Proof Mining in Bounded Arithmetic

Amirhossein Akbar Tabatabai *†
Utrecht University

Abstract

A computational flow is a pair consisting of a sequence of computational problems of a certain sort and a sequence of computational reductions among them. In this paper we will develop a theory for these computational flows to design a classical direct reading of the Dialectica interpretation. This theory makes a sound and complete interpretation for bounded theories of arithmetic as well as the low complexity statements of strong unbounded systems. We will first use this fact to extract the computational content of the low complexity statements in some bounded theories of arithmetic such as IU_k, T^k_n, $I\Delta_0(\exp)$ and PRA. And then we will apply the theory on some strong unbounded mathematical theories such as PA and $\mathrm{PA} + \mathrm{TI}(\alpha)$ using the bridge of continuous cut elimination technique.

1 Introduction

Intuitively speaking, proofs are information carriers that transfer the informational content of the assumptions to the informational content of the conclusion. This open notion of content admits many different interpretations in many different disciplines. The most trivial one is the truth value which is preserved along any sound proof and consequently is the least informative one. But there are more useful examples. *The computational content* is one of them and it is no exaggeration to state that this type of content is one of the main players in proof theory and theoretical computer science. The reason is its widespread incarnations, from the witnesses of existential quantifiers a la Herbrand to the Gödel's Dialectica interpretation of higher order arithmetical statements. In this paper, we follow this line to introduce

*The author is supported by the ERC Advanced Grant 339691 (FEALORA)
†amir.akbar@gmail.com

another computational interpretation which can be seen as a classical and a more direct reading of the Dialectica interpretation and in the rest of this introduction we will try to explain its basic ideas.

Let us begin with the general idea of how any computational interpretation works by summarizing the process behind it: First, any interpretation needs to interpret a sentence as a computational problem for which the computational content roughly means *any way* that can solve the problem computationally. Then it should define a computational flow as a sequence of certain type of *simple methods* to transfer the previously defined content from one point to another. And finally, it should find a way to translate any formal proof of a given system to such a computational flow.

To implement these three stages in the case of our own interpretation, we need to define the game theoretic interpretation of formulas. The basic idea is the following: First interpret any quantifier-free formula $A(x_1, y_1, x_2, \ldots)$ as a game between the players $\forall$ and $\exists$ in which the first player, $\forall$, plays x_1 and then the second player, $\exists$, plays y_1 and they continue this process alternately. At the end of the game, if $A(x_1, y_1, x_2, \ldots)$ holds, the second player wins and otherwise the first one is the winner. Now, the sentence $\forall x_1 \exists y_1 \forall x_2 \ldots A(x_1, y_1, x_2, \ldots)$ simply means that the second player has a winning strategy and this strategy is exactly the *computational content* of the sentence $\forall x_1 \exists y_1 \forall x_2 \ldots A(x_1, y_1, x_2, \ldots)$. Now let us define the *simple methods* or the *ways that the information flows.* Assume that we have a second player's strategy to win the game $A(x_1, y_1, x_2, \ldots)$ and we want to provide a second player's strategy to win the game $B(u_1, v_1, u_2, \ldots)$. For this purpose, we define the computational reduction from $\forall u_1 \exists v_1 \forall u_2 \ldots B(u_1, v_1, u_2, \ldots)$ to $\forall x_1 \exists y_1 \forall x_2 \ldots A(x_1, y_1, x_2, \ldots)$ as a tuple of functions (f_i, g_i) with the lowest possible complexity such that f_i reads all u_j's for $j \leq i$ and y_k for $k < i$ and finds x_i, and g_i reads the same data plus y_i and computes v_{i+1} such that

$$A(f_1(u_1), y_1, f_2(u_1, u_2, y_1), \ldots) \to B(u_1, g_1(u_1, y_1), u_2, \ldots).$$

It is clear that these functions find a way to transfer any move from the game B to the game A and vice versa to transfer the winning strategy of the second player for the game A to the winning strategy for the game B. This completes the definition of a reasonable method that we sought for. But what about the simplicity of these methods? At the first glance, it seems that the low complexity of the reductions ensures the expected simplicity but unfortunately the reality is far from that. In fact, in some cases, while the complexity of the functions can be extremely low, verifying the truth of

$$A(f_1(u_1), y_1, f_2(u_1, u_2, y_1), \ldots) \to B(u_1, g_1(u_1, y_1), u_2, \ldots)$$

can be extremely high, non-trivial and non-syntactical. This is clearly not what we expect from a simple reduction. Hence, we also add a base weak theory $\mathcal{B}$ to the definition and we force the implication to be provable in $\mathcal{B}$. This condition makes the reductions simple and syntactical as we expect them to be.

Based on these reductions, it is now natural to define a computational flow as a sequence of reductions and try to transform any proof in any appropriate theory to a flow. This transformation is the core of a new proof mining method tailored specifically for the weak arithmetical theories. It can be seen as a more direct version of the Dialectica interpretation in which the usual higher type computability is replaced by the more faithful many-one reductions. It is important to mention that this theory is only applicable to bounded theories because of some technical reasons. However, we will use some classical proof theoretic methods to reduce strong unbounded theories to the bounded ones to apply the theory indirectly also on these stronger systems.

As the last part in this introduction, let us review the possible applications of the theory of flows. The theory will be useful to reprove some recent characterizations of search problems in Buss' hierarchy of bounded arithmetic via game induction principle [?], [?] or higher PLS problems [1]. We will also use it to generalize these results to prove some new characterizations of low-complexity search problems from higher order bounded theories of arithmetic and stronger theories such as $I\Delta_0(\exp)$ to extremely strong theories like PA and PA + TI(α) or any other theory for which we have a reasonable ordinal analysis.

2 Preliminaries

In this section we will introduce the language, the basic theory and the general notion of bounded theory of arithmetic. We will also provide some examples to show how these general notions unify the usual well-known examples.

Definition 2.1. By convention, we will assume that throughout this paper first order formulas are constructed by *literals*, meaning atomic formulas

$P(t_1, \cdots, t_n)$ and their negations $\neg P(t_1, \cdots, t_n)$, using the following positive connectives $\wedge$, $\vee$, $\forall$, $\exists$. By $\neg A$ we mean the formula defined inductively by De Morgan laws and $A \to B$ is just an abbreviation for $\neg A \vee B$. When the language has two constants 0 and 1, by $\top$ and $\bot$ we mean $0 = 0$ and $0 = 1$, respectively.

Definition 2.2. Let $\mathcal{L}$ be a first order language of arithmetic extending $\mathcal{L}_{\mathcal{R}} = \{0, 1, +, \dot{-}, \cdot, d(-,-), \leq\}$, where $x \dot{-} y$ and $d(x, y)$ are interpreted as $max\{0, x-y\}$ and $\lfloor \frac{x}{y+1} \rfloor$, in the standard model, respectively. By $\mathcal{R}$ we mean the first order theory in the language $\mathcal{L}_{\mathcal{R}}$ consisting of the following axioms:

- Axioms of the commutative semirings, namely the usual axioms of commutative rings except the existence of additive inverses,
- Axioms of discrete total orders. By totality we mean $x \leq y \vee y \leq x$ and by discreteness we mean

$$x \leq y + 1 \leftrightarrow (x \leq y) \vee (x = y),$$

- Axioms of compatibility, i.e., the axioms to state that addition and multiplication with non-zero elements respect the strict order $<$, where $x < y$ if $(x \leq y) \wedge (x \neq y)$,
- Defining axioms for $\dot{-}$ and d:

$$[x \geq y \to (x \dot{-} y) + y = x] \wedge [x < y \to x \dot{-} y = 0],$$

$$[(y + 1) \cdot d(x, y) \leq x] \wedge [x \dot{-} (y + 1) \cdot d(x, y) < y + 1].$$

  Note that to avoid division by zero and to have a total function symbol in the language, we defined division as $d(x, y) = \lfloor \frac{x}{y+1} \rfloor$ and not $\lfloor \frac{x}{y} \rfloor$,
- Axiom of non-triviality, i.e., $0 \neq 1$.

Remark 2.3. Here are some remarks on the basic properties that the theory $\mathcal{R}$ can represent:

- For any pair x and y either $x < y$ or $x = y$ or $y < x$, and only one of these cases happen. This is just a logical consequence of the definition of the strict order and the totality axiom.
- The stability conditions clearly extend to non-strict inequality $\leq$. This is also a logical consequence of the equivalence between $x \leq y$ and $x < y \vee x = y$.

- The discreteness of the order also implies that if $x < y$, then $x+1 \leq y$, because otherwise, $y < x+1$ which implies $y \leq x+1$ and by discreteness of the order we have either $y \leq x$ or $y = x$, both of which contradicts with $x < y$.

- The theory $\mathcal{R}$ proves the cancellation laws for addition and multiplication by non-zero elements, i.e.,

$$\mathcal{R} \vdash (x + y \leq x + z) \to (y \leq z) \qquad (*)$$

$$\mathcal{R} \vdash (x \neq 0 \wedge 0 \leq u, v < x) \to (xy + u \leq xz + v \to y \leq z) \qquad (**)$$

The argument is as usual. For $(*)$, assume $x + y \leq x + z$. Then by the totality of the order, we have either $y \leq z$ or $z < y$. If $z < y$, by the stability under the addition, we have $x + z < x + y$, which is a contradiction. For $(**)$, if $z < y$, we have $z + 1 \leq y$. Since $x \neq 0$, by the stability under multiplication, we have $x(z + 1) \leq xy$. Hence, by distributivity and commutativity we have

$$xz + u + x = xz + x + u \leq xy + u \leq xz + v$$

Therefore, by $(*)$, we have $x = 0 + x \leq u + x \leq v$, which is a contradiction. Hence, $y \leq z$. Note that the second cancelation law implies

$$\mathcal{R} \vdash (x \neq 0 \wedge 0 \leq u, v < x) \to [(xy + u = xz + v) \to (y = z) \wedge (u = v)]$$

because if $xy + u = xz + v$, then $xy + u \leq xz + v$. By $(**)$, we have $y \leq z$. Similarly, we have $z \leq y$, which implies $y = z$. Therefore, $xy = xz$ and by $(*)$, we have $u = v$.

- The theory $\mathcal{R}$ proves that $x0 = 0$, for any x, since by distributivity we have $x0 = x(0 + 0) = x0 + x0$ and then by the cancellation law we reach $x0 = 0$. Moreover, $\mathcal{R}$ proves that $x \geq 0$ for any x. First, note that $0 < 1$, because by the discreteness of the order we have $x \leq x + 1$ by $x \leq x$. Hence, $0 \leq 1$. But $0 \neq 1$ which means $0 < 1$. Then, since multiplying by non-zero x preserves the strict order, by $0 < 1$, we have $0 = x0 < x1 = x$. Hence, either $x = 0$ or $0 < x$, which imply $0 \leq x$.

- The cancellation laws provides a way to compute $x \dot{-} y$ and $d(x, y)$ by the following equalities:

$$\mathcal{R} \vdash (x + z = y) \to (x \dot{-} y = z),$$

$$\mathcal{R} \vdash (x = (y + 1)z + w \wedge w < y + 1) \to (d(x, y) = z).$$

The first holds because using $z \geq 0$, the equation $x + z = y$ implies $x \geq y$ and by the defining axiom of $\dot{-}$ we have $x = y + (x \dot{-} y)$. The claim then follows from the cancellation law for the addition. The proof of the second equality is similar to the first.

- The language $\mathcal{L}_{\mathcal{R}}$ is powerful enough to represent the conditional function:

$$C(x, y, z) = \begin{cases} y & x = 0 \\ z & x > 0 \end{cases}$$

by a term $c(x, y, z)$ such that

$$\mathcal{R} \vdash c(0, y, z) = y \wedge (x > 0 \to c(x, y, z) = z).$$

First note that the term $\chi_{=0}(x) = d(x+2, x) \dot{-} 1$ (computing $\lfloor \frac{x+2}{x+1} \rfloor \dot{-} 1$) represents the characteristic function of the relation $\{0\}$, provably in $\mathcal{R}$, i.e.,

$$\chi_{=0}(x) = \begin{cases} 1 & x = 0 \\ 0 & x > 0 \end{cases}$$

It is enough to show that $d(2, 0) \dot{-} 1 = 1$ and $d(x + 2, x) \dot{-} 1 = 1$ for $x > 0$. This is clear by the previous remark and the equalities

$$2 = 1 \cdot 1 + 1 \quad \text{and} \quad x + 2 = (x + 1) \cdot 1 + 1,$$

where $1 < x + 1$ because $0 < x$. Then define the representation term $c(x, y, z) = (\chi(x)y + (1 \dot{-} \chi(x))z$. It is clear that this c represents C. The only needed ingredients are the equalities $1 \dot{-} 0 = 1$ and $1 \dot{-} 1 = 0$, both of which are easy to prove from the previous remark.

- The relation $x \leq y$ is equivalent to $x \dot{-} y = 0$. One direction is clear from the defining axiom for $\dot{-}$. For the other direction, if $x > y$, by the defining axiom we have $x = (x \dot{-} y) + y$ which implies $x = y$; a contradiction. Given this equivalence, we can use the term $\chi_{=0}(x \dot{-} y)$ to represent $\chi_{\leq}(x, y)$, the characteristic function of $\leq$. Finally, since we have the power to simulate all boolean operators (multiplication for the conjunction and $x \mapsto 1 \dot{-} x$, for the negation) and $x = y$ is equivalent to $x \leq y \wedge y \leq x$, we can represent the characteristic functions of all the quantifier-free formulas of the language $\mathcal{L}_{\mathcal{R}} = \{0, 1, +, \dot{-}, \cdot, d(-, -), \leq\}$, meaning that for any quantifier-free formula $A(\vec{x}) \in \mathcal{L}_{\mathcal{R}}$, there exists an $\mathcal{L}_{\mathcal{R}}$-term $t(\vec{x})$ such that

$$\mathcal{R} \vdash [t(\vec{x}) = 0 \to A(\vec{x})] \wedge [t(\vec{x}) \neq 0 \to \neg A(\vec{x})].$$

This fact plays a crucial role later in the paper.

Convention. Let A, B, C be some formulas whose characterisitic functions are representable in the language $\mathcal{L}$. When we define a term as

$$p(\vec{x}) = \begin{cases} t(\vec{x}) & A(\vec{x}) \\ s(\vec{x}) & B(\vec{x}) \\ r(\vec{x}) & C(\vec{x}) \end{cases}$$

we mean $p(\vec{x}) = \chi_A(\vec{x})t(\vec{x}) + \chi_B(\vec{x})s(\vec{x}) + \chi_C(\vec{x})r(\vec{x})$ and when we define a formula

$$D(\vec{x}) = \begin{cases} A'(\vec{x}) & A(\vec{x}) \\ B'(\vec{x}) & B(\vec{x}) \\ C'(\vec{x}) & C(\vec{x}) \end{cases}$$

we mean $D(\vec{x}) = [A(\vec{x}) \to A'(\vec{x})] \wedge [B(\vec{x}) \to B'(\vec{x})] \wedge [C(\vec{x}) \to C'(\vec{x})]$. Note that if Φ is a class of formulas including all quantifier-free formulas and closed under disjunction and conjunction, then it is also closed under this definition by cases.

In the following, we define a general notion of bounded arithmetic we want to investigate in this paper. For that purpose, we first need to define some complexity classes:

Definition 2.4. Let $\mathcal{L} \supseteq \mathcal{L}_{\mathcal{R}}$ be a first-order language. Then the hierarchy $\{\Sigma_k^b(\mathcal{L}), \Pi_k^b(\mathcal{L})\}_{k=0}^{\infty}$ is recursively defined in the following manner:

(*i*) $\Pi_0^b(\mathcal{L}) = \Sigma_0^b(\mathcal{L})$ is the class of all quantifier-free formulas in $\mathcal{L}$,

(*ii*) $\Sigma_k^b(\mathcal{L}) \cup \Pi_k^b(\mathcal{L}) \subseteq \Sigma_{k+1}^b(\mathcal{L}) \cap \Pi_{k+1}^b(\mathcal{L})$,

(*iii*) $\Pi_k^b(\mathcal{L})$ and $\Sigma_k^b(\mathcal{L})$ are closed under conjunction and disjunction,

(*iv*) If $B(x) \in \Sigma_k^b(\mathcal{L})$ then $\exists x \leq t\ B(x) \in \Sigma_k^b(\mathcal{L})$ and $\forall x \leq t\ B(x) \in \Pi_{k+1}^b(\mathcal{L})$ and

(*v*) If $B(x) \in \Pi_k^b(\mathcal{L})$ then $\forall x \leq t\ B(x) \in \Pi_k^b(\mathcal{L})$ and $\exists x \leq t\ B(x) \in \Sigma_{k+1}^b(\mathcal{L})$.

A formula is called bounded if it is in $\bigcup_{k=0}^{\infty} \Sigma_k^b(\mathcal{L}) = \bigcup_{k=0}^{\infty} \Pi_k^b(\mathcal{L})$.

Remark 2.5. Note that the classes $\Sigma_k^b(\mathcal{L})$ and $\Pi_k^b(\mathcal{L})$ include all the quantifier-free formulas. They are also closed under sub-formulas and substitutions. We will need these properties in the free-cut elimination theorem for the arithmetical theories we are interested in.

Example 2.6. The classes $\Sigma_k^b(\mathcal{L}_{\mathcal{R}})$ and $\Pi_k^b(\mathcal{L}_{\mathcal{R}})$ are essentially the same as the classes E_k and U_k in the linear hierarchy. Moreover, if we use the language of bounded arithmetic, $\mathcal{L}_m$, augmented with subtraction, division and $\#_i$ for $2 \leq i \leq m$, the classes $\Pi_k^b(\mathcal{L}_m)$ and $\Sigma_k^b(\mathcal{L}_m)$ captures $\hat{\Pi}_k^b(\#_m)$ and $\hat{\Sigma}_k^b(\#_m)$, respectively.

