Theorem 1

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

for all T\_e at least one of the following properties is false (T\_e is consistent, T\_e is complete, T\_e is an extension of T, T\_e has the same constants as T, T\_e is recursively enumerably axiomatizable).

This implies

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

There exists T\_e (  
 T\_e is paraconsistent,

T\_e is complete

T\_e is an extension of T

T\_e has the same constants as T

T\_e is recursively enumerably axiomatizable

).

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

There exists T\_e (  
 T\_e is consistent,

T\_e is paracomplete

T\_e is an extension of T

T\_e has the same constants as T

T\_e is recursively enumerably axiomatizable

).

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

There exists T\_e (  
 T\_e is consistent,

T\_e is complete

T\_e is a non-conservative extension of T

T\_e has the same constants as T

T\_e is recursively enumerably axiomatizable

).

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

There exists T\_e (  
 T\_e is consistent,

T\_e is complete

T\_e is an extension of T

T\_e has the different constants than T

T\_e is recursively enumerably axiomatizable

).

For all theories T in standard formalization.

T is essentially undecidable if and only if T is consistent and:

There exists T\_e (  
 T\_e is consistent,

T\_e is complete

T\_e is an extension of T

T\_e has the same constants as T

T\_e is non-axiomatizable

).

Tarskian Decidability -> Recursive Axiomatizability

Einstein Univocality/Categoricity -> Completeness

Tarskian undecidability and completeness <-> non-axiomatizability;

Tarskian undecidability and univocality <-> non-axiomatizability.

Tarskian decidability and completeness <-> axiomatizability;

Tarskian decidability and completeness </> non-axiomatizability;

non-(Tarskian decidability) or non-completeness </> non-axiomatizability.

We care whether a theory or language is algorithmically decidable vs non-algorithmically decidable.

A decision procedure for T is an automated or mechanical procedure that takes some input and produces binary or bivalent output.

A decision algorithm for T is an automated or mechanical procedure that takes some input and produces binary or bivalent outputs with only finite resources such as space or time; there exists a decision algorithm for all Type 1 languages that depends only on the length-increasing property of recursive language grammars (monotonicity?).

A non-algorithmically decidable problem is a semi-decidable problem.

Every decidable language is recursively axiomatizable. Every algorithmically decidable language is finitely axiomatizable.

Not every axiomatizable language is decidable.

Not every finitely axiomatizable language is decidable.

There exists non-bivalent languages.

A non-bivalent procedure is a procedure that takes some input or inputs and produces non-binary or many-valued output or outputs.

A many-valued decision algorithm for T is an algorithm that halts with true, false, or at least one distinct other value such as possible, undefined, indeterminate, unknown, neither true nor false, both true and false.