Using the Theory of Institutions for semiotic–argumentational treatment of mathematical proof

Georgios V. Pitsiladis
gpitsiladis@mail.ntua.gr

Petros S. Stefaneas

Department of Mathematics,
School of Applied Mathematical and Physical Science,
National Technical University of Athens, Greece

18 September 2025
International Conference on
Mathematical and Computational Linguistics for Proofs

Section 1

What are proving events?

Joseph Goguen on proof events

> *Mathematicians talk of "proofs" as real things. But the only things that can actually happen in the real world are proof events, or provings, which are actual experiences, each occurring at a particular time and place, and involving particular people, who have particular skills as members of an appropriate mathematical community.*
>
> *A proof event minimally involves a person having the relevant background and interest, and some mediating physical objects, such as spoken words, gestures, hand written formulae, 3D models, printed words, diagrams, or formulae [. . .]. None of these mediating signs can be a "proof" in itself, because it must be interpreted in order to come alive as a proof event; we will call them proof objects.*
> *[. . .]*
> *One simple step foward would be to allow alternative proofs that are incomplete, or even incorrect.*

Joseph A. Goguen. *What is a Proof?* July 2001. URL: https://cseweb.ucsd.edu/~goguen/papers/proof.html

A first formalisation of proof events

A *proof event* contains:

1. two agents, *prover* and *interpreter*,
2. an *intention* (insight or idea or proof sketch, or mathematical argument, etc.) linguistically articulated for
3. a (time-independent) *problem*
4. at *time t*,
5. with a *value* expressing
   the subjective conviction
   in the truth of the outcome.

Petros Stefaneas and Ioannis M. Vandoulakis. "On Mathematical Proving". In: *Journal of Artificial General Intelligence* 6.1 (2015), pp. 130–149. DOI: 10.1515/jagi-2015-0007

A first formalisation of proof events

A *proof event* contains:

1. two agents, *prover* and *interpreter*,
2. an *intention* (insight or idea or proof sketch, or mathematical argument, etc.) linguistically articulated for
3. a (time-independent) *problem*
4. at *time t*,
5. with a *value* expressing
   the subjective conviction
   in the truth of the outcome.

The prover states the proof attempt in a semiotic space.

Petros Stefaneas and Ioannis M. Vandoulakis. "On Mathematical Proving". In: *Journal of Artificial General Intelligence* 6.1 (2015), pp. 130–149. DOI: 10.1515/jagi-2015-0007

Joseph A. Goguen. "An Introduction to Algebraic Semiotics, with Application to User Interface Design". In: *Computation for Metaphors, Analogy, and Agents*. Ed. by Chrystopher L. Nehaniv. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1999, pp. 242–291. DOI: 10.1007/3-540-48834-0_15

A first formalisation of proof events

A *proof event* contains:

1. two agents, *prover* and *interpreter*,
2. an *intention* (insight or idea or proof sketch, or mathematical argument, etc.) linguistically articulated for
3. a (time-independent) *problem*
4. at *time t*,
5. with a *value* expressing
   the subjective conviction
   in the truth of the outcome.

The prover states the proof attempt in a semiotic space.

The interpreter interprets and validates the proof attempt
in another semiotic space.

For practicality, start by considering only one semiotic space.

Petros Stefaneas and Ioannis M. Vandoulakis. “On Mathematical Proving”. In: *Journal of Artificial General Intelligence* 6.1 (2015), pp. 130–149. DOI: 10.1515/jagi-2015-0007

A first formalisation of proof events: possible extensions

A *proof event* contains:

1. two agents, *prover* and *interpreter*,
2. an *intention* (insight or idea or proof sketch, or mathematical argument, etc.) linguistically articulated for
3. a (time-independent) *problem*
4. at *time t* (multiple timestamps may be relevant: presentation, validation, inclusion in some library),
5. with a *value* expressing
   the subjective conviction
   in the truth of the outcome.