From now on, whenever the first-order language $\mathcal{L} \supseteq \mathcal{L}_{\mathcal{R}}$ is fixed and there is no risk of confusion, we will drop the letter $\mathcal{L}$ in $\Sigma_k^b(\mathcal{L})$ and $\Pi_k^b(\mathcal{L})$.

Definition 2.7. Let $\mathcal{A} \supseteq \mathcal{R}$ be a set of quantifier-free formulas and Φ be a class of bounded formulas extending the class of all quantifier-free formulas and closed under substitution and subformulas. By the first-order bounded arithmetic $\mathfrak{B}(\Phi, \mathcal{A})$, we mean the first-order theory in the language $\mathcal{L}$ that consists of the axioms $\mathcal{A}$, and the Φ-induction axiom, i.e.,

$$A(0) \wedge \forall x(A(x) \to A(x+1)) \to \forall x A(x),$$

where $A \in \Phi$.

Example 2.8. With our definition of bounded arithmetic, different kinds of theories can be considered as bounded theories of arithmetic, for instance $I\Delta_0$, T_n^k, $I\Delta_0(\exp)$ and PRA augmented with subtraction and division in the language and the axioms of $\mathcal{R}$ in the theory, are just some of the well-known examples.

It is possible to represent the theory $\mathfrak{B}(\Phi, \mathcal{A})$, by the following sequent-style calculus:

Axioms:

$$\overline{L \Rightarrow L} \qquad \overline{P, \neg P \Rightarrow} \qquad \overline{\Rightarrow P, \neg P} \qquad \overline{\Rightarrow A}$$

where L is a literal, P is an atomic formula and in the rightmost sequent, the formula A is a substitution of a formula in $\mathcal{A}$.

Equality:

$$\overline{\Rightarrow t = t} \qquad \overline{t_1 = s_1, \ldots, t_n = s_n \Rightarrow f(t_1, \ldots, t_n) = f(s_1, \ldots, s_n)}$$

$$\overline{t_1 = s_1, \ldots, t_n = s_n, L(t_1, \ldots, t_n) \Rightarrow L(s_1, \ldots, s_n)}$$

where f is a function symbol and L is a literal.

Structural Rules:

$$\frac{\Gamma \Rightarrow \Delta}{\Gamma, A \Rightarrow \Delta}\,(wL) \qquad \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, A}\,(wR)$$

$$\frac{\Gamma, A, B, \Sigma \Rightarrow \Delta}{\Gamma, B, A, \Sigma \Rightarrow \Delta}\,(eL) \qquad \frac{\Gamma \Rightarrow \Delta, A, B, \Lambda}{\Gamma \Rightarrow \Delta, B, A, \Lambda}\,(eR)$$

$$\frac{\Gamma, A, A \Rightarrow \Delta}{\Gamma, A \Rightarrow \Delta}\,(cL) \qquad \frac{\Gamma \Rightarrow \Delta, A, A}{\Gamma \Rightarrow \Delta, A}\,(cR)$$

$$\frac{\Gamma \Rightarrow \Delta, A \qquad \Gamma, A \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(cut)$$

Propositional Rules:

$$\frac{\Gamma, A \Rightarrow \Delta}{\Gamma, A \wedge B \Rightarrow \Delta}\,\wedge L_1 \quad \frac{\Gamma, B \Rightarrow \Delta}{\Gamma, A \wedge B \Rightarrow \Delta}\,\wedge L_2 \quad \frac{\Gamma \Rightarrow \Delta, A \qquad \Gamma \Rightarrow \Delta, B}{\Gamma \Rightarrow \Delta, A \wedge B}\,\wedge R$$

$$\frac{\Gamma, A \Rightarrow \Delta \qquad \Gamma, B \Rightarrow \Delta}{\Gamma, A \vee B \Rightarrow \Delta}\,\vee L \quad \frac{\Gamma \Rightarrow \Delta, A}{\Gamma \Rightarrow \Delta, A \vee B}\,\vee R_1 \quad \frac{\Gamma \Rightarrow \Delta, B}{\Gamma \Rightarrow \Delta, A \vee B}\,\vee R_2$$

Quantifier rules:

$$\frac{\Gamma, A(s) \Rightarrow \Delta}{\Gamma, \forall y\ A(y) \Rightarrow \Delta}\,\forall L \quad \frac{\Gamma \Rightarrow \Delta, A(a)}{\Gamma \Rightarrow \Delta, \forall y\ A(y)}\,\forall R$$

$$\frac{\Gamma, A(a) \Rightarrow \Delta}{\Gamma, \exists y\ A(y) \Rightarrow \Delta}\,\exists L \quad \frac{\Gamma \Rightarrow \Delta, A(s)}{\Gamma, \Rightarrow \Delta, \exists y\ A(y)}\,\exists R$$

where in the rules ($\forall R$) and ($\exists L$), the variable a must not occur in the lower sequent of the rules.

Induction:

$$\frac{\Gamma, A(a) \Rightarrow \Delta, A(a+1)}{\Gamma, A(0) \Rightarrow \Delta, A(t)}\,(\Phi - Ind)$$

for every $A \in \Phi$, where the variable a must not occur in the lower sequent of the rule.

Theorem 2.9. *([2]) Let $\Gamma \cup \Delta \subseteq \Phi$, where Φ is a class of bounded formulas extending the class of all quantifier-free formulas and closed under substitution and subformulas. Then if $\mathfrak{B}(\Phi, \mathcal{A}) \vdash \Gamma \Rightarrow \Delta$, there exists a proof of the sequent $\Gamma \Rightarrow \Delta$ in the previously defined sequent-style system such that all formulas occurring in the proof are in the class Φ.*

Theorem 2.10. *For any set $\mathcal{A} \supseteq \mathcal{R}$ of quantifier-free axioms and any $k \geq 0$, we have $\mathfrak{B}(\Pi_k^b(\mathcal{L}), \mathcal{A}) = \mathfrak{B}(\Sigma_k^b(\mathcal{L}), \mathcal{A})$.*

Proof. We show that $\mathfrak{B}(\Pi_k^b(\mathcal{L}), \mathcal{A}) \subseteq \mathfrak{B}(\Sigma_k^b(\mathcal{L}), \mathcal{A})$. The other case is similar. For that matter, it is enough to prove that the $\Pi_k^b(\mathcal{L})$-induction axiom

$$A(0) \wedge \forall x(A(x) \to A(x+1)) \to \forall x A(x),$$

is provable in $\mathfrak{B}(\Sigma_k^b(\mathcal{L}), \mathcal{A})$. For the sake of contradiction, let us assume that $A(0) \wedge \forall x(A(x) \to A(x+1))$ but $\neg A(b)$, for some b. Define $B(y) = \neg A(b \dot{-} y)$. It is clear that $B(y) \in \Sigma_k^b(\mathcal{L})$. Now, we need some computations. By Remark **??**, it is easy to see that $b \dot{-} b = 0$, because $0 + b = b$. Moreover, observe that if $b < y+1$, we have $b \dot{-} (y+1) = b \dot{-} y = 0$, by the defining axioms of $\mathcal{R}$, also available in $\mathfrak{B}(\Sigma_k^b(\mathcal{L}), \mathcal{A})$. And if $y + 1 \leq b$, then we have $[b \dot{-} (y+1)] + 1 = (b \dot{-} y)$, because $b \dot{-} (y+1) + (y+1) = b$ by the defining axiom and by Remark **??**, since $[b \dot{-} (y+1)] + 1 + y = b$, we can compute $b \dot{-} y$ as $[b \dot{-} (y+1)] + 1$. Now, since $\neg A(b)$, we have $B(0)$. And since $\forall x(A(x) \to A(x+1))$, we have $\neg A(x+1) \to \neg A(x)$. If $y + 1 \leq b$, put $x = b \dot{-} y = [b \dot{-} (y+1)] + 1$ which implies $B(y) \to B(y+1)$. And if $b < y + 1$, then $= b \dot{-} y = [b \dot{-} (y+1)]$ which implies again that $B(y) \to B(y+1)$. By $\Sigma_k^b(\mathcal{L})$-induction on $B(y)$, we can prove $B(b)$ which implies $\neg A(b \dot{-} b)$. Therefore, we have $\neg A(0)$ which is a contradiction. □

3 Reductions and Flows

In this section, we will introduce the two main notions of the paper. The first is the notion of a reduction as the building block of the flow of the computational information. These reductions are the generalization of the usual polynomial-time reductions between total NP search problems and the deterministic reductions between k-turn games as introduced in [5]. Then we will generalize reductions to their iterated version that we call flows. In Section 4, we will use these flows to transform any proof in a bounded theory of arithmetic to a term-length sequence of provably simple steps.

Definition 3.1. Let $\mathcal{L} \supseteq \mathcal{L}_{\mathcal{R}}$ be a first-order language. A theory $\mathcal{B} \supseteq \mathcal{R}$ is called a *base theory* for the language $\mathcal{L}$ if:

(i) There exists a $\mathcal{B}$-provable *family of monotone majorizing $\mathcal{L}$-terms*, i.e., a set M of the $\mathcal{L}$-terms such that for any $t(\vec{x}) \in M$ we have $\mathcal{B} \vdash \vec{x} \leq \vec{y} \to t(\vec{x}) \leq t(\vec{x})$ and for any arbitrary $\mathcal{L}$-term $s(\vec{x})$ there exists $r(\vec{x}) \in M$ such that $\mathcal{B} \vdash s(\vec{x}) \leq r(\vec{x})$.

(*ii*) For any quantifier-free formula $A(\vec{x}) \in \mathcal{L}$, there exists an $\mathcal{L}$-term $t(\vec{x})$ such that $\mathcal{B} \vdash [t(\vec{x}) = 0 \to A(\vec{x})] \wedge [t(\vec{x}) \neq 0 \to \neg A(\vec{x})]$. We call this term the *characteristic function for the formula $A(\vec{x})$.*

Example 3.2. Note that the theory $\mathcal{R}$ is a base theory for the language $\mathcal{L}_{\mathcal{R}}$. It is just enough to use the class of all polynomials as the majorizing family M. The first part of the definition is clear, using the inequalities $x \dot{-} y \leq x$ and $d(x, y) \leq x$ and the fact that all the polynomials are monotone in $\mathcal{R}$. For the second part, see Remark 2.3.

Definition 3.3. By a prenex formula A we mean any formula in the form $\vec{Q}\vec{x}B(\vec{x}, \vec{y})$, where $\vec{Q}$ is a block of quantifiers over distinct variables $\vec{x}$ and B is a quantifier-free formula. Let C be any formula in the language $\mathcal{L}$. By the σ-prenex form of C, we mean the result of the following process: After changing the quantified variables to make them distinct, if necessary, first bring out all the existential quantifiers in any preferred order and then all the universal quantifiers again in any preferred order, alternatively till they end. If we begin by the universal quantifiers, the result is called the π-prenex form of C.

Remark 3.4. Notice that the σ- and π-prenex forms of a formula are not unique and depend on the new names of the variables and the order of quantifiers in each block. However, we will use these prenex forms in the forthcoming notion of the reduction which totally ignores these types of differences. Therefore, for the sake of simplicity, we will also ignore the differences and pretend that these forms are unique.

Example 3.5. Consider the formula

$$A = \forall y B(x, y) \vee \exists z \neg B(x, z)$$

where B is quantifier-free. Then the σ- and π-prenex forms of A are

$$\exists z \forall y [B(x, y) \vee \neg B(x, z)] \qquad \forall y \exists z [B(x, y) \vee \neg B(x, z)]$$

respectively. Note that if the quantifiers are also bounded, the prenex form leaves the bounds inside the formula and just brings out the quantifier. For instance, the σ-prenex forms of $\forall y \leq t(x) B(x, y) \vee \exists z \leq t(x) \neg B(x, z)$ is

$$\exists z \forall y [[y \leq t(x) \to B(x, y)] \vee [z \leq t(x) \wedge \neg B(x, z)]]$$

Definition 3.6. Let $\alpha \in \{\sigma, \pi\}$ and $A(\vec{x})$ and $B(\vec{x})$ be some formulas in the prenex form with at most k many alternations of quantifiers, $F = \{\vec{F}_i\}_{i=1}^{k}$

be a sequence of sequences of terms and $\mathcal{B}$ a base theory. By recursion on the number of quantifier alternations, we will define $F = \{\vec{F}_i\}_{i=1}^k$ as an $(\mathcal{B}, \alpha)$-reduction from $B(\vec{x})$ to $A(\vec{x})$ and we will denote it by $A(\vec{x}) \geq_{\alpha}^{\mathcal{B},F} B(\vec{x})$, when:

- (i) If $A(\vec{x}), B(\vec{x})$ are quantifier-free, any sequence of sequences of terms is both a $(\mathcal{B}, \sigma)$- and a $(\mathcal{B}, \pi)$-reduction from B to A iff $\mathcal{B} \vdash A(\vec{x}) \to B(\vec{x})$.
- (ii) If $\alpha = \pi$ and $A = \forall \vec{u} C(\vec{x}, \vec{u})$, $B = \forall \vec{v} D(\vec{x}, \vec{v})$, where the universal quantifiers are the whole block of left-most universal quantifiers (possibly empty), then $A(\vec{x}) \geq_{\pi}^{\mathcal{B},F} B(\vec{x})$ iff

$$C(\vec{x}, \vec{F}_{k+1}(\vec{x}, \vec{v})) \geq_{\sigma}^{\mathcal{B},\hat{F}} D(\vec{x}, \vec{v})$$

  where $\hat{F} = \{\vec{F}_i\}_{i=1}^{k-1}$.
- (iii) If $\alpha = \sigma$ and $A = \exists \vec{u} C(\vec{x}, \vec{u})$, $B = \exists \vec{v} D(\vec{x}, \vec{v})$, where the existential quantifiers are the whole block of left-most existential quantifiers (possibly empty), then $A(\vec{x}) \geq_{\sigma}^{\mathcal{B},F} B(\vec{x})$ iff

$$C(\vec{x}, \vec{u}) \geq_{\pi}^{\mathcal{B},\hat{F}} D(\vec{x}, \vec{F}_{k+1}(\vec{x}, \vec{u}))$$

  where $\hat{F} = \{\vec{F}_i\}_{i=1}^{k-1}$.

It is possible to extend the definition to all formulas $A(\vec{x})$ and $B(\vec{x})$ in the following way: We say $F = \{\vec{F}_i\}_{i=1}^k$ is an $(\mathcal{B}, \alpha)$-reduction from $B(\vec{x})$ to $A(\vec{x})$ iff $F = \{\vec{F}_i\}_{i=1}^k$ is an $(\mathcal{B}, \alpha)$-reduction from $\tilde{B}(\vec{x})$ to $\tilde{A}(\vec{x})$, where $\tilde{A}(\vec{x})$ and $\tilde{B}(\vec{x})$ are the α-prenex forms of A and B, respectively.
Finally, we say that B is $(\mathcal{B}, \pi)$-reducible to A and we write $A \geq_{\pi}^{\mathcal{B}} B$, when there exists a sequence of sequences of terms F such that $A \geq_{\pi}^{\mathcal{B},F} B$. Moreover, by the equivalence $A \equiv_{\pi}^{\mathcal{B},E,F} B$, we mean the conjunction of $A \geq_{\pi}^{\mathcal{B},E} B$ and $B \geq_{\pi}^{\mathcal{B},F} A$ and we define $(\mathcal{B}, \sigma)$-reducibility and equivalence dually by replacing π to σ, everywhere. Note that whenever the theory $\mathcal{B}$ is clear from the context, we drop it from the superscripts, everywhere.

The computational way to interpret a $(\mathcal{B}, \pi)$-reduction $F = \{\vec{F}_i\}_{i=1}^k$ from $B(\vec{x})$ to $A(\vec{x})$ is the following: We first read the first block of the universal quantifiers in $B(\vec{x})$ as the new input, added to the base input $\vec{x}$. Then using the all the inputs so far, we use the terms $\vec{F}_k$ as our computational instructions to witness the first block of the universal quantifiers in $A(\vec{x})$. Then we start with the first block of the existential quantifiers in $A(\vec{x})$. We read and then add them as the new input to our previously aggregated inputs. Then using these inputs and the term $\vec{F}_{k-1}$, we witness the first block of the existential quantifiers in $B(\vec{x})$. Then we continue with the universal quantifiers in $B(\vec{x})$ again, till we reach the last block of quantifiers in the formulas. How do we know that our witnessing process is sound? When we reach the quantifier-free case, then the implication $C(\vec{x}, \vec{y}) \to D(x, \vec{y})$ must be provable in $\mathcal{B}$. The similar process also works for a $(\mathcal{B}, \sigma)$-reduction with the difference that it now begins with the existential block of $A(\vec{x})$.

