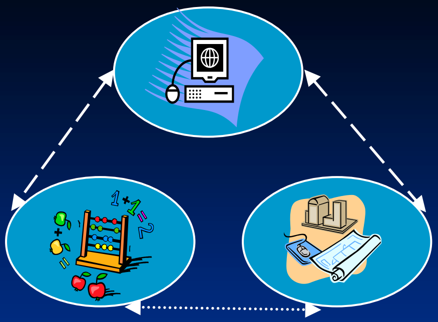


Reasoning systems from descriptive complexity

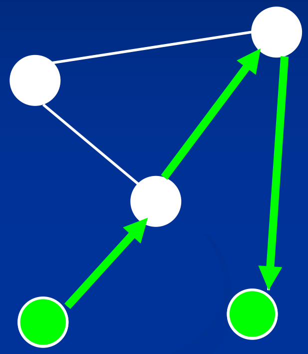
Antonina Kolokolova,

Memorial U. of Newfoundland

Simons Institute, April 7th, 2021

How hard is it ?

To find ?



What is the path from s to t in G?

To prove ?

$T \vdash$ "Reachability is transitive"

To describe ?

ϕ *(G) = "Graph G is connected"*

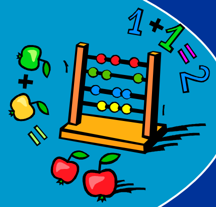
How hard is it ?

To find ?

Algorithms and complexity

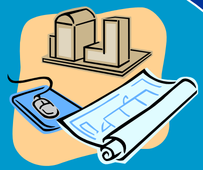
To prove ?

Bounded arithmetic (and proof systems)

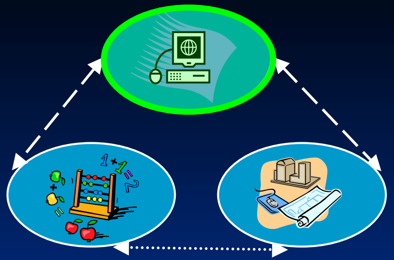


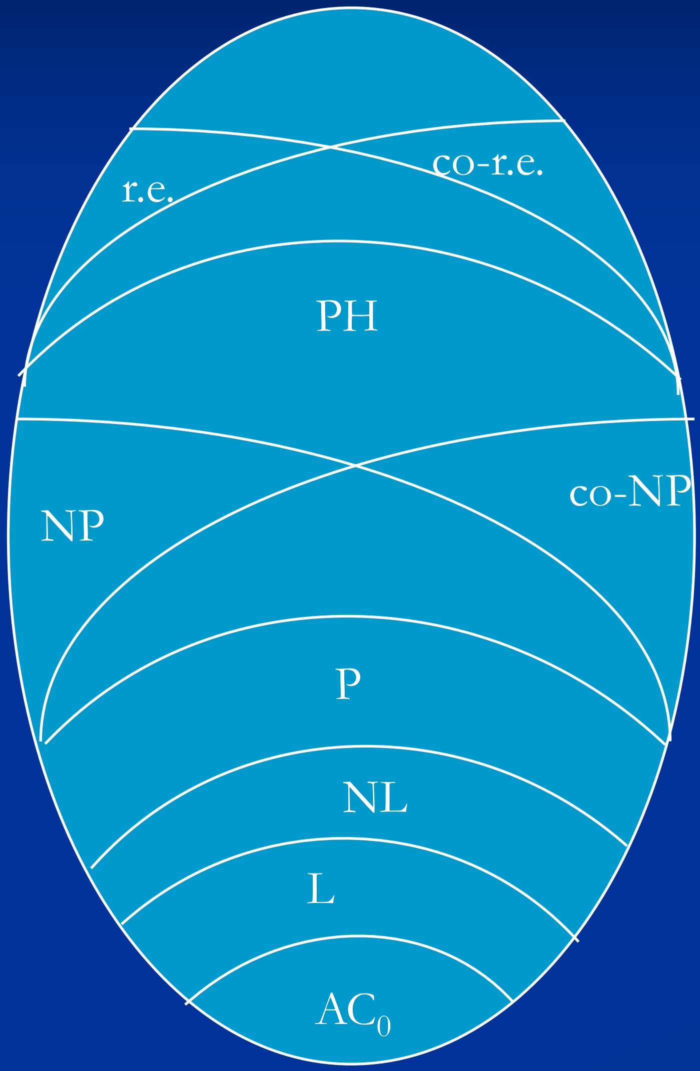
To describe ?

