Totality, Provability, and Feasibility

Sam Buss
U.C. San Diego

ASL Annual Meeting
Gödel Lecture
New York City
May 22, 2019

Topics:

- Formal theories of weak fragments of Peano arithmetic
  - First- and second-order theories of bounded arithmetic
- $\forall\exists$ consequences: Provably total functions
  - Computational complexity characterizations
- $\forall$ consequences: Universal statements
  - Cook translation to propositional logic
  - Paris-Wilkie translation to propositional logic

Underlying philosophy:

- A feasibly constructive proof that a function is total should provide a feasible method to compute it.
- A feasibly constructive proof of a universal statement should provide a feasible method to verify any given instance.

Some quotes

Cook, 1975, Feasibly constructive proofs and the propositional calculus

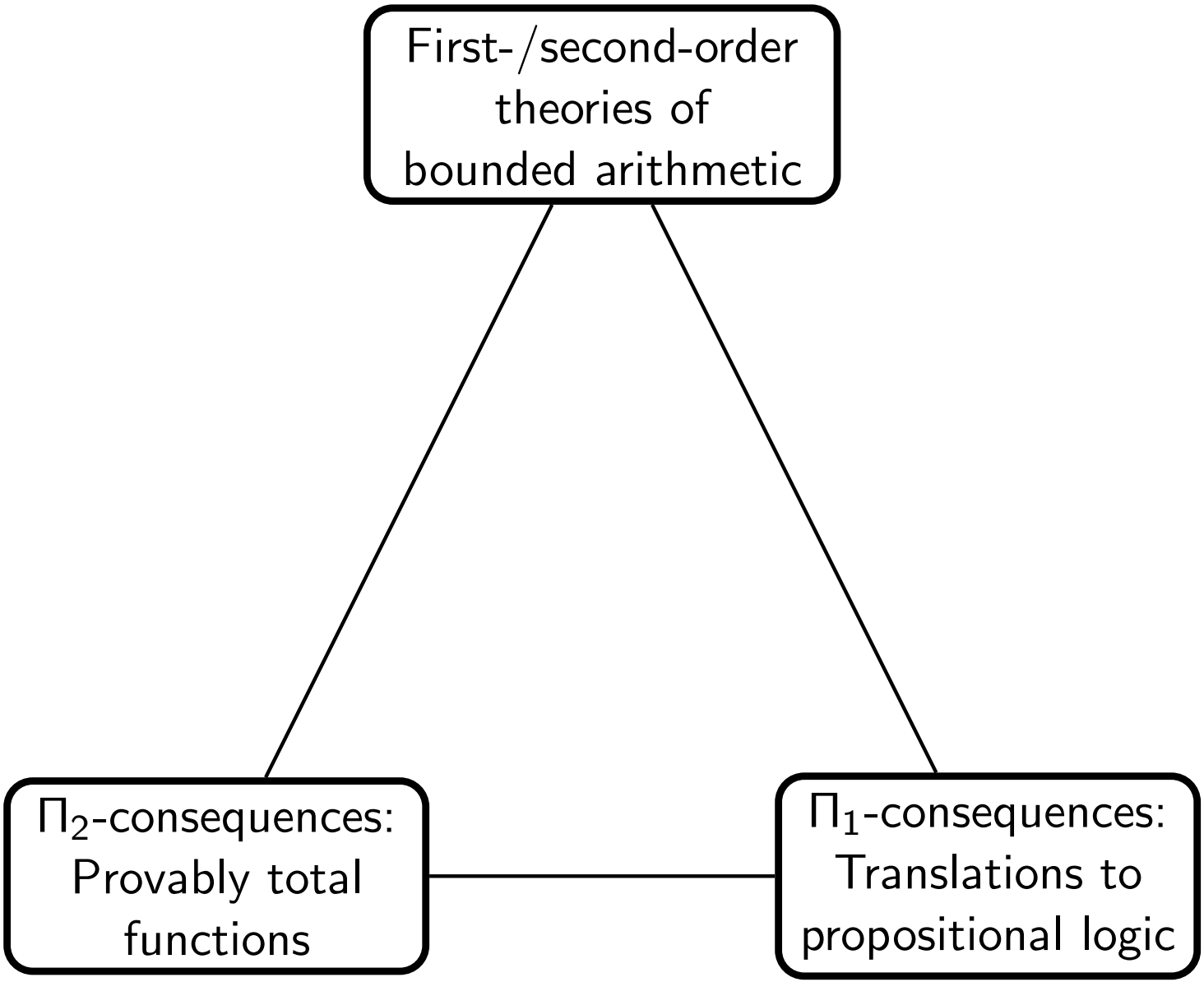
A constructive proof of, say, a statement $\forall x A$ must provide an effective means of finding a proof of A for each value of x, but nothing is said about how long this proof is as a function of x. If the function is exponential or super exponential, then for short values of x the length of the proof of the instance of A may exceed the number of electrons in the universe.