A first formalisation of proof events: possible extensions

A *proof event* contains:

1. two (or more) agents, *prover* and *interpreter*,
2. an *intention* (insight or idea or proof sketch, or mathematical argument, etc.) linguistically articulated for
3. a (time-independent) *problem*
4. at *time t* (multiple timestamps may be relevant: presentation, validation, etc. per agent, inclusion in some library),
5. with a *value* expressing
   the subjective conviction of the community
   in the truth of the outcome.

Problems vs. proof (and disproof) attempts

We discern between two proving intentions:

- proof: proof object, proof event
- disproof: disproof object, disproof event

Observe that a (rationally reconstructed) proving event
(i.e. a proof or disproof attempt)
contains two main elements:

- (formal) problem,
- (possibly informal) proof/disproof.

Section 2

Defining problems

On institutions

Abstract model theoretic framework by J. Goguen and R. Burstall.

Definition

A (set–cat) institution $I = \left(\mathrm{Sig}_I, \mathrm{Sen}_I, \mathrm{Mod}_I, \models^I\right)$ consists of:

1. a category Sig_I of *signatures*,
2. a functor $\mathrm{Sen}_I : \mathrm{Sig}_I \longrightarrow \mathbb{S}\mathrm{et}$ of *sentences*,
3. a functor $\mathrm{Mod}_I : \mathrm{Sig}_I \longrightarrow \mathbb{C}\mathrm{at}^{\mathrm{op}}$ of *models*,
4. for each $\Sigma \in \mathrm{Sig}_I$, a *satisfaction* relation $\models^I_\Sigma \subseteq |\mathrm{Mod}_I(\Sigma)| \times \mathrm{Sen}_I(\Sigma)$,

such that for all $f : \Sigma \longrightarrow \Sigma' \in \mathrm{Sig}_I$, $M' \in |\mathrm{Mod}_I(\Sigma')|$, $e \in \mathrm{Sen}_I(\Sigma)$, the *satisfaction condition* $M' \models^I_{\Sigma'} \mathrm{Sen}_I(f)(e)$ iff $\mathrm{Mod}_I(f)(M') \models^I_\Sigma e$, i.e. that "truth is invariant under change of notation" is validated.

$$\begin{tikzcd} \mathrm{Sig}_I \arrow[r, "\mathrm{Sen}_I"] \arrow[dr, "\mathrm{Mod}_I"'] & \mathbb{S}\mathrm{et} \arrow[d, dotted, no head, "\models^I"] \\ & \mathbb{C}\mathrm{at}^{\mathrm{op}} \end{tikzcd}$$

$$\begin{tikzcd} \Sigma \arrow[d, "f"] \\ \Sigma' \end{tikzcd}$$

$$\begin{tikzcd} \mathrm{Mod}(\Sigma) \arrow[r, dashed, no head, "\models_\Sigma"] & \mathrm{Sen}(\Sigma) \arrow[d, "\mathrm{Sen}(f)"] \\ \mathrm{Mod}(\Sigma') \arrow[u, "\mathrm{Mod}(f)"] \arrow[r, dashed, no head, "\models_{\Sigma'}"] & \mathrm{Sen}(\Sigma') \end{tikzcd}$$

Răzvan Diaconescu. *Institution-independent Model Theory*. 2nd edition. Studies in Universal Logic. Birkhäuser Cham, 2025. DOI: 10.1007/978-3-031-68854-6

What is a (mathematical) problem?

Definition

A *problem* P consists of:

1. an institution $\mathtt{I}(P)$,
2. a signature $\mathtt{sig}(P) \in \mathrm{Sig}_I$,
3. a set $\Gamma(P) \subseteq \mathrm{Sen}_I(\Sigma)$ of *premises*,
4. a *claim* $\mathtt{claim}(P) \in \mathrm{Sen}_I(\Sigma)$.

or, intuitively, to state a problem:

1. pick a logical framework,
2. pick a logic of the framework,
3. state your premises,
4. state your claim.

What is a (mathematical) problem?