Descriptive complexity (finite model theory)



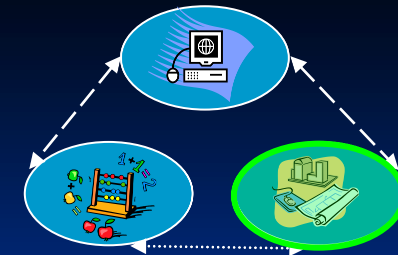
Complexity classes

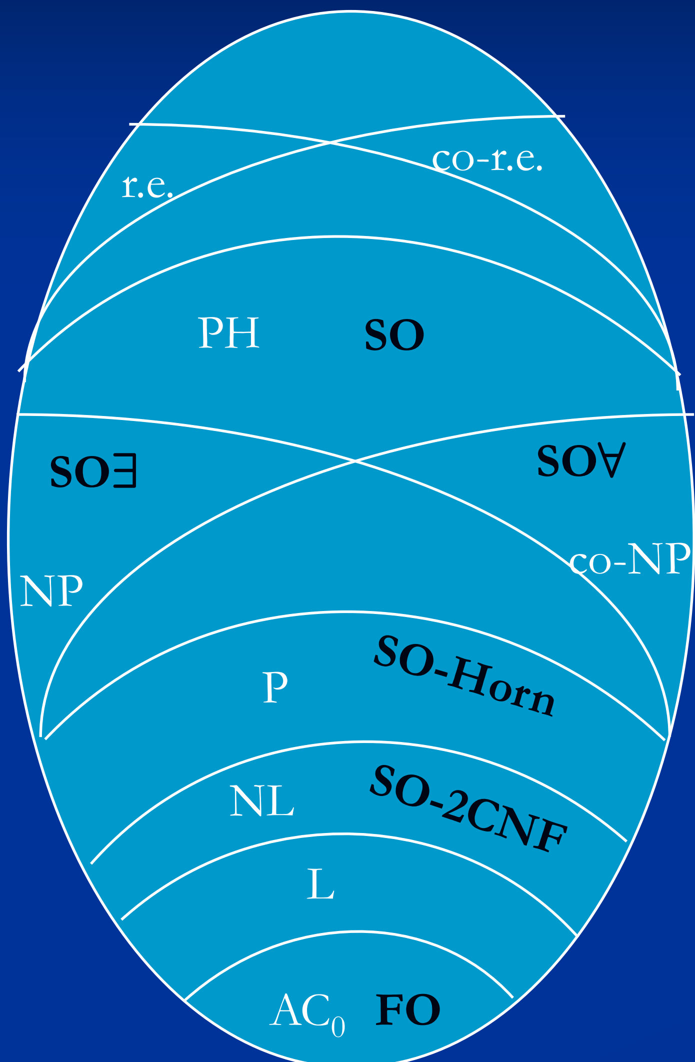




- Here, only uniform classes.
- In particular, DLOGTIME-uniform AC_0

Fragments of SO





- On finite structures with arithmetic
  - + and *, as well as ≤

- AC_0 : first-order logic [BIS]
- NL : second-order 2CNF [Grädel]
- P : second-order Horn [Grädel]
- NP : second-order ∃ logic [Fagin]
- PH: second-order logic [Stockmeyer]

Fagin's Theorem [Fagin 1974]

Iso-invariant Non-deterministic Polynomial-time (NP) $\equiv$ Existential Second-Order Logic (ESO)

a semantic class!

a syntactic class!

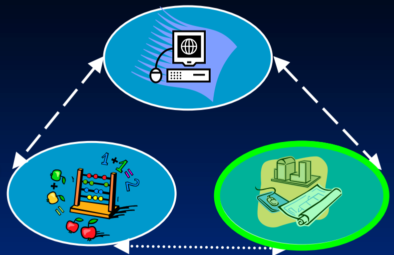
$$\exists \bar{X} \ \phi(\bar{R}, \bar{X})$$