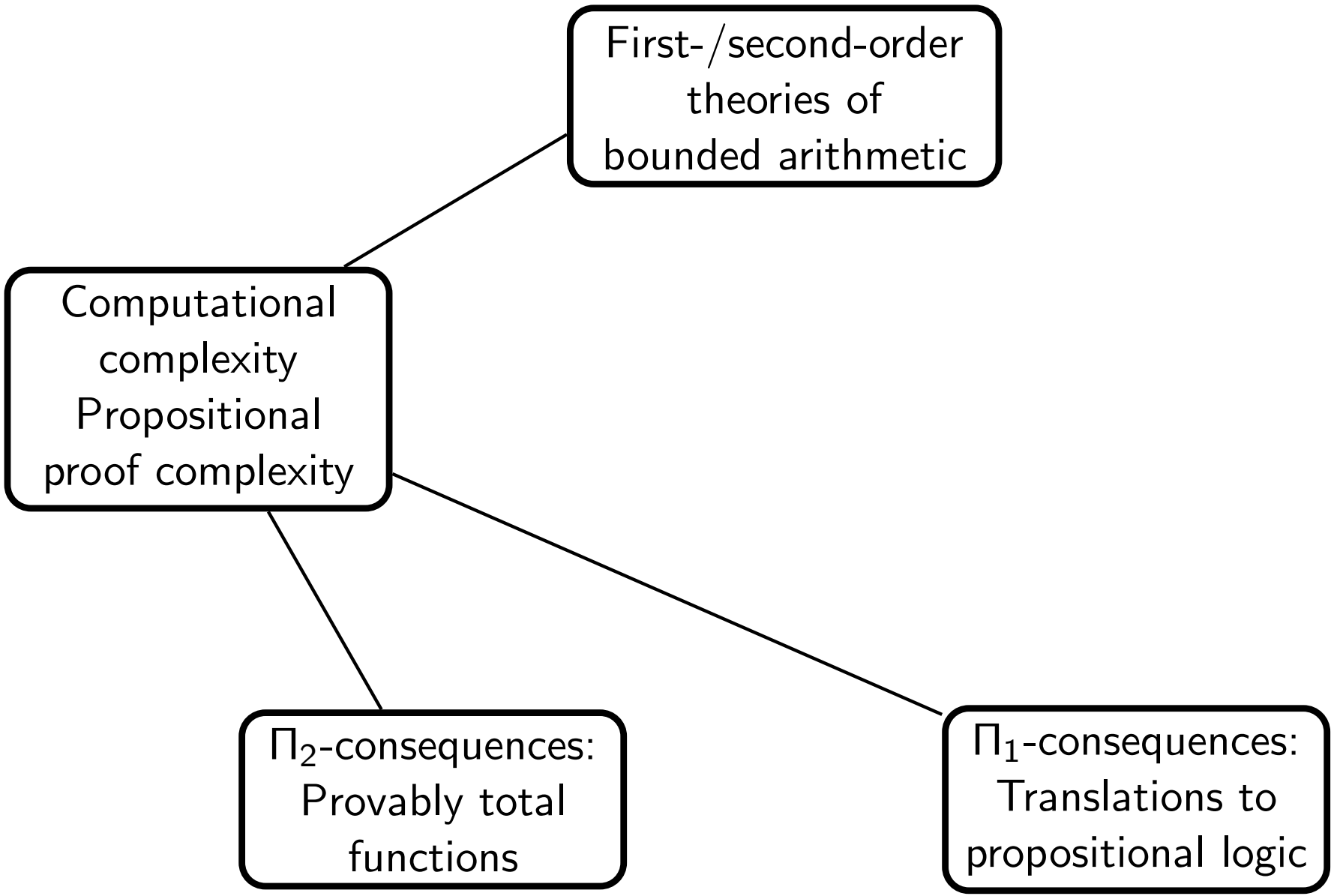
Introducing PV and the Cook translation

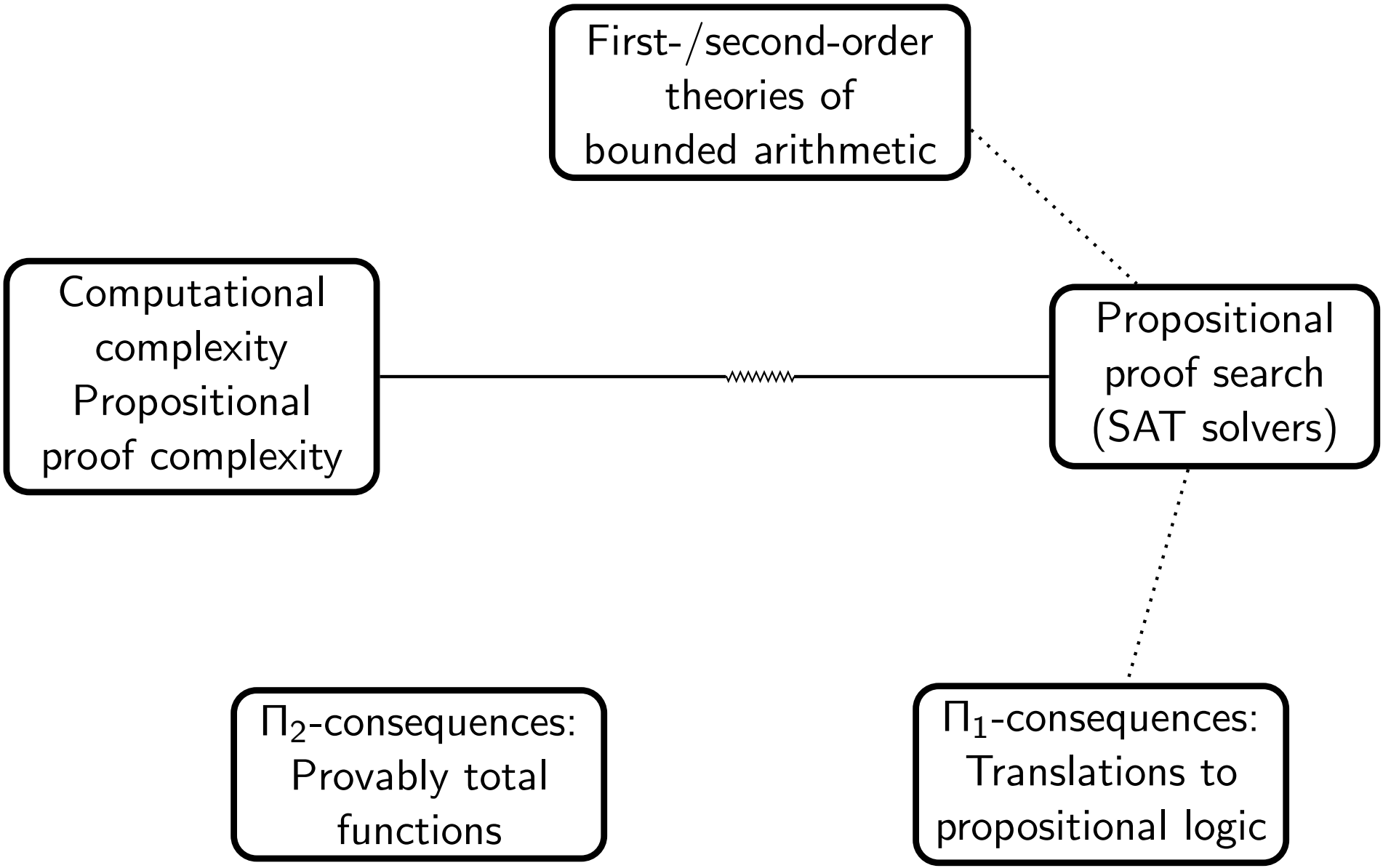
Parikh, 1971, Existence and feasibility in arithmetic