Example 3.7. (*Self-witnessing* and *Ignoring* Techniques)
Let $A(x) = \forall y \exists z B(x, y, z)$, $A'(x) = \forall y' \exists z' B'(x, y', z')$, $C(x) = \exists u D(x, u)$ and $C'(x) = \exists u' D'(x, u')$ be some formulas, where B, B', D, D' are quantifier-free and $\mathcal{B}$ be a base theory. Since both of the formulas are in the prenex form, a $(\mathcal{B}, \pi)$-reduction from $A'(x)$ to $A(x)$ is a pair of terms like (t, s) such that

$$\forall y \exists z B(x, y, z) \geq_{\pi}^{\mathcal{B},(s,t)} \forall y' \exists z' B'(x, y', z')$$

meaning that t reads x, y' to witness $y = t(y')$ and s reads x, y', z to witness z' such that

$$\mathcal{B} \vdash B(x, t(x, y'), z) \to B'(x, y', s(x, y', z))$$

Now, we will assume the existence of such a reduction, to construct other reductions via the simple techniques that we call *self-witnessing* and *ignoring.* Let $A(x) \geq_{\pi}^{\mathcal{B},(s,t)} A'(x)$. We want to provide a canonical term reduction constructed from (s, t) to show $A(x) \wedge C(x) \geq_{\pi}^{\mathcal{B}} A'(x) \wedge C(x)$. To show how, we have to write both sides in their π-prenex form, meaning that we have to show

$$\forall y \exists z \exists u [B(x, y, z) \wedge D(x, u)] \geq_{\pi}^{\mathcal{B},t,s,r} \forall y' \exists z' \exists v [B'(x, y', z') \wedge D(x, v)]$$

Note that we changed the name of the bounded variable of right-hand side $C(x)$ to v, as the process of π-prenexing dictates. Now, the canonical $(\mathcal{B}, \pi)$-reduction here is the following: Use $t(x, y')$ to witness y and $s(x, y', z)$ to witness z' as before. But to witness v, the most natural candidate is u. The last thing to check is whether

$$\mathcal{B} \vdash [B(x, t(x, y), z) \wedge D(x, u)] \to [B'(x, y', s(x, y', z)) \wedge D(x, u)]$$

which is an easy consequence of

$$\mathcal{B} \vdash B(x, t(x, y), z) \to B'(x, y', s(x, y', z))$$

It is possible to describe the whole argument in the following rough line: To provide a reduction for $A(x) \wedge C(x) \geq_{\pi}^{\mathcal{B}} A'(x) \wedge C(x)$, use the reduction for $A(x) \geq_{\pi}^{\mathcal{B}} A'(x)$ and witness the quantifiers in C by themselves. This is the simple technique we call self-witnessing. To see how the technique simplifies the explanation, note that if $A(x) \geq_\pi^{\mathcal{B}} A'(x)$ then we also have $\forall x A(x) \geq_\pi^{\mathcal{B}} \forall x A'(x)$. The new reduction is just the previous one, plus witnessing x in the left by itself, read on the right.

The second technique, the ignoring, works as follows. Let $A(x) \geq_\pi^{\mathcal{B},(s,t)} A'(x)$. We want to provide a canonical term reduction constructed from (s, t) to show $A(x) \wedge C(\vec{x}) \geq_\pi^{\mathcal{B}} A'(x) \vee C'(x)$. After making both sides π-prenex, we have to show that

$$\forall y \exists z \exists u [B(x, y, z) \wedge D(x, u)] \geq_\pi^{\mathcal{B},t,s,r} \forall y' \exists z' \exists u [B'(x, y', z') \vee D'(x, u')]$$

Again, it is enough to witness y by $t(x, y')$ and z' by $s(x, y', z)$. But when it comes to witness u', We do not need u and we do not care how to witness u', because the implication holds, regardless the extra data of $D(x, u)$ and the value of u'. Therefore, it is enough first to ignore the input u and then to witness u' by some arbitrary term, usually the constant zero. Then, we have to have

$$\mathcal{B} \vdash [B(x, t(x, y), z) \wedge D(x, u)] \rightarrow [B'(x, y', s(x, y', z)) \vee D'(x, 0)]$$

which is clearly provable. This is what we call the ignoring technique. We say ignore some variables, both as the inputs and the ones we have to witness, when such a situation happens.

Example 3.8. In this example we will draw the reader's attention to the difference between π- and σ-reductions. Consider the formula

$$A = \forall y B(x, y) \vee \exists z \neg B(x, z)$$

where $B(x, y)$ is quantifier-free. Working with π-reductions, it is clear that we have $\top \geq_\pi A$, because reading the variable y, it is enough to witness the variable z by the term $t(y) = y$. Then we will reach the $\mathcal{B}$-provable formula

$$\top \rightarrow B(x, y) \vee \neg B(x, y)$$

However, if we work with the σ-reductions, the order of the variables changes and we have to first witness z, only using x and without any knowledge of the value y. This highly depends on the terms of the language and it usually is not the case. The reason is as follows. Suppose that $\top \geq_\sigma^{\mathcal{R}} A$. Then, there must be a term $t(x)$ such that

$$\mathcal{R} \vdash \top \rightarrow [B(x, y) \vee \neg B(x, t(x))]$$

Therefore, checking the truth value of $B(x, t(x))$, which is a computable task, we can decide whether $\forall y B(x, y)$ holds, because if $B(x, t(x))$ is true, then since $\mathcal{R}$ is sound, $\forall y(x,y) \vee \neg B(x,t(x))]$ must be true and hence $\forall y B(x,y)$ will be true. If $B(x,t(x))$ is false then the formula $\forall y B(x,y)$ is clearly false. This contradicts the existence of the uncomputable Diophantine sets.

Looking at the computational process that we explained, it is not hard to see that a reduction form B to A are some specific computational methods to witness the existential quantifiers of $A \to B$ by its universal quantifiers, in some certain order and only using the terms of the language, in the eyes of a base theory $\mathcal{B}$. Therefore, it is reasonable to expect that:

Theorem 3.9. *Let $\alpha \in \{\sigma, \pi\}$, $\mathcal{B}$ be a base theory and $A(\vec{x})$ and $B(\vec{x})$ be some formulas. Then, if $A(\vec{x}) \geq_{\alpha}^{\mathcal{B}} B(\vec{x})$ then $\mathcal{B} \vdash A(\vec{x}) \to B(\vec{x})$.*

Proof. First note that the α-prenex form of any formula is logically equivalent to itself. Hence, w.l.o.g we can assume that both $A(\vec{x})$ and $B(\vec{x})$ are in α-prenex form. Then, the claim is easy by an induction of the maximum number of the quantifier alternations of the formulas $A(\vec{x})$ and $B(\vec{x})$. For quantifier-free formulas there is nothing to prove. If $\alpha = \pi$, then we have $A = \forall \vec{u} C(\vec{x}, \vec{u})$ and $B = \forall \vec{v} D(\vec{x}, \vec{v})$, where the universal quantifiers are the whole block of left-most universal quantifiers (possibly empty)and terms $\vec{t}(\vec{x}, \vec{v})$ such that

$$C(\vec{x}, \vec{t}(\vec{x}, \vec{v})) \geq_{\sigma}^{\mathcal{B}} D(\vec{x}, \vec{v})$$

By the induction hypothesis, we have

$$\mathcal{B} \vdash C(\vec{x}, \vec{t}(\vec{x}, \vec{v})) \to D(\vec{x}, \vec{v})$$

But since the formula $\forall \vec{u} C(\vec{x}, \vec{u})$ logically implies $C(\vec{x}, \vec{t}(\vec{x}, \vec{v}))$, we have

$$\mathcal{B} \vdash \forall \vec{u} C(\vec{x}, \vec{u}) \to D(\vec{x}, \vec{v})$$

Since $\vec{v}$ is not free in $\forall \vec{u} C(\vec{x}, \vec{u})$, we finally reach $\mathcal{B} \vdash A(\vec{x}) \to B(\vec{x})$. The case $\alpha = \sigma$ is similar. □

A natural question is whether the converse of Theorem **??** also holds, meaning whether the computational method of reduction faithfully extract the computational information from the proofs in $\mathcal{B}$? The answer is negative. As an example, it is just enough to use the formula A in the Example **??**. This formula is a logical tautology and hence provable in any base theory including $\mathcal{R}$. But, as we observed there is no $(\mathcal{R}, \sigma)$-reduction from A to $\top$. Although, it may not be clear now, we will see that the main obstruction to have the converse is exactly such a situation: There are some formulas whose characteristic functions are not representable by the terms of the language.

As it is very well-known, the gap between the computable functions (a reasonable arithmetical term represent such a function) and the decision problem for the arithmetical formulas is enormous and hence there is no hope to somehow modify the statement of Theorem **??** such that its converse also becomes true. However, for bounded formulas A and B, this modification is somehow imaginable. Here, the natural setting is using the low complexity terms (polynomial-time computable function for instance) as the basic functions and then try to use them to extract the computational information from the proofs. By the same line of argument as in Example **??**, we can again show that the converse of Theorem **??** for bounded formulas also lead to the low complexity decision procedures for the bounded existential problems like the NP problems. However, in the bounded case, we have a modification to handle these complex decision problems. The ides is the following: One computational method to decide bounded formulas is the brute force technique to open all bounded quantifiers and check all the simple possibilities once at a time. This is a very long sequence of simple decision problems. We know that it is impossible to simulate this process by one reduction. But what if we use a long sequence of them. It may be possible to simulate any simple decision problem by a reduction and then using a uniform sequence of reductions to do the decision for any bounded formula. As we will see throughout this paper, this is possible and when we handle these decision problems, we can witness any proof not only in a base theory but in any bounded theory of arithmetic. The following notion of a flow is the sequence of reductions that we need:

Definition 3.10. Let $A(\vec{x}), B(\vec{x}) \in \Pi_k^b(\mathcal{L})$ be two formulas and $\alpha \in \{\sigma, \pi\}$. A $(\Pi_k(\mathcal{L}), \mathcal{B}, \alpha)$-flow from $A(\vec{x})$ to $B(\vec{x})$ is the following data: A term $t(\vec{x})$, a formula $H(u, \vec{x}) \in \Pi_k^b(\mathcal{L})$ and sequences of terms E_0, E_1, G_0, G_1 and $F(u)$ such that:

(i) $H(0, \vec{x}) \equiv_{\alpha}^{\mathcal{B}, E_0, E_1} A(\vec{x})$.

(ii) $H(t(\vec{x}), \vec{x}) \equiv_{\alpha}^{\mathcal{B}, G_0, G_1} B(\vec{x})$.

(iii) $H(u, \vec{x}) \geq_{\alpha}^{\mathcal{B}, F(u)} H(u+1, \vec{x})$.

If there exists a $(\Pi_k^b(\mathcal{L}), \mathcal{B}, \alpha)$-flow from $A(\vec{x})$ to $B(\vec{x})$, we will write $A(\vec{x}) \rhd_{\alpha}^{(\Pi_k^b(\mathcal{L}), \mathcal{B})} B(\vec{x})$. Moreover, if Γ and Δ are sequences of formulas in $\Pi_k^b(\mathcal{L})$, by $\Gamma \rhd_{\alpha}^{(\Pi_k^b(\mathcal{L}), \mathcal{B})} \Delta$ we mean $\bigwedge \Gamma \rhd_{\alpha}^{(\Pi_k^b(\mathcal{L}), \mathcal{B})} \bigvee \Delta$. The case for $(\Sigma_k^b(\mathcal{L}), \mathcal{B}, \alpha)$-flows is defined similarly by replacing $\Pi_k^b(\mathcal{L})$ with $\Sigma_k^b(\mathcal{L})$.

Convention. Let $\phi \in \{\sigma, \pi\}$. Since throughout the paper, except in the applications, the language is fixed and we want to address some statements more compact, we use the notation Φ_k by which we mean $\Pi_k^b(\mathcal{L})$ if $\phi = \pi$ and $\Sigma_k^b(\mathcal{L})$ if $\phi = \sigma$. Moreover, when we work with a fixed choice for $\mathcal{B}$, we drop the letter $\mathcal{B}$ from $\rhd_\alpha^{(\Phi,\mathcal{B})}$ or the tuple $(\Phi, \mathcal{B}, \alpha)$. Moreover, we will use the notation $\mathcal{C}(\mathcal{L})$ for $\{\Sigma_k^b(\mathcal{L}), \Pi_k^b(\mathcal{L})\}_{k=0}^{\infty}$ and Φ for a variable over $\mathcal{C}(\mathcal{L})$.

Theorem 3.11. *(Completeness) Let $\phi \in \{\sigma, \pi\}$, $A(\vec{x}), B(\vec{x}) \in \Phi_k$ and $\mathcal{B} \subseteq \mathfrak{B}(\Phi_k, \mathcal{A})$ be a base theory. If $A(\vec{x}) \rhd_\phi^{(\Phi_k,\mathcal{B})} B(\vec{x})$, then $\mathfrak{B}(\Phi_k, \mathcal{A}) \vdash A(\vec{x}) \to B(\vec{x})$.*

Proof. If $A(\vec{x}) \rhd_\phi^{(\Phi_k,\mathcal{B})} B(\vec{x})$, then by Theorem 3.9, there exist a term $t(\vec{x})$, and a formula $H(u, \vec{x}) \in \Phi_k$ such that:

(*i*) $\mathcal{B} \vdash H(0, \vec{x}) \leftrightarrow A(\vec{x})$,

(*ii*) $\mathcal{B} \vdash H(t(x), \vec{x}) \leftrightarrow B(\vec{x})$,

(*iii*) $\mathcal{B} \vdash H(u, \vec{x}) \to H(u+1, \vec{x})$.

Since $\mathcal{B} \subseteq \mathfrak{B}(\Phi_k, \mathcal{A})$, all of these three facts are also provable in $\mathfrak{B}(\Phi_k, \mathcal{A})$. Since $H(u, \vec{x}) \in \Phi_k$, by induction we have

$$\mathfrak{B}(\Phi_k, \mathcal{A}) \vdash H(0, \vec{x}) \to H(t(\vec{x}), \vec{x}).$$

Therefore, by (*i*) and (*ii*) we have $\mathfrak{B}(\Phi_k, \mathcal{A}) \vdash A(\vec{x}) \to B(\vec{x})$. □

4 The Main Theorem

Now we are ready to state the main theorem of the paper. The theorem relates the provability of an implication between a pair of bounded formulas in a bounded theory of arithmetic to the existence of a flow between the formulas, meaning a uniform term-length sequence of reductions between them. The latter can also be interpreted as the existence of a uniform term-length sequence of games with a uniform term-based sequence of methods to transfer the winning strategies along them.

Theorem 4.1. *(Main Theorem) Let $\phi \in \{\sigma, \pi\}$, $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Phi_k$ and $\mathcal{B}$ be a base theory such that $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathfrak{B}(\Phi_k, \mathcal{A})$. Then, $\mathfrak{B}(\Phi_k, \mathcal{A}) \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff $\Gamma \rhd_\phi^{(\Phi_k,\mathcal{B})} \Delta$.*

We have already proved one direction in Theorem 3.11. For the other half, we need a high-level calculus to argue about the existence of the flows. After a sequence of lemmas to bootstrap the calculus, we will come back to the main theorem as in Theorem 4.1.

Lemma 4.2. *(Conjunction Application) Let $\alpha \in \{\sigma, \pi\}$, $\Phi \in \mathcal{C}(\mathcal{L})$ and $A(\vec{x}), B(\vec{x}), C(\vec{x}) \in \Phi$ be some formulas. If $A(\vec{x}) \rhd_\alpha^\Phi B(\vec{x})$ then $A(\vec{x}) \wedge C(\vec{x}) \rhd_\alpha^\Phi B(\vec{x}) \wedge C(\vec{x})$.*

Proof. Since $A(\vec{x}) \rhd_\alpha^\Phi B(\vec{x})$, by Definition 3.10, there exists a term $t(\vec{x})$, a formula $H(u, \vec{x}) \in \Phi$ and sequences of terms E_0, E_1, G_0, G_1 and $F(u)$ such that:

- $A(\vec{x}) \equiv_\alpha^{\mathcal{B}, E_0, E_1} H(0, \vec{x})$,
- $B(\vec{x}) \equiv_\alpha^{\mathcal{B}, G_0, G_1} H(t(\vec{x}), \vec{x})$,
- $H(u, \vec{x}) \geq_\alpha^{\mathcal{B}, F(u)} H(u+1, \vec{x})$.

Now define $t'(\vec{x}) = t(\vec{x})$ and $H'(u, \vec{x}) = H(u, \vec{x}) \wedge C(\vec{x})$. Since Φ is closed under conjunction, we have $H'(u, \vec{x}) \in \Phi$. Define the sequences of terms E'_0, E'_1, G'_0, G'_1 and $F'(u)$ using their corresponding sequence of terms augmented by some self-witnessing as introduced in Example 3.7. It means that to witness a variable in $C(\vec{x})$, use the same variable read on the other side of the reduction and to witness a variable outside of $C(\vec{x})$ use the corresponding term from the given appropriate reduction. Therefore, it is clear that the new data is a $(\Phi, \mathcal{B}, \alpha)$-flow from $A(\vec{x}) \wedge C(\vec{x})$ to $B(\vec{x}) \wedge C(\vec{x})$. $\square$

Lemma 4.3. *(Disjunction Application) Let $\alpha \in \{\sigma, \pi\}$, $\Phi \in \mathcal{C}(\mathcal{L})$ and $A(\vec{x}), B(\vec{x}), C(\vec{x}) \in \Phi$ be some formulas. If $A(\vec{x}) \rhd_\alpha^\Phi B(\vec{x})$ then $A(\vec{x}) \vee C(\vec{x}) \rhd_\alpha^\Phi B(\vec{x}) \vee C(\vec{x})$.*

Proof. The proof is similar to that of 4.2. The only difference here is the definition $H'(u, \vec{x})$ as $H(u, \vec{x}) \vee C(\vec{x})$, which is clearly in Φ. $\square$

Lemma 4.4. *(Gluing) Let $\alpha \in \{\sigma, \pi\}$, $\Phi \in \mathcal{C}(\mathcal{L})$ and $A(\vec{x}), B(\vec{x}), C(\vec{x}), D(y, \vec{x}) \in \Phi$ be some formulas and $s(\vec{x})$ be a term:*

(i) (Weak Gluing) If $A(\vec{x}) \rhd_\alpha^\Phi B(\vec{x})$ and $B(\vec{x}) \rhd_\alpha^\Phi C(\vec{x})$ then $A(\vec{x}) \rhd_\alpha^\Phi C(\vec{x})$.