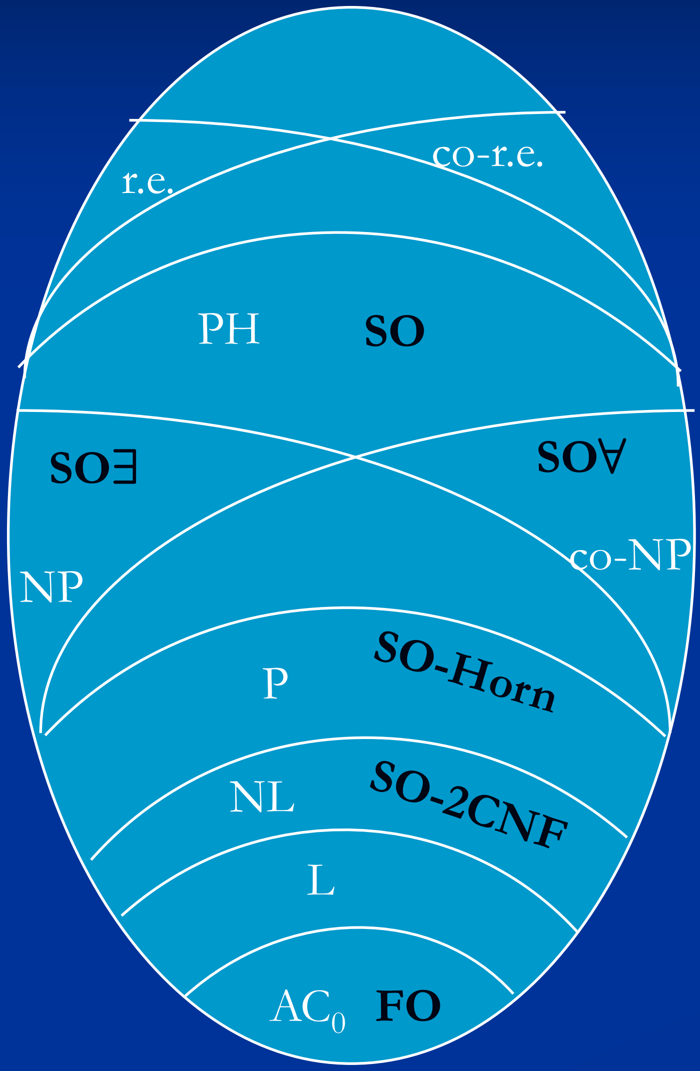
variables that range over relations!