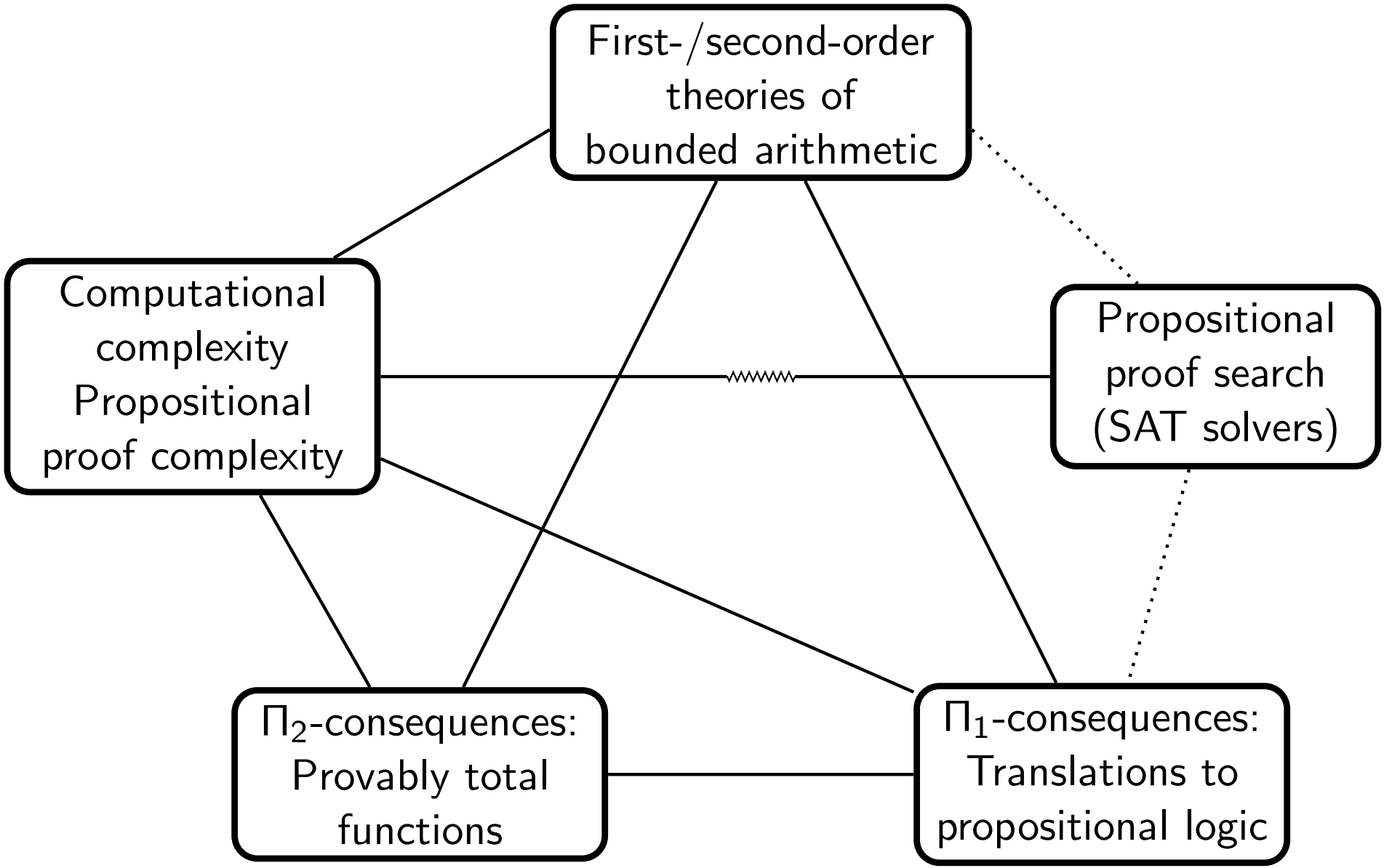
> *All this would tend to give some weight to the contentions that there is a definite concrete or anthropomorphic point of view possible in mathematics and that exponentiation would be excluded by this view as a genuine computable function. [...] We formulate a subsystem of Peano arithmetic which is related to the anthropomorphic viewpoint and investigate some of its properties.*

Introducing $\mathrm{I}\Delta_0$.


