Logic and Proof Hilary 2024

Decidable Theories

James Worrell

1 Logical Theories

In this lecture we work exclusively with first-order logic with equality.

Fix a signature σ. A *theory* $\boldsymbol{T}$ is a set of σ-sentences that is closed under semantic entailment, i.e., if $\boldsymbol{T} \models F$ then $F \in \boldsymbol{T}$. Given a σ-structure $\mathcal{A}$ it is clear that the set of sentences that hold in $\mathcal{A}$ is a theory. We denote this theory by $\mathrm{Th}(\mathcal{A})$ and call it the *theory of* $\mathcal{A}$. For example, below we will consider the theory of the ordered set $(\mathbb{Q}, <)$.

Another important source of theories is from sets of axioms. Given a set of sentences $\boldsymbol{S}$, the set $\boldsymbol{T} = \{F : \boldsymbol{S} \models F\}$ is a theory. We call $\boldsymbol{S}$ a set of *axioms* for the theory $\boldsymbol{T}$. For example, if $\boldsymbol{S}$ comprises the group axioms (over a suitable signature σ) then $\boldsymbol{T}$ is the theory of groups, i.e., the set of all σ-sentences that are true in every group.

We say that a theory $\boldsymbol{T}$ is *complete* if for any sentence F, either $F \in \boldsymbol{T}$ or $\neg F \in \boldsymbol{T}$. Clearly the theory of any particular structure is complete; however the theory of an axiomatically presented class of structures can easily fail to be so. For example, the theory of groups is not complete: if m denotes the binary multiplication operation then the theory of groups neither contains the sentence $\forall x \, \forall y \, (m(x, y) = m(y, x))$ nor its negation (some groups are abelian and other groups are non-abelian). More simply, the set of valid σ-formulas is an example of a theory that is not complete.

We say that a theory $\boldsymbol{T}$ admits *quantifier elimination* if for any formula $\exists x \, F$, with F quantifier-free, there exists a quantifier-free formula G with the same free variables as $\exists x \, F$ such that $\boldsymbol{T} \models \exists x \, F \leftrightarrow G$, that is, for any assignment $\mathcal{A}$ that is a model of $\boldsymbol{T}$, $\mathcal{A} \models \exists x \, F$ if and only if $\mathcal{A} \models G$. (It is worth emphasising that quantifier elimination is defined on formulas that may have free variables.) We furthermore say that $\boldsymbol{T}$ has a *quantifier elimination procedure* if there is an algorithm to obtain G given F.

Example 1. Let $\boldsymbol{T}$ denote the theory of the structure $(\mathbb{R}, +, \cdot, 0, 1)$ and consider the formula $F := \exists x \, (ax^2 + bx + c = 0)$ in free variables a, b, c. This formula asserts that the quadratic equation $ax^2 + bx + c = 0$ has a real solution. By the quadratic formula we have $\boldsymbol{T} \models F \leftrightarrow b^2 \geq 4ac$. As another example, consider the formula

$$F := (x_1 a + x_2 c = 1) \wedge (x_1 b + x_2 d = 0) \wedge (x_3 a + x_4 c = 0) \wedge (x_3 b + x_4 d = 1) \, .$$

F can be written $\begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ in matrix notation. Thus $\exists x_1 \exists x_2 \exists x_3 \exists x_4 F$ asserts that the matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ has a multiplicative inverse. Thus $\boldsymbol{T} \models \exists x_1 \exists x_2 \exists x_3 \exists x_4 F \leftrightarrow ad - bc \neq 0$.

The definition of quantifier elimination refers only to the existential quantifier. The universal quantifier can be handled using duality. Consider a formula $\forall x \, F$ with F quantifier-free. If a theory

$\boldsymbol{T}$ has quantifier elimination then we can find a quantifier-free formula G such that $\boldsymbol{T} \models \exists x \neg F \leftrightarrow G$. But then $\boldsymbol{T} \models \forall x\, F \leftrightarrow \neg G$.