(ii) (Strong Gluing) If $D(y, \vec{x}) \rhd_\alpha^\Phi D(y+1, \vec{x})$ then $D(0, \vec{x}) \rhd_\alpha^\Phi D(s, \vec{x})$.

Proof. For (i), since $A(\vec{x}) \rhd_\alpha^\Phi B(\vec{x})$, there exists a term $t(\vec{x})$, a formula $H(u, \vec{x}) \in \Phi$ and sequences of terms E_0, E_1, G_0, G_1 and $F(u)$ satisfying the conditions in the Definition 3.10. With the same reason, since $B(\vec{x}) \rhd_\alpha^\Phi C(\vec{x})$, we have the corresponding data denoted by $t'(\vec{x})$, $H'(u, \vec{x})$, E'_0, E'_1, G'_0, G'_1 and $F'(u)$. Define $r(\vec{x}) = t(\vec{x}) + t'(\vec{x}) + 1$, the formula $I(u, \vec{x})$ as

$$I(u, \vec{x}) = \begin{cases} H(u, \vec{x}) & u \leq t(\vec{x}) \\ B(\vec{x}) & u = t(\vec{x}) + 1 \\ H'(u \dot{-} t(\vec{x}) \dot{-} 2, \vec{x}) & t(\vec{x}) + 1 < u \leq t(\vec{x}) + t'(\vec{x}) + 2 \end{cases}$$

and the sequence of reduction terms in the same pointwise manner: $E''_0 = E_0$, $E''_1 = G'_1$ and

$$F''(u) = \begin{cases} F(u) & u < t(\vec{x}) \\ E_1 & u = t(\vec{x}) \\ E'_0 & u = t(\vec{x}) + 1 \\ F'(u \dot{-} t(\vec{x}) \dot{-} 2, \vec{x}) & t(\vec{x}) + 1 < u < t(\vec{x}) + t'(\vec{x}) + 2 \end{cases}$$

It is easy to see that this new data is nothing but the result of gluing the two given sequences of reductions, meaning a $(\Phi, \mathcal{B}, \alpha)$-flow from $A(\vec{x})$ to $C(\vec{x})$. However, to be more precise we will discuss the details of the construction here. Later, we will use the informal style of reasoning more often.
First note that as mentioned in Convention **??**, the formula $I(u, \vec{x})$ is technically defined as the following formula

$$([u \leq t(\vec{x})] \to H(u, \vec{x})) \wedge ([u = t(\vec{x}) + 1] \to B(\vec{x}))$$

$$\wedge([t(\vec{x}) + 1 < u \leq t(\vec{x}) + t'(\vec{x}) + 2] \to H'(u \dot{-} t(\vec{x}) \dot{-} 2, \vec{x}))$$

and since Φ includes all the quantifier-free formulas and it is closed under disjunctions and conjunctions, we have $I(u, \vec{x}) \in \Phi$.
To show how the reductions work, we will handle the first case, where $u < t(\vec{x})$. Here, by Remark **??**, we have $F''(u) = F(u)$. It is easy to see that it would be enough to show that $I(u, \vec{x}) \geq^F I(u+1, \vec{x})$. Since $u, u+1 \leq t(\vec{x})$, in both $I(u, \vec{x})$ and $I(u+1, \vec{x})$, the second and the third conjuncts in $I(u, \vec{x})$ are true, regardless of the witnesses we may use for the quantifiers involved in those parts. Therefore, we can use the ignoring technique here, to say that the original $F(u)$ is also an $(\mathcal{B}, \alpha)$-reduction here.

For (ii), if we have $A(y, \vec{x}) \rhd^{\Phi}_{\alpha} A(y+1, \vec{x})$, then to reach the intended consequence, it is just enough to glue all copies of the sequences of reductions for $0 \leq y \leq s$ after each other, to have a reduction to show $A(0, \vec{x}) \rhd^{\Phi}_{\alpha} A(s, \vec{x})$.

$$A(0) \geq H(0,0) \geq H(1,0) \ldots H(t',0) \geq A(1) \geq H(0,1) \geq \ldots \geq H(t', s \dot{-} 1) \geq A(s)$$

Spelling out the details, first note that we will intuitively use the data from the reduction $A(y, \vec{x}) \rhd^{\Phi}_{\alpha} A(y+1, \vec{x})$ only for $y \leq s$. Therefore, w.l.o.g we can assume that after s, the flow becomes stationary with fix formula and identity as the reduction. Now, assume that all the reductions have the same length $t'(\vec{x})$ greater than $t(s, \vec{x})$. This is an immediate consequence of the existence of a majorizing family. First find a monotone majorization for $t(y, \vec{x})$ like $r(y, \vec{x})$. Since $y \leq s$, we have $t(y, \vec{x}) \leq r(y, \vec{x}) \leq r(s, \vec{x})$. Now it is enough to repeat the last formula in the flow to make the flow longer to reach the length $t'(\vec{x}, \vec{z}) = r(s, \vec{x})$ where $\vec{z}$ is a vector of variables in s. Now, define $t''(\vec{x}, \vec{z}) = s(t'(\vec{x})+2)$, $Y(u, \vec{x}) = d(u, t'+1)$, $U(u, \vec{x}) = u \dot{-} d(u, t'+1)(t'+2)$ and

$$I(u, \vec{x}) = \begin{cases} H(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x}), \vec{x}) & U(u, \vec{x}) > 0 \\ A(y, \vec{x}) & U(u, \vec{x}) = 0 \end{cases}$$

and

$$F'(u) = \begin{cases} F(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x})) & 0 < U(u, \vec{x}) < t'+1 \\ E_0(Y(u, \vec{x})) & U(u, \vec{x}) = 0 \\ G_1(Y(u, \vec{x})) & U(u, \vec{x}) = t'+1 \end{cases}$$

and $E'_0 = E'_1 = G'_0 = G'_1 = id$. It is easy to see that this new sequence is a $(\Phi, \mathcal{B}, \alpha)$-flow from $A(0, \vec{x})$ to $A(s, \vec{x})$. First note that since $H, A \in \Phi$, we have $I \in \Phi$. For the reductions, we only check one case. If $0 < U(u, \vec{x}) < t'+1$, we want to show that

$$I(u, \vec{x}) \geq_{\alpha}^{\mathcal{B}, F'(u)} I(u+1, \vec{x})$$

By definition the only part of the formula $I(u, \vec{x})$ that plays a role in the witnessing process is the part $H(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x}), \vec{x})$. But since $F(u, y)$ is a $(\mathcal{B}, \alpha)$-reduction from $H(u+1, y, \vec{x})$ to $H(u, y, \vec{x})$, for any $y \leq s$, and since $Y(u, \vec{x}) \leq s$, the reduction $F(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x}))$ provides a reduction

$$H(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x}), \vec{x}) \geq^{F(U(u, \vec{x}) \dot{-} 1, Y(u, \vec{x}))} H(U(u, \vec{x}), Y(u, \vec{x}), \vec{x})$$

□

Lemma 4.5. *(Quantifier Application) Let* $\alpha \in \{\sigma, \pi\}$. *Then:*

(i) If $A(\vec{x}, y), B(\vec{x}, y) \in \Pi_k^b(\mathcal{L})$ *and* $A(\vec{x}, y) \rhd_{\pi}^{\Pi_k^b(\mathcal{L})} B(\vec{x}, y)$ *then*

$$\forall y \leq t(\vec{x}) A(\vec{x}, y) \rhd_{\alpha}^{\Pi_k^b(\mathcal{L})} \forall y \leq t(\vec{x}) B(\vec{x}, y).$$

(ii) If $A(\vec{x}, y), B(\vec{x}, y) \in \Sigma_k^b(\mathcal{L})$ *and* $A(\vec{x}, y) \rhd_{\sigma}^{\Sigma_k^b(\mathcal{L})} B(\vec{x}, y)$ *then*

$$\exists y \leq t(\vec{x}) A(\vec{x}, y) \rhd_{\alpha}^{\Sigma_k^b(\mathcal{L})} \exists y \leq t(\vec{x}) B(\vec{x}, y).$$

Proof. For (i), since $A(\vec{x}, y) \rhd_{\pi}^{\Pi_k^b(\mathcal{L})} B(\vec{x}, y)$, there exists a term $s(\vec{x}, y)$, a formula $H(u, \vec{x}, y) \in \Pi_k^b(\mathcal{L})$ and sequences of sequences of terms E_0, E_1, G_0, G_1 and $F(u)$ satisfying the conditions of Definition 3.6. W.l.o.g we can assume that s is monotone, because any term is majorizable by a monotone term and we can extend the sequence by repeating the last formula to reach that majorization as the length of the flow. Define $t'(\vec{x})$ as $s(\vec{x}, t(\vec{x}))$ and $H'(u, \vec{x}) = \forall y \leq t(\vec{x}) I(u, \vec{x}, y)$ where

$$I(u, \vec{x}, y) = \begin{cases} H(u, y, \vec{x}) & u \leq s(\vec{x}, y) \\ H(s(\vec{x}, y), y, \vec{x}) & u \nleq s(\vec{x}, y) \end{cases}$$

and also define

$$F'(u) = \begin{cases} F(u) & u + 1 \leq s(\vec{x}, y) \\ Id & u + 1 \nleq s(\vec{x}, y) \end{cases}$$

It is clear that $H'(0, \vec{x}) \equiv_{\pi}^{\mathcal{B}} \forall y \leq t \ A(\vec{x}, y)$. Because, it is possible to self-witness y and then to witness the other quantifiers according to the reductions E_0 and E_1. The reason is that $0 \leq s(\vec{x}, y)$ and hence the main part of $\forall y \leq t \ I(0, \vec{x}, y)$ in the witnessing process is just $H(0, y, \vec{x})$.

Secondly, we have $H'(u, \vec{x}) \geq_{\pi}^{\mathcal{B}} H'(u + 1, \vec{x})$, by self-witnessing the outmost quantifier $\forall y$ and then applying $F'(u)$. Thirdly, $H'(t'(\vec{x}), \vec{x}) \equiv_{\pi}^{\mathcal{B}} \forall y \leq tB(\vec{x}, y)$ by the reductions which read y and witness it by itself and then apply the reductions G_0 and G_1. To prove this claim, first note that we can assume $y \leq t$, because otherwise, both sides of the reduction will be false regardless of the reduction. Then using $y \leq t$ and the monotonicity of s we have $s(\vec{x}, y) \leq t'(\vec{x})$ which implies $I(t'(\vec{x}), \vec{x}, y) = I(s(\vec{x}, y), y, \vec{x})$ and since $I(s(\vec{x}, y), y, \vec{x}) = H(s(\vec{x}, y), y, \vec{x})$ is π-equivalent to B by the reductions G_0, G_1, the claim follows. Finally note that all the formulas in the flow begin with a universal quantifier, therefore, we can also claim that all the reductions are σ-reductions and hence the flow is also a σ-flow. The proof of (ii) is similar. □

The following lemma provides a machinery to compute the value of the formula $A \in \Phi_k \in \{\Pi_k, \Sigma_k\}$ by a deterministic $(\Sigma_{k+1}, \mathcal{B}, \alpha)$-flow of reductions for any $\alpha \in \{\pi, \sigma\}$. This is a very important tool to reduce the complexity of deciding a complex formula to just deciding one equality. We will see its use in full force in the case of handling the contraction rule.

Lemma 4.6. *(Computability of the characteristic functions) Let $\alpha \in \{\pi, \sigma\}$, $A(\vec{x}) \in \Phi_k$ and $\mathcal{B}$ be a base theory. Then:*

$$\rhd_{\alpha}^{(\Sigma_{k+1}, \mathcal{B})} \exists i \leq 1 \ [(i = 1 \to A) \wedge (i = 0 \to \neg A)]$$

Proof. We say a bounded quantifier is constant if it has the form $\forall z \leq s(z = s \to D(z))$ or $\exists z \leq t(z = s \wedge D(z))$ for some term s. We denote these quantifiers by $\forall\{z = s\}$ and $\exists\{z = s\}$. To prove the theorem, use induction on the sum of the number of non-constant quantifiers of A and the number of disjunctions and conjunctions of A.

If all the quantifiers in A are constant, then it is enough to first eliminate all the quantifiers in A by substituting the variables by the constant terms that the constant quantifiers suggest, i.e., substituting the variable z in the quantifier $Q\{z = s\}$ by s. Call this quantifier-free formula B and put $i = \chi_B$. If we witness all the essentially existential quantifiers by the terms that they suggest, then we reach the implication

$$(\chi_B = 1 \to B) \wedge (\chi_B = 0 \to \neg B)$$

which is provable in $\mathcal{B}$ by the assumption.

If $A = \vec{Q}\{\vec{z} = \vec{s}\}(B \wedge C)$ where $Q_n \in \{\forall, \exists\}$, then by IH,

$$\rhd_{\alpha}^{(\Sigma_{k+1}, \mathcal{B})} \exists j \leq 1\ [(j = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}B) \wedge (j = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}B)]$$

and

$$\rhd_{\alpha}^{(\Sigma_{k+1}, \mathcal{B})} \exists k \leq 1\ [(k = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}C) \wedge (k = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}C)].$$

On the other hand, it is possible to reduce

$$\exists i \leq 1\ [(i = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}(B \wedge C)) \wedge (i = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}(B \wedge C))]$$

to the conjunction of two statements

$$\exists j \leq 1\ [(j = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}B) \wedge (j = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}B)]$$

and

$$\exists k \leq 1\ [(k = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}C) \wedge (k = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}C)].$$

To prove this, witness i by jk, the quantifiers in $\vec{Q}$ by the terms that they suggest and the other quantifiers with themselves. Therefore, by conjunction application and then gluing, we have

$$\rhd_{\alpha}^{(\Sigma_{k+1}, \mathcal{B})} \exists i \leq 1\ [(i = 1 \to \vec{Q}\{\vec{z} = \vec{s}\}(B \wedge C)) \wedge (i = 0 \to \neg\vec{Q}\{\vec{z} = \vec{s}\}(B \wedge C))].$$

The case for disjunction is similar to the conjunction case.

If $A = \vec{Q}\{\vec{v} = \vec{s}\}\forall z \leq t(\vec{x})B(\vec{x}, z)$ where $Q_n \in \{\forall, \exists\}$, then define $G(u)$ as

$$\exists k \leq 1\ [(k = 1 \to \vec{Q}\tilde{B}(\vec{x}, u)) \wedge (k = 0 \to \neg\vec{Q}\tilde{B}(\vec{x}, u))].$$

where $\tilde{B}(\vec{x}, u)$ is $\forall\{z = u\}B(\vec{x}, z)$ and $\vec{Q}$ stands for $\vec{Q}\{\vec{v} = \vec{s}\}$. By IH we have a $(\Sigma_{k+1}, \mathcal{B}, \alpha)$-flow from $\top$ to $G(u+1)$ which is

$$\exists k \leq 1 \; [(k = 1 \rightarrow \vec{Q}\tilde{B}(\vec{x}, u+1)) \wedge (k = 0 \rightarrow \neg\vec{Q}\tilde{B}(\vec{x}, u+1))]$$

Define $H(u)$ as

$$\exists i \leq 1 \; [(i = 1 \rightarrow \vec{Q}\forall z \leq u \; B(\vec{x}, z)) \wedge (i = 0 \rightarrow \neg\vec{Q}\forall z \leq u \; B(\vec{x}, z))].$$

Now, we want to prove the existence of a reduction from $H(u+1)$ which is

$$\exists j \leq 1 \; [(j = 1 \rightarrow \vec{Q}\forall z \leq u+1 \; B(\vec{x}, z)) \wedge (j = 0 \rightarrow \neg\vec{Q}\forall z \leq u+1 \; B(\vec{x}, z))].$$

to the conjunction of $G(u+1)$ and $H(u)$. For this purpose, witness j by ik. Then for the other quantifiers use the following scheme: Note that we have three possible cases, the case when $i = k = 1$, the case $i = 1$, $k = 0$ and the case $i = 0$. In each case, some parts of the formulas, will be true regardless of the reduction that we will present. Hence, we ignore them altogether and we call the other formulas the main formulas.

Now, if $i = k = 1$, then the main formulas are $\vec{Q}\forall z \leq u \; B(\vec{x}, z)$, $\tilde{B}(\vec{x}, u+1)$ and $\vec{Q}\forall z \leq u+1 \; B(\vec{x}, z)$. To reduce $\vec{Q}\forall z \leq u+1 \; B(\vec{x}, z)$ to the conjunction of $\vec{Q}\forall z \leq u \; B(\vec{x}, z)$ and $\vec{Q}\tilde{B}(\vec{x}, u+1)$, first witness constant quantifiers in $\vec{Q}$ by the terms that they suggest. Then read $z \leq u+1$, if $z = u+1$ use $\tilde{B}(\vec{x}, u+1)$ and $\forall z \leq u+1 \; B(\vec{x}, z)$ as the main formulas and ignore $\forall z \leq u \; B(\vec{x}, z)$. Then witness the last universal quantifier of $\tilde{B}(\vec{x}, u+1)$ by $u+1$ and all the other quantifiers in $\forall z \leq u+1 \; B(\vec{x}, z)$ and $\tilde{B}(\vec{x}, u+1)$ with themselves. If $z < u+1$, then use $\forall z \leq u \; B(\vec{x}, z)$ and $\forall z \leq u+1 \; B(\vec{x}, z)$ as the main formulas and again witness everything with themselves. If $i = 1$ and $k = 0$, then use $\neg\vec{Q}\forall z \leq u+1 \; B(\vec{x}, z)$ and $\neg\vec{Q}\tilde{B}(\vec{x}, u+1)$ as the main formulas and witness constant quantifiers in $\vec{Q}$ by the terms that they suggest. Then use $u+1$ for z and witness all the variables with themselves. Finally if $i = 0$, then use $\neg\vec{Q}\forall z \leq u \; B(\vec{x}, z)$ and $\neg\vec{Q}\forall z \leq u+1 \; B(\vec{x}, z)$ as the main formulas and witness constant quantifiers in $\vec{Q}$ by the terms that they suggest and all the other variables with themselves.