input relations

FO formula

Grädel's characterizations



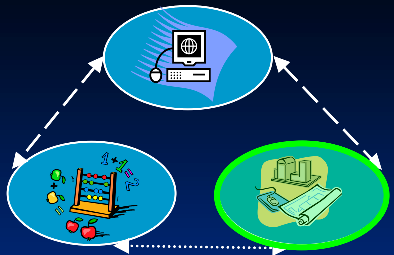


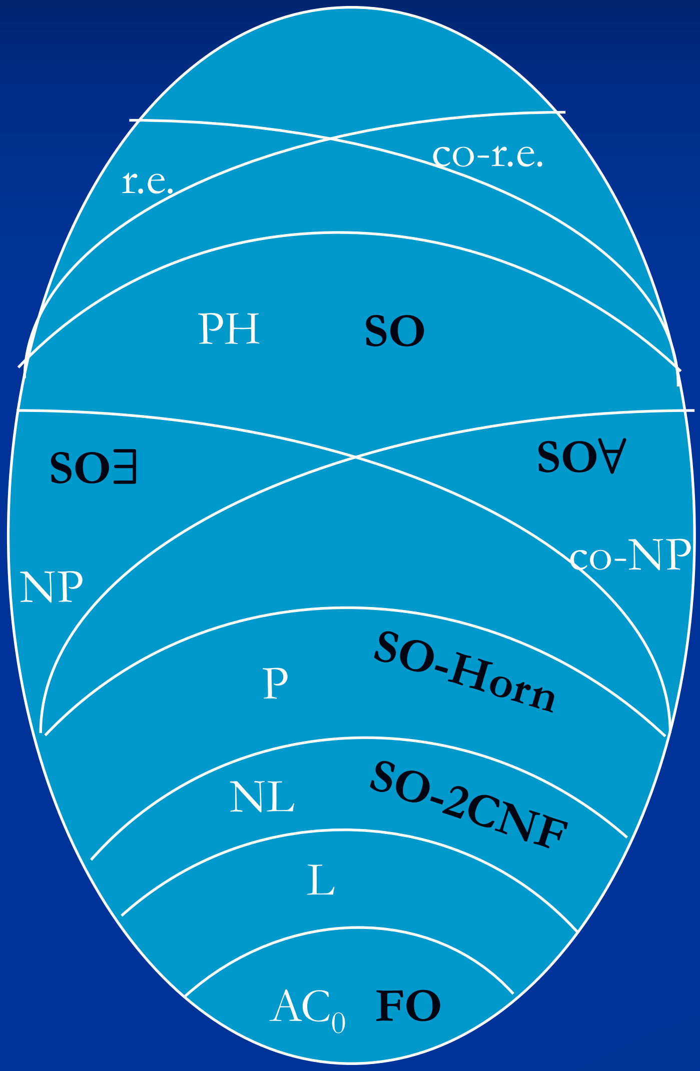
SNP: formulas of the form

$$\exists P_1 \ldots \exists P_k \forall x_1 \ldots \forall x_\ell\ \varphi$$

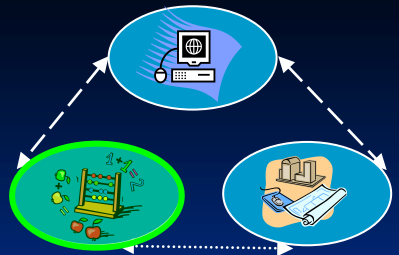
- Grädel'91:
  - Restrict φ to Horn, 2CNF,..
    - Resulting logics: $SO\exists\ Horn, SO\exists\ Krom$
  - Over successor structures
    - $SO\exists\ Horn$ captures P
    - $SO\exists\ Krom$ captures NL

Descriptive complexity





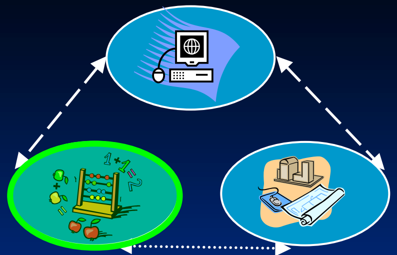
- Need arithmetic (<,+,*) for FO vs. AC_0
  - Successor for $SO\exists$ *Horn* vs P and $SO\exists$ *Krom* vs NL
  - $SO\exists$ captures NP over general finite structures
- Two logics are equivalent iff the corresponding complexity classes are.



From expressing to proving: Theories of arithmetic

The uniform side of proof complexity

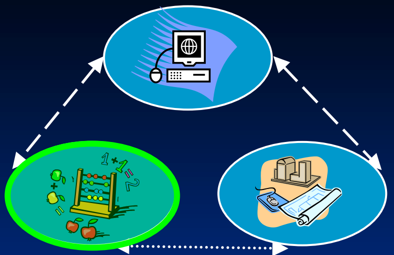
A little history



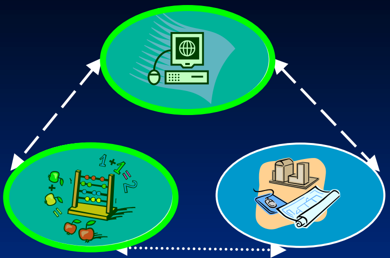
- Peano arithmetic
  - Axioms of numbers + induction
  - Too strong for efficient computation!
- Parikh's bounded version $I\Delta_0$
  - Axioms of numbers + bounded induction
  - Too weak: can only do linear time hierarchy
- Cook's PV:
  - Exactly polynomial-time by design; equational.
- Buss' bounded arithmetic

Much more on this in the next week session of this workshop

Bounded arithmetic



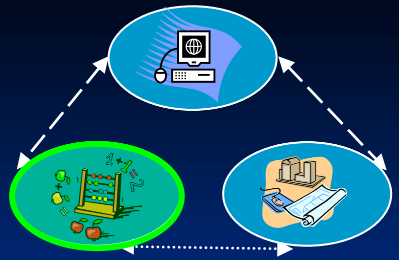
- All quantifiers are bounded by terms in free variables.
- Power of a theory of arithmetic ~ how complex are the functions it proves total.
  - Complexity of formulas defining the functions also matters
- Caveat: Two theories capturing the same class of functions may not be fully conservative over each other.
  - A theory is conservative over another if it can prove the other theory's theorems



Systems of arithmetic are uniform counterparts to propositional proof systems.

- Direct translations of the form "a theory proves soundness of a proof system, and each proof in the theory can be done in the proof system".
- AC_0 theory corresponds to Bounded Depth Frege proof system; P-theory to Extended Frege.