A theory $\boldsymbol{T}$ is *decidable* if there is an algorithm that, given a sentence F, determines whether or not $F \in \boldsymbol{T}$. A theory $\boldsymbol{T}$ is decidable if it has a quantifier elimination-procedure and a procedure for determining whether or not $F \in \boldsymbol{T}$ for a variable-free atomic formula F. Given an arbitrary formula F, to determine whether $F \in \boldsymbol{T}$, first convert F to an equivalent formula in prenex normal form, and eliminate quantifiers from the inside out. In particular, if $\boldsymbol{T} \models \exists x\, F^* \leftrightarrow G$ then $\boldsymbol{T} \models Q_1 x_1 \ldots Q_n x_n Q x\, F^* \leftrightarrow Q_1 x_1 \ldots Q_n x_n\, G$, where $Q_i, Q \in \{\exists, \forall\}$. Eventually one obtains a sentence F' such that $\boldsymbol{T} \models F \leftrightarrow F'$. Thus $F \in \boldsymbol{T}$ if and only if $F' \in \boldsymbol{T}$. But by assumption we have a procedure to decide this last membership query.

2 Unbounded Dense Linear Orders

Consider a signature with a single binary relation $<$. The theory $\boldsymbol{T}_{UDLO}$ of *unbounded dense linear orders* is the set of sentences entailed by the following set of axioms:

$$
\begin{array}{ll}
F_1 & \forall x \, \neg(x < x) \\
F_2 & \forall x \, \forall y \, \forall z \, (x < y \wedge y < z \rightarrow x < z) \\
F_3 & \forall x \, \forall y \, (x < y \vee y < x \vee x = y) \\
F_4 & \forall x \, \forall y \, (x < y \rightarrow \exists z \, (x < z \wedge z < y)) \\
F_5 & \forall x \, \exists y \, \exists z \, (y < x < z) \, .
\end{array}
$$

Theorem 2. The theory $\boldsymbol{T}_{UDLO}$ of unbounded dense linear orders is complete, decidable, and has quantifier elimination.

Proof. The main step of the proof is to show that $\boldsymbol{T}_{UDLO}$ has an effective quantifier-elimination procedure. Consider a formula $\exists x\, F$, with F quantifier-free. We give a quantifier-free formula G with the same free variables as $\exists x\, F$ such that for any assignment $\mathcal{A}$ that is a model of $\boldsymbol{T}_{UDLO}$, $\mathcal{A} \models \exists x\, F$ if and only if $\mathcal{A} \models G$. The quantifier-elimination procedure has two phases: first we simplify the formula F through logical manipulations and then we show how to eliminate quantifiers within formulas in simplified form.

As a first step, we can convert F into a logically equivalent formula in DNF. We can moreover eliminate negative literals by replacing the subformula $\neg(x_i < x_j)$ with $x_i = x_j \vee x_j < x_i$ and replacing the subformula $\neg(x_i = x_j)$ with $x_i < x_j \vee x_j < x_i$.

Henceforth we assume that F is in DNF and negation-free. Now using the equivalence $\exists x\, (F_1 \vee F_2) \equiv \exists x\, F_1 \vee \exists x\, F_2$ it suffices that we be able to eliminate the quantifier $\exists x$ in case F is a conjunction of atomic formulas. Finally, using the equivalence $\exists x\, (F_1 \wedge F_2) \equiv \exists x\, F_1 \wedge F_2$ in case x is not free in F_2, it suffices that we be able to eliminate the quantifier $\exists x$ in case F is a conjunction of atomic formulas all of which mention x. Such formulas have the form $x = y$, $x < y$ or $y < x$ for some variable y.

For the final case above, we proceed as follows. If F contains a conjunct $x < x$ then we have $\boldsymbol{T}_{UDLO} \models \exists x\, F \leftrightarrow$ **false**. Otherwise, if F contains a conjunct $x = y$ for some other variable y then we have that $\boldsymbol{T}_{UDLO} \models \exists x\, F \leftrightarrow F[y/x]$.

If neither of the above applies then (after deleting conjuncts of the form $x = x$ if present) we can write F in the form

$$F = \bigwedge_{i=1}^{m} l_i < x \wedge \bigwedge_{j=1}^{n} x < u_j \, ,$$

where the l_i and u_j are variables different from x. Now if $m = 0$, i.e., there are no lower bounds on x, then $\boldsymbol{T}_{UDLO} \models \exists x\, F \leftrightarrow \textbf{true}$ (since we're considering the theory of unbounded orders). Likewise if $n = 0$, i.e., there are no upper bounds on x, then $\boldsymbol{T}_{UDLO} \models \exists x\, F \leftrightarrow \textbf{true}$. Otherwise, by density of the order relation, we have

$$\boldsymbol{T}_{UDLO} \models \exists x\, F \leftrightarrow \bigwedge_{i=1}^{m} \bigwedge_{j=1}^{n} l_i < u_j \, .$$