Therefore $G(u+1) \wedge H(u) \rhd H(u+1)$. By IH, $\rhd G(u+1)$. Hence, by conjunction application $H(u) \rhd G(u+1) \wedge H(u)$ and then by gluing $H(u) \rhd H(u+1)$ and finally by strong gluing $H(0) \rhd H(t(\vec{x}))$. Since $H(0) \equiv G(0)$ and $\rhd G(0)$, hence $\rhd H(0)$ which means $\rhd H(t(\vec{x}))$.

The case $A = \vec{Q}\{\vec{v} = \vec{s}\}\exists z \leq t(\vec{x})B(\vec{x}, z)$ is similar to the universal case. □

Lemma 4.7. *(Canonical Normal Form) For any formula $C(\vec{x}) \in \Sigma_{k+1}$, there exists $\tilde{C}(\vec{x}, \vec{u}) \in \Pi_k$ and some terms $\vec{s}$ such that the σ-prenex form of $\tilde{C}(\vec{x}, \vec{u})$ is quantifier-free or it begins with a universal quantifier and $\exists \vec{u} \leq \vec{s}\, \tilde{C}(\vec{x}, \vec{u})$ is σ-deterministic equivalent to C. The same also holds for universal quantifiers, Π_{k+1} and π-equivalence.*

Proof. We will prove the claim by induction on the complexity of C. If C is quantifier-free, then pick $\tilde{C} = C$ and pick $\vec{u}$ as the empty vector. If C begins with a universal formula then $\tilde{C} = C$ and pick $\vec{u}$ as the empty vector again. If $C = \exists y \leq t\; D$, then pick $\tilde{C} = \tilde{D}$ and add y to $\vec{u}$ and t to $\vec{s}$. For the cases $C = D \wedge E$ or $C = D \vee E$, since their proofs are similar, we will only check the disjunction case. If $C = D \vee E$, by IH, there exists $\tilde{C}$ and $\tilde{D}$ and $\vec{s}$ and $\vec{r}$ such that $C \equiv_\sigma \exists \vec{u} \leq \vec{s}\; \tilde{C}$ and $D \equiv_\sigma \exists \vec{v} \leq \vec{r}\; \tilde{D}$. Hence by propositional rules, it is clear that

$$C \vee D \equiv_\sigma \exists \vec{u} \leq \vec{s}\; \tilde{C} \vee \exists \vec{v} \leq \vec{r}\; \tilde{D}$$

But

$$\exists \vec{u_0} \leq \vec{s}\; \tilde{C} \vee \exists \vec{v_0} \leq \vec{r}\; \tilde{D} \equiv_\sigma \exists \vec{u_1} \leq \vec{s}\; \exists \vec{v_1} \leq \vec{r}\; (\tilde{C} \vee \tilde{D})$$

(For the moment we put some indices for the variables $\vec{u}$ and $\vec{v}$ for the referring purpose.) To show the latter, for both reductions, when we read an existential quantifier $w \in \vec{u} \cup \vec{v}$ with the bound p, if $w_i \leq p$ use w_i to witness w_{1-i}, if not just use zero. From right to left, if at least for one variable w_1 we have $w_1 > p$, then this choice for the variable w_1 makes the left hand-side of the reduction false, regardless the choice of the other variables, which implies the reduction. If for all the variables we have $w_1 \leq p$, then after using identity reduction both sides will be equal and there is nothing to prove. For the other direction, let $\vec{u_0}'$ and $\vec{v_0}'$ be the variables that do not meet their bounds in $\vec{u_0}$ and $\vec{v_0}$, respectively. If both $\vec{u_0}'$ and $\vec{v_0}'$ have some variables, as before, it makes both $\exists \vec{u_0} \leq \vec{s}\; \tilde{C}$ and $\exists \vec{v_0} \leq \vec{r}\; \tilde{D}$ false and hence we have the reduction. If $\vec{u_0}$ is non-empty and $\vec{v_0}$ is empty, then $\exists \vec{u_0} \leq \vec{s}\; \tilde{C}$ is false, regardless of the other parts of the reduction. Since we choose zero to witness the variables $\vec{u_1}'$, all $\vec{u_1}$ meet their bounds and therefore the reduction is reduced to the fact that the disjunction of $\tilde{D}$ and a substitution of $\tilde{C}$ is reducible to $\tilde{D}$. The proof for the other cases are similar. □

Lemma 4.8. *(Negation Rules) If $\Gamma, \Delta \subseteq \Phi_{k+1}$ and $A \in \Pi_k \cup \Sigma_k$ then*

(*i*) *If $\Gamma, A \rhd_\phi^{\Phi_{k+1}} \Delta$ then $\Gamma \rhd_\phi^{\Phi_{k+1}} \Delta, \neg A$.*

(*ii*) *If* $\Gamma \rhd_{\phi}^{\Phi_{k+1}} \Delta, A$ *then* $\Gamma, \neg A \rhd_{\phi}^{\Phi_{k+1}} \Delta$.

Proof. Since we have conjunction and disjunction application, it is enough to prove the claim:

Claim. If $A(\vec{x}) \in \Pi_k \cup \Sigma_k$, then

$(*)$ $\top \rhd_{\phi}^{\Phi_{k+1}} A(\vec{x}) \vee \neg A(\vec{x})$.

$(**)$ $A(\vec{x}) \wedge \neg A(\vec{x}) \rhd_{\phi}^{\Phi_{k+1}} \bot$.

The reason for this sufficiency is the following:

For (i), if we have $\Gamma, A \rhd \Delta$ then $\bigwedge \Gamma \wedge A \rhd \bigvee \Delta$, hence by disjunction application we have $(\bigwedge \Gamma \wedge A) \vee \neg A \rhd \bigvee \Delta \vee \neg A$. By the claim we have $\rhd A \vee \neg A$, therefore by conjunction application $\bigwedge \Gamma \rhd \bigwedge \Gamma \wedge (A \vee \neg A)$. But, it is easy to see that $\bigwedge \Gamma \wedge (A \vee \neg A) \geq (\bigwedge \Gamma \wedge A) \vee \neg A$. Hence by gluing we have $\bigwedge \Gamma \rhd \bigvee \Delta \vee \neg A$.

For (ii), we have $\bigwedge \Gamma \rhd \bigvee \Delta \vee A$. By conjunction application $\bigwedge \Gamma \wedge \neg A \rhd (\bigvee \Delta \vee A) \wedge \neg A$. By the claim we have $A \wedge \neg A \rhd \bot$ therefore by disjunction application $\bigvee \Delta \vee (A \wedge \neg A) \rhd \bigvee \Delta$. But, it is clear that $(\bigvee \Delta \vee A) \wedge \neg A \geq \bigvee \Delta \vee (A \wedge \neg A)$. Hence by gluing, $\bigwedge \Gamma \wedge \neg A \rhd \bigvee \Delta$.

Now, we will prove both $(*)$ and $(**)$ for the class Σ_{k+1}. For the other two cases for Π_{k+1}, we will use the following duality argument: Note that using negation on all the elements of a $(\Sigma_{k+1}, \mathcal{B}, \sigma)$-flow from C to D provides a $(\Pi_{k+1}, \mathcal{B}, \pi)$-flow from $\neg D$ to $\neg C$. Therefore, the Π_{k+1} case of $(*)$ is provable from the Σ_{k+1} case of $(**)$ and the Π_{k+1} case of $(**)$ is provable from the Σ_{k+1} case of $(*)$.

Assume $\Phi_{k+1} = \Sigma_{k+1}$. For $(*)$, notice that

$$\exists i \leq 1 \; [(i = 1 \to A) \wedge (i = 0 \to \neg A)] \geq_{\sigma} A \vee \neg A$$

it is enough to witness A and $\neg A$ in both sides with themselves. But since

$$\rhd_{\sigma}^{(\Sigma_{k+1}, \mathcal{B})} \exists i \leq 1 \; [(i = 1 \to A) \wedge (i = 0 \to \neg A)]$$

by propositional rules and gluing we can deduce $\rhd_{\sigma}^{(\Sigma_{k+1}, \mathcal{B})} A \vee \neg A$.

For $(**)$, use induction on the complexity of A. If A is quantifier-free, then there is nothing to prove. If $A = B \wedge C$, by IH, $B \wedge \neg B \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$ and $C \wedge \neg C \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$ since

$$(B \wedge C) \wedge \neg(B \wedge C) \geq_{\sigma} (B \wedge \neg B) \vee (C \wedge \neg C)$$

witnessing any quantifier by itself, using gluing we will have

$$(B \wedge C) \wedge \neg(B \wedge C) \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$$

The case for the disjunction is similar.

If A begins with a universal quantifier, by Lemma 4.7, there exists A' such that $A \equiv_{\pi} A' = \forall \vec{z} \leq \vec{t}\; B(\vec{z}) \in \Pi_k \cup \Sigma_k$ where $\forall \vec{z} \leq \vec{t}$ is the whole left-most block of bounded universal quantifiers and $B \in \Sigma_{k-1}$. Then by the above considerations on duality, since we have

$$\rhd_{\sigma}^{(\Sigma_k,\mathcal{B})} B(\vec{w}) \vee \neg B(\vec{w})$$

hence

$$B(\vec{w}) \wedge \neg B(\vec{w}) \rhd_{\pi}^{(\Pi_k,\mathcal{B})}$$

Now by Lemma 4.5 we have

$$\exists \vec{w} \leq \vec{t} \forall \vec{z} \leq \vec{t}\, [B(\vec{w}) \wedge \neg B(\vec{w})] \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$$

Now note that

$$\exists \vec{w} \leq \vec{t} \neg B(\vec{w}) \wedge \forall \vec{z} \leq \vec{t} B(\vec{z}) \geq_{\sigma} \exists \vec{w} \leq \vec{t} \forall \vec{z} \leq \vec{t} [B(\vec{w}) \wedge \neg B(\vec{w})]$$

because we can witness $\vec{w}$ by itself and $\vec{z}$ by $\vec{w}$. The main point here is that σ-prenex form of $\neg B(\vec{w})$ do not begin with an existential quantifier and hence after reading the first block of existential quantifiers, the formula $\neg B(\vec{w})$ remains intact. Therefore,

$$\exists \vec{w} \leq \vec{t} \neg B(\vec{w}) \wedge \forall \vec{z} \leq \vec{t} B(\vec{z}) \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$$

hence $A' \wedge \neg A' \rhd_{\sigma}^{(\Sigma_{k+1},\mathcal{B})}$. Finally, since A begins with at least one universal quantifier and $A \equiv_{\pi} A'$ we have $A \equiv_{\sigma} A'$. On the other hand, $\neg A \equiv_{\sigma} \neg A'$ and hence $A \wedge \neg A \equiv_{\sigma} A' \wedge \neg A'$ which completes the proof.

The case for the existential quantifier is similar. □

In the following lemma, we will show that it is possible to simulate the contraction rule by deterministic reductions in the cost of extending one reduction to a sequence of them, i.e., a flow.

Lemma 4.9. *(Structural rules)*

(*i*) *If* $\Gamma, A, B, \Sigma \rhd^{\Phi}_{\alpha} \Delta$ *then* $\Gamma, B, A, \Sigma \rhd^{\Phi}_{\alpha} \Delta$.

(*ii*) *If* $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A, B, \Sigma$ *then* $\Gamma \rhd^{\Phi}_{\alpha} \Delta, B, A, \Sigma$.

(*iv*) *If* $\Gamma \rhd^{\Phi}_{\alpha} \Delta$ *then* $\Gamma, A \rhd^{\Phi}_{\alpha} \Delta$.

(*v*) *If* $\Gamma \rhd^{\Phi}_{\alpha} \Delta$ *then* $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A$.

(*iii*) *If* $\Gamma, A, A \rhd^{\Phi}_{\alpha} \Delta$ *then* $\Gamma, A \rhd^{\Phi}_{\alpha} \Delta$.

(*vi*) *If* $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A, A$ *then* $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A$.

Proof. The weakening and the exchange cases are trivial. For the contraction case notice that in the presence of conjunction and disjunction applications and also the gluing rule, it is enough to prove the following claim:

Claim. For any $\alpha \in \{\pi, \sigma\}$, if $A \in \Phi$, then:

(*i*) $A(\vec{x}) \vee A(\vec{x}) \rhd^{\Phi}_{\alpha} A(\vec{x})$.

(*ii*) $A(\vec{x}) \rhd^{\Phi}_{\alpha} A(\vec{x}) \wedge A(\vec{x})$.

For (*i*), use induction on the complexity of A. If A is quantifier-free, then there is nothing to prove, because $A \vee A \equiv^{\Phi}_{\alpha} A \equiv^{\Phi}_{\alpha} A \wedge A$.

If $A = B \wedge C$, then by IH, $B \vee B \rhd^{\Phi} B$ and $C \vee C \rhd^{\Phi} C$. But $(B \wedge C) \vee (B \wedge C) \geq (B \vee B) \wedge (C \vee C)$ because it is just enough to witness any quantifier with itself. Hence, by gluing and conjunction application, $(B \wedge C) \vee (B \wedge C) \rhd^{\Phi} B \wedge C$. The case for disjunction is easy.

Now assume $A = \forall z \leq t(\vec{x})\ B(\vec{x}, z)$. If Φ is the class Π_k, by IH we have $B(\vec{x}, z) \vee B(\vec{x}, z) \rhd^{\Pi_k}_{\pi} B(\vec{x}, z)$ and if Φ is Σ_k, since $\forall z \leq t(\vec{x}) B(\vec{x}, z) \in \Phi$, then it actually lives in the lower class Π_{k-1}, which again by IH means $B(\vec{x}, z) \vee B(\vec{x}, z) \rhd^{\Pi_{k-1}}_{\pi} B(\vec{x}, z)$. Hence, in either case

$$B(\vec{x}, z) \vee B(\vec{x}, z) \rhd^{\Pi_k}_{\pi} B(\vec{x}, z)$$

By Lemma 4.5 we have

$$\forall z \leq t(\vec{x})\ [B(\vec{x}, z) \vee B(\vec{x}, z)] \rhd^{\Phi}_{\alpha} \forall z \leq t(\vec{x})\ B(\vec{x}, z).$$

for any $\alpha \in \{\sigma, \pi\}$. But $\forall z \leq t(\vec{x})\ [B(\vec{x}, z) \vee B(\vec{x}, z)]$ is α-reducible to

$$\forall u \leq t(\vec{x})\ B(\vec{x}, u) \vee \forall v \leq t(\vec{x}) B(\vec{x}, v)$$

using the variable z as the witness for both of u and v, hence the claim follows from gluing.

For the existential case, w.l.o.g we can assume $\Phi = \Sigma_k$ for some k. The reason is that if $\Phi = \Pi_k$, then since A begins with an existential quantifier, $A \in \Sigma_{k-1}$ and hence we can work with Σ_{k-1}. Therefore, we assume $\Phi = \Sigma_k$ for some k. First note that by the Lemma 4.7, there exists $A' = \exists \vec{z} \leq \vec{t}(\vec{x})\ B(\vec{x}, \vec{z})$ such that $A \equiv_\sigma A'$. But since both of the formulas A and A' begin with an existential quantifier, $A \equiv_\pi A'$. Therefore, it is enough to prove the claim for A'. Note that by this assumption we can assume that the σ-prenex form of B is quantifier-free or begins with universal quantifiers and hence $B \in \Pi_{k-1}$. Then by the Lemma 4.8, we have $B(\vec{u}) \wedge \neg B(\vec{u}) \rhd_\sigma^{\Sigma_k} \bot$ and $B(\vec{v}) \wedge \neg B(\vec{v}) \rhd_\sigma^{\Sigma_k} \bot$ and then by the propositional rules

$$(B(\vec{u}) \vee B(\vec{v})) \wedge \neg B(\vec{u}) \wedge \neg B(\vec{v}) \rhd_\sigma^{\Sigma_k} \bot \ \ (*)$$

Assume the length of this flow is s. Then, there is a $(\Sigma_k, \mathcal{B}, \sigma)$-flow from

$$(i \leq 1 \wedge j \leq 1) \wedge [B(\vec{u}) \vee B(\vec{v})] \wedge (\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)$$

to

$$(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)] \wedge (i = 1 \vee j = 1)$$

with the length s where $\chi_B(\vec{u}) = i$ means $(i = 1 \to B(\vec{u})) \wedge (i = 0 \to \neg B(\vec{u}))$. It is enough to use the formula $G(w, i, j, \vec{u}, \vec{v})$ to fill in between, where G is defined by the following scheme: If $i > 1$ or $j > 1$ then use $\bot$. If $i = j = 1$, then use $G(w, i, j, \vec{u}, \vec{v}) = B(\vec{u}) \wedge B(\vec{v})$. If $i = 1$ and $j = 0$ use $G(w, i, j, \vec{u}, \vec{v}) = B(u) \wedge \neg B(v)$. If $i = 0$ and $j = 1$ use $G(w, i, j, \vec{u}, \vec{v}) = \neg B(u) \wedge B(v)$. And finally if $i = j = 0$, use the flow from $(*)$. Moreover, in the first three cases, use identity reductions, ignoring the $B(\vec{u}) \vee B(\vec{v})$.