Definition

A *problem* P consists of:

1. an institution $\mathtt{I}(P)$,
2. a signature $\mathtt{sig}(P) \in \mathrm{Sig}_I$,
3. a set $\Gamma(P) \subseteq \mathrm{Sen}_I(\Sigma)$ of *premises*,
4. a *claim* $\mathtt{claim}(P) \in \mathrm{Sen}_I(\Sigma)$.

Example ($P^{\text{Peano}}_{\text{Frmt}}$)

1. **FOL** (the institution of first-order logic)
2. $(0, \mathsf{s}, +, n$-th power$)$
3. (first-order) Peano axioms & axioms for the operations
4. $\forall x, y, z, n \,.\, n > 2 \Rightarrow x^n + y^n \neq z^n$

“Correctness” of problem statements

Definition

A problem P *holds* (for brevity, $\mathrm{H}(P)$) iff

$$\Gamma(P) \text{ is consistent}$$

and

$$\Gamma(P) \models^{\mathtt{I}(P)}_{\mathtt{sig}(P)} \mathtt{claim}(P)$$

Spoiler #1

If we find a formal proof of a P with consistent premises, then $\mathrm{H}(P)$.

Spoiler #2

If we find a formal counterexample for P, then P does not hold.

Argumentational notions: support and attack

Definition

If H(P) implies H(Q), then Q is *supported* by P (notation $Q \preceq P$).
This is a preorder.
Denote the equivalence relation $Q \preceq P \ \wedge \ P \preceq Q$ as $P \approx Q$.

Definition

If H(P) implies not H(Q), then Q is *attacked* by P (notation $Q \pitchfork P$).
This is symmetric.

A differently constructed argumentational treatment (APEC) of some kinds of proof events can be found in
Sofia Almpani, Petros Stefaneas, and Ioannis Vandoulakis. “Formalization of Mathematical Proof Practice Through an Argumentation-Based Model”. In: *Global Philosophy* 33.3 (2023), p. 33. DOI: 10.1007/s10516-023-09685-z

Argumentational notions: support and attack

Definition

If H(P) implies H(Q), then Q is *supported* by P (notation $Q \preceq P$).
This is a preorder.
Denote the equivalence relation $Q \preceq P \ \wedge\ P \preceq Q$ as $P \approx Q$.

Definition

If H(P) implies not H(Q), then Q is *attacked* by P (notation $Q \pitchfork P$).
This is symmetric.

Remark

Attacking an attacker of P does not necessarily support P.
Supporting an attacker of P attacks P.

A differently constructed argumentational treatment (APEC) of some kinds of proof events can be found in
Sofia Almpani, Petros Stefaneas, and Ioannis Vandoulakis. "Formalization of Mathematical Proof Practice Through an Argumentation-Based Model". In: *Global Philosophy* 33.3 (2023), p. 33. DOI: 10.1007/s10516-023-09685-z

Simple support example

Recall:

Example ($P^{\text{Peano}}_{\text{Frmt}}$)

1. **FOL**
2. $(0, \mathrm{s}, +, n\text{-th power})$
3. (first-order) Peano axioms & axioms for the operations
4. $\forall x, y, z, n \;.\; n > 2 \Rightarrow x^n + y^n \neq z^n$

Simple support example

Example ($P^{\text{Peano}}_{\text{Frmt}}$)

1. **FOL**
2. $(0, \mathrm{s}, +, n\text{-th power})$
3. (first-order) Peano axioms & axioms for the operations
4. $\forall x, y, z, n \,.\, n > 2 \Rightarrow x^n + y^n \neq z^n$

Example ($P^{\mathbb{N}}_{\text{Frmt}}$)

1. **FOL**
2. $(0, \mathrm{s}, +, n\text{-th power})$
3. $\mathbb{N}^{\bullet}$ (i.e. all sentences satisfied in the natural numbers model)
4. $\forall x, y, z, n \,.\, n > 2 \Rightarrow x^n + y^n \neq z^n$

Remark

$P^{\mathbb{N}}_{Frmt} \preceq P^{Peano}_{Frmt}$

Support sub-relations

In fact, various special cases of support can be pinpointed, corresponding to usual claim-supporting strategies.

