Logic and Computational Complexity

A Perfect match

The unity of logic and computation has manifested itself in the development of computability theory from the 1930s onward and the development of computational complexity from the 1960s onward. Computability theory delineates the boundary between decidability and undecidability. Computational complexity delineates the boundary between tractability and intractability.
Logic provides prototypical complete problems for complexity classes and led to descriptive complexity, a framework for characterising complexity classes using logical resources.

COMPLETE PROBLEMS

1936
Church-Turing Theorem
First-Order Validity is computably enumerable (c.e.)-complete.

1949
Trakhtenbrot's Theorem
First-Order Finite Satisfiability is computably enumerable (c.e)-complete.

1971
Cook-Levin Theorem
SAT is NP-complete.

DESCRIPTIVE COMPLEXITY

1974
Fagin's Theorem
NP = ESO. In words, a decision problem Q is in NP if and only if Q is expressible in existential second-order logic ESO.

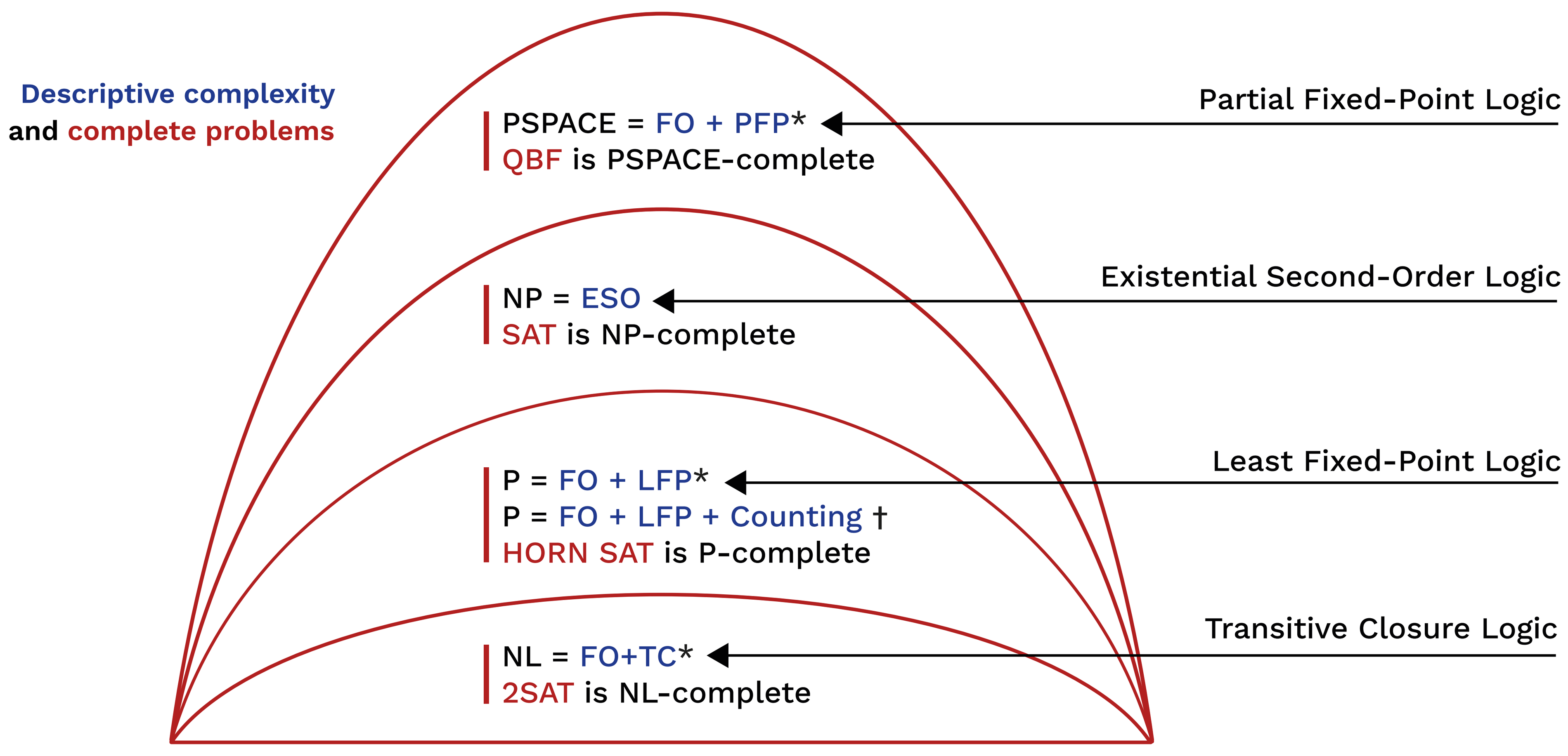
"machine-free characterisation of NP with no mention of polynomial"

Example: SAT is definable by the ESO-formula

$$\exists S \forall c \exists v \big((P(c,v) \wedge S(v)) \vee (N(c,v) \wedge \neg S(v))\big)$$

1982
Immerman-Vardi Theorem
P = FO+LFP on classes of ordered finite structures.

2010
Grohe's Theorem
If **C** is a class of graphs with at least one excluded minor, then on **C**
P = FO + LFP + Counting.
Key Property: Linear order definable in FO + LFP + Counting on C.



*on classes of *ordered* finite structures
† on classes of finite structures *excluding at least one minor*

Long-standing Open Problem in Descriptive Complexity
[Chandra & Harel (1982) – Gurevich (1988)]

Is there a logic for P on the class of all finite structures?