First-/second-order theories of bounded arithmetic
Π_2-consequences: Provably total functions
Π_1-consequences: Translations to propositional logic


First-/second-order theories of bounded arithmetic
Computational complexity Propositional proof complexity
Π_2-consequences: Provably total functions
Π_1-consequences: Translations to propositional logic


First-/second-order theories of bounded arithmetic
Computational complexity Propositional proof complexity
Propositional proof search (SAT solvers)
Π_2-consequences: Provably total functions
Π_1-consequences: Translations to propositional logic


First-/second-order theories of bounded arithmetic
Computational complexity Propositional proof complexity
Propositional proof search (SAT solvers)
Π_2-consequences: Provably total functions
Π_1-consequences: Translations to propositional logic

S^1_2, PV — Polynomial time — $\mathrm{e}\mathcal{F}$ [B'85; C'76]

First-order theory S^1_2 of arithmetic:

- Terms have polynomial growth rate (smash, #, is used).
- Bounded quantifiers $\forall x \leq t$, $\exists x \leq t$.
- Sharply bounded quantifiers $\forall x \leq |t|$, $\exists x \leq |t|$,

  bound x by *log* (or *length*) of t.
- Classes Σ^{b}_i and Π^{b}_i of formulas are defined by counting bounded quantifiers, ignoring sharply bounded quantifiers.
- Σ^{b}_1 formulas express exactly the NP predicates.

  Σ^{b}_i, Π^{b}_i - express exactly the predicates at the i-th level of the polynomial time hierarchy.
- S^1_2 has *polynomial induction* PIND, equivalently *length induction* (LIND), for Σ^{b}_1 formulas A (i.e., NP formulas):

$$A(0) \wedge (\forall x)(A(x) \rightarrow A(x{+}1)) \rightarrow (\forall x)A(|x|)$$

(1) Provably total functions of S^1_2:

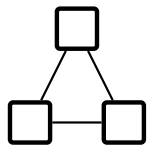
- The $\forall\Sigma^{\mathrm{b}}_1$-definable functions (aka: *provably total functions*) are precisely the polynomial time computable functions.
- PV: equational theory over polynomial time functions. [C'75]
- $\mathrm{S}^1_2(\mathrm{PV})$ is conservative over both S^1_2 and PV.

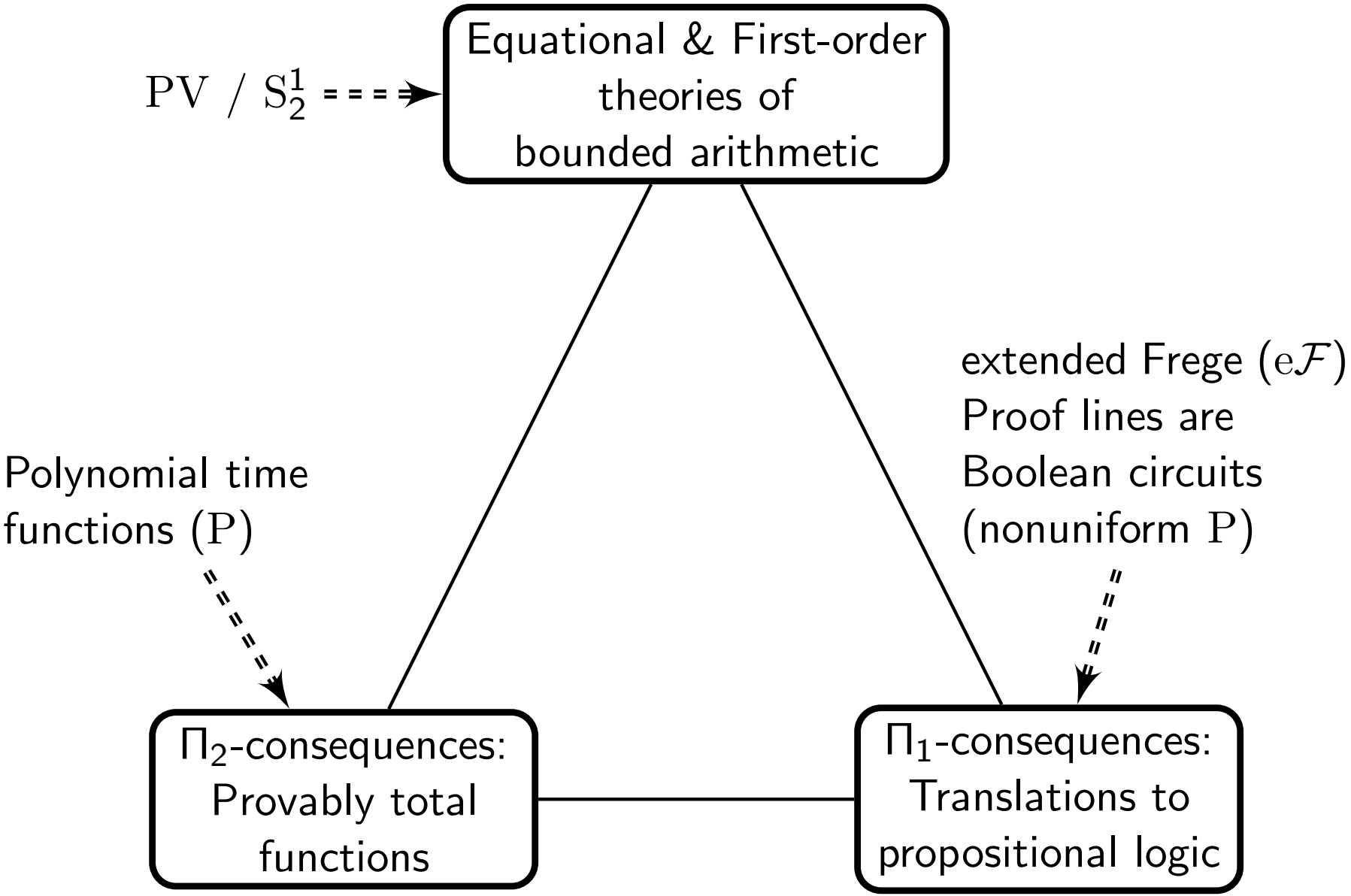
(2) Translation to propositional logic ("Cook translation")