For example:

- If P and Q have same institution I, signature Σ, and claim:

Support sub-relations

In fact, various special cases of support can be pinpointed, corresponding to usual claim-supporting strategies.

For example:

- If P and Q have same institution I, signature Σ, and claim:
  - If $\Gamma(P) \subseteq \Gamma(Q)$
    and consistency of $\Gamma(P)$ implies consistency of $\Gamma(Q)$,
    then $Q \preceq P$.

Support sub-relations

In fact, various special cases of support can be pinpointed, corresponding to usual claim-supporting strategies.

For example:

- If P and Q have same institution I, signature Σ, and claim:
    - If $\Gamma(P) \subseteq \Gamma(Q)$
      and consistency of $\Gamma(P)$ implies consistency of $\Gamma(Q)$,
      then $Q \preceq P$.
    - If $\Gamma(Q) \subseteq \Gamma(P)$
      and $\Gamma(Q) \models^{I}_{\Sigma} \Gamma(P) \setminus \Gamma(Q)$,
      then $P \approx Q$.

Support sub-relations

In fact, various special cases of support can be pinpointed, corresponding to usual claim-supporting strategies.

For example:

- If P and Q have same institution I, signature Σ, and claim:
  - If $\Gamma(P) \subseteq \Gamma(Q)$
    and consistency of $\Gamma(P)$ implies consistency of $\Gamma(Q)$,
    then $Q \preceq P$.
  - If $\Gamma(Q) \subseteq \Gamma(P)$
    and $\Gamma(Q) \models^{I}_{\Sigma} \Gamma(P) \setminus \Gamma(Q)$,
    then $P \approx Q$.
- If P and Q have same institution I and signature Σ:

Support sub-relations

In fact, various special cases of support can be pinpointed, corresponding to usual claim-supporting strategies.

For example:

- If P and Q have same institution I, signature Σ, and claim:
  - If $\Gamma(P) \subseteq \Gamma(Q)$
    and consistency of $\Gamma(P)$ implies consistency of $\Gamma(Q)$,
    then $Q \preceq P$.
  - If $\Gamma(Q) \subseteq \Gamma(P)$
    and $\Gamma(Q) \models^{I}_{\Sigma} \Gamma(P) \setminus \Gamma(Q)$,
    then $P \approx Q$.
- If P and Q have same institution I and signature Σ:
  - If $\Gamma(P)$ are semantically equivalent to $\Gamma(Q)$
    and `claim`(P) is semantically equivalent to `claim`(Q),
    then $P \approx Q$.

Attack sub-relations

Similarly, special cases of attack can be pinpointed, corresponding to usual claim-attacking strategies.

For example:

- If P and Q have same institution I, signature Σ, and premises Γ:

Attack sub-relations

Similarly, special cases of attack can be pinpointed, corresponding to usual claim-attacking strategies.

For example:

- If P and Q have same institution I, signature Σ, and premises Γ:
  - If $\texttt{claim}(P)$ is a semantic negation of $\texttt{claim}(Q)$, then $P \pitchfork Q$.

Attack sub-relations

Similarly, special cases of attack can be pinpointed, corresponding to usual claim-attacking strategies.

For example:

- If P and Q have same institution I, signature Σ, and premises Γ:
  - If $\mathtt{claim}(P)$ is a semantic negation of $\mathtt{claim}(Q)$, then $P \pitchfork Q$.
- If P and Q have same institution I and signature Σ:

Attack sub-relations

Similarly, special cases of attack can be pinpointed, corresponding to usual claim-attacking strategies.

For example:

- If P and Q have same institution I, signature Σ, and premises Γ:
  - If $\texttt{claim}(P)$ is a semantic negation of $\texttt{claim}(Q)$, then $P \pitchfork Q$.
- If P and Q have same institution I and signature Σ:
  - If $\Gamma(P) \subseteq \Gamma(Q)$ and $\texttt{claim}(P)$ is a semantic negation of some $q \in \Gamma(Q)$, then $P \pitchfork Q$.

