Interpreting a field in its Heisenberg group

Julia F. Knight[1]

August 28, 2020

[1] Joint work with Rachael Alvir, Wesley Calvert, Grant Goodman, Valentina Harizanov, Andrey Morozov, Russell Miller, Alexandra Soskova, and Rose Weisshaar

Outline

1. Formulas of different kinds
2. Defining, interpreting one structure in another
3. Effective interpretations and computable functors
4. $H(F)$
5. Maltsev's definition of F in $H(F)$, with parameters
6. Effective interpretation of F in $H(F)$, without parameters
7. Generalizing
8. Questions

Elementary first order formulas

Elementary first order formulas are formulas of the usual kind—finitely long, with quantifiers ranging over elements. Here are some of the familiar elementary first order axioms for ordered fields.

- $(\forall x)(\forall y)(\forall z)\ x + (y + z) = (x + y) + z$
- $(\forall x)[x \neq 0 \rightarrow (\exists y) x \cdot y = 1]$
- $(\forall x)(\forall y)(\forall z)((x < y \ \& \ y < z) \rightarrow x < z)$

Complexity

Fact: We can put any elementary first order formula into *prenex normal form*, bringing the quantifiers outside.

We measure complexity of formulas in prenex normal form by the number of alternations of $\exists$ and $\forall$.

Classification. A formula in prenex normal form is Σ_n, Π_n, if it has n blocks of like quantifiers, starting with $\exists$, $\forall$.

Definable sets, relations

Convention. We identify a structure $\mathcal{A}$ with its atomic diagram $D(\mathcal{A})$. For $\mathcal{A}$ with universe a subset of ω, we identify $D(\mathcal{A})$ (via Gödel numbering) with a subset of ω, or with $\chi_{D(\mathcal{A})}$.

Notation. For a formula $\varphi(\bar{x})$, we write $\varphi^{\mathcal{A}}$ for the relation on $\mathcal{A}$ defined by $\varphi(\bar{x})$.

Facts:

1. If $\varphi(\bar{x})$ is existential, then $\varphi^{\mathcal{A}}$ is computably enumerable *relative to* $\mathcal{A}$.
2. For an *operation* F, if the relation $F(\bar{x}) = y$ is defined by an existential formula, then so is the relation $F(\bar{x}) \neq y$—we say $(\exists z)(z \neq y \ \& \ F(\bar{x}) = z)$.

Formulas of $L_{\omega_1\omega}$

$L_{\omega_1\omega}$ is a version of infinitary logic that allows countably infinite disjunctions and conjunctions, but only finite strings of quantifiers.

Example. To say of an ordered field that it is Archimedean, we write the $L_{\omega_1\omega}$-formula $(\forall x) \bigvee_n x < \underbrace{1 + \cdots + 1}_{n}$.

Normal form. For an $L_{\omega_1\omega}$-formula, we cannot, in general, bring the quantifiers to the front. However, we can bring the negations inside. This gives a kind of normal form.

We classify $L_{\omega_1\omega}$-formulas in this normal form according to the number of alternations of $\bigvee(\exists)$ and $\bigwedge(\forall)$.

Complexity of $L_{\omega_1\omega}$-formulas

1. $\varphi(\bar{x})$ is Σ_0 and Π_0 if it is finitary quantifier-free,
2. for a countable ordinal $\alpha > 0$,
   (a) $\varphi(\bar{x})$ is Σ_α if it has form $\bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where each ψ_i is Π_{β_i} for some $\beta_i < \alpha$,
   (b) $\varphi(\bar{x})$ is Π_α if it has form $\bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where each ψ_i is Σ_{β_i} for some $\beta_i < \alpha$

Computable infinitary formulas

The *computable infinitary formulas* are infinitary formulas in which the infinite disjunctions and conjunctions are over c.e. sets. Such formulas are in some sense "comprehensible." We classify these formulas as *computable* Σ_α, *computable* Π_α, where α ranges over computable ordinals.