- Any polynomial identity ($\forall\Sigma^{\mathrm{b}}_0$-property) provable in PV / S^1_2, has a natural translation to a family F of propositional formulas. These formulas have polynomial size extended Frege ($\mathrm{e}\mathcal{F}$) proofs.

(3) S^1_2 proves the consistency of $\mathrm{e}\mathcal{F}$. Conversely, any propositional proof systems (p.p.s.) S^1_2 proves is consistent(provably) polynomially simulated by $\mathrm{e}\mathcal{F}$.

(4) Lines (formulas) in an $\mathrm{e}\mathcal{F}$ proof correspond to Boolean circuits. The circuit value problem is complete for P (polynomial time).




PV / S_2^1
Equational & First-order theories of bounded arithmetic
Polynomial time functions (P)
extended Frege ($e\mathcal{F}$) Proof lines are Boolean circuits (nonuniform P)
Π_2-consequences: Provably total functions
Π_1-consequences: Translations to propositional logic

Example of Cook translation: S^1_2, $\mathrm{e}\mathcal{F}$, PHP.

The first-order theory S^1_2 proves:
$(\forall x, n)$["The bits of x do not code an incidence matrix of a bipartite graph on $[n+1] \cup [n]$ violating the Pigeonhole Principle PHP^{n+1}_n"]

Propositional translations PHP^{n+1}_n: $(n \geq 1)$

$$\bigwedge_{i=0}^{n} \bigvee_{j=0}^{n-1} p_{i,j} \quad \rightarrow \quad \bigvee_{i=0}^{n-1} \bigvee_{i'=i+1}^{n} \bigvee_{j=0}^{n-1} (p_{i,j} \wedge p_{i',j})$$

The propositional variables $p_{i,j}$ correspond to the bits of the first-order variable x.

Cook translation yields:
The PHP^{n+1}_n formulas have polynomial size $\mathrm{e}\mathcal{F}$ proofs. [CR]

Propositional proof systems ($\mathcal{F}$, $\mathrm{e}\mathcal{F}$, ...)

Frege proofs ($\mathcal{F}$): Sequent calculus propositional system. Equivalent to a 'textbook style' proof system using modus ponens.

Extended Frege proofs ($\mathrm{e}\mathcal{F}$): Frege systems augmented with extension rule allowing (iterated) introduction of new variables x abbreviating formulas:

$$\textit{Extension axiom:} \qquad x \leftrightarrow \varphi.$$

$\mathbf{AC^0}$-Frege, aka constant-depth Frege: Frege proofs over $\wedge, \vee, \neg$ with a constant bound on the number of alternations of $\wedge$'s and $\vee$'s. (Negations applied only to variables.)

Quantified sequent calculus $\mathbf{QBF}$ with $\forall p$, $\exists p$ Boolean quantifiers. G_i is QBF restricted to i-levels of quantifiers.

Proof size = number of symbols in the proof.
(The purpose of extension is to reduce proof size.)

Open problems:

(1) Does the Frege system ($\mathcal{F}$) allow polynomial size proofs of tautologies? (Subexponential size?)

(2) Does the Frege system quasipolynomially simulate the extended Frege ($\mathrm{e}\mathcal{F}$) system?

- No good combinatorial candidates for separation are known. [BBP,HT,B,AB,...]

(3) QBF versus $\mathrm{e}\mathcal{F}$?

- ($\mathrm{e}\mathcal{F}$ is equivalent to G_1^*, i.e., tree-like G_1).

More theories with Cook translations

Theories for polynomial space

- PSA - Equational theory for PSPACE functions [D]
- U^1_2 - Second-order theory for polynomial space [B]
- The $\Sigma^{1,b}_1$-definable functions of U^1_2 are precisely the PSPACE functions.
- $\mathrm{U}^1_2(\mathrm{PSA})$ is conservative over both U^1_2 and PSA. [**]
- PSPACE identities provable in U^1_2 have natural translations to QBF formulas which have polynomial size QBF proofs.