Using the Lemma 4.5, for any $\alpha \in \{\sigma, \pi\}$ we have a $(\Sigma_k, \mathcal{B}, \alpha)$-flow from

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, \exists i, j \leq 1\ [(i \leq 1 \wedge j \leq 1) \wedge [B(\vec{u}) \vee B(\vec{v})] \wedge (\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)]$$

to

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, \exists i, j \leq 1\ [(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)] \wedge (i = 1 \vee j = 1)]$$

Since the first element of the flow is α-equivalent to

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, [[B(\vec{u}) \vee B(\vec{v})] \wedge \exists i, j \leq 1\ [(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)]]$$

for any $\alpha \in \{\sigma, \pi\}$ we will have $(\Sigma_k, \mathcal{B}, \alpha)$-flow from

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, [[B(\vec{u}) \vee B(\vec{v})] \wedge \exists i, j \leq 1\, [(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)]]$$

to

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, \exists i, j \leq 1\, [(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)] \wedge (i = 1 \vee j = 1)]$$

On the other hand, by the Lemma 4.6 and the Lemma 4.5, for any $\alpha \in \{\sigma, \pi\}$ we know that there is a $(\Sigma_k, \mathcal{B}, \alpha)$-flow from

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, B(\vec{u}) \vee B(\vec{v})$$

to

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, [[B(\vec{u}) \vee B(\vec{v})] \wedge \exists i, j \leq 1\, [(\chi_B(\vec{u}) = i) \wedge (\chi_B(\vec{v}) = j)]]$$

Now, since

$$\exists \vec{u}, \vec{v} \leq \vec{t}\, B(\vec{u}) \vee B(\vec{v})$$

and

$$\exists \vec{u} \leq \vec{t}\, B(\vec{u}) \vee \exists \vec{v} \leq \vec{t}\, B(\vec{v})$$

are α-equivalent, it is enough to show that $\exists \vec{y} \leq \vec{t}(\vec{x})\ B(\vec{x}, \vec{y})$ is α-reducible to

$$\exists \vec{u}, \vec{v} \leq \vec{t}(\vec{x})\ \exists i, j \leq 1\ (i = 1 \vee j = 1) \wedge (\chi_B(u) = i) \wedge (\chi_B(v) = j)$$

It is enough to read i and j and decide between the cases that $i = 1$ or $(i = 0,\ j = 1)$. Then if $i = 1$, use $\vec{u}$ to witness $\vec{y}$ and reduce $B(\vec{y})$ to $B(\vec{u})$ in $\chi_B(\vec{u}) = i$ by identity reduction. If $(i = 0,\ j = 1)$ then use $\vec{v}$ to witness $\vec{y}$ and reduce $B(\vec{y})$ to $B(\vec{v})$ in $\chi_B(\vec{v}) = j$ by identity reduction.

The case (ii) is the dual of (i) and provable by just using (i) on $\neg A$ and then taking negations. □

Lemma 4.10. *(Conjunction and Disjunction Rules) Let $\alpha \in \{\sigma, \pi\}$, $\Phi \in \mathcal{C}(\mathcal{L})$ and $\Gamma \cup \Delta \cup \{A(\vec{x}), B(\vec{x})\} \subseteq \Phi$ be some formulas. Then:*

(i) If $\Gamma, A \rhd^{\Phi}_{\alpha} \Delta$ or $\Gamma, B \rhd^{\Phi}_{\alpha} \Delta$ then $\Gamma, A \wedge B \rhd^{\Phi}_{\alpha} \Delta$.

(ii) If $\Gamma_0 \rhd^{\Phi}_{\alpha} \Delta_0, A$ and $\Gamma_1 \rhd^{\Phi}_{\alpha} \Delta_1, B$ then $\Gamma_0, \Gamma_1 \rhd^{\Phi}_{\alpha} \Delta_0, \Delta_1, A \wedge B$.

(iii) If $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A$ or $\Gamma \rhd^{\Phi}_{\alpha} \Delta, B$ then $\Gamma \rhd^{\Phi}_{\alpha} \Delta, A \vee B$.

(iv) If $\Gamma, A \rhd^{\Phi}_{\alpha} \Delta$ and $\Gamma, B \rhd^{\Phi}_{\alpha} \Delta$ then $\Gamma, A \vee B \rhd^{\Phi}_{\alpha} \Delta$.

Proof. (*i*) and (*iii*) are trivial, because we have the reductions

$$\bigwedge \Gamma \wedge A \wedge B \geq_\alpha^\Phi \bigwedge \Gamma \wedge A \qquad \bigwedge \Gamma \wedge A \wedge B \geq_\alpha^\Phi \bigwedge \Gamma \wedge B$$

$$\bigvee \Delta \vee A \geq_\alpha^\Phi \bigvee \Delta \vee A \vee B \qquad \bigvee \Delta \vee B \geq_\alpha^\Phi \bigvee \Delta \vee A \vee B$$

by self-witnessing the common quantifiers on both sides and ignoring the irrelevant parts. Then glue the given flow to the the flow corresponding to the reduction, using the weak gluing, Lemma 4.4. For (*ii*), if $\Gamma_0 \rhd_\alpha^\Phi \Delta_0, A$, then by conjunction application with $\bigwedge \Gamma_1$ we have $\bigwedge \Gamma_0 \wedge \bigwedge \Gamma_1 \rhd_\alpha^\Phi (\bigvee \Delta_0 \vee A) \wedge \bigwedge \Gamma_1$. Moreover, we have $\bigwedge \Gamma_1 \rhd_\alpha^\Phi \bigvee \Delta_1 \vee B$ and again by conjunction application $\bigwedge \Gamma_1 \wedge (\bigvee \Delta_0 \vee A) \rhd_\alpha^\Phi (\bigvee \Delta_1 \vee B) \wedge (\bigvee \Delta_0 \vee A)$. Therefore by weak gluing

$$\bigwedge \Gamma_0 \wedge \bigwedge \Gamma_1 \rhd_\alpha^\Phi (\bigvee \Delta_1 \vee B) \wedge (\bigvee \Delta_0 \vee A).$$

But it is easy to see that

$$(\bigvee \Delta_1 \vee B) \wedge (\bigvee \Delta_0 \vee A) \geq_\alpha \bigvee \Delta_1 \vee \bigvee \Delta_0 \vee (A \wedge B).$$

Hence

$$\Gamma_0, \Gamma_1 \rhd_\alpha^\Phi \Delta_0, \Delta_1, (A \wedge B).$$

For (*iv*), if $\Gamma_0, A \rhd_\alpha^\Phi \Delta_0$ then by disjunction application with $\bigwedge \Gamma_1 \wedge B$ we have

$$(\bigwedge \Gamma_0 \wedge A) \vee (\bigwedge \Gamma_1 \wedge B) \rhd_\alpha^\Phi \bigvee \Delta_0 \vee (\bigwedge \Gamma_1 \wedge B).$$

Moreover, we have $\bigwedge \Gamma_1 \wedge B \rhd_\alpha^\Phi \bigvee \Delta_1$, hence again by disjunction application

$$(\bigwedge \Gamma_1 \wedge B) \vee \bigvee \Delta_0 \rhd_\alpha^\Phi \bigvee \Delta_0 \vee \bigvee \Delta_1.$$

Hence, by weak gluing,

$$(\bigwedge \Gamma_0 \wedge A) \vee (\bigwedge \Gamma_1 \wedge B) \rhd_\alpha^\Phi \bigvee \Delta_0 \vee \bigvee \Delta_1.$$

However, it is clear that

$$\bigwedge \Gamma_0 \wedge \bigwedge \Gamma_1 \wedge (A \vee B) \geq_\alpha (\bigwedge \Gamma_0 \wedge A) \vee (\bigwedge \Gamma_1 \wedge B).$$

Hence,

$$\Gamma_0, \Gamma_1, (A \vee B) \rhd_\alpha^\Phi \Delta_0, \Delta_1.$$

□

Lemma 4.11. *(Cut and Induction) Let* $\alpha \in \{\sigma, \pi\}$, $\Phi \in \mathcal{C}(\mathcal{L})$ *and* $\Gamma \cup \Delta \cup \{A\} \subseteq \Phi$ *be some formulas:*

(*i*) *If* $\Gamma \rhd^{\Phi}_{\alpha} A, \Delta$ *and* $\Gamma, A \rhd^{\Phi}_{\alpha} \Delta$, *then we have* $\Gamma \rhd^{\Phi}_{\alpha} \Delta$.

(*ii*) *If* $\Gamma, A(y, \vec{x}) \rhd^{\Phi}_{\alpha} \Delta, A(y+1, \vec{x})$ *then* $\Gamma, A(0, \vec{x}) \rhd^{\Phi}_{\alpha} \Delta, A(s(\vec{z}, \vec{x}), \vec{x})$.

Proof. For (*i*), Since $\Gamma_0 \rhd^{\Phi}_{\alpha} \Delta_0, A$ and $\Gamma_1, A \rhd^{\Phi}_{\alpha} \Delta_1$ then

$$\bigwedge \Gamma_0 \rhd^{\Phi}_{\alpha} \bigvee \Delta_0 \vee A$$

and $\bigwedge \Gamma_1 \wedge A \rhd^{\Phi}_{\alpha} \bigvee \Delta_1$. Apply conjunction with $\bigwedge \Gamma_1$ on the first one and disjunction with $\bigvee \Delta_0$ on the second one to prove $\bigwedge \Gamma_1 \wedge \bigwedge \Gamma_0 \rhd^{\Phi}_{\alpha} (\bigvee \Delta_0 \vee A) \wedge \bigwedge \Gamma_1$ and $(\bigwedge \Gamma_1 \wedge A) \vee \bigvee \Delta_0 \rhd \bigvee \Delta_1 \vee \bigvee \Delta_0$. Since $(\bigvee \Delta_0 \vee A) \wedge \bigwedge \Gamma_1 \geq_{\alpha} (\bigwedge \Gamma_1 \wedge A) \vee \bigvee \Delta_0$, by using gluing we will have $\bigwedge \Gamma_1 \wedge \bigwedge \Gamma_0 \rhd^{\Phi}_{\alpha} \bigvee \Delta_0 \vee \bigvee \Delta_1$.

For (*ii*) we reduce the induction case to the strong gluing case. Since

$$\Gamma, A(y, \vec{x}) \rhd^{\Phi}_{\alpha} \Delta, A(y+1, \vec{x})$$

by definition, $\bigwedge \Gamma \wedge A(y, \vec{x}) \rhd^{\Phi}_{\alpha} \bigvee \Delta \vee A(y+1, \vec{x})$. Therefore, by the Lemma **??** we have

$$(\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta \rhd^{\Phi}_{\alpha} \bigvee \Delta \vee A(y+1, \vec{x}) \vee \bigvee \Delta$$

and by contraction for $\bigvee \Delta$ we know

$$\bigvee \Delta \vee A(y+1, \vec{x}) \vee \bigvee \Delta \rhd^{\Phi}_{\alpha} \bigvee \Delta \vee A(y+1, \vec{x}).$$

Hence,

$$(\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta \rhd^{\Phi}_{\alpha} \bigvee \Delta \vee A(y+1, \vec{x}).$$

Then by conjunction introduction and the fact that $(\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta) \rhd^{\Phi}_{\alpha} \bigwedge \Gamma \vee \bigvee \Delta$,

$$((\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta), (\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta) \rhd^{\Phi}_{\alpha} (\bigvee \Delta \vee A(y+1, \vec{x})) \wedge (\bigwedge \Gamma \vee \bigvee \Delta)$$

By using the propositional, structural and the cut rule, it is easy to prove

$$(\phi \vee \psi) \wedge (\sigma \vee \psi) \rhd^{\Phi}_{\alpha} (\phi \wedge \sigma) \vee \psi.$$

Hence, by using the contraction we have

$$(\bigwedge \Gamma \wedge A(y, \vec{x})) \vee \bigvee \Delta \rhd^{\Phi}_{\alpha} (\bigwedge \Gamma \wedge A(y+1, \vec{x})) \vee \bigvee \Delta.$$

Now by strong gluing we have

$$(\bigwedge \Gamma \wedge A(0, \vec{x})) \vee \bigvee \Delta \rhd^{\Phi}_{\alpha} (\bigwedge \Gamma \wedge A(s(\vec{z}, \vec{x}), \vec{x})) \vee \bigvee \Delta.$$

But since $\Gamma \wedge A(0, \vec{x}) \rhd_{\alpha}^{\Phi} (\bigwedge \Gamma \wedge A(0, \vec{x})) \vee \bigvee \Delta$ and

$$(\bigwedge \Gamma \wedge A(s(\vec{x}), \vec{x})) \vee \bigvee \Delta \geq_{\alpha} \bigvee \Delta \vee A(s(\vec{z}, \vec{x}), \vec{x}),$$

we have

$$\Gamma(\vec{x}), A(0, \vec{x}) \rhd_{\alpha}^{\Phi} \Delta(\vec{x}), A(s(\vec{z}, \vec{x}), \vec{x}).$$

□

The following theorem is the main theorem of the theory of flows for bounded theories of arithmetic:

Theorem 4.12. *(Soundness) If* $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Phi$, $\mathfrak{B}(\Phi, \mathcal{A}) \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ *and* $\mathcal{A} \subseteq \mathcal{B}$ *has a characteristic function for any quantifier-free formula then* $\Gamma \rhd_{\phi}^{(\Phi, \mathcal{B})} \Delta$.

Proof. We assume Φ is a π-type class. The other case is similar. To prove the lemma we use induction on the length of the free-cut free proof of $\Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$.

1. (Axioms). If $\Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ is a logical axiom then the claim is trivial. If it is a non-logical axiom then the claim will be also trivial because all non-logical axioms are quantifier-free and provable in $\mathcal{B}$. Therefore there is nothing to prove.

2. (Structural Rules). It is proved in the Lemma 4.9.

3. (Cut). It is proved by Lemma 4.11.

4. (Propositional). The conjunction and disjunction cases are proved in the Lemma 4.10. The implication and negation cases are proved in the Lemma **??**.

5. (Bounded Universal Quantifier Rules, Right). If $\Gamma(\vec{x}) \Rightarrow \Delta(\vec{x}), \forall z \leq p(\vec{x}) B(\vec{x}, z)$ is proved by the $\forall^{\leq} R$ rule by $\Gamma(\vec{x}), b \leq p(\vec{x}) \Rightarrow \Delta(\vec{x}), B(\vec{x}, b)$, then by IH, $\Gamma(\vec{x}), b \leq p(\vec{x}) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x}), B(\vec{x}, b)$. By the Lemma 4.5, we have a $(\Pi_k, \mathcal{B}, \pi)$-flow from $\forall b \leq p(\vec{x})(b \leq p(\vec{x}) \wedge \bigwedge \Gamma)$ to $\forall b \leq p(\vec{x})[B(\vec{x}, b) \vee \bigvee \Delta]$. Since Γ does not have a free b, it is easy to see that $\bigwedge \Gamma \geq_{\pi} \forall b \leq p(\vec{x})(b \leq p(\vec{x}) \wedge \bigwedge \Gamma)$. Hence it is enough to add $\bigwedge \Gamma$ to the beginning of the flow. Do the same for the right side to reach $\forall b \leq p(\vec{x}) B(\vec{x}, b) \vee \bigvee \Delta$. Finally note that changing the name of a bounded variable does not change the nature of deterministic flows which complete the proof.

6. (Bounded Universal Quantifier Rules, Left). Suppose

$$\Gamma(\vec{x}), s(\vec{x}) \leq p(\vec{x}), \forall z \leq p(\vec{x}) B(\vec{x}, z) \Rightarrow \Delta(\vec{x})$$

is proved by the $\forall^{\leq} L$ rule by $\Gamma(\vec{x}), B(\vec{x}, s(\vec{x})) \Rightarrow \Delta(\vec{x})$. Then by IH,

$$\Gamma(\vec{x}), B(\vec{x}, s(\vec{x})) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x})$$

But by witnessing z by s and the rest by themselves, we have

$$\bigwedge \Gamma(\vec{x}) \wedge s(\vec{x}) \leq p(\vec{x}) \wedge \forall z \leq p(\vec{x}) B(\vec{x}, z) \geq_{\pi} \bigwedge \Gamma(\vec{x}) \wedge B(\vec{x}, s(\vec{x}))$$

hence by gluing

$$\Gamma(\vec{x}), s(\vec{x}) \leq p(\vec{x}), \forall z \leq p(\vec{x}) B(\vec{x}, z) \geq_{\pi} \Delta(\vec{x}).$$

7. (Bounded Existential Quantifier Rules, Right). If $\Gamma(\vec{x}), s(\vec{x}) \leq p(\vec{x}) \Rightarrow \Delta(\vec{x}), \exists z \leq p(\vec{x}) B(\vec{x}, z)$ is proved by the $\exists^{\leq} R$ rule by $\Gamma(\vec{x}) \Rightarrow \Delta(\vec{x}), B(\vec{x}, s(\vec{x}))$ then by IH

$$\Gamma(\vec{x}) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x}), B(\vec{x}, s(\vec{x})).$$

Since $\exists z \leq p(\vec{x}) B(\vec{x}, z) \in \Pi_k$, it is also in Σ_{k-1}. Therefore, by Lemma 4.8, $\Gamma(\vec{x}), \neg B(\vec{x}, s(\vec{x})) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x})$. By 6, $\Gamma(\vec{x}), s(\vec{x}) \leq p(\vec{x}), \forall z \leq p(\vec{x}) \neg B(\vec{x}, z) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x})$ and again by the Lemma 4.8 we will have

$$\Gamma(\vec{x}), s(\vec{x}) \leq p(\vec{x}) \rhd_{\pi}^{\Pi_k} \Delta(\vec{x}), \exists z \leq p(\vec{x}) B(\vec{x}, z).$$

8. (Bounded Existential Quantifier Rules, Left). If $\Gamma, \exists y \leq p(\vec{x}) B(\vec{x}, y) \Rightarrow \Delta$ is proved by the $\exists^{\leq} L$ rule by $\Gamma, b \leq p(\vec{x}), B(\vec{x}, b) \Rightarrow \Delta$, by IH we have $\Gamma, b \leq p(\vec{x}), B(\vec{x}, b) \rhd_{\pi}^{\Pi_k} \Delta$ then since $\exists b \leq p(\vec{x}) B(\vec{x}, b) \in \Pi_k$, it is also in Σ_{k-1}. Therefore, by the Lemma 4.8

$$\Gamma, b \leq p(\vec{x}) \rhd_{\pi}^{\Pi_k} \Delta, \neg B(\vec{x}, b)$$

by 5, we have

$$\Gamma \rhd_{\pi}^{\Pi_k} \Delta, \forall y \leq p(\vec{x}) \neg B(\vec{x}, y)$$

Finally again by Lemma 4.8 we have

$$\Gamma, \exists y \leq p(\vec{x}) B(\vec{x}, y) \rhd_{\pi}^{\Pi_k} \Delta.$$

9. (Induction). It is proved in Lemma 4.11. □

5 Applications

In this subsection we will use the soundness and completeness theorems that we have proved in the previous subsection to extract the computational content of the low complexity statements of some concrete weak bounded theories such as Buss' hierarchy of bounded theories of arithmetic and some strong theories such as $I\Delta_0(\exp)$, PRA and $\mathrm{PA} + \mathrm{TI}(\alpha)$.