Decidability of $\boldsymbol{T}_{UDLO}$ follows straightforwardly from the existence of a quantifier-elimination procedure. Starting from a sentence F, after eliminating all quantifiers from F we are left with a variable-free formula G such that $\boldsymbol{T} \models F \leftrightarrow G$. But G must be a propositional combination of **true** or **false**, and therefore logically equivalent to either **true** or **false**. The same reasoning shows *inter alia* that $\boldsymbol{T}_{UDLO}$ is complete: given a sentence F, either F holds on all unbounded dense linear orders, or its negation holds on all unbounded dense linear orders. $\square$

Theorem 2 shows that $(\mathbb{Q}, <)$ and $(\mathbb{R}, <)$ satisfy the same first-order sentences. (This finally answers Exercise 7 from the lecture introducing first-order logic.) You may recall that $(\mathbb{R}, <)$ is *Dedekind complete*: any non-empty set of reals that is bounded above has a least upper bound. This property fails for the rationals since, e.g., $\{x \in \mathbb{Q} : x^2 < 2\}$ has no least upper bound in the rationals. Evidently Dedekind completeness cannot be expressed in first-order logic in the language of linear orders.

3 Ordered Divisible Abelian Groups

Consider a signature with a binary relation symbol $<$, binary function symbol $+$, and a constant symbol 0. Via an obvious notational shortcut, it will be convenient to admit $\mathbb{Z}$-linear expressions in variables as terms. For example, we write $3x + y$ for the term $x + (x + (x + y))$ and we write $x - 2y < z$ for the formula $x < (z + y) + y$.

The set of axioms $\{F_1, \ldots, F_8\} \cup \{G_n : n \in \mathbb{N}_+\}$, shown below, determines the theory $\boldsymbol{T}_{\mathrm{ODAG}}$ of (non-trivial) ordered divisible abelian groups.

$$\begin{array}{ll}
F_1 & \forall x\, \neg(x < x) \\
F_2 & \forall x\, \forall y\, \forall z\, (x < y \wedge y < z \to x < z) \\
F_3 & \forall x\, \forall y\, (x < y \vee y < x \vee x = y) \\
F_4 & \forall x\, \forall y\, (x + y = y + x) \\
F_5 & \forall x\, \forall y\, \forall z\, ((x + y) + z = x + (y + z))) \\
F_6 & \forall x\, \exists y\, (x + y = 0) \\
F_7 & \forall x\, \forall y\, \forall z\, (x < y \to x + z < y + z) \\
F_8 & \exists x\, \neg(x = 0) \\
G_n & \forall x\, \exists y\, (ny = x)\, .
\end{array}$$

A model of $\boldsymbol{T}_{\mathrm{ODAG}}$ is a structure $(A, <, +, 0)$ such that $(A, <)$ is a linearly ordered set, $(A, +, 0)$ is a divisible Abelian group, addition $+$ is monotone in both variables, and A is non-trivial as a group (it has some non-zero element). Examples include $(\mathbb{R}, <, +, 0)$ and $(\mathbb{Q}, <, +, 0)$. We leave it as an exercise to show that for any model $(A, <, +, 0)$ of $\boldsymbol{T}_{\mathrm{ODAG}}$ the order $<$ is unbounded and dense.

Theorem 3. $\boldsymbol{T}_{\mathrm{ODAG}}$ has quantifier elimination.

Proof. Following the proof of Theorem 2, it suffices to show how to eliminate the quantifier $\exists x$ in $\exists x\, F$, where F is a conjunction of atomic formulas all of which mention x. Each such atomic formula has the form $t_1 < t_2$ or $t_1 = t_2$ for terms t_1 and t_2, where at least one of t_1 or t_2 mentions x. We can further simplify to assume that there exists a positive integer m such that each formula in F has the form $mx < t$, $t < mx$, or $mx = t$ for some term t (where some variables may occur in t with negative coefficients). Suppose that F has the form

$$\bigwedge_{i=1}^{n_1} t_i < mx \wedge \bigwedge_{j=1}^{n_2} mx < s_j \wedge \bigwedge_{k=1}^{n_3} mx = u_k \,.$$