Weak second-order theories for weaker complexity [I,Z,...,CN]
These second-order theories use
(a) first-order objects playing the role of sharply bounded objects,
(b) second-order objects playing the role of inputs and outputs.
Base theory V^0 has comprehension and induction for bounded first-order formulas (with second order free variables).

Theories for $\mathbf{ALogTime}$ (uniform $\mathbf{NC^1}$): [CT, A, CM, CN]

- Complexity class NC^1 - properties expressible by polynomial size Boolean formulas.
- VNC^1 - is V^0 plus axioms asserting the totality of the Boolean Formula Value Problem or log-bounded tree recursion. These are in NC^1 [B] and complete for NC^1.
- Provably total functions are precisely the functions of polynomial growth rate with NC^1 bit graph.
- Cook translation is to Frege proofs $\mathcal{F}$.

Theories for L (log space) [Z, P, CN]

- VL - is V^0 plus axioms asserting the totality of log-bounded recursion.
- Provably total functions are precisely the log-space computable functions.
- Cook translation is a tree-like p.p.s. GL^* for Σ-$\mathrm{CNF}(2)$ formulas, a class of QBF formulas complete for log space. [J]

Theories for NL (nondeterministic log space) [CK, P, CN]

- VNL - is V^0 plus axioms asserting the existence of a distance predicate for graph reachability.
- Provably total functions are precisely the polynomial growth rate functions with NL bit graph.
- Cook translation is a tree-like p.p.s. GNL^* for ΣKrom formulas, a class of QBF formulas complete for NL. [G]

In progress: New p.p.s.'s eLDT and eLNDT for branching programs and nondeterministic branching programs as Cook translations for VL and VNL. [B-Das-Knop, following Cook]

Formal Theory	Propositional Proof System	Total Functions	
PV, S^1_2, VPV	$\mathrm{e}\mathcal{F}$, G^*_1	P	[C, B, CN]
T^1_2, S^2_2	G_1, G^*_2	$\leq_{1\text{-}1}(\mathrm{PLS})$	[B, KP, KT, BK]
T^2_2, S^3_2	G_2, G^*_3	$\leq_{1\text{-}1}(\mathrm{CPLS})$	[B, KP, KT, KST]
T^i_2, S^{i+1}_2	G_i, G^*_{i+1}	$\leq_{1\text{-}1}(\mathrm{LLI}_i)$	[B, KP, KT, KNT]
PSA, U^1_2, W^1_1	QBF	*PSPACE*	[D, B, S]
V^1_2	**	*EXPTIME*	[B]
VNC^1	Frege ($\mathcal{F}$)	$\mathrm{ALogTime}$	[CT, A; CM, CN]
VL	GL^*	L	[Z, P, CN]
VNL	GNL^*	NL	[CK, P, CN]

PV, PSA - equational theories.
S^i_2, T^i_2 - first order
$\mathrm{U}^1_2, \mathrm{V}^1_2$, $\mathrm{VNC}^1, \mathrm{VL}, \mathrm{VNL}, \mathrm{VPV}$ - second order

Formal Theory	Propositional Proof System	Total Functions	
PV, S^1_2, VPV	$\mathrm{e}\mathcal{F}$, G^*_1	P	[C, B, CN]
T^1_2, S^2_2	G_1, G^*_2	$\leq_{1\text{-}1}(\mathrm{PLS})$	[B, KP, KT, BK]
T^2_2, S^3_2	G_2, G^*_3	$\leq_{1\text{-}1}(\mathrm{CPLS})$	[B, KP, KT, KST]
T^i_2, S^{i+1}_2	G_i, G^*_{i+1}	$\leq_{1\text{-}1}(\mathrm{LLI}_i)$	[B, KP, KT, KNT]
PSA, U^1_2, W^1_1	QBF	*PSPACE*	[D, B, S]
V^1_2	**	*EXPTIME*	[B]
VNC^1	Frege ($\mathcal{F}$)	$\mathrm{ALogTime}$	[CT, A; CM, CN]
VL	GL^*	L	[Z, P, CN]
VNL	GNL^*	NL	[CK, P, CN]

Using Cook translation to propositional proof systems (p.p.s.'s)

$\mathcal{F}, \mathrm{e}\mathcal{F}$ - Frege and extended Frege.

G_i, QBF - quantified propositional logics.

Starred (*) propositional systems are tree-like.

Formal Theory	Propositional Proof System	Total Functions	
PV, S^1_2, VPV	$\mathrm{e}\mathcal{F}$, G^*_1	P	[C, B, CN]
T^1_2, S^2_2	G_1, G^*_2	$\leq_{\text{1-1}}(\mathrm{PLS})$	[B, KP, KT, BK]
T^2_2, S^3_2	G_2, G^*_3	$\leq_{\text{1-1}}(\mathrm{CPLS})$	[B, KP, KT, KST]
T^i_2, S^{i+1}_2	G_i, G^*_{i+1}	$\leq_{\text{1-1}}(\mathrm{LLI}_i)$	[B, KP, KT, KNT]
PSA, U^1_2, W^1_1	QBF	*PSPACE*	[D, B, S]
V^1_2	**	*EXPTIME*	[B]
VNC^1	Frege ($\mathcal{F}$)	$\mathrm{ALogTime}$	[CT, A; CM, CN]
VL	GL^*	L	[Z, P, CN]
VNL	GNL^*	NL	[CK, P, CN]

PLS = Polynomial local search [JPY]
CPLS = "Colored" PLS [ST]
LLI = Linear local improvement

Second order arithmetic & Paris-Wilkie translations

Paris-Wilkie translation: is a second kind of translation to propositional logic.