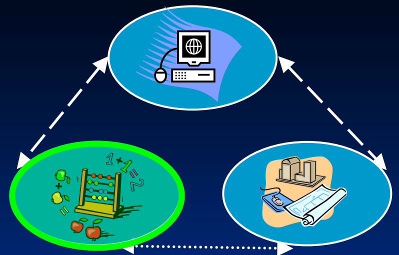
Let's build a theory



- Language: 2-sorted arithmetic (numbers + strings)
- Axioms:
  - For numbers: standard ($x + 1 \neq 0$, etc)
  - For strings: defining length and string equality
    - $X(y) \rightarrow y < |X|$, $y + 1 = |X| \rightarrow X(y)$,..
  - **Comprehension**: for a class of formulas Φ
    - $\exists X \leq n\, \forall z < n\, \big(X(z) \leftrightarrow \varphi(z)\big)$ for $\varphi \in \Phi$

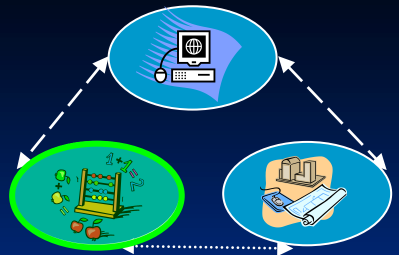
  - Can also add induction (provable in all our theories):
    - $X(0) \wedge \forall y < n\, \big(X(y) \rightarrow X(y+1)\big) \rightarrow X(n)$

Bounded arithmetic



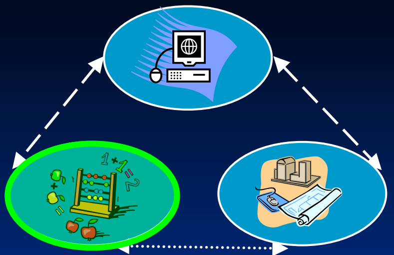
- For Φ the levels of SO, get (2-sorted analogues) of Buss' hierarchy S_2^i
  - Does it capture the corresponding classes?
- **Buss' witnessing:** $SO\exists$-theory captures P.
  - If it proves that a function is in NP ∩ co-NP, the function is in P.
  - Generalizes to levels of PH
- What would it take to capture a class of functions exactly?

First vs. Second-order



- **First-order:** Buss's basic theories S^i_2, T^i_2. Have $x\#y = 2^{|x|*|y|}$ in the language. Do not capture AC_0.
- **Second-order**: First, Buss's theories for PSPACE and beyond (with x#y).
- By Razborov-Takeuti's RSUV isomorphism, removing x#y and adding second sort (strings) get two-sorted theory V^i_1 for the same class.

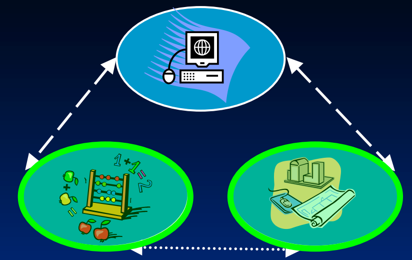
Sorts are strings and numbers indexing string positions. No operations on strings other than length and index.