Then, by the divisibility axioms, $\boldsymbol{T}_{\mathrm{ODAG}} \models \exists x\, F \leftrightarrow \exists y\, G$ where

$$G := \bigwedge_{i=1}^{n_1} t_i < y \wedge \bigwedge_{j=1}^{n_2} y < s_j \wedge \bigwedge_{k=1}^{n_3} y = u_k \,,$$

for some fresh variable y not occurring in F. If $n_3 > 0$ then $\exists y\, G \equiv G[u_1/y]$. If $n_3 = 0$ then $T_{\mathrm{ODAG}} \models \exists y\, G \leftrightarrow \bigwedge_{i=1}^{n_1} \bigwedge_{=1}^{n_2} t_i < s_j$. $\square$

The quantifier elimination procedure for $\boldsymbol{T}_{\mathrm{ODAG}}$ can be used to solve certain elementary problems in convex geometry. In this context quantifier elimination is sometimes called *Fourier-Motzkin elimination*. For example, given matrices A and C and vectors $\boldsymbol{b}$ and $\boldsymbol{d}$, all with rational entries, determining the whether the polygon $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \leq \boldsymbol{b}\}$ is included in the polygon $\{\boldsymbol{x} \in \mathbb{R}^n : C\boldsymbol{x} \leq \boldsymbol{d}\}$ can straightforwardly be reduced to the decision problem for $\boldsymbol{T}_{\mathrm{ODAG}}$.

4 The Random Graph

Let σ be the signature with a single binary relation symbol R. The σ-theory $\boldsymbol{T}_{\mathrm{RG}}$ is axiomatised by the set of sentences $\{F_1, F_2, F_3\}$, which axiomatise the class of undirected graphs wIth at least two vertices, and the *extension axioms* $\{H_{m,n} : m, n \in \mathbb{N}\}$, given as follows.

$$\begin{array}{ll}
F_1 & \exists x\, \exists y\, \neg(x = y) \\
F_2 & \exists x \neg R(x, x) \\
F_3 & \forall x\, \forall y (R(x, y) \to R(y, x)) \\
H_{m,n} & \forall x_1 \dots \forall x_m\, \forall y_1 \dots \forall y_n \left(\bigwedge_{i=1}^{m} \bigwedge_{j=1}^{n} \neg(x_i = y_j) \to \exists z \bigwedge_{i=1}^{m} R(x_i, z) \wedge \bigwedge_{j=1}^{n} \neg R(y_j, z) \right)
\end{array}$$

We call $\boldsymbol{T}_{\mathrm{RG}}$ the *theory of the random graph.*

The so-called *Rado Graph* is a model of $\boldsymbol{T}_{\mathrm{RG}}$. This has the positive integers as its set of vertices and two integers $m < n$ are connected by an undirected edge iff the m-th bit in the infinite binary expansion of n is 1, i.e., writing $n = \sum_{i=0}^{\infty} b_i 2^i$ with $b_i \in \{0, 1\}$, we have $b_m = 1$.

Theorem 4. $\boldsymbol{T}_{\mathrm{RG}}$ is complete, decidable, and has quantifier elimination.

Proof. Completeness and decidability follow from the existence of an effective quantifier-elimination procedure, since every quantifier-free σ-sentence is equivalent to either **true** of **false**. To eliminate quantifiers in general it suffices to eliminate quantifiers in the case of a formula $\exists x\, F$, where F arises as a conjunction of atoms and negated atoms. See Exercise Sheet 5 for details. $\square$

For a positive integer N, let $\boldsymbol{G}_N$ be the set of all graphs with set of vertices $\{1, \ldots, N\}$. For a σ-sentence φ, we denote by $\mathrm{Pr}_N(\varphi)$ the probability that φ is satisfied by a graph drawn uniformly at random from $\boldsymbol{G}_N$, that is,

$$\mathrm{Pr}_N(\varphi) := \frac{|\{G \in \boldsymbol{G}_N : G \models \varphi\}|}{|\boldsymbol{G}_N|}\,.$$

Proposition 5. For all $m, n \in \mathbb{N}$ we have $\lim_{N\to\infty} \mathrm{Pr}_N(H_{m,n}) = 1$.