More advanced support sub-relations: signature expansion

Example (Signature expansion)

Consider a problem $P_1 = (I, \Sigma_1, \Gamma_1, c_1)$.
If we think that Σ_1 is not “adequate”
and given a morphism $f : \Sigma_1 \longrightarrow \Sigma_2$
such that every model of Γ_1 is a translation of a model of Σ_2 via f,
we can translate the problem into a $P_2 = (I, \Sigma_2, \Gamma_2, c_2)$
such that $P_1 \preceq P_2$.

TL;DR

This mechanism can relate statements (and thus proofs)
of the “same” problem in different logics of the same framework.

Example

- Additive to multiplicative notation in groups.
- Inclusion of a new symbol.

More advanced support sub-relations: logical expansion

Example (Logical expansion)

Consider a problem $P_1 = (\mathbf{EQL}, \Sigma_1, \Gamma_1, c_1)$ specified in equational logic.
If we think that **EQL** is not “rich” enough,
and given the correct signature Σ_2 in **FOL** corresponding to Σ_1,
we can translate the problem into a $P_2 = (\mathbf{FOL}, \Sigma_2, \Gamma_2, c_2)$
such that $P_1 \preceq P_2$.

This uses the notion of *institution morphisms*.
And the fact that there exists such a morphism **FOL** $\longrightarrow$ **EQL**.

Under some conditions, we may also move from “richer” to “simpler” logics.

TL;DR

This mechanism can relate statements (and thus proofs)
of the “same” problem in different logical frameworks.

For institution morphisms and comorphisms, see relevant chapter of
Răzvan Diaconescu. *Institution-independent Model Theory*. 2nd edition. Studies in Universal Logic. Birkhäuser Cham, 2025. DOI: 10.1007/978-3-031-68854-6

Generalised support and attack (a.k.a. sub-problems)

Definition

Let P be a problem and P_i be a family of problems.
P_i *jointly support* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.
P_i *jointly attack* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.

Generalised support and attack (a.k.a. sub-problems)

Definition

Let P be a problem and P_i be a family of problems.
P_i *jointly support* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.
P_i *jointly attack* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.

Example

(I, Σ, Γ, p) is jointly supported by

1. (I, Σ, Γ, p_1),
2. $(I, \Sigma, \Gamma \cup \{ p_1 \}, p_2)$,
3. $\ldots$,
4. $(I, \Sigma, \Gamma \cup \{ p_1, \ldots, p_{n-1} \}, p_n)$,
5. $(I, \Sigma, \Gamma \cup \{ p_1, \ldots, p_n \}, p)$,

for arbitrary $p_1, \ldots, p_n$.

Generalised support and attack (a.k.a. sub-problems)

Definition

Let P be a problem and P_i be a family of problems.
P_i *jointly support* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.
P_i *jointly attack* P iff, when $\mathrm{H}(P_i)$ for all i, this implies $\mathrm{H}(P)$.

Example

(I, Σ, Γ, p) is jointly supported by

1. $(I, \Sigma, \Gamma, \owedge S)$,
2. $(I, \Sigma, \Gamma \cup S, p)$,

where $\owedge S$ is a semantic conjunction of S
($\owedge S$ does not necessarily exist for every I and S).

Section 3

Defining proving objects

Proving objects

An open-ended hierarchy:

- Proof object
  - Plain string (or image, or . . .)
  - Formal proof
  - Other proof event(s)
  - . . .
- Disproof object
  - Plain string (or image, or . . .)
  - Formal counterexample
  - Other disproof event(s)
  - . . .

Proving objects

An open-ended hierarchy:

- Proof object
  - Plain string (or image, or . . .)
  - Formal proof
  - Other proof event(s)
  - . . . (various levels of non-formal proof of increasing formality)
  - . . . (formal proof with annotations)
  - . . .