For the first application, consider the theories $IU_k = \mathfrak{B}(\Pi_k(\mathcal{L}_{\mathcal{R}}), \mathcal{R})$ for $k \geq 1$. These theories are the fragments of the theory $I\Delta_0$ corresponding to the computational world of the linear time hierarchy. Moreover, consider the class of all functions constructed from zero, projections and closed under successor, addition, production, subtraction and division and call it R:

Corollary 5.1. *Let $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq U_k$. Then, $IU_k \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff $\Gamma \rhd_{\pi}^{(U_k, \mathcal{R})} \Delta$. The second condition means that there exists a sequence of length $t \in R$ of formulas in U_k beginning from $\bigwedge \Gamma$ ending with $\bigvee \Delta$ such that each formula is $(\pi, \mathcal{R})$-reducible to its successor using just the functions in R.*

Proof. The only thing that we have to check is the fact that $\mathcal{R}$ has the characteristic functions for any quantifier-free formula in the language $\mathcal{L}_{\mathcal{R}}$. It has been proved in the Remark 2.3. □

The second application, and maybe the more important one, is the case of Buss' hierarchy of bounded arithmetic, in which we assume the language has a symbol for any PV function and we denote the class of all strict Σ_k^b and Π_k^b formulas with $\hat{\Sigma}_k^b$ and $\hat{\Pi}_k^b$.

Corollary 5.2. *Let $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \hat{\Pi}_k^b(\#_n)$. Then, $T_n^k \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff $\Gamma \rhd_{\pi}^{(\hat{\Pi}_k^b(\#_n), \mathrm{BASIC}_n(\mathrm{PV}))} \Delta$, where* $\mathrm{BASIC}_n(\mathrm{PV}))$ *is the theory* BASIC_n *plus all the defining axioms of* PV. *Specifically, for $n = 2$, $T_2^k \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff $\Gamma \rhd_{\pi}^{(\hat{\Pi}_k^b, \mathrm{PV})} \Delta$. The second condition in the latter case means the existence of a uniform sequence of length $2^{p(|\vec{x}|)}$ of formulas in Π_k^b starting with $\bigwedge \Gamma$ and ending in $\bigvee \Delta$ such that each formula is (π, PV)-reducible to its successor, using just the polynomial time computable functions.*

Proof. Observe that in the presence of all PV functions, any formula in $\hat{\Pi}_k^b(\#_n)$ is equivalent to a formula in Π_k. Therefore, since T_n^k is axiomatizable by $\hat{\Pi}_k^b(\#_n)$-induction, it is also axiomatizable by Π_k-induction. □

And also we can apply the soundness theorem on stronger theories with full exponentiation like $I\Delta_0(\exp)$ and PRA. Consider the theory $\mathcal{R}$ augmented with a function symbol for exponentiation with the usual recursive definition and denote it by $\mathcal{R}(\exp)$ and also denote the union of $\mathcal{R}$ and the induction-free part of PRA by PRA^-. Then:

Corollary 5.3. *Let* $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Pi_k$. *Then:*

(i) $I\Delta_0(\exp) \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ *iff* $\Gamma \rhd_{\pi}^{(\Pi_k, \mathcal{R}(\exp))} \Delta$.

(ii) $\mathrm{PRA} \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ *iff* $\Gamma \rhd_{\pi}^{(\Pi_k, \mathrm{PRA}^-)} \Delta$.

Proof. The only point to mention is that both of the theories $I\Delta_0(\exp)$ and PRA are axiomatizable by $I\Pi_k$ for any k. Hence we can apply the theory of deterministic flows here. □

We can also use the theory of flows to extract the computational content of low complexity sentences of the very strong theories of arithmetic such as PA and PA + TI(α). But this is not what we can implement in a very direct way. The reason is that our method is tailored for bounded theories while these theories are unbounded. Hence, to use our theory, we have to find a way to transfer low complexity statements from these theories to some corresponding bounded theories. This is what the continuous cut elimination method makes possible in its very elegant enterprise. It transfers all Π_2^0 consequences of a strong theory T to some quantifier-free extensions of PRA and then makes it possible to apply the flow decomposition technique. To explain how it works, we need some definitions:

Definition 5.4. Let T be a theory of arithmetic. We say that α is a Π_2^0-proof theoretical ordinal of T when $\prec$ is the primitive recursive representation of the order on α and $T \equiv_{\Pi_2^0} \mathrm{PRA} + \bigcup_{\beta \prec \alpha} \mathrm{TI}(\prec_\beta)$ where $\mathrm{TI}(\prec_\beta)$ means full transfinite induction up to the ordinal β.

Convention. From now on wherever we have a proof theoretic ordinal, we always assume that it is closed under the operation $\beta \mapsto \omega^\beta$.

Definition 5.5. Let $\prec$ be a quantifier-free formula in the language of PRA. By theory $\mathrm{PRA} + \bigcup_{\beta \prec \alpha} \mathrm{PRWO}(\prec_\beta)$ we mean PRA plus the axiom schema

$$\mathrm{PRWO}(\prec_\beta) : \forall \vec{x} \exists y \; f(\vec{x}, y+1) \not\prec_\beta f(\vec{x}, y)$$

for any function symbol f.

The following theorem uses continuous cut elimination technique to reduce transfinite induction to PRWO.

Theorem 5.6. *[4] Let T be a theory of arithmetic and α its Π_2^0-proof theoretical ordinal. Then*

$$T \equiv_{\Pi_2^0} \mathrm{PRA} + \bigcup_{\beta \prec \alpha} \mathrm{PRWO}(\prec_\beta)$$

The following theory is the skolemization of $\mathrm{PRA} + \bigcup_{\beta \prec \alpha} \mathrm{PRWO}(\prec_\beta)$:

Definition 5.7. The language of the theory $\mathrm{PRA}_\prec$ consists of the language of PRA plus the scheme which says that for any PRA-function symbol $f(\vec{x}, y)$ and any $\beta \prec \alpha$, there exists a function symbol $[\mu_\beta y.f](\vec{x})$. Then $\mathrm{BASIC}_\prec$ is the theory axiomatized by the axioms of PRA plus the theory $\mathcal{R}$ and the following definitional equations: $f(\vec{x}, 1 + [\mu_\beta y.f](\vec{x})) \not\prec_\beta f(\vec{x}, [\mu_\beta y.f](\vec{x}))$ and $z < [\mu_\beta y.f](\vec{x}) \to f(\vec{x}, z+1) \prec_\beta f(\vec{x}, z)$. Finally, $\mathrm{PRA}_\prec$ is $\mathrm{BASIC}_\prec$ plus the usual quantifier-free induction.

Combining all of these steps together we can reduce a theory T to a bounded arithmetical theory $\mathrm{PRA}_\prec$.

Corollary 5.8. *Let T be a theory of arithmetic and α its Π_2^0-proof theoretical ordinal. Then $T \equiv_{\Pi_2^0} \mathrm{PRA}_\prec$.*

Now we are ready to have the following corollary:

Corollary 5.9. *Let $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Pi_k$, and α_T is the Π_2^0-ordinal of T with the primitive recursive representation $\prec_{\alpha_T}$, then $T \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff*

$$\Gamma(\vec{x}) \rhd_\pi^{(\Pi_k, \mathrm{BASIC}_{\prec_{\alpha_T}})} \Delta(\vec{x}).$$

Proof. Note that the existence of the flow is equivalent to the provability of $\Gamma \Rightarrow \Delta$ in $\mathrm{PRA}_{\prec_{\alpha_T}}$ because $\mathrm{PRA}_{\prec_{\alpha_T}}$ is a bounded theory axiomatizable by the usual induction on formulas in Π_k. On the other hand, we have $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Pi_k$. Hence the sequent is bounded and is in Π_2^0. Therefore, by the definition of Π_2^0-ordinals, we know that $\mathrm{PRA}_{\prec_{\alpha_T}} \vdash \Gamma \Rightarrow \Delta$ iff $T \vdash \Gamma \Rightarrow \Delta$, which completes the proof. □

Corollary 5.10. *Let $\Gamma(\vec{x}) \cup \Delta(\vec{x}) \subseteq \Pi_k$, and $\epsilon(\alpha)$ be the least ϵ number after α with a primitive recursive representation. Then $\mathrm{PA}+\mathrm{TI}(\alpha) \vdash \Gamma(\vec{x}) \Rightarrow \Delta(\vec{x})$ iff*

$$\Gamma(\vec{x}) \rhd_\pi^{(\Pi_k, \mathrm{BASIC}_{\prec_{\epsilon(\alpha)}})} \Delta(\vec{x}).$$

So far, we have used the theory of deterministic flows to decompose first order proofs of bounded theories. In the following we will introduce two different kinds of characterizations and we will use them to reprove some recent results for some specific classes of formulas. The types that we want to use are generalizations of some recent characterizations of some low complexity statements in Buss' hierarchy of bounded arithmetic by Game induction principles [5], [6] and some kind of PLS problems [?].

First let us generalize our game interpretation of the Remark ?? to interpret any formula of the form

$$A = \forall \vec{y_1} \leq \vec{p_1}(\vec{x}) \exists \vec{z_1} \leq \vec{q_1}(\vec{x}) \forall \vec{y_2} \leq \vec{p_2}(\vec{x}) \ldots G_A(\vec{x}, \vec{y_1}, \vec{z_1}, \vec{y_2}, \vec{z_2}, \ldots)$$

as a k-turn game $\mathcal{G}_A$ in which the players can have some but fixed predefined number of simultaneous moves. More precisely, in the game $\mathcal{G}_A$, the first player begins by choosing the moves $\vec{y_1} \leq \vec{p_1}(\vec{x})$ altogether, then the second player chooses the moves $\vec{z_1} \leq \vec{q_1}(\vec{x})$ and they continue alternately. Again if $G_A(\vec{x}, \vec{y_1}, \vec{z_1}, \vec{y_2}, \vec{z_2}, \ldots)$ becomes true the second player wins and otherwise the first player is the winner. Note that in this multi-move version, we still have the equivalence between the truth of A and the existence of the winning strategy for the second player. What we want to add to this fact is its explicit version which states that any deterministic reduction from A to $\top$ is nothing but an explicit winning strategy for the second player in the game $\mathcal{G}_A$.

Definition 5.11. Let $\mathcal{L} \supseteq \mathcal{L}_{\mathcal{R}}$ be a language. An instance of the (j, k)-game induction principle, $GI^j_k(\mathcal{L})$, is given by size parameters a and b, a quantifier-free formula $G(u, \vec{v})$ with a fixed partition of the variables $\vec{v}$ into k groups, a sequence of terms V and a uniform sequence W_u of sequences of terms. The instance $GI(G, V, W, a, b)$ states that, interpreting $G(u, \vec{v})$ as a k-turn game on moves $\vec{v}$ in which all moves are bounded by b, the following cannot all be true:

- (i) Deciding the winner of the game $G(0, \vec{v})$ depends only on the first j moves,
- (ii) The second player has a winning strategy for $G(0, \vec{v})$ (expressed as a Π_j formula.)
- (iii) For $u \leq a \dot{-} 2$, W_u gives a deterministic reduction from $G(u+1, \vec{v})$ to $G(u, \vec{v})$,
- (iv) V is an explicit winning strategy for the first player in $G(a \dot{-} 1, \vec{v})$.

Notation. Let $\mathcal{C}$ and $\mathcal{D}$ be two classes of formulas and $\mathcal{B}$ be a theory. By $\mathcal{C} \equiv_{\mathcal{B}} \mathcal{D}$ we mean that for any $A \in \mathcal{C}$ there exists $B \in \mathcal{D}$ such that $B \geq_\pi A$ and for any $A \in \mathcal{D}$ there exists $B \in \mathcal{C}$ such that $B \geq_\pi A$.

Theorem 5.12. *Let $j \leq k$. Then,*

$$\forall \Sigma_j[\mathfrak{B}(\Pi_k, \mathcal{B})] \equiv_{\mathcal{B}} GI_k^j(\mathcal{L}).$$

Proof. It is clear that $\mathfrak{B}(\Pi_k, \mathcal{B}) \vdash GI_k^j(\mathcal{L})$ and $GI_k^j(\mathcal{L})$ is expressible by a $\forall \Sigma_j$ sentence. For the converse, assume $\mathfrak{B}(\Pi_k, \mathcal{B}) \vdash \forall \vec{x} A(\vec{x})$ where $A \in \Sigma_j$ and $j \leq k$. Then, we know that $\mathfrak{B}(\Pi_k, \mathcal{B}) \vdash \neg A(\vec{x}) \Rightarrow \bot$ and $\neg A \in \Pi_j$. By Corollary 5.2, there exist a term $t(\vec{x})$, a formula $H(u, \vec{x}) \in \Pi_k$ and sequences of terms E_0, E_1, I_0, I_1 and $F(u)$ such that the following statements are provable in $\mathcal{B}$:

(*i*) $H(0, \vec{x}) \equiv_{\pi}^{(E_0, E_1)} \neg A(\vec{x})$.

(*ii*) $H(t(\vec{x}), \vec{x}) \equiv_{\pi}^{(I_0, I_1)} \bot$.

(*iii*) $\forall u < t(\vec{x})[H(u, \vec{x}) \geq_{\pi}^{F_u} H(u+1, \vec{x})]$.

First of all, note that we can change the definition of H in the following way:

$$H'(u, \vec{x}) = (u = 0 \rightarrow \neg A(\vec{x})) \wedge (u \neq 0 \rightarrow H(u \dot{-} 1, \vec{x})).$$

And, it is possible to shift also the reductions to have (*i*) to (*iii*) for H'. Call these reductions E'_0, E'_1, F'_u, I'_0 and I'_1. Note that the truth of $H'(0, x)$ depends only on first j blocks of quantifiers when we write it in the Π_k form.

W.l.o.g., we assume that all bounds in $H'(u, \vec{x})$ are the same and depend only on $\vec{x}$. Call this bound $s(\vec{x})$. This is possible because any term is majorizable by a monotone term. Again w.l.o.g we can assume that H' is in the following prenex form:

$$H'(u, \vec{x}) = \forall \vec{z}_1 \leq s \exists \vec{y}_1 \leq s \forall \vec{z}_2 \leq s \ldots G(u, \vec{x}, \vec{z}_1, \vec{y}_1, \vec{z}_2, \ldots)$$

where G is quantifier-free and the number of quantifier groups are k. Define $a = t(\vec{x})$, $b = s(\vec{x})$, $W_u = F'_u$ and $V = I'_0$ and pick G for the game predicate with its natural partition of variables. Therefore, we have an instance of the game induction. Now we want to show that $A(\vec{x})$ is reducible to this game induction provably in $\mathcal{B}$. Since $\mathcal{B} \vdash \forall u < t(\vec{x})[H'(u, \vec{x}) \geq_{\pi}^{F'_u} H'(u+1, \vec{x})]$ and $\mathcal{B} \vdash H(t(\vec{x}), \vec{x}) \equiv^{(I_0, I_1)} \bot$, the false part of $GI_k^j(\mathcal{L})$ is the part which states "The second player has a winning strategy for $G(0, \vec{v})$." which means that $H'(0, \vec{x})$ is false. Since $H'(0, \vec{x})$ is equivalent with $\neg A(\vec{x})$ provably in $\mathcal{B}$, the reduction of the sentence $A(\vec{x})$ to the game induction principle is proved. $\square$

Using this generalization it is trivial to reprove the case for Buss' hierarchy of bounded arithmetic:

Corollary 5.13. *([5], [6]) For all $j \leq k$, $\forall\Sigma_j(T_2^k) \equiv_{\mathrm{PV}} GI_k^j(\mathcal{L}_{\mathrm{PV}})$.*

Now, let us explain the second type of problems, i.e., the generalized local search problems:

Definition 5.14. A formalized $(\Psi, \Lambda, \mathcal{B}, \prec)$-GLS problem consists of the following data:

(i) A sequence of terms $\vec{N}(\vec{x}, \vec{s}) \in \mathcal{L}_{\mathcal{B}}$ as the local improvements.

(ii) A term $c(\vec{x}, \vec{s}) \in \mathcal{L}_{\mathcal{B}}$ as a cost function.