Proof. Let $N > m+n$. Fix $a_1, \ldots, a_m, b_1, \ldots, b_n \in \{1, \ldots, N\}$. We claim that for a graph G drawn uniformly at random from $\boldsymbol{G}_N$ the probability that

$$H_{[x_1\mapsto a_1,\ldots,x_m\mapsto a_m,y_1\mapsto b_1,\ldots,y_n\mapsto b_n]} \not\models \exists z \left(\bigwedge_{i=1}^{m} E(x_i, z) \wedge \bigwedge_{j=1}^{n} E(y_j, z) \right)$$

is at most q^{N-m-n}, where $q := 1 - 2^{-(n+m)} < 1$. Indeed, for each possible choice of c from $\{1, \ldots, N\} \setminus \{a_1, \ldots, a_m, b_1, \ldots, b_n\}$, the probability that

$$H_{[x_1\mapsto a_1,\ldots,x_m\mapsto a_m,y_1\mapsto b_1,\ldots,y_n\mapsto b_n,z\mapsto c]} \not\models \bigwedge_{i=1}^{m} E(x_i, z) \wedge \bigwedge_{j=1}^{n} E(y_j, z)$$

is at most q. Since these are independent events for the (at least $N-m-n$ many) different choices of c, the claim follows. Given the claim, taking a union bound over the N^{n+m} possible choices of $a_1, \ldots, a_m, b_1, \ldots, b_n \in \{1, \ldots, N\}$ we have that $\mathrm{Pr}_N(\neg H_{m,n}) \leq N^{n+m} q^{N-(n+m)}$. Since $q < 1$ we have $\lim_{N\to\infty} \mathrm{Pr}_N(H_{m,n}) = 1$. $\square$

.

We can now prove the following zero-one law for first-order logic over the language of graphs.

Theorem 6. For every σ-formula φ the limit $\lim_{N\to\infty} \mathrm{Pr}_N(\varphi)$ exists and is either zero or one. Moreover $\boldsymbol{T}_{\mathrm{RG}} = \{\varphi : \lim_{N\to\infty} \mathrm{Pr}_N(\varphi) = 1\}$.

Proof. We have already established that $\boldsymbol{T}_{\mathrm{RG}}$ is complete. Thus to prove the theorem it suffices to show that $\lim_{N\to\infty} \mathrm{Pr}_N(\varphi) = 1$ for every formula φ in $\boldsymbol{T}_{\mathrm{RG}}$. But, by the compactness theorem for first-order logic, there exist $m, n \in \mathbb{N}$ such that $\{F_1, F_2, F_3, H_{m,n}\}$ entails φ (we can take a single extension axiom here since $H_{m,n} \models H_{m',n'}$ whenever $m \geq m'$ and $n \geq n'$). Hence $\mathrm{Pr}_N(\varphi) \geq \mathrm{Pr}_N(H_{m,n})$, which entails $\lim_{N\to\infty} \mathrm{Pr}_N(\varphi) = 1$. $\square$

5 Presburger Arithmetic

Our final decidability result concerns the theory of the structure $(\mathbb{N}, 0, 1, +, <)$, sometimes called *Presburger arithmetic*. In this case the proof of decidability does not proceed via quantifier elimination, but instead exploits closure properties of the class of regular languages. In fact $\mathrm{Th}(\mathbb{N}, 0, 1, +, <)$ does not have quantifier elimination since, e.g., the formula $\exists y\,(x = y + y)$ is not equivalent to a quantifier-free formula

Recall that a *regular language* is a language accepted by a *nondeterministic finite automaton (NFA)*. Recall also that the class of regular languages is closed under intersection and complementation, and under direct and inverse images with respect to renaming functions. Amplifying the last two closure properties, recall that a renaming function is a map $f : \Sigma \to \Gamma$ between two alphabets. We extend such a function pointwise to a map $f : \Sigma^* \to \Gamma^*$ by defining $f(\sigma_1 \ldots \sigma_m) = f(\sigma_1) \ldots f(\sigma_m)$. Then given a regular language $L \subseteq \Gamma^*$, its *inverse image* $f^{-1}(L) = \{w \in \Sigma^* : f(w) \in L\}$ is also regular. Likewise given a regular language $L \subseteq \Sigma^*$, its *direct image* $f(L) = \{f(w) : w \in L\}$ is also regular.

Importantly the above closure properties are all effective. For example, let $A = (\Gamma, Q, Q_0, \Delta, F)$ be a NFA for a given language $L \subseteq \Gamma^*$, with set of states Q, initial states Q_0, final states F, and transition relation $\Delta \subseteq Q \times \Gamma \times Q$. Then, given a renaming map $f : \Sigma \to \Gamma$, an NFA for the inverse image $f^{-1}(L)$ is $B = (\Sigma, Q, Q_0, \Delta', F)$, with transition relation Δ' given by $\Delta' = \{(p, \sigma, q) : (p, f(\sigma), q) \in \Delta\}$. We leave an an exercise the straightforward proof that this construction does the job.