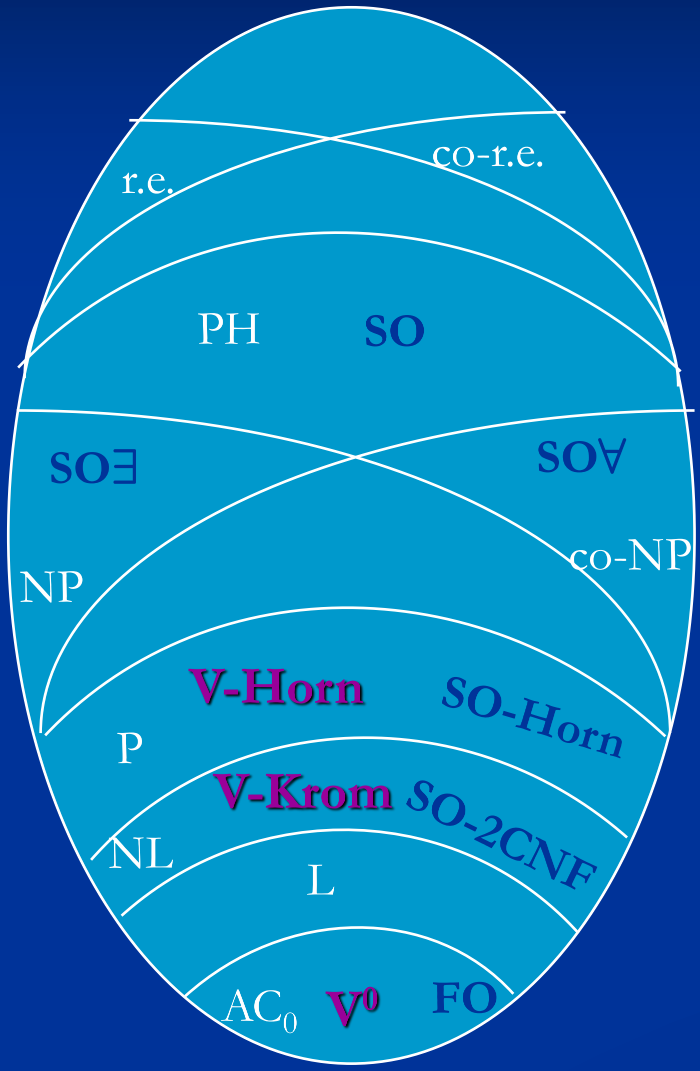


Build theories from logics of known descriptive complexity

- To create a theory, take basic axioms of arithmetic, and add an axiom stating “all objects definable in logic L exist”.
- For levels of PH, get the same theories as before.
- For non-deterministic classes, so far provably get the functions in the deterministic level of PH.

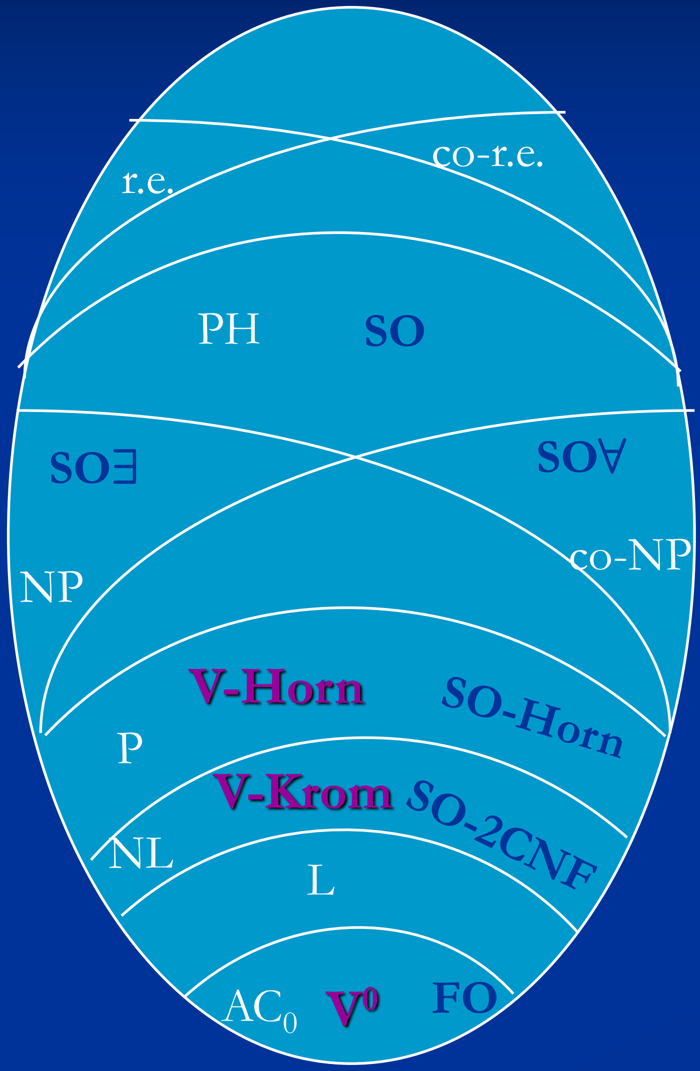
Systems of bounded arithmetic





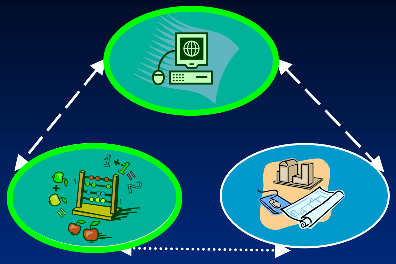
- First-order formulas give a theory for AC_0.
- $\Phi = SO\exists\ Krom$ gives a theory for NL.
- $\Phi = SO\exists\ Horn$ gives a (minimal) theory for P.