(iii) A predicate $F(\vec{x}, \vec{s}) \in \Psi$ which intuitively means that $\vec{s}$ is a feasible solution for the input $\vec{x}$.

(iv) An initial sequence of terms $\vec{i}(\vec{x}) \in \mathcal{L}_{\mathcal{B}}$.

(v) A goal predicate $G(\vec{x}, \vec{s'}) \in \Lambda$.

(vi) A quantifier-free predicate $\prec \in \mathcal{L}_{\mathcal{B}}$ as a well-ordering.

(vii) A sequence of bounding terms $\vec{t}(\vec{x})$.

$(viii)$ A projection function I.

such that $\mathcal{B}$ proves that $\prec$ is a total order and

$$\mathcal{B} \vdash \forall \vec{x}\, F(\vec{x}, \vec{i}(\vec{x}))$$

$$\mathcal{B} \vdash \forall \vec{x}\vec{s}\, (F(\vec{x}, \vec{s}) \rightarrow F(\vec{x}, \vec{N}(\vec{x}, \vec{s})))$$

$$\mathcal{B} \vdash \forall \vec{x}\vec{s}\, (\vec{N}(\vec{x}, \vec{s}) = \vec{s} \vee c(\vec{x}, \vec{N}(\vec{x}, \vec{s})) \prec c(\vec{x}, \vec{s}))$$

$$\mathcal{B} \vdash \forall \vec{x}\vec{s}\, ((\vec{N}(\vec{x}, \vec{s}) = \vec{s} \wedge F(\vec{x}, \vec{s})) \rightarrow G(\vec{x}, I(\vec{s})))$$

$$\mathcal{B} \vdash \forall \vec{x}\vec{s'}\, (G(\vec{x}, \vec{s'}) \rightarrow \vec{s'} \leq \vec{t}(\vec{x}))$$

By the computational problem associated to a GLS problem, we mean finding $\vec{s'} \leq \vec{t}(\vec{x})$ such that $G(\vec{x}, \vec{s'})$.

If there is also a sequence of terms $\vec{b}(\vec{x})$ such that

$$\mathcal{B} \vdash \forall \vec{x}\vec{s}\, (F(\vec{x}, \vec{s}) \rightarrow \vec{s} \leq \vec{b}(\vec{x}))$$

The GLS-problem is called bounded and their class is denoted by $\mathrm{BGLS}(\Psi, \Lambda, \mathcal{B}, \prec)$. Moreover, if $\mathcal{L}_{\mathrm{PV}} \subseteq \mathcal{L}_{\mathcal{B}}$ and $\vec{t}(\vec{x}) = 2^{\vec{p}(|x|)}$ for some polynomials $\vec{p}$ we denote the class by $\mathrm{PLS}(\Psi, \Lambda, \prec, \mathcal{B})$ and if we have also the conditions that F is quantifier-free in the language of $\mathcal{B}$ and G is quantifier-free in the language of PV, we denote the class by $\mathrm{PLS}(\prec, \mathcal{B})$. Finally if we also add $\mathcal{B} = \mathrm{PV}$, then we write $\mathrm{PLS}(\prec)$ for the class of these GLS-problems.

Theorem 5.15. *(i) For any* BGLS$(\Pi_k, \Lambda, \mathcal{B}, \leq)$*-problem we have:*

$$\mathfrak{B}(\Pi_{k+1}, \mathcal{B}) \vdash \forall \vec{x} \exists \vec{s'} G(\vec{x}, \vec{s'})$$

(ii) Let $\Lambda \subseteq \Psi$ be a class of formulas, $A \in \Lambda$ a formula and $\vec{t}(\vec{x})$ are terms such that $\vec{z} \leq \vec{t}(\vec{x}) \in \Lambda$ for all variables $\vec{z}$. Then if

$$\mathfrak{B}(\Pi_{k+1}, \mathcal{B}) \vdash \forall \vec{x} \exists \vec{y} \leq \vec{t}(\vec{x}) A(\vec{x}, \vec{y})$$

then there exists a BGLS$(\Pi_k, \Lambda, \mathcal{B}, \leq)$*-problem with the condition that*

$$G(\vec{x}, \vec{y}) = A(\vec{x}, \vec{y}) \wedge \vec{y} \leq \vec{t}(\vec{x})$$

Proof. For (i), argue inside $\mathcal{B}$ and assume that there is no $\vec{s'}$ such that $G(\vec{x}, \vec{s'})$. It implies that $\forall \vec{s}(F(\vec{x}, \vec{s}) \to \vec{N}(\vec{x}, \vec{s}) \neq \vec{s})$. Use induction on the formula

$$\forall \vec{s} \leq \vec{r}(\vec{x})[F(\vec{x}, \vec{s}) \to c(\vec{x}, \vec{s}) \geq n]$$

where $\vec{r}(\vec{x})$ is the bound for F. This bound exists because the GLS problem is bounded. First note that the formula is in Π_{k+1}. Hence in $\mathfrak{B}(\Pi_{k+1}, \mathcal{B})$ we can afford such an induction. For $n = 0$ the claim is clear. For $n+1$, assume $F(\vec{x}, \vec{s})$, therefore by the assumption $\vec{N}(\vec{x}, \vec{s}) \neq \vec{s}$ which implies

$$c(\vec{x}, \vec{N}(\vec{x}, \vec{s})) < c(\vec{x}, \vec{s})$$

On the other hand, by $F(\vec{x}, \vec{s})$ we know that $F(\vec{x}, \vec{N}(\vec{x}, \vec{s}))$ and hence $\vec{N}(\vec{x}, \vec{s}) \leq r(\vec{x})$. By IH, we have $c(\vec{x}, \vec{N}(\vec{x}, \vec{s})) \geq n$ which implies $c(\vec{x}, \vec{s}) \geq n+1$. Therefore, we have

$$\forall n \forall \vec{s} \leq \vec{r}(\vec{x})[F(\vec{x}, \vec{s}) \to c(\vec{x}, \vec{s}) \geq n]$$

Define $c_0 = c(\vec{x}, \vec{i}(\vec{x}))$. For $n = c_0 + 1$ and $\vec{s} = \vec{i}(\vec{x})$ we will have $c_0 \geq c_0 + 1$ which is a contradiction. Hence there exists $\vec{s}$ such that $Goal(\vec{x}, \vec{s})$ which also implies that $\vec{s} \leq \vec{t}(\vec{x})$.

For (ii), assume

$$\mathfrak{B}(\Pi_{k+1}, \mathcal{B}) \vdash \forall \vec{x} \exists \vec{y} \leq \vec{t}(\vec{x}) A(\vec{x}, \vec{y}).$$

Then, we know that $\forall \vec{y} \leq \vec{t}(\vec{x}) \neg A(\vec{x}, \vec{y}) \Rightarrow \bot$ is provable in the theory. Since $A \in \Lambda \subseteq \Pi_k$, we have $\forall \vec{y} \leq \vec{t}(\vec{x}) \neg A(\vec{x}, \vec{y}) \in \Pi_{k+1}$. By soundness theorem 4.12, there exist a term $s(\vec{x})$, a formula $H(u, \vec{x}) \in \Pi_{k+1}$ and sequences of terms E_0, E_1, G_0, G_1 and $F(u)$ such that the following statements are provable in $\mathcal{B}$:

(*i*) $H(0,\vec{x}) \equiv_{\pi}^{(E_0,E_1)} \forall \vec{y} \leq \vec{t}(\vec{x}) \neg A(\vec{x},\vec{y})$.

(*ii*) $H(s(\vec{x}),\vec{x}) \equiv_{\pi}^{(G_0,G_1)} \bot$.

(*iii*) $\forall u < s(\vec{x})\ H(u,\vec{x}) \geq_{\pi}^{F_u} H(u+1,\vec{x})$.

Since $H \in \Pi_{k+1}$, w.l.o.g we can assume $H(u,\vec{x}) = \forall \vec{v} \leq \vec{r}(\vec{x},u) G(u,\vec{v},\vec{x})$ where $G(u,\vec{v},\vec{x}) \in \Sigma_k$ and $\vec{r}$ are monotone. Use the deterministic reductions to show the existence of terms U, V and Z such that

(*i*) $\mathcal{B} \vdash [\vec{Z}(\vec{x},\vec{v}) \leq \vec{t}(\vec{x}) \to \neg A(\vec{Z}(\vec{x},\vec{v}),\vec{x})] \to [\vec{v} \leq \vec{r}(\vec{x},0) \to G(0,\vec{v},\vec{x})]$.

(*ii*) $\mathcal{B} \vdash [\vec{U}(\vec{x}) \leq \vec{r}(\vec{x},s(\vec{x})) \to G(s(\vec{x}),\vec{U}(\vec{x}),\vec{x})] \to \bot$.

(*iii*) $\mathcal{B} \vdash \forall u < s(\vec{x})[\vec{V}(u,\vec{v},\vec{x}) \leq \vec{r}(\vec{x},u) \to G(u,\vec{V}(u,\vec{v},\vec{x}),\vec{x})] \to [\vec{v} \leq \vec{r}(\vec{x},u+1) \to G(u+1,\vec{v},\vec{x})]$.

Now define $B(u,\vec{v},\vec{z}) = [u \leq s(\vec{x}) \wedge \vec{v} \leq \vec{r}(\vec{x},s(\vec{x})) \wedge \vec{z} \leq \vec{t}(\vec{x})]$

$$F(\vec{x};u,\vec{v},\vec{z}) = \begin{cases} \vec{v} \leq \vec{r}(\vec{x},u \dot{-} 1) \wedge \neg G(u \dot{-} 1,\vec{v},\vec{x}) \wedge B(u,\vec{v},\vec{z}) & u > 0 \\ \vec{z} \leq \vec{t}(\vec{x}) \wedge A(\vec{x},\vec{z}) \wedge B(u,\vec{v},\vec{z}) & u = 0 \end{cases}$$

and

$$\vec{N}(\vec{x};u,\vec{v},\vec{z}) = \begin{cases} (u \dot{-} 1,\vec{V}(u,\vec{v},\vec{x}),\vec{z}) & u > 1 \\ (0,\vec{v},\vec{Z}(\vec{x},\vec{v})) & u = 1 \\ (u,\vec{v},\vec{z}) & u = 0 \end{cases}$$

and $Goal(\vec{x};\vec{z}) = [\vec{z} \leq \vec{t}(\vec{x}) \wedge A(\vec{x},\vec{z})]$, $\vec{i}(\vec{x}) = (s(\vec{x})+1,\vec{U}(\vec{x}),0)$, and $c(\vec{x};u,\vec{v},\vec{z}) = u$. It is clear to see that this data is a $\mathrm{BGLS}(\Pi_k,\Lambda,\mathcal{B},\leq)$-problem. The reason is that $F \in \Pi_k$ and $Goal \in \Lambda$ by the assumption. The answer to this problem is $\vec{z}$ such that $\vec{z} \leq \vec{t}$ and $A(\vec{x},\vec{z})$ which completes the proof. □

Corollary 5.16.

$$\forall\Sigma_{j+1}[\mathfrak{B}(\Pi_{k+1},\mathcal{B})] \equiv_{\mathcal{B}} \mathrm{BGLS}(\Pi_k,\Pi_j,\mathcal{B},\leq).$$

for all $j \leq k$.

And again the special case for Buss' hierarchy will be:

Corollary 5.17. *([?]) For all* $j \leq k$, $\forall\Sigma_{j+1}(T_2^{k+1}) \equiv_{\mathrm{PV}} \mathrm{PLS}(\Pi_k,\Pi_j,\mathrm{PV},\leq)$.

Remark 5.18. Local search problems and the game induction principles provide weaker characterizations than what the theory of flows has to offer. The game induction principle relaxes the $\mathcal{B}$-provability condition of the reductions to make the statement purely combinatorial at the expense of missing some useful information about the provability. The GLS problems, though, keep the base theories present, but instead they reduce their reductions to unwind only the outmost block of bounded universal quantifiers, sweeping the rest under the carpet of the feasibility predicate. This is helpful to simplify the formalization, but it clearly misses the huge reduction information that lies in the witnessing of the other quantifiers.

Using this characterization by the GLS problems, we can also capture the class of all low complexity search problems in strong theories. For the remaining part of this subsection, assume that the languages $\mathcal{L}_{I\Delta_0(\exp)}$ and $\mathcal{L}_{\mathrm{PRA}(\prec)}$ has a separate copy of the language of PV and define $\tilde{\Sigma}^b_j$ and $\tilde{\Pi}^b_j$ as Σ_j and Π_j in the language of PV. For instance, a formula in $\tilde{\Sigma}^b_1$ is essentially in the form $\exists \vec{y} \leq \vec{t}(\vec{x}) A(\vec{x}, \vec{y})$ where $\vec{t}$ are polynomial-time computable functions and A is a polynomial-time computable predicate. Hence, $\tilde{\Sigma}^b_1$ represents the NP predicates in our greater languages. Moreover, assume that our theories have access to all definitional axioms of PV for their separate language. To emphasize on this modification, we will denote the new version of any theory by the superscript p.

Corollary 5.19. *(i)* $\forall \tilde{\Sigma}^b_{j+1}[I\Delta^p_0(\exp)] \equiv_{\mathcal{R}^p(\exp)} \mathrm{BGLS}(\Pi_j, \tilde{\Pi}^b_j, \mathcal{R}^p(\exp), \leq)$.

(ii) $\forall \tilde{\Sigma}^b_{j+1}(\mathrm{PRA}^p) \equiv \mathrm{PLS}((\mathrm{PRA}^-)^p, \leq) \equiv_{(\mathrm{PRA}^-)^p,} \mathrm{BGLS}(\Pi_j, \tilde{\Pi}^b_j, (\mathrm{PRA}^-)^p, \leq)$.

Since we have $\forall \tilde{\Sigma}^b_{j+1}(\mathrm{PRA}^p_{\prec}) \equiv_{\mathrm{BASIC}^p_{\prec}} \mathrm{BGLS}(\Pi_j, \tilde{\Pi}^b_j, \mathrm{BASIC}^p_{\prec}, \leq)$, by the fact that $T \equiv_{\Pi^0_2} \mathrm{PRA}_{\prec_{\alpha_T}}$ we will have:

Theorem 5.20. *Let T be a theory of arithmetic with Π^0_2-ordinal α_T with a primitive recursive representation $\prec_{\alpha_T}$, then*

$$\forall \tilde{\Sigma}^b_{j+1}(T^p) \equiv_{\mathrm{BASIC}^p_{\prec_{\alpha_T}}} \mathrm{BGLS}(\Pi_j, \tilde{\Pi}^b_j, \mathrm{BASIC}^p_{\prec_{\alpha_T}}, \leq)$$

Corollary 5.21. *Let $\epsilon(\alpha)$ be the least ϵ number after α with a primitive recursive representation. Then*

$$\forall \tilde{\Sigma}^b_{j+1}([\mathrm{PA} + \mathrm{TI}(\alpha)]^p) \equiv_{\mathrm{BASIC}^p_{\prec_{\epsilon(\alpha)}}} \mathrm{BGLS}(\Pi_j, \tilde{\Pi}^b_j, \mathrm{BASIC}^p_{\prec_{\epsilon(\alpha)}}, \leq)$$

Remark 5.22. These characterizations of the low complexity consequences of the strong theories of arithmetic may seem a bit counter-intuitive. The reason is the paradoxical situation in which we have full access to a class of extremely complex functions while the search problems that we try to solve are much easier. A typical example of such a mismatch is our characterization of the total $\tilde{\Sigma}_1^b = \mathrm{NP}$ search problems of the theory $I\Delta_0(\exp)$. What the Lemma 5.19 presents is an algorithm based on a sequence of elementary computable reductions, while our NP search problem is just a very low complexity problem solvable by a brute force search in exponential time. Based on this mismatch, it may seem natural to conclude the sufficiency of one obvious reduction which implies the triviality of our characterizations. This is not a sound argument. It is correct that we have full access to a certain class of complex functions but it does not mean that we have full access to their complete theory about their behavior. What we know is usually a very basic theory consisting of the defining axioms of the function symbols. These complex functions behave as oracles to which we can impose our questions, but we can't fully understand their behavior, and hence we can't be sure about the correctness of their computations. Here is where the long sequences of reductions come to rescue. They consist of very simple computational steps based on the definitional axioms of the functions so that in each reduction we can ensure that our computation works correctly. In fact, reductions decompose a computation to simple verifiable steps which actually simulates the application of the induction axiom in the proof of the totality of the search problem.

Acknowledgment. We wish to thank Pavel Pudlak for his support, his suggestions and the invaluable discussions that we had. We are also genuinely grateful to Sam Buss, Neil Thapen, Emil Jeřabek and Raheleh Jalali for their constructive suggestions and the helpful discussions on the crucial and primitive stages of developing the theory.

References

[1] A. Beckmann, S. R. Buss, Polynomial Local Search in the Polynomial Hierarchy and Witnessing in Fragments of Bounded Arithmetic, Journal of Mathematical Logic 9, 1 (2009) 103-138.

[2] Buss, S. R. (1998). An introduction to proof theory. Handbook of proof theory, 137, 1-78.

[3] Buss, S. R. (1986). Bounded arithmetic (Vol. 86). Napoli: Bibliopolis.

[4] H. Friedman and S. Sheard, Elementary descent recursion and proof theory, Annals of Pure and Applied Logic 71 (1995) 145.

[5] A. Skelley, N. Thapen, The provably total search problems of bounded arithmetic, Proceedings of the London Mathematical Society, Vol 103:1, pages 106-138, 2011.

[6] N. Thapen, Higher complexity search problems for bounded arithmetic and a formalized no-gap theorem, Archive for Mathematical Logic, Vol 50:7-8, pages 665-680, 2011.