- Disproof object
  - Plain string (or image, or . . .)
  - Formal counterexample
  - Other disproof event(s)
  - . . . (various levels of non-formal disproof of increasing formality)
  - . . . (formal disproof with annotations)
  - . . .

Formal proof objects

Definition

An *institution with proofs* is a tuple $\left(\mathrm{Sig}_I, \mathrm{Sen}_I, \mathrm{Mod}_I, \models^I, \mathrm{Prf}_I\right)$ such that

- $\left(\mathrm{Sig}_I, \mathrm{Sen}_I, \mathrm{Mod}_I, \models^I\right)$ is an institution,
- Prf_I is a functor (with some properties)
  that maps every $\Sigma \in \left|\mathrm{Sig}_I\right|$ to a category of proofs
  (sets of Σ-sentences as objects; proofs as morphisms).

An institution with proofs is *sound* when,
for each signature $\Sigma \in \mathrm{Sig}_I$ and morphism $E \longrightarrow E'$ in $\mathrm{Prf}_I(\Sigma)$,

$$E \models^I_\Sigma E'.$$

For full details of institutions with proofs, see relevant chapter of
Răzvan Diaconescu. *Institution-independent Model Theory*. 2nd edition. Studies in Universal Logic. Birkhäuser Cham, 2025. DOI: 10.1007/978-3-031-68854-6

Formal proof objects

Definition

An *institution with proofs* is a tuple $\left(\mathrm{Sig}_I, \mathrm{Sen}_I, \mathrm{Mod}_I, \models^I, \mathrm{Prf}_I\right)$ such that

- $\left(\mathrm{Sig}_I, \mathrm{Sen}_I, \mathrm{Mod}_I, \models^I\right)$ is an institution,
- Prf_I is a functor (with some properties) that maps every $\Sigma \in |\mathrm{Sig}_I|$ to a category of proofs (sets of Σ-sentences as objects; proofs as morphisms).

An institution with proofs is *sound* when,
for each signature $\Sigma \in \mathrm{Sig}_I$ and morphism $E \longrightarrow E'$ in $\mathrm{Prf}_I(\Sigma)$,

$$E \models^I_\Sigma E'.$$

Definition

A *formal proof object* consists of

- a functor Prf, making $\mathtt{I}(P)$ a sound institution with proofs,
- a morphism $\Gamma(P) \xrightarrow{\pi} \mathtt{claim}(P)$ in $\mathrm{Prf}(\mathtt{sig}(P))$.

Section 4

Families of proving events

Relations of proving events

Holding, support and attack can be lifted to proving events.

Definition

For every proof event e, define $\mathtt{H}(e)$ as $\mathtt{H}(\mathtt{problem}(e))$.
For every disproof event e, define $\mathtt{H}(e)$ as not $\mathtt{H}(\mathtt{problem}(e))$.
Joint support/attack are defined as expected.

Fluents

Definition

A *fluent* (at time t) is a hypergraph:

- Every node is a proof event (stated until t).
- Two kinds of edges:
  - Support hyperedges: encode that e is jointly supported by e_i,

    sources: e_i
    target: e
  - Attack hyperedges: similar for joint attack.

Fluents evolve over time
(recall that proving events have timestamps).

Sometimes, a fluent may be accepted as a proving object.

Section 5

Conclusion

Conclusion

Proving events, with institutional foundations,

- can serve for rational reconstruction of (some) mathematical proving,
- are (quite more) logic agnostic (than usual),
- discern between syntax and semantics,
- discern between problem and solution,
- can be extended to discern between presentation and interpretation,
- may allow for incomplete and annotated proofs,
- enable the emergence of argumentational notions.

Still in preliminary stage.

- All proof events of the same problem support each other. Is this OK?
- Different but related proofs are not linked as such.
- When can a fluent be accepted as a proving event?
- What about the valuation proving events?
  Bayesian probabilities have been proposed.
- Add support for multiple agents.
- Add support for more “exotic” stuff; e.g. multi-valued institutions.
- . . .

Thank you!