Systems of bounded arithmetic



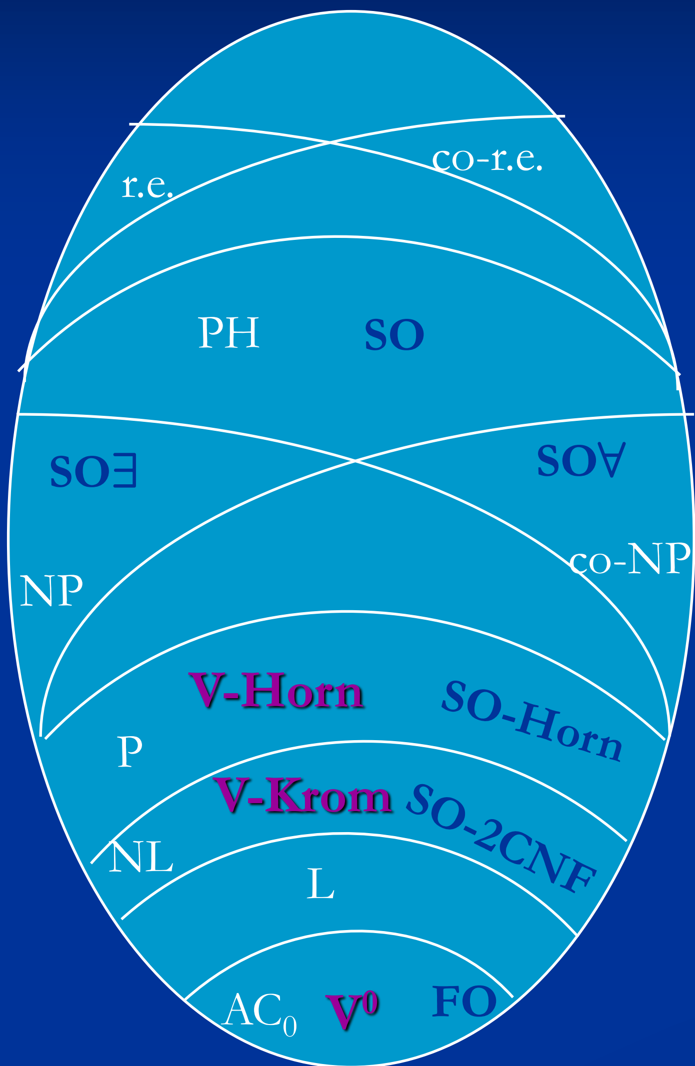
- The correspondences are not automatic: recall that a system based on NP formulas captured functions in P.
- Need additional conditions on provability of properties.

Closure properties



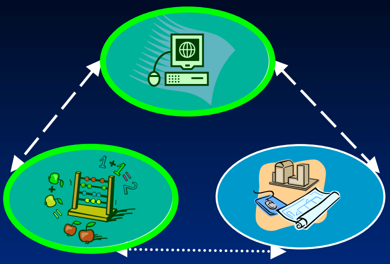
- We want robust definitions of complexity classes.
- Closure under first-order operations: AND, OR, NOT (*hardest one*), bounded quantification, and function composition.
- NP is not known to be closed under complementation. However, P is robust.
- Closure properties should be "easy" to prove.

Closure properties



- *Theorem:* If proving that a class is closed can be done with the reasoning inside the class, then the resulting system of arithmetic captures that class.

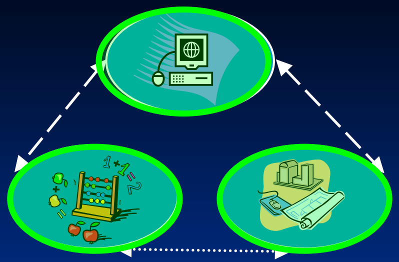
Closure properties



If proving that a class is closed can be done inside the class, then the resulting system of arithmetic captures that class.