A computable Σ_1 formula is a c.e. disjunction of finitary existential formulas.

Fact: The relation defined in a structure $\mathcal{A}$ by a computable Σ_1 formula is c.e. relative to $\mathcal{A}$.

Second order formulas

Second order formulas allow quantifiers ranging over sets.

Example. To characterize the ordered field of reals, up to isomorphism, we add to the elementary first order axioms for ordered fields the second order *Completeness Axiom*, stated below using some abbreviations.

- $N(X) = (\exists y)Xy$ says X is non-empty.
- $B(X, u) = (\forall y)(Xy \rightarrow y \leq u)$ says u is an upper bound for X.
- $L(X, v) = B(X, v) \ \& \ (\forall u)(B(X, u) \rightarrow v \leq u)$.
- $(\forall X)[(N(X) \ \& \ (\exists u)B(X, u)) \rightarrow (\exists v)L(X, v)]$.

Examples of definable sets

1. **Lagrange**. The set $\mathbb{N}$ is defined in the ring $\mathbb{Z}$ by the existential (Σ_1) formula $(\exists y_1)(\exists y_2)(\exists y_3)(\exists y_4)\ x = y_1^2 + y_2^2 + y_3^2 + y_4^2$.
2. The set of integers is defined in the field $\mathbb{Q}$.
   (a) There is a computable Σ_1 definition $\bigvee_i \varphi_{i \in \mathbb{Z}}(x)$, where $\varphi_i(x)$ is a natural quantifier-free formula saying that x is the integer i,
   (b) **J. Robinson**. There is an elementary first order definition, with several alternations of quantifiers.
   (c) **Königsmann**. There is a universal (Π_1) definition.

Defining one structure in another

For simplicity, suppose $\mathcal{A} = (A, R_i)$ is relational.

Definition. $\mathcal{A}$ is *defined* in $\mathcal{B}$ if there are formulas that in $\mathcal{B}$ define a set D and relations R_i^* on D s.t. $(D, R_i^*) \cong \mathcal{A}$.

Example. The field $\mathbb{C}$ is defined in the field $\mathbb{R}$ as follows:

- D is the set of ordered pairs $(a, b) \in \mathbb{R}^2$—think of (a, b) as $a + bi$,
- $+^*$ is the set of $(a, b)(a', b')(a'', b'')$ s.t. $a'' = a + a'$ and $b'' = b + b'$,
- $\cdot^*$ is the set of $(a, b)(a', b')(a'', b'')$ s.t. $a'' = aa' - bb'$ and $b'' = ab' + a'b$.

Interpreting one structure in another

Definition. $\mathcal{A} = (A, R_i)$ is *interpreted* in $\mathcal{B}$ if there are formulas that in $\mathcal{B}$ define a set D, relations R_i^* on D, and a congruence relation $\sim$ on (D, R_i^*), s.t. $(D, R_i^*)/_\sim \cong \mathcal{A}$.

Example. The field $\mathbb{Q}$ is interpreted in the ring $\mathbb{Z}$ as follows:

- $D = \{(a, b) : b \neq 0\}$—think of (a, b) as $\frac{a}{b}$,
- $(a, b) \sim (a', b')$ if $ab' = a'b$,
- $+^*$ is the set of $(a, b)(a', b')(a'', b'')$ s.t. $(a'', b'') \sim (ab' + a'b, bb')$,
- $\cdot^*$ is the set of $(a, b)(a', b')(a'', b'')$ s.t. $(a'', b'') \sim (a \cdot a', b \cdot b')$.

Generalized computable Σ_1 formulas

Definition. A *generalized* computable Σ_1 formula is a c.e. disjunction of finitary existential formulas, possibly of different arities.

Example. Linear dependence in $\mathbb{Q}$-vector spaces is defined by a generalized computable Σ_1 formula.

