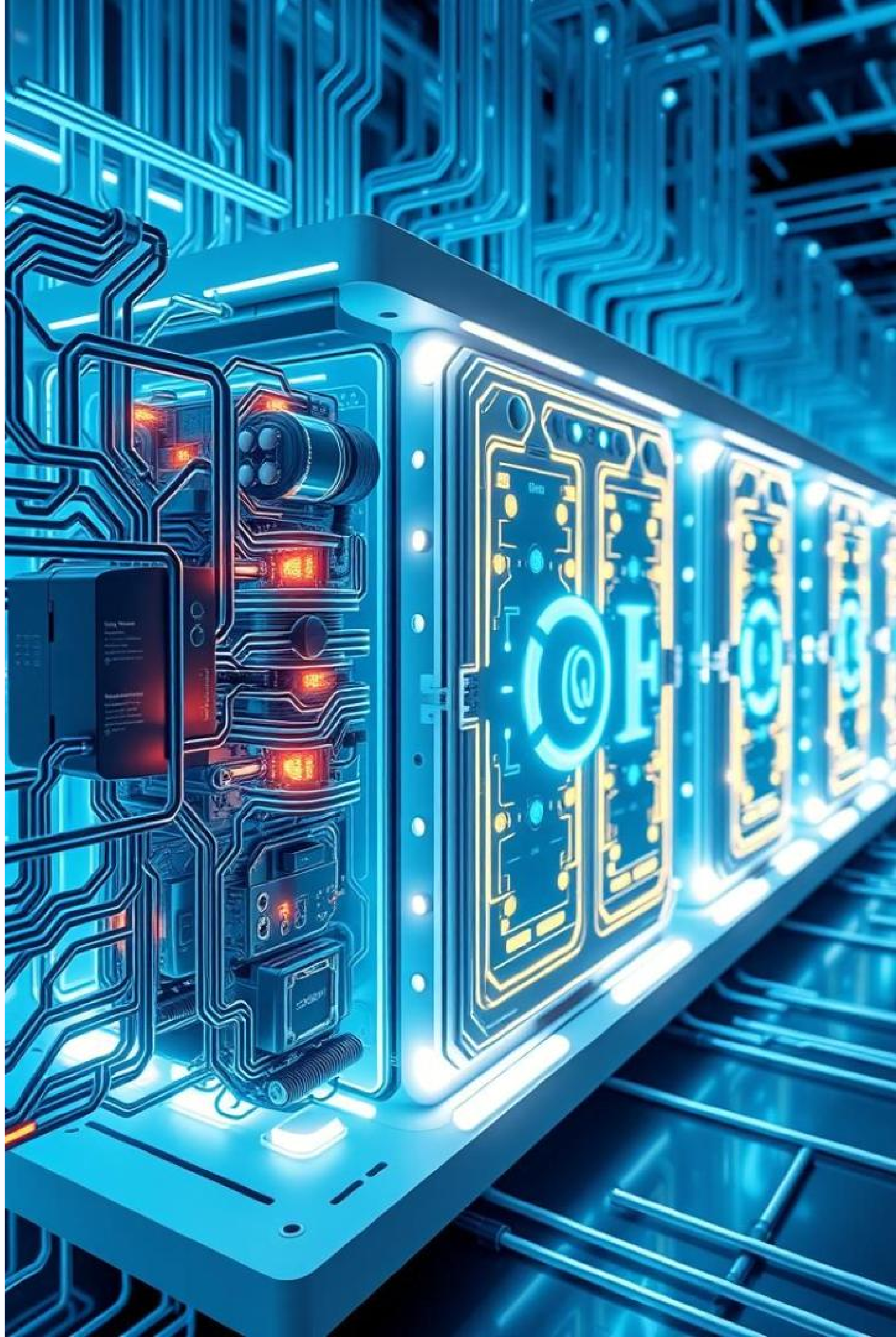
Modern Challenges and Open Problems

Quantum Computing

Complexity of ML

Explainable AI

Challenges include quantum computing. Also, the complexity of machine learning. Explainable AI is another focus.



The Future of Theoretical Computing

1

New Models

Exploring unconventional paradigms.

2

Interdisciplinary

Applying TCS to other fields.

3

New Algorithms

Efficient, scalable algorithms.

4

Theory and Practice

Bridging the gap.

The future involves new models. Also interdisciplinary applications. New algorithms are needed. Bridging theory and practice is vital.