Theorem 7. $\mathrm{Th}(\mathbb{N}, 0, 1, +, <)$ is decidable.

Proof. It will suffice to show that $\mathrm{Th}(\mathbb{N}, +)$ is decidable, since any formula over the richer signature can be rewritten to a formula using only $+$ (and equality) that defines the same property on $\mathbb{N}$. (We leave it as an exercise to check this.)

Consider a quantifier-free formula F that mentions variables $x_1, \ldots, x_n$. We show how to define an automaton A_F over the alphabet of n-dimensional bit vectors

$$\Sigma_n = \left\{ \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \right\}$$

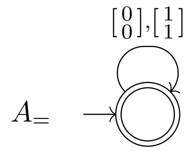
whose language is in one-to-one correspondence with the set of values of the free variables $x_1, \ldots, x_n$ that satisfy F. Here each natural number is encoded in binary, with the value for x_i represented in the i-th component of each tuple in Σ_n. For example, the valuation $x_1 = 1, x_2 = 4, x_3 = 9$ is encoded by the word

$$\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

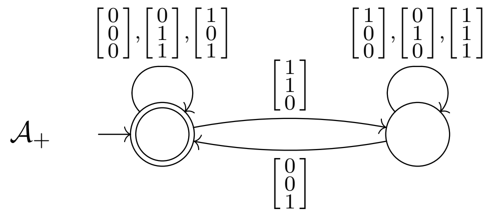
where the least significant bits occur on the left. Note that Σ_0 is a singleton set consisting of the empty vector $\{[]\}$ (the only 0-dimensional bit vector).

The construction of A_F is predicated on the following two *basic automata*.

We have the following one-state automaton $A_=$ over the alphabet Σ_2, corresponding to the equality relation $x_1 = x_2$:



And we have the a two-state automata A_+ over the alphabet Σ_3, corresponding to the addition function $x_1 + x_2 = x_3$:



We now define the automaton A_F by induction on the structure of the formula F. The construction proceeds from the atoms $A_=$ and A_+ using only the closure properties of the class of regular languages.

Base cases: Suppose F is the formula $x_i = x_j$. Then the automaton A_F is defined to be automaton whose language is $\pi^{-1}(L(A_=))$, where $\pi : \Sigma_n \to \Sigma_2$ is the projection map

$$\pi : \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \mapsto \begin{bmatrix} x_i \\ x_j \end{bmatrix}$$

Likewise, if F is the formula $x_i + x_j = x_k$, then A_F is defined to be an automaton whose language is $\pi^{-1}(L(A_+))$, where $\pi : \Sigma_n \to \Sigma_3$ is the projection map

$$\pi : \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \mapsto \begin{bmatrix} x_i \\ x_j \\ x_k \end{bmatrix}$$

Case: $F = F_1 \wedge F_2$. Then we define A_F to be an automaton whose language is $L(A_{F_1}) \cap L(A_{F_2})$.

Case: $F = \neg G$. Then we define A_F to be the automaton with language $\Sigma_n^* \setminus L(A_G)$.

This completes the definition of the automaton A_F corresponding to a quantifier-free formula F. Now consider a sentence $Q_1 x_1 \ldots Q_n x_n\, F$ in prenex form. For $k = 0, \ldots, n$, we write $F_k := Q_{k+1} x_{k+1} \ldots Q_n x_n F^*$ and define a corresponding automaton A_k over alphabet Σ_k such that A_k accepts the set of values of the variables $x_1, \ldots, x_k$ that satisfy F_k. In particular, an invariant of this construction is that A_k has non-empty language if and only if formula F_k is satisfiable.

We start by defining A_n to be the automaton A_F, as constructed above.

Now suppose that $F_{k-1} = \exists x_k F_k$. By induction we have an automaton A_k on alphabet Σ_k corresponding to F_k. Then we define A_{k-1} to be an automaton whose language is $\pi(L(A_k))$, where $\pi : \Sigma_k \to \Sigma_{k-1}$ is the map the projects out the k-th coordinate of each tuple in Σ_k.

Finally we handle the universal quantifier $\forall x_k$ by treating it as shorthand for $\neg \exists x_k \neg$.

We end up with an automaton A_0 for the sentence F_0 (which is $Q_1 x_1 \ldots Q_n x_n\, F$) over the alphabet Σ_0. This automaton has non-empty language if and only if $(\mathbb{N}, +)$ satisfies F_0. $\square$