Fact: The relation defined in a structure $\mathcal{B}$ by a generalized computable Σ_1 formula is c.e. relative to $\mathcal{B}$.

Effective interpretation

Definition (Montalbán). $\mathcal{A} = (A, R_i)$ is *effectively interpreted* in $\mathcal{B}$ if there exist $D \subseteq \mathcal{B}^{<\omega}$, relations R_i^* and a congruence relation $\sim$ s.t. $(D, R_i^*)/_\sim \cong \mathcal{A}$, and D, R_i^*, $\neg R_i^*$, $\sim$, and $\not\sim$ are all defined by generalized computable Σ_1 formulas, with no parameters.

Note: If R_i is the relation $F(\bar{x}) = y$, where F is an operation, we get a generalized Σ_1-definition for $\neg R_i^*$ from those for R_i^* and $\not\sim$.

Fact. If there is an effective interpretation of $\mathcal{A}$ in $\mathcal{B}$, then there is a Turing operator Φ that takes each copy of $\mathcal{B}$ to a copy of $\mathcal{A}$.

Somewhat disturbing

Fact. If $\mathcal{A}$ is a computable structure, then $\mathcal{A}$ is effectively interpreted in any structure $\mathcal{B}$.

Proof: For simplicity, suppose $\mathcal{A} = (\omega, R)$, where R is binary. Let $D = \mathcal{B}^{<\omega}$. Let $\bar{b} \sim \bar{b}'$ if $\bar{b}$ and $\bar{b}'$ have the same length. Let R^* consist of the pairs of tuples $(\bar{b}, \bar{c})$ s.t. $\bar{b}$ has length m, $\bar{c}$ has length n, and $(m, n) \in R$.

Computable functor

Definition (R. Miller). A *computable functor* from $\mathcal{B}$ to $\mathcal{A}$ is a pair of Turing operators Φ, Ψ s.t.

1. Φ takes copies of $\mathcal{B}$ to copies of $\mathcal{A}$,
2. for each triple $(\mathcal{B}', f, \mathcal{B}'')$ s.t. $\mathcal{B}', \mathcal{B}'' \cong \mathcal{B}$ and $\mathcal{B}' \cong_f \mathcal{B}''$, $\Psi(\mathcal{B}', f, \mathcal{B}'')$ is an isomorphism from $\Phi(\mathcal{B}')$ to $\Phi(\mathcal{B}'')$,
3. Ψ preserves identity and composition; i.e.,
   - if f is the identity on $\mathcal{B}'$, then $\Psi(\mathcal{B}', f, \mathcal{B}')$ is the identity on $\Phi(\mathcal{B}')$,
   - if $\Psi(\mathcal{B}, f, \mathcal{B}') = g$ and $\Psi(\mathcal{B}', f', \mathcal{B}'') = g'$, then $\Psi(\mathcal{B}, f' \circ f, \mathcal{B}'') = g' \circ g$.

Equivalence

Theorem (Harrison-Trainor-Melnikov-R. Miller-Montalbán). There is an effective interpretation of $\mathcal{A}$ in $\mathcal{B}$ iff there is a computable functor from $\mathcal{B}$ to $\mathcal{A}$.

Note: In the proof, it is important that D consists of tuples of arbitrary length—the formulas that define $\sim$, $\not\sim$, R_i^*, $\neg R_i{}^*$ are generalized computable Σ_1.

Heisenberg group

For a field F, the Heisenberg group $H(F)$ is the group of matrices

$$h(a, b, c) = \begin{bmatrix} 1 & a & c \\ 0 & 1 & b \\ 0 & 0 & 1 \end{bmatrix},$$

where $a, b, c \in F$.

Operation: matrix multiplication.

Identity: $h(0, 0, 0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

More on the Heisenberg group

1. $H(F)$ is not Abelian; for example,

$$h(1,0,0) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; h(0,1,0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

   do not commute.