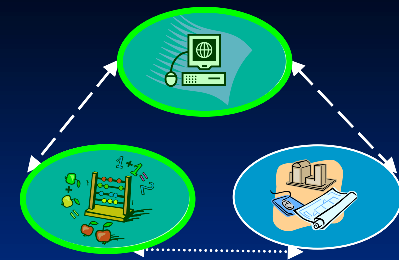
- Holds for AC_0 from the definitions.
- For P, need to formalize algorithms. [Cook, K '01,'03]
- Surprisingly, proof that NL=coNL can be done with NL reasoning. [Cook, K'04]
- LogCFL done from its circuit (SAC_1) definition (Kuroda)

Proof idea

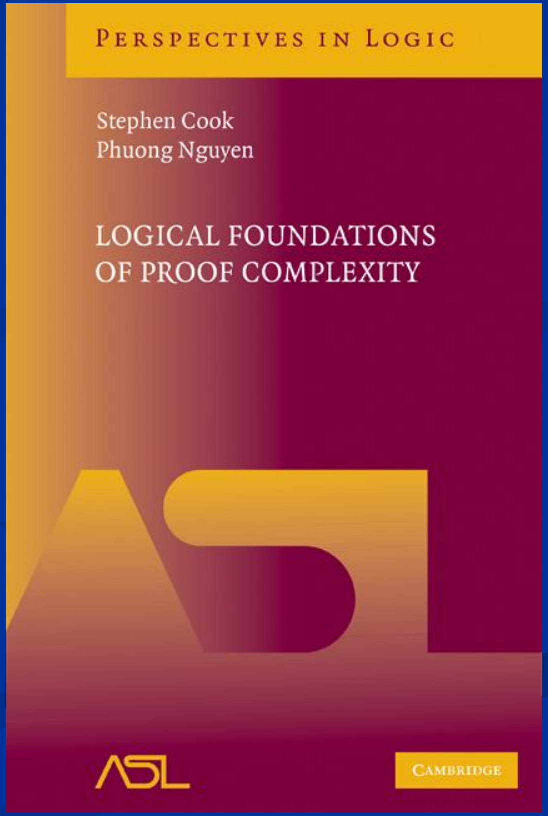


- Translate logics from descriptive complexity setting to the language of arithmetic.
- Define class of theories based on the logics, and show that basic properties (e.g., induction) hold.
- Introduce functions into the theory by defining their bit graphs by formulas (not the usual recursion-theoretic definitions).
- Generalize Buss' witnessing theorem to apply to this setting (complicated base case).

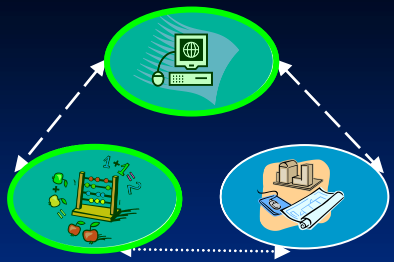
Other approaches



- *Constructing systems by adding to V^0 an axiom asserting the existence of a solution to a complete problem (Nguyen/Cook).*
  - E.g., based on versions of reachability problems
  - Different minimal theories for P, NL, L, etc.
  - Universally axiomatizable theories
  - Applicable to small circuit classes such as TC_0



Provability of separations



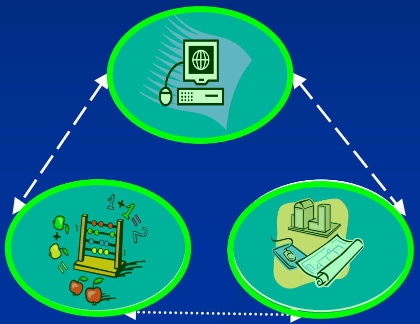
Maybe it is easier to separate theories than classes?

- Ajtai showed that Parity Principle is not provable in an AC_0 theory.

  The proof uses heavy model-theoretic machinery: forcing, non-standard models of arithmetic.
- Furst, Saxe, Sipser proved that Parity function is not computable by AC_0 circuits.

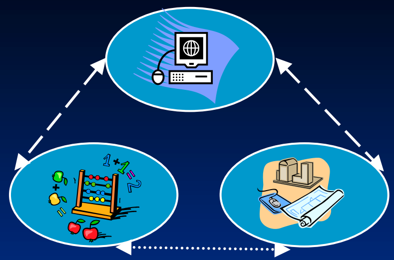
Conclusions

- There is a natural connection between the realms of *descriptive complexity* and *bounded arithmetic*, each of which is closely related to *complexity theory*.
- This gives a general method for constructing theories of arithmetic with predefined power.



Much more on arithmetic, etc next Wednesday!

Open questions



- Prove that the theories corresponding to different complexity classes are different.
- Which techniques are formalizable in weak theories?
- Connecting from bounded arithmetic back to descriptive complexity?

- In which theory can SL=L be formalized?
  - Existence of expander graphs is provable in an NC_1 theory

Thank You!