

A Paper Presented at Society of Biblical Literature International Meeting, Tartu, Estonia, July 2010

Title: Biblical Paradox and Coinductive Reasoning

Author: Richard K. Min, and Gopal Gupta
The University of Texas at Dallas, Texas, USA

Abstract:

The study of paradox has been one of the most neglected areas in contemporary biblical scholarship. One of the landmark examples is the Liar's Paradox (Titus 1:12) with circular reasoning, along with the name of God (Exodus 3:14) and the identity of Jesus (John 14:10). Yet biblical paradox is one of the most ignored areas in biblical study for the latter half of the 20th century. The study of the paradox in formal logic and philosophy has been pioneered by Russell in the early 20th century, followed by Tarski in Mathematics and Wittgenstein in Philosophy. The scholarly consensus and trend since Tarski was to exclude circular reasoning from the formal logic and to negate it as valid reasoning, to avoid a paradox occurring. The consequence of this mainstream decision has been somewhat devastating, especially in biblical scholarship. As a result, there was no basis of formal reasoning or logic to support any literary or logical construct of circularity, found in the biblical text. However, there has been a renewed interest due to the innovative approach and breakthrough in the study of circularity and paradox pioneered by Kripke since 1975. This paper presents this new perspective and paradigm of coinductive reasoning and its application to biblical texts. The new perspective and paradigm brings a renewed interest and excitement toward the study of biblical paradox with its linguistic construct, and its modal and nonmonotonic reasoning.

Keywords: biblical paradox; coinductive reasoning; paradox of circularity; modal reasoning; nonmonotonic reasoning

Biblical Paradox and Coinductive Reasoning

Richard K. Min and Gopal Gupta

The University of Texas at Dallas, Texas, USA

A Paper Presented at Society of Biblical Literature International Meeting, Tartu, Estonia, July 2010

1. Introduction

The study of paradox has been one of the most neglected areas in contemporary biblical scholarship. However, it has been recognized as one of the most active and controversial areas in the field of Philosophy, Mathematical Logic, Computer Science, Linguistics, and Literary Study. One of the landmark examples is the Liar's Paradox (Titus 1:12) with circular reasoning. Yet biblical paradox is one of the most ignored areas in biblical study for the latter half of the 20th century. The study of paradox in formal logic and philosophy was pioneered by Russell in the early 20th century, followed by Tarski in Mathematics and Wittgenstein in Philosophy. The scholarly consensus and trend set by Tarski was to exclude circular reasoning from the formal logic, to treat it as invalid (nonsense). The goal was to avoid a paradox of circularity occurring, in the system of formal logic and reasoning. Its intention was pragmatic, to keep the system of formal logic manageable at that time.

The scholarly trend since Tarski has set not only the direction and mainstream of formal logic and reasoning but it was soon to be followed by biblical scholarship as well. However, the consequence of this mainstream decision has been somewhat devastating, especially in biblical scholarship. One of the most damaging and devastating impacts to biblical scholarship has been a prevalent trend and consensus that there is no basis in formal reasoning or logic to support circular reasoning, including a literary or logical construct of circularity in the Bible. As a result, biblical paradox has been one of the most confused, ignored, neglected, or misunderstood areas in biblical and theological scholarship in the latter half of the 20th century. However, there has been a renewed interest due to the innovative approach and breakthrough in the study of circularity and paradox pioneered by Kripke (1975), followed by Fitting (1985). The new paradigm has been recognized as one of the most active and controversial areas of Philosophy, Mathematical Logic, and Computer Science, to name a few. This new paradigm is the primary method in this paper.

In this paper, we present this new perspective and paradigm dealing with circularity in reasoning and logic toward coinductive reasoning and its application to biblical texts. We present and examine some of these well-known examples of biblical paradox of circularity. Their linguistic constructs, literary and rhetoric features, and discourse analysis are presented and examined. In doing so, the modal and nonmonotonic characteristics, imbedded in these paradoxes of circularity (e.g., Matthew 22:15-46), are analyzed and elaborated. Among many paradoxes found in the literature, the most well-known paradox is the Liar's Paradox in the Bible (Titus 1:12). The Liar's Paradox has been identified with circular reasoning (circularity) which is the primary scope of this paper. There are abundant examples of paradox of circularity throughout the Bible. Some of these landmark examples are: (1) the Liar's Paradox, (2) the name (the meaning of the name) of God in Exodus 3:14, (3) the Father-Son relationship with the identity of Jesus in John 14:10, and (4) the proof-method and testimony by Jesus in John 8:12-18.

The definition for the term "paradox" in biblical scholarship has been unsettled and controversial, far from any scholarly consensus. Current working definition of biblical paradox is "contrary to opinion" in this paper. This definition is somewhat flexible and pragmatic to serve our purpose and scope of this paper, conservatively following the meaning of the early Greek and *Koine* tradition of the New Testament. This definition has been commonly used in English until the 18th century. Other meanings or similar notions for paradox found in contemporary discussions (for example, mystery, apparent or actual self-contradiction, etc.) are also to be examined as we elaborate some of the biblical paradoxes.

Our current approach is distinctively and deliberately computational. The approach and its method have been implemented as a programming language. It is termed as "coinductive logic programming" ("co-LP" for brevity), atop of a logic programming language. As noted earlier, the scope of this paper is limited to one class of biblical paradox dealing with circularity and its solution with coinductive reasoning. Our current approach does not solve all the problems, nor does it address all the issues in the area of the biblical paradox. However, the new paradigm is very promising, to understand many confusions, failures, misunderstandings, limitations, or even ignorance among the various scholarly attempts and trends with respect to the biblical paradox in the past. Further it is the authors' hope through this study to bring a renewed interest, understanding, and excitement toward the study of biblical paradox in the dawn of the 21st century.

2. Inductive versus Coinductive Reasoning

Induction is a familiar term along with inductive reasoning, inductive logic, or even inductive bible study using the logic of induction. Induction corresponds to well-founded structures from which a basis serves as the foundation for building more complex structures. For example, natural numbers are inductively defined via the base element (zero) and the successor function (by adding one). Thus one may construct any natural number from zero by adding one repeatedly. For example, three is constructed from zero by adding one three times. Similarly one may check whether a given number is a natural number, inductively by undoing the addition of one (that is, by subtracting one) repeatedly to reach its base of zero, in a finite number of steps. An object constructed in inductive definition is called a “well-founded” object for there is a well-founded base, and a set of such objects is called a well-founded set. Hence the set of natural numbers constructed by induction is a well-founded set. And it does not include any infinite number (because the infinitely many iterations of adding one onto zero will never be terminated in finite steps). Conversely one cannot check an infinite (natural) number inductively for it takes an infinite time to reach the base (by subtracting one repeatedly). For this reason, an inductively-defined set of natural numbers is minimal (smallest) of all sets of natural numbers (including arbitrarily large but still finite). Further one can view any finite number with respect to “the number of steps of adding one repeatedly” to its base of zero (that is, as a list of “adding ones for the number of steps”). Thus minimality implies that any infinite numbers (that is, infinite-length lists of numbers) are not members of the inductively defined set of lists of numbers. Inductive definitions correspond to “least fixed point interpretation” of so-called “recursive” definitions. In summary, inductive definitions have three components: initiality, iteration, and minimality.

In contrast, coinduction eliminates the initiality condition and replaces the minimality condition with maximality. No requirement for initiality means that there is no need for a base-case in coinductive definitions. Coinductive definitions have two components: iteration and maximality. Any object constructed in coinductive definition is called a “not well founded” object because there is no base. Further iteration of coinductive definition (without a base) is achieved by circular construct. Thus, while these examples and definition may appear to be circular (or meaningless, as it seems to be), the definition is well formed since coinduction corresponds to “greatest fixed point interpretation” of recursive definitions. The resulting formal system of reasoning (logic) is termed as “coinduction” (in coinductive reasoning or logic), in contrast to the traditional “induction” (in inductive reasoning or logic).

For a note, we use the terms of “induction” and “coinduction” as proof method, the terms of “recursion” and “co-recursion” as definition (or mapping), and the terms of “least fixed point interpretation” and “greatest fixed point interpretation” as formal meaning (semantics). The detailed and formal account is found in Barwise and Moss (1996), and Aczel (1977; 1988). Coinduction has been used in many areas including computer science, mathematics, philosophy and linguistics, as noted in Barwise and Moss (1996). Some of the exemplary applications of coinduction are bisimulation, bisimilarity proof and concurrency (Park 1981; Milner 1989), process algebras (Milner 1980) such as π -calculus (Milner 1999), programming language semantics (Milner and Tofte, 1991), model checking (Clarke, Grumberg, and Doron 1999), situation calculus (Reiter 2001), description logic (Baader et al. 2003), and game theory and modal logic (Barwise and Moss, 1996). The extension of logic programming with coinduction allows for both recursion and co-recursion (Simon et al. 2006).

One misleading view on minimality in induction is a tendency for one and only one best model or interpretation (if one exists). The reflection of this misconception in biblical scholarship is the pervasive and persistent tendency toward one and only one (best) interpretation in the contemporary biblical exegesis and interpretation. However, the minimality requirement for induction does not warrant one and only one best model or interpretation but there could be many optimal interpretations (as a minimal set), as long as each interpretation is not implied by the other. Further, if one allows the “possible world” semantics (of modal reasoning) in coinduction, then it is possible to justify an array of possible interpretations in exegesis (where one interpretation could be even in conflict with another interpretation). The scholarly tradition of inductive reasoning and its opposition against modal reasoning could be traced back to Kant (1781), and the omission of modality by Frege (1879) in his pioneering groundwork of modern logic for propositional and higher-order logic.

In the Bible, we find a noteworthy example in John 8:12-18. Here Jesus uses both inductive and coinductive reasoning to defend the validity of his testimony. Accused by the Jews, Jesus uses a self-reference (a circular reasoning) to validate his own testimony (John 8:12 which is one of the “I am” sayings in John and a self-claim), and the objection by the Pharisees (John 8:13). In his defense, Jesus uses both coinductive reasoning (John 8:14)

and then inductive reasoning (John 8:17). First (1), using coinductive reasoning (John 8:14) Jesus says that his testimony is valid even if he testifies on his own behalf (as he is using a circular or coinductive reasoning of self-referencing). Moreover Jesus is providing a justification for himself to use the self-referencing (a coinductive reasoning) due to his own (supernatural and omniscient) knowledge about himself to know where he came from and where he is going. In contrast to his own defense and qualification, Jesus qualifies further that they (the accusers) are not qualified due to the fact that they do not know where Jesus came from or where Jesus is going. This claim further reveals that Jesus knows not only their inner thoughts (John 2:24-25) but also their origin and destination (John 8:44). Second (2), using inductive reasoning (John 8:17) Jesus provides two witnesses (the Father and the Son himself) for the requirement imposed by the Law (Deuteronomy 19:15). Interestingly the accused (Jesus) himself is also qualified as a witness to defend himself. The accusers take at least the part of Jesus but then seek the claimed second witness (the father of Jesus) to be in a witness stance (John 8:19). Later we find that there is at least one more witness (John 9:29-33) to stand up in the witness stance. He who was born blind but healed by Jesus boldly comes forward to testify for Jesus and where Jesus came from.

Note that circular reasoning (coinductive reasoning) can be used as a sound method of reasoning or logic (just as induction as a sound method of proof). However, if an assumption or material (testimony or witness) of the argument (proof or reasoning) is invalid (whether it is inductive or coinductive), then the whole argument and thus the proof itself is invalid even though the proof method itself (whether it is inductive or coinductive) is correctly applied. For example, if a stranger says to me, "Trust me with all of your money and your life". I should tell and challenge him to prove it. That is, an unwarranted plain response (of "simply because I say so" from a stranger) is not good enough to be a credible proof for any cautious human being, for the entrustment of his life and all of his fortune.

A similar provision in biblical law is explicitly stated and mandated for safeguarding against false testimony (Exodus 20:16), and for the proof method in the authentication process of prophet (Deuteronomy 18:21-22). One may find many biblical examples for the "challenge-response" model or "identification-authentication" model of security, for example, for the identity and proof of Christ (John 1:19-27, 6:30) and toward the secure model of revelation. Thus the secure system of biblical reasoning and revelation warrants the challenge-response model, using the "sign" as one of the most prominent proof methods in the Bible (Deuteronomy 18:19-22, Isaiah 7:10-17, and John 20:30-31). This elevates the necessity and interest toward the biblical concept of "sign" as a proof method in the "identification and authentication" process (cf. John 2:11). One may find the stages or the process of the faith (for example, of Peter in John) in formation, growth, and maturity through (1) an indirect but a credible personal testimony of one's teacher and prophet (John 1:35-42), (2) a direct and personal self-experience of the unshakable "sign" as a proof (John 2:11), (3) a doubt and controversy (John 6:60-71), (4) a confirmation of the faith (John 16:29-31), (5) the ultimate shake-up test (John 13:36-38 and 16:32-33 for John 18:25-27), and (6) the commencement (John 21:15-18). One may extend the contemporary view and scope of paradox beyond the literary genre of discourse and rhetoric into the realm of action. Then one may view the miraculous signs in the Bible under the category of paradox in action, as paradox is in either word or deed (as also noted in Luke 5:26). Moreover the secure system is also found in the case of the biblical dream-vision and its interpretation. For example, a dream is used as a secure means of transmission of a hidden message to a particular person who may or may not know its hidden message at the time of the reception. It is to be interpreted, only by a qualified or authenticated secure interpreter to reveal its hidden message securely, thus effectively protecting its trust system (cf. 2 Peter 3:16). Some of the classical and well-known examples of the secure dream-vision model are found in Genesis 40-41, Daniel 2 and 4, Matthew 1:20-25 and Matthew 2:19-21.

The secure system of the biblical message and communication is also found and applicable to biblical paradox (1) as a means to bring out a hidden message and divine wisdom, sealed in or through the paradox, (2) for the identification and authentication of the wise messenger sent by God. The paradoxical examples in Matthew 22:15-46 clearly demonstrate the model of the secure biblical message and communication. Whether a paradox is used by those who are not aware of its hidden message (that is, its solution to a question and query of the paradox) and the one who knows both (that is, a paradox as a question and its answer), begging for its hidden (theological) message to be disclosed, explained, revealed, and thus its paradoxical quest to be completed (as a unit of a discourse or a narrative). Similar to the biblical dreams or parables, one may find the biblical model of the paradox (1) applied as the secure message and communication, (2) with a motif of wisdom and paradigm-shift and (3) as an offensive and shock-wave rhetoric device, provoking crisis and conflict, and intellectual dilemma to stir up the audience, (4) with a mind-boggling and controversial question (seemingly so naïve at a first sight, but so profound theologically in

reality), (5) with an impending suspension and thrill followed by a breath-taking silence of the audience waiting for a moment of victory or defeat (for a glory or a shame), (6) to reveal a hidden divine wisdom through a seemingly so effortless resolution and novel answer for the paradox as theological challenge and quest, (7) to identify and authenticate the divine wisdom teacher (sage), and (8) for praise to the wisdom and authority of God, with wonder and amazement.

As a discourse model and means of rhetoric, biblical paradox generates a series of life-and-death crisis and conflict, breath-taking suspension and thrill, unexpected resolution and enlightening excitement, and out-bursting joy and praising finale through stimulation and unrest among the intellectuals (cf. Proverbs 30:4, John 3:3-10). Further one may find the discourse model of “question-answer” in all three paradoxical examples in Matthew 22:15-46, concluded by the overwhelming response to reveal and demonstrate the divine wisdom and biblical authority through a divine messenger to reveal the hidden message. In this regard, one may find a unifying motif and theme of the biblical revelation as the “secret” and “mystery”, hidden (even before the creation of the world) then but now revealed and known (Romans 16:25-26). This motif is not only inherent in the distinctive genre of parable, dream-vision, and paradox, but also clear and abundant in the prophetic writings (for example, Psalm 110:1 with Matthew 22:41-46). Some of the well-known and landmark (paradoxical) examples of “mystery” (hidden but now revealed and known in the New Testament) include: (1) the mystery of God in Christ (Colossians 2:2), (2) the mystery of the gospel of Jesus Christ (Romans 16:25-27), (3) the mystery of God’s will, set before the creation (Ephesians 1:9), and (4) the mystery of the corporate unity and relationship of Christ and the church (Ephesians 5:29-32).

3. Coinductive Logic Programming (Co-LP)

Coinductive logic has been used in many contemporary systems which require 24x7 operations (an infinite process designed to run forever). Some of these exemplar systems include: web server, wireless mobile telecommunication system, 24x7 life-support system in hospital, GPS satellite systems for navigation, computer operating systems, network systems, and so on. These systems are designed to run forever (without an end) once they are up and running. The process of these systems is characterized as an infinite loop (circularity or cycle). For example, the process of the wireless telecommunication system (server) is to wait for an incoming call request (by a client wireless caller), to service the call request (connection and then communication), to terminate the call, and then to go back to the wait state for the next call. If you have taken an introductory computer programming course, you may remember vividly how severely you are warned against a devastating mistake of making an infinite loop in your program. An infinite loop is taught to be a pain (*enigma*) and a curse (*anathema*) in programming. However, one would be shocked later to discover infinite loop as a valid and useful control structure in an advanced graduate programming course (such as operating system, web server, or network programming) for “24 by 7” systems designed to run forever once they are up and running. Next we present a brief informal introduction to coinductive logic programming, to implement coinductive logic as a programming language. The reader is referred to Lloyd (1987) for logic programming, Simon *et al* (2006) for co-LP, Min and Gupta (2009) for co-LP with negation, for a detailed account.

Logic (or logic-based) programming language was first proposed by McCarthy (1959). The proposal was to use logic to represent declarative knowledge to be processed by an automated theorem prover. The major breakthrough and progress in automated theorem proving for first order logic is marked by *resolution principle* by Robinson (1965) along with its subsequent developments. In 1972, the fundamental idea that “logic can be used as a programming language” is conceived by Kowalski and Colmerauer (Lloyd, 1987), with the proposal of logic programming language (Kowalski, 1974) and the implementation of Prolog (PROgramming in LOGic) interpreter by Colmerauer and his students in 1972 (Colmerauer and Roussel, 1996). We term this type of logic programming as traditional (or “inductive”) *logic programming* (LP) for its basis of “induction” as a proof method, in contrast to *coinductive logic programming* (co-LP) due to its extension with “coinduction”, following the terminology of Simon (2006).

A logic program consists of rules (called *Horn’s clauses*) of the form:

$$A_0 \leftarrow B_1, \dots, B_m \text{ not } B_{m+1}, \dots, \text{not } B_n. \quad (4.1)$$

for some $n \geq 0$ where A_0, B_1, \dots, B_n are called *atoms* (or *predicates*), A_0 is called the *head* (or *conclusion*) of rule (4.1), and B_1, \dots, B_n is called the *body* (or *premises*) of rule (4.1). The body is the conjunction of atoms B_1, \dots, B_n where each atom is separated by a comma corresponding to the logical operator “ \wedge ” (that is, “and”). Each clause is terminated by a period and the connective “ \leftarrow ” between the head and the body is corresponding to the logical operator of implication (that is, “if”). For a convenience of our notation, we also use “ $:-$ ” with “ \leftarrow ” interchangeably.

A rule without a body is called a *fact* (or a *unit clause*) (for example, { A_0 }). One may view a program as a theory (of a formal system) with a semantics assigning a collection of sets (or models where each model is a valid interpretation, and referred to as a solution of the program) to a theory.

For example, consider a parent relationship, $\text{parent}(X, Y)$ where it means “X is a parent of Y”. For example, the parent (“begot”) relations of Genesis 5:3-9 can be represented as the following facts.

<pre>parent(adam, seth). parent(seth, enosh). parent(enoth, kenan). parent(kenan, mahalalel).</pre>

Table 1. parent program in Genesis 5:3-9

Next, let’s consider an ancestor relationship, $\text{ancestor}(X,Y)$, where X is an ancestor of Y. Then the ancestor relationship can be defined as two rules: (1) a parent X of Y is an ancestor X of Y, and (2) an ancestor X of Y is a parent of X of Z where Z is an ancestor of Y, as follows.

<pre>ancestor(X, Y) :- parent(X,Y). ancestor(X, Y) :- parent(X, Z), ancestor(Z, Y).</pre>

Table 2. ancestor program

The “ancestor” relations are inductive, not coinductive (circular). If a circular definition of “ancestor” is allowed, then it would be invalid (nonsense). The query to this knowledgebase (combining parent program and ancestor program) is expressed as “:- p” where p is a predicate (or a sequence of predicates such as “:- p, q, r, ...”). For example, a query of “:- parent(adam, seth)” will be true as there is a fact, “parent(adam, seth)” found in Table 1. Another query of “:- parent(adam, cain)” would be resulted in a failure (to be false) for there is no fact or rule to satisfy the current query in the current knowledgebase of the “parent and ancestor” program in Table 1 and Table 2. This says something about the knowledgebase (of the facts and the rules) with respect to a “closed-world” semantics (in contrast to a “possible-world” semantics to allow “the possibility of something that one does not know”). The query of “parent(adam, cain)” results in a failure (that is, false). This is equivalent to say that a negation of “parent(adam, cain)” is true, with respect to the current knowledgebase based on Genesis 5:3-9. One may note that this type of negation is “negation by failure”. However, “parent(adam, cain)” is true if one may expand current knowledgebase to those facts in Genesis 4. In fact, one may add many more parent-facts (from Genesis 4): “parent(adam, cain)”, “parent(adam, abel)”, “parent(eve, cain)”, “parent(eve, abel)”, and so on. Next, consider the query of “parent(adam, X)” where X is a variable to be instantiated with a constant if there is any matching fact to accommodate this query. This query will be true as there is a fact of “parent(adam, seth)” where X is instantiated by “seth”. Further the query of “parent(eve, seth)” will be failed (“false”, that is, “not true”) because the current knowledge of the parent program does not warrant this fact. From this, one should notice that the known world of the parent program is restricted by the “closed-world assumption” (CWA). That is, “the things that I do not know” means “the truth value of these things are false”.

Further the query of “ancestor(adam, enosh)” is true for “ancestor(adam, enosh)” will be replaced by the second rule of ancestor, “ancestor(X,Y) :- parent(X, Z), ancestor(Z,Y)” with X for “adam”, Y for “enosh”, and Z for “seth”, resulting in the next goal of “ancestor(seth, Y)” to be resolved. Next, “ancestor(seth, Y)” will be resolved by the first rule of ancestor program, “ancestor(X, Y) :- parent(X,Y)” with the current goal of “ancestor(seth, Y)” where current X of the head will be replaced by “seth” and current Y will be replaced by “enosh”, resulting in a successful inference to be well-founded.

Extending the traditional LP, one may declare a coinductive predicate. For example, consider the following program with a coinductive predicate “i_am” with the following coinductive definition.

<pre>i_am :- i_am.</pre>

Table 3. i_am program with a coinductive predicate “i_am”.

This program seems to capture and represent the circular construct and meaning of “I am who I am” in Exodus 3:14, even though the predicate “i_am” is somewhat simplified and seems in need of further refinement. Note that “i_am” is a coinductive predicate; otherwise, the computational evaluation (execution) of “i_am” results in an infinite loop. Moreover, this program can be refined with a coinductive predicate “who(X)” to represent “who is X” where “X” is “i_am” as follow.

```
i_am :- who(i_am).
who(i_am) :- i_am.
```

Table 4. i_am 1 program: the revised i_am program with two coinductive predicates for “i_am” who “i_am”.

Moreover, the predicate “i_am” is propositional, to be further refined by making the coinductive predicate “i_am” with “i_am(X)” to be predicated, that is, in First-order logic, as follows.

```
i_am(X) :- who(i_am(X)).
who(i_am(X)) :- i_am(X).
```

Table 5. i_am 2 program: another revised coinductive predicates for “i_am” who “i_am”.

Thus one may further qualify the variable X to be a certain person (for example, “Jesus Christ”) for an instance of X in “i_am(X)”. Next we consider the “in” relationship of the son and the father, in John 14:10, as follow.

```
in(the_son, the_father).
in(the_father, the_son).
in(X, Y) :- in(Y, X).
```

Table 4. in(X,Y) program with a coinductive Predicate “in(X, Y)”

With negation “not”, one may express the friend and enemy relations with the following coinductive definitions.

```
friend(X, Y) :- not enemy(X, Y).
enemy(X, Y) :- not friend(X, Y).
```

Table 5. friend-enemy program, with two coinductive predicates friend(X,Y) and enemy(X,Y).

The coinductive predicate “friend(X, Y)” means that X is a friend of Y, and enemy(X, Y) means that X is an enemy of Y. One may extend the friend-enemy program by adding a few facts, as follows.

```
friend(X, Y) :- not enemy(X, Y).
enemy(X, Y) :- not friend(X, Y).
friend(ester, mordecai).
enemy(ester, haman).
friend(ester, xerxes).
```

Table 6. friend-enemy program with a few facts.

Three facts of friend or enemy are added to the program. Based on the fact of “friend(ester, mordecai)” alone, it is known (in this theory) that “ester” is a friend of “mordecai”. However, whether “mordecai” is a friend of “ester” is not known for there is no known fact or rule of inference to deduce this fact. Thus one may add two more rules of friendship so that friendship is commutative and transitive, as follows.

```
friend(X, Y) :- not enemy(X, Y).
enemy(X, Y) :- not friend(X, Y).
friend(X,Y) :- friend(Y,X).
friend(X,Y) :- friend(X, Z), Y is not = Z, friend(Z, Y).
friend(ester, mordecai).
enemy(ester, haman).
friend(ester, xerxes).
```

Table 7. friend-enemy program with a few facts.

From this program (theory), one may deduce that “ester” is a friend of “mordecai” and vice versa, based on the rule of “friend(X,Y) :- friend(Y,X)”, and that (1) “ester” is a friend of “mordecai”, (2) “mordecai” is a friend of “xerxes”, and thus (3) “ester” is a friend of “xerxes” based on (1) and (2) with the transitive rule of “friend(X,Y) :- friend(X,Z), Y is not = Z, friend(Z, Y)”. Moreover, one may assert (1) that “ester” is an enemy of “haman”, (2) that “ester” is not a friend of “haman”, and so on. One may try the “love-hate” relationship similar to “friend-enemy” relationship.

Next we present the following four examples to illustrate some of the important features and characteristics of co-LP.

Example 1. Consider the first program NP1 with one coinductive predicate of “p”.

NP1: p :- p.

Each query of “p” and “not p” succeed with the program NP1. The program NP1 has two models. One model holds for “p” and the other model holds for “not p”. That is, “p” is true in the first model whereas nothing is true in the second model (that is, “not p” is true). This type of the behavior seems to be confusing (seemingly contradictory) and thus counter-intuitive. However, as noted earlier, this type of behavior is indeed advantageous as we extend traditional LP into the realm of modal reasoning. Clearly, the addition of a circular clause (e.g., “p :- p.” as in the program NP1) to a program extends each of its initial models into two models where one includes “p” and the other does not include “p” (that is, “not p”). Further, co-LP enforces the consistency of the query result causing the query of “p and not p” to fail. However, the query of “p or not p” will succeed (in fact, there are two models: one for “p” and the other for “not p”).

Example 2. Consider the second program NP2 with two coinductive predicates “p” and “q”.

NP2: p :- not q.
 q :- not p.

Each query of “p” and “q” succeeds with the program NP2. In fact, the program NP2 has two models where one model holds for “p” and “not q”, and the other model holds for “not p” and “q”. Further they are mutually exclusive to each other. That is, they are not consistent with each other. Thus the query “p” is true with the first model of “p” and false with the second model of “q” while the query “not p” is true with the second model of “q” but not true with the first model of “p”. The results of these two queries are “seemingly” contradictory, if conjoined carelessly. Thus one should be careful not to join these results as one query holds with one model but not with the other model. Computing with the “possible world” semantics in presence of negation can be troublesome and seemingly lead to contradictions, similar to the first program NP1. Moreover, the query of “p and not p” will never succeed if we are aware of the context (of a particular model) because there is no model that holds for both “p” and “not p”.

Example 3. Consider the following program NP3 with one coinductive predicate “p”.

NP3: p :- not p.

The program NP3 has no model, in contrast to the program NP1 which has two models. Further, the query of “p or not p” provides a validation test for NP3 with respect to the predicate “p” whether the program NP3 is consistent or not.

Example 4. Consider the following program NP4 with the two coinductive predicates “p” and “q”.

NP4: p :- not q.

The program NP4 has a model of “p” whereas the program NP1 has two models of “p” and “q” as we noted earlier. Further the program NP4 has a definition of “p” (that is, p is defined as the left-hand of the rule with its body) but not for “q”, whereas the program NP2 has the definition for “p” and “q”. The query of “p” succeeds with the program NP2 and NP4 whereas the query of “q” succeeds with the program NP1 but not with the program NP4. To put it simply, we have some facts and rules defined for “p” and “q” in the program NP2. But only “p” is defined in the program NP4. That is, there is nothing known about “q” in NP4, except through “p”. In a “closed-world”

semantics, “q” is “not true (that is, false)” with the program NP4 for “q” is not defined (not known). This is an example of “negation as failure” with “closed-world assumption” (CWA) as noted earlier.

Logic programming (LP) language has two types of semantics: declarative semantics and operational semantics. The declarative semantics gives the denotational or mathematical description (meaning) of the objects of a program expressed with the syntax, that is, a formal specification of what it is meant for what is being expressed. The operational semantics expresses the formal description (meaning) of how to compute the objects of a program expressed with the syntax, that is, to define the meaning by a formal specification of how to compute what is being expressed. Ideally the (intended) meaning of a program by the declarative semantics should be equal to the (computed or extended) meaning of the program by the operational semantics. Given the model of declarative semantics, the model of operational semantics that holds is called “soundness”. Given the model of operational semantics, the model of declarative semantics that also holds is called “completeness”. “Correctness” therefore consists of soundness and completeness. The reader is referred to Lloyd (1987) for inductive LP, Simon *et al* (2006) for co-LP, and Min and Gupta (2009) for co-LP with negation. To summarize, co-LP is a recent and first attempt to implement coinductive reasoning as a formal programming language in LP paradigm. Co-LP is the first attempt to implement coinductive logic. By no means can the current state of coinductive reasoning (and co-LP) handle or solve all the problems and challenges in the paradox of circularity and coinductive reasoning. With this summary on coinductive reasoning and logic, the selected landmark examples in the Bible of the paradox of the circularity are presented and analyzed next.

4. Selected Biblical Examples and Analysis

A few illustrating and motivating paradoxical examples of circularity from the Bible are presented and examined for its conceptual construction.

(1) The first example is the text (“I am who I am”) from Exodus 3:14, as shown in Figure 1.

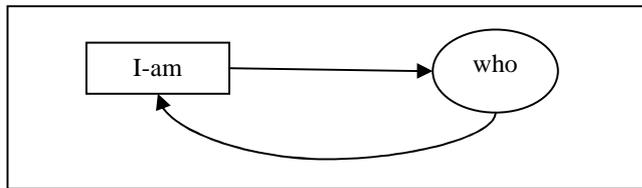


Figure 1. “I am who I am.” Exodus 3:14

In this example, a simple lexical (linguistic) or conceptual diagram clearly reveals its circular construct. A cycle (loop) is formed to reference itself (self-referencing). Its exact semantic meaning or interpretation is still debatable. However, one can easily see its circular meaning from the lexical or conceptual construct such as “I am who I am who I am who ...”. Further one may suggest further its (logical or theological) meaning as the one who is self-defining, self-referencing or self-revealing (that is, God who has no beginning and no end).

One may find many similar examples in the Bible, such as in Proverbs 1:22 saying that the naïve (foolish) ones love naïve (foolish) ways and the mockers love mockery. Here the passage defines or characterizes a person with the very character or description that it tries to describe. Similarly it is also noted that a dictionary definition of a concept may be defined by the very concept (through a circular definition) where one concept is defined by the second concept which is indeed defined by the first concept, and so on. One of the comparable examples outside of the Bible is the definition of “male with only male-offspring” found in Description Logic (Baader et al. 2003). The first definition without circularity is “Mos” where “Mos” is defined as “a male who has offspring where any of his offspring is male”. The second definition with circularity is “Momo” where “Momo” is defined as “a male who has offspring where any of his offspring is Momo”. One may argue that this type of the characterization (or a dictionary definition) adds nothing (no meaning), thus, is meaningless (or not inductive).

(2) The second example is the text from John 14:10-11. This illustrates again a circular construct. Here Jesus (the son) says: “I am in the Father and the father is in me.” (John 14:10) and “The father is in me and I in him.” (John 14:11, as shown in Figure 2.

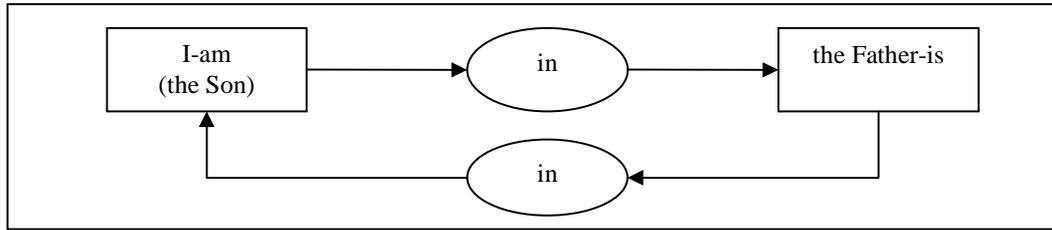


Figure 2. "I am in the Father and the Father is in me" in John 14:10

In contrast with the first example of "I am who I am", this construct is again circular with two concepts (persons) referencing each other (dual-referencing) with a preposition ("in") of indwelling-relationship, creating a cycle. One may view this indwelling or "be-in" relationship as a "part-whole" or "element-set" relationship. Next we examine a few more similar examples in the Bible whose circular constructs are not so obvious at lexical level.

(3) Third example is the well-known the Liar's Paradox in Titus 1:12 ("One of themselves, even a prophet of their own, said, the Cretans are always liars, ...") as shown in Figure 3.

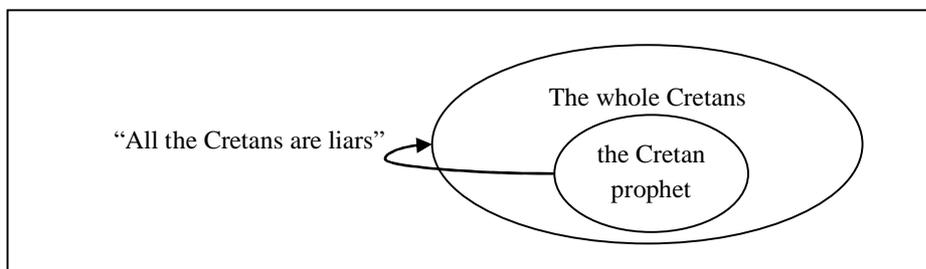


Figure 3. The Liar's Paradox in Titus 1:12

The problem is that the prophet himself is a Cretan, a member of the community of the whole Cretans (of the liars whose statement is false) to whom he is referring in his prophecy (which is true). Thus the prophet is referring to himself (creating a paradox of circularity) by referring to the group of which he is a member. A paradoxical question is whether the prophet is a liar and thus whether his prophetic assertion (which is true) is a lie (false). This is a classical example of circular relationship dealing with a set and its member (that is, an element in the set) of "set-membership". Outside of the biblical context, there are many examples similar to the Liar's Paradox. One famous and classical example is "Russell's paradox (barber's paradox)". Here a barber claims that he shaves (or hair-cuts) everyone in his village. Then a paradoxical question is then whether he (the barber) can shave (or hair-cut) for himself.

One noteworthy and similar biblical example is found in 2 Timothy 3:16. Here Paul's letter which is a holy scripture (2 Timothy) refers to "all the scriptures" (that is, the canonic bible yet to be completed) which includes the very scripture (the whole letter of 2 Timothy, yet to be completely written down) and the very verse (2 Timothy 3:16) that Paul is writing. A similar but more generalized philosophical example is a tea-cup where this tea-cup is the biggest tea-cup in the universe which contains all the tea-cups in the universe. Then a paradoxical question is whether this biggest tea-cup is contained in itself or not. These examples do not have a negation in circularity, in contrast to the Liar's Paradox. One controversial question dealing with this type of the Liar's Paradox (Titus 1:12) is whether a paradox contains an inherent self-contradiction due to the presence of negation in circularity. One similar example to the Liar's Paradox is a set "R" whose element is a set which does "not" contain itself. Then a paradoxical question is whether the set "R" contains itself or not. There are two possibilities. First (1), let's assume that the set "R" does not contain itself. Since the set "R" does not contain itself, then the set "R" should be an element of the set "R" by the very definition of the set "R". This is a contraction to the assumption (1). Second (2), let's assume that the set "R" does contain itself. Then since the set "R" does contain itself, then the set "R" does not contain itself by the very definition of the set "R", which is again a contradiction. This is one of the difficulties dealing with a paradox of circularity with negation.

In this regards, one may note the difficulty in the Liar's Paradox, dealing with the circularity in negation. Since the

Cretan prophet asserts that all Cretans are liars, this Cretan prophet is then also a liar and thus to assert his own statement (the prophecy cited by Paul in Titus 1:12) to be a lie (that is, its truth-value to be “false”). In other words, it is the Cretan prophet whose prophecy (which is true and is even affirmed by the apostle Paul) denies truthfulness of a message of any Cretan (including himself) as a liar. If granted and extended, this line of reasoning further shakes the credibility of Paul’s assertion of Titus 1:12 in Titus 1:13. This line of reasoning seems to create a theological controversy and an easy target of a suicidal trap or a witch hunt of being a heresy. However, one should also note that a liar needs not to tell a lie all the time. A liar may tell truth all the time except a few times to tell a lie. This leads to the discovery of yet another important feature of modal reasoning (logic) as one works with a paradox of circularity. Modal logic deals with the modal aspect of logic, qualifying the truth of judgment. The most important and well-known pair is that of “necessarily” and “possibly” modal qualifiers for “alethic” (truth) modality. There are many more modalities and associated modal logics as one may imagine. For example, temporal aspect of past, present, and future is dealt with “temporal” modalities, the aspect of permission and obligation is dealt with “deontic” modalities, and the aspect of the belief and knowledge is with “epistemic” modalities. As noted earlier, once ignored in the past, modal reasoning and logic have become an important and emerging area of scholarship in recent years. Some of these emerging new fields in the last one or two decades are, for example, computational ethics with the advance of autonomous system (e.g., robots and intelligent agents) and computational law in automated intelligent legal tutoring systems to train law students. There are plenty of examples of the biblical paradoxes dealing with the legal aspect. For example, Matthew 22:15-22 is for the tax law, Matthew 22:23-33 is for marriage (family) law, Matthew 22:34-40 is for the greatest law, Matthew 22:41-26 is for the legal aspect of the lord-servant relationship, the chain of command, and obedience.

Second noteworthy observation is that this type of paradox (Titus 1:12) deals with not only a circularity but also a negation. Even though a negation is not explicitly stated, one may discover a negative implication imbedded in this paradox. That is, a lie is a “false” statement (proposition) and a liar tells a lie all the time. A similar example is a friend-relationship along with an enemy-relationship as noted previously. Thus one may define a friend-relationship which is by default a negation of an enemy-relationship. For example, if X is a friend of Y, then X is “not” an enemy of Y. Similarly, if X is an enemy of Z, then X is “not” a friend of Z. However, there are many hate-love relationships in the Bible (and many more in the history of mankind in politics or love-triangle). Thus, negation in a circular reasoning presents not only a challenge but also it complicates the meaning and its validity. There are many examples of paradoxical negations in the biblical system. Some of the noteworthy examples in Mark (Santos 1997) include: self-denial (Mark 8:34), saving-or-losing one’s life (Mark 8:35), and servant-leadership (Mark 9:35), along with many other examples throughout the Bible.

(4) The fourth example is found in Matthew 22:41-46, extending the number of the constituents in a cycle, as shown in Figure 4.