2. The *center* consists of the elements

$$h(0,0,a) = \begin{bmatrix} 1 & 0 & a \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Maltsev definition, using parameters

We write 1 for the group identity and $[x, y]$ for the *commutator* $x^{-1}y^{-1}xy$.

Maltsev. For a field F, let $u = h(u_1, u_2, u_3)$, $v = h(v_1, v_2, v_3)$ be a non-commuting pair in $H(F)$. There is a copy $F_{(u,v)}$ of F defined in $H(F)$, with parameters (u, v), as follows:

1. D is the center—$x \in D$ iff $[x, u] = [x, v] = 1$,
2. $+$ is the group operation,
3. $x \cdot y = z$ if there exist x', y' s.t. $[x', u] = [y', v] = 1$, $[x', v] = x$, $[u, y'] = y$, and $[x', y'] = z$.

Isomorphism from F to $F_{(u,v)}$

Let $F_{(u,v)}$ be the copy of F defined in $H(F)$ with parameters (u,v), where $u = h(u_1, u_2, u_3), v = h(v_1, v_2, v_3)$. Let

$$\Delta_{(u,v)} = \begin{vmatrix} u_1 & v_1 \\ u_2 & v_2 \end{vmatrix}.$$

Proposition (Maltsev, Morozov). $F \cong_{g_{(u,v)}} F_{(u,v)}$, where

$$g_{(u,v)}(\alpha) = h(0, 0, \alpha \cdot \Delta_{(u,v)}).$$

Parameters needed

Proposition. For all fields F, F cannot be defined in $H(F)$ without parameters (using formulas with fixed arity).

Proof: If we had a copy of F defined in $H(F)$, then every automorphism of $H(F)$ would induce an automorphism of the copy, fixing the elements (or tuples), that represent 0 and 1. However, the only element of $H(F)$ fixed by all automorphisms is 1, and the only n-tuple fixed by all automorphisms is $(\underbrace{1, \ldots, 1}_{n})$.

Recovering a copy of F from a copy of $H(F)$

Proposition. There is a uniform Turing operator Φ that, for all F, takes copies of $H(F)$ to copies of F.

Proof: We look for a non-commuting pair (u, v) in G, and, for the first we find, take the copy of F defined using these parameters.

Computable functor from $H(F)$ to F

We have a uniform Turing operator Φ that takes copies of $H(F)$ to copies of F. This is half of a computable functor. Using the following, we get the other half.

Lemma. There is an existential formula $\psi(u, v, u', v', x, y)$ that, for any non-commuting pairs (u, v) and (u', v') in $H(F)$, defines an isomorphism $f_{(u,v),(u'v')}$ from $F_{(u,v)}$ to $F_{(u',v')}$. Moreover, the family of isomorphisms is functorial—$f_{(u,v),(u,v)}$ is the identity and $f_{(u',v')(u'',v'')} \circ f_{(u,v),(u',v')} = f_{(u,v)(u'',v'')}$.

Proposition. There is a computable functor from $H(F)$ to F.

Effective interpretation of F in $H(F)$

Completing the First Proof: We have a computable functor from $H(F)$ to F. Applying the theorem of HTMMM, we get the existence of an effective interpretation of F in $H(F)$.

The proof of HTMMM gives an interpretation in which D consists of tuples of arbitrary arity, and D, $\sim$, $\not\sim$, and the operations are defined by generalized computable Σ_1-formulas.

There is a second proof, explicitly defining an interpretation.

Explicit definition

Proposition. There are finitary existential formulas, with no parameters, that for all fields F, define an interpretation of F in $H(F)$.

Proof:

1. D is the set of $(u, v, x) \in H(F)$ s.t. $[u, v] \neq 1$ and $[x, u] = [x, v] = 1$,
2. $(u, v, x) \sim (u', v', x')$ if $f_{(u,v)(u',v')}(x) = x'$,
3. $(u, v, x) \not\sim (u', v', x')$ if $f_{(u,v)(u',v')}(x) \neq x'$,
4. $+^*((u, v, x), (u', v', y), (u'', v'', z))$ if there exist y', z' s.t. $(u, v, y') \sim (u', v', y)$, $(u, v, z') \sim (u'', v'', z)$, and $M_{(u,v)} \models x + y' = z'$,
5. $\cdot^*((u, v, x), (u', v', y), (u'', v'', z))$ if there exist y', z' s.t. $(u, v, y') \sim (u', v', y)$, $(u, v, z') \sim (u'', v'', z)$, and $M_{(u,v)} \models x \cdot y' = z'$.

What have we accomplished?

Starting with Maltsev's definition of a copy $F_{(u,v)}$ of F in $H(F)$, which used an arbitrary non-commuting pair (u, v) as parameters, we found uniform finitary existential formulas, with no parameters, that, for all fields F, define an interpretation of F in $H(F)$.

We used the fact that there are existential formulas defining:

1. the set of parameter pairs (u, v),
2. a nice family of isomorphisms $f_{(u,v)(u',v')}$.

General result

Theorem. Suppose there are existential formulas, with parameters $\bar{b}$, that effectively define a copy of $\mathcal{A}$ in $\mathcal{B}$.[2] Suppose the orbit of $\bar{b}$ is defined by an existential formula. For $\bar{c}$ in the orbit of $\bar{b}$, let $\mathcal{A}_{\bar{c}}$ be the copy of $\mathcal{A}$ obtained with parameters $\bar{c}$. Suppose that there is a formula $\psi(\bar{u}, \bar{v}, x, y)$ s.t. for all $\bar{c}, \bar{d}$ in the orbit of $\bar{b}$, $\psi(\bar{c}, \bar{d}, x, y)$ defines an isomorphism $f_{\bar{c},\bar{d}}$ from $\mathcal{A}_{\bar{c}}$ onto $\mathcal{A}_{\bar{d}}$. Finally, suppose the family of isomorphisms is functorial (preserving identity and composition). Then there is an interpretation of $\mathcal{A}$ in $\mathcal{B}$ defined by existential formulas with no parameters.

[2] If the language of $\mathcal{A}$ includes relation symbols R_i, we require existential formulas defining both R_i and $\neg R_i$.

Generalizing further

We may replace the given definition (with parameters) by an interpretation (with parameters). We may replace the existential formulas by computable Σ_1 formulas, or generalized computable Σ_1 formulas, or $L_{\omega_1\omega}$ formulas, or generalized $L_{\omega_1\omega}$ formulas. In each case, the complexity of the output formulas, with no parameters, matches that of the input formulas.

$SL_2(\mathbb{C})$

$SL_2(\mathbb{C})$ is the set of 2×2 matrices over $\mathbb{C}$ with determinant 1. We can define $\mathbb{C}$ in $SL_2(\mathbb{C})$ with parameters.

The theory of $SL_2(\mathbb{C})$ is ω-stable. Old results of Poizat yield (according to Pillay) an interpretation using elementary first order formulas without parameters. But, we don't know the complexity of the interpreting formulas.

Question. Is $\mathbb{C}$ interpreted in $SL_2(\mathbb{C})$ using existential formulas, with no parameters?

Bi-interpretability

We have (uniform) formulas that define $H(F)$ in F and interpret F in $H(F)$. Bi-interpretability asks more. We need definable isomorphisms from F to the copy of F interpreted in the natural copy of $H(F)$ defined in F, and from $H(F)$ to the copy of $H(F)$ defined in the copy of F interpreted in $H(F)$. Montalbán asked whether we have effective bi-interpretability. If we had bi-intepretability, then the automorphism groups of F and $H(F)$ would be isomorphic. For $\mathbb{Q}$, the automorphism group is rigid, while $H(F)$ is never rigid.

Question. Is there any field F such that F and $H(F)$ are bi-interpretable?