- The Paris-Wilkie translation applies to first-order theories with second-order predicates (free variables, α), essentially oracles.
- Propositional variables now represent values of the second order objects α.
  In contrast, the Cook translation uses variables for the bits of first-order objects (the function's inputs).
- Paris-Wilkie translations are most commonly applied to fragments of $\mathrm{I}\Delta_0(\#, \alpha)$. [P, PW, ...].

  α denotes an uninterpreted second-order object (a predicate, or oracle),

  and $\#$ is the polynomial growth rate function $x\#y = 2^{|x|\cdot|y|}$

Example of Paris-Wilkie translation

Let T be the theory $\mathrm{I}\Delta_0$ or $\mathrm{I}\Delta_0(\#)$.

Thm: [PW] If $T(\alpha)$ proves the pigeonhole principle

$$(\forall x{\leq}a)(\exists y{<}a)\alpha(x,y) \rightarrow (\exists x{<}x'{\leq}a)(\exists y{<}a)(\alpha(x,y) \wedge \alpha(x',y))$$

then PHP_n^{n+1} has polynomial (quasipolynomial, resp) size AC^0-Frege proofs.

Recall PHP_n^{n+1}:

$$\bigwedge_{i=0}^{n} \bigvee_{j=0}^{n-1} p_{i,j} \quad \rightarrow \quad \bigvee_{i=0}^{n-1} \bigvee_{i'=i+1}^{n} \bigvee_{j=0}^{n-1} (p_{i,j} \wedge p_{i',j})$$

Propositional variables $p_{i,j}$ correspond to truth values of $\alpha(x,y)$.

On the other hand, [A,BPI,KPW],

Thm: PHP_n^{n+1} requires exponential size AC^0-Frege proofs.

Proof idea: apply a Hastad-style switching lemma, to reduce to a proof in which all formulas are decision trees.

Corollary: Neither $\mathrm{I}\Delta_0$ nor $\mathrm{I}\Delta_0(\#)$ proves the pigeonhole principle.

But, [PWW,MPW], ...

Thm: $\mathrm{I}\Delta(\#)$ proves the weak pigeonhole principle (replacing "$\exists y{<}a$" with "$\exists y{<}a/2$").

Corollary: The propositional weak pigeonhole principle PHP_n^{2n} has quasipolynomial size AC^0-Frege proofs.

Theories of arithmetic for Paris-Wilkie translations

A hierarchy of fragments of $\mathrm{I}\Delta_0(\#)$: [B]

- T_2^i - induction for Σ_i^{b} predicates (the i-th level of the polynomial time hierarchy).
- S_2^i - length induction for Σ_i^{b} predicates.
- $\mathrm{S}_2^1 \subseteq \mathrm{T}_2^1 \preccurlyeq_{\forall\Sigma_2^{\mathrm{b}}} \mathrm{S}_2^2 \subseteq \mathrm{T}_2^2 \preccurlyeq_{\forall\Sigma_3^{\mathrm{b}}} \mathrm{S}_2^3 \subseteq \mathrm{T}_2^3 \preccurlyeq_{\forall\Sigma_4^{\mathrm{b}}} \cdots$

Thm: [KPT]

- If $\mathrm{T}_2^i = \mathrm{S}_2^{i+1}$, then the polynomial time hierarchy collapses.
- In fact, if $\mathrm{T}_2^i \preccurlyeq_{\forall\Sigma_{i+2}^{\mathrm{b}}} S_2^{i+1}$, then the polynomial time hierarchy collapses.
- $\mathrm{T}_2^i(\alpha) \neq \mathrm{S}_2^{i+1}(\alpha)$; i.e., relative to an oracle.

$$\mathrm{S}^1_2(\alpha) \subseteq \mathrm{T}^1_2(\alpha) \preccurlyeq_{\forall\mathbf{\Sigma}^{\mathrm{b}}_2(\alpha)} \mathrm{S}^2_2(\alpha) \subseteq \mathrm{T}^2_2(\alpha) \preccurlyeq_{\forall\mathbf{\Sigma}^{\mathrm{b}}_3(\alpha)} \cdots$$

Formal Theory	Paris-Wilkie translation Propositional Proof System [K]	Total Functions
$\mathrm{T}^1_2(\alpha)$, $\mathrm{S}^2_2(\alpha)$	**	$\leq_{1\text{-}1}(\mathrm{PLS}(\alpha))$
$\mathrm{T}^2_2(\alpha)$, $\mathrm{S}^3_2(\alpha)$	**res(log)**	$\leq_{1\text{-}1}(\mathrm{CPLS}(\alpha))$
$\mathrm{T}^i_2(\alpha)$, $\mathrm{S}^{i+1}_2(\alpha)$	**depth $(i-\frac{3}{2})$-Frege**	$\leq_{1\text{-}1}(\mathrm{LLI}_i(\alpha))$

Depth $(n+\frac{1}{2})$-Frege means LK proofs with formulas having at most $n+1$ alternations, the bottom level having only logarithmic fanin. res(log) $=$ depth $\frac{1}{2}$-Frege.

Sample application: $\mathrm{T}^2_2 \vdash \mathrm{PHP}^{2n}_n$. Hence, the bit-graph weak PHP has res(log) refutations of quasipolynomial size. Likewise, any sparse instance of the weak PHP. [MPW]

Open problem:

(4) Do the theories $\mathrm{T}_2^i(\alpha)$ have distinct (increasing) $\forall\Sigma_0^{\mathrm{b}}(\alpha)$-consequences?

- Note this would not have any (known) computational complexity implications.

(5) For $i \geq 1$, does depth i-Frege quasipolynomially simulate depth $(i{+}1)$-Frege with respect to refuting sets of clauses?

- Note that this is the nonuniform version of Question (4).

For (5): Best results to-date are a superpolynomial separation, based on upper and lower bounds for the pigeonhole principle. [IK]

Hastad switching lemma gives exponential separation of *expressibility* in depth i versus depth $i{+}1$. (!)
(5) asks: Does this extra expressiveness allow shorter proofs?

TFNP, Provably total functions

It is also interesting to study the $\forall\Sigma^{\mathrm{b}}_1$-consequences of the theories T^i_2. These define a subset of the TFNP problems:

Definition: [MP, P] A **Total NP Search Problem (TFNP)** is a polynomial time relation $R(x, y)$ so that R is

- *Total:* For all x, there exists y s.t. $R(x, y)$,
- *Polynomial growth rate:*
  If $R(x, y)$, then $|y| \leq p(|x|)$ for some polynomial p.
- The TFNP problem is:

  Given an input x, output a y s.t. $R(x, y)$.

Note the solution y may not be unique!

TFNP classes need to come with a proof of totality, usually either a combinatorial principle or a formal proof.

Pigeonhole Principle (PPP) [P]
Input: $x \in \mathbb{N}$ and a purportedly injective $f : [x] \to [x-1]$.
Output: $a, b \in [x]$ s.t. either $f(a) \notin [x-1]$ or $f(a) = f(b)$.

Parity principle (PPAD) [P]
Input: A directed graph G with in- and out-degrees ≤ 1, and a vertex v of total degree 1.
Output: Another vertex v' of total degree 1.

Polynomial Local Search (PLS) [JPY]
Input: A directed graph with out-degree ≤ 1, and a nonnegative cost function which strictly decreases along directed edges
Output: A sink vertex.

Proofs in bounded arithmetic also establish TFNP problems:

PLS - same as before
CPLS - PLS with a Herbrandized coNP (Π^{b}_1) accepting condition.

RAMSEY
Input: an undirected graph on n nodes.
Output: a clique or co-clique of size $\frac{1}{2}\log n$.

But, now the inputs are coded with a second-order object α.
The output is a first-order object.

Thm. The PLS function is provably total in $\mathrm{T}^1_2(\alpha)$, and is many-one complete for the provably total relations of $\mathrm{T}^1_2(\alpha)$. [BK]

Thm. The same holds for CPLS and $\mathrm{T}^2_2(\alpha)$. [KST]

Thm. $\mathrm{T}^3_2(\alpha)$ proves the totality of RAMSEY. [P]

See also: Game Induction [ST], Local Improvement [KNT,BB], ...

Open problems:

(6) Do the $\forall\Sigma^{\mathrm{b}}_1(\alpha)$ consequences (or, the provably total functions) of T^i_2 form a proper hierarchy (for $i = 2, 3, 4, \ldots$)?

(7) Does $\mathrm{T}^2_2(\alpha)$ prove the totality of RAMSEY?

The $\mathrm{T}^3_2(\alpha)$ proof of RAMSEY is essentially a refinement of the usual inductive combinatorial proof of the Ramsey theorem (via a reduction to the pigeonhole principle). It appears that proving RAMSEY in $\mathrm{T}^2_2(\alpha)$ would require a new method proof for Ramsey's theorem.

See also related results and questions for the theory of approximate counting, APC^2. [J,KT]

TFNP problems for stronger theories:

Consistency search problem for Frege proofs: [BB]
Input: A (purported) Frege proof of $\perp$.
Output: A local error in the proof.

Also introduced as the **Wrong proof** search problem [GP].

Thm.

- The Frege Consistency Search problem is provable in $\mathrm{U}^1_2(\alpha)$ and many-one complete for its provably total functions. [BB]
- The same holds for extended Frege and $\mathrm{V}^1_2(\alpha)$. [K, BB]

Here the input is coded by a second-order object; i.e., algorithms have *oracle* access to the Frege "proof" and seek a local error.

The "standard" TFNP problems are all included in the Consistency Search/Wrong Proof search classes for all these theories. [BB, GP]

Finis

Finis

Thank you!

Tseitin, 1968

> *The question of the minimum complexity of derivation of a given formula in classical propositional calculus is considered in this article and it is proved that estimates of complexity may vary considerably among the various forms of propositional calculus.*

Initiating the modern study of proof complexity.

Davis-Putnam, 1960

> *The idea of a refutation-algorithm, [...], is not new. In essence, it goes back to Herbrand, and [is] based on the idea of generating a sequence of quantifier-free lines, and then testing the conjunction of the first n lines for consistency as* $n = 1, 2, 3, \ldots$. *[...] However, the crucial difficulty, to which little attention appears to have been given in this connection, is that of finding a feasible technique for testing the conjunction of the first n lines for consistency when n is large.*

Anticipating the hardness of satisfiability for SAT solvers.

Many formal theories have some kind of translation to propositional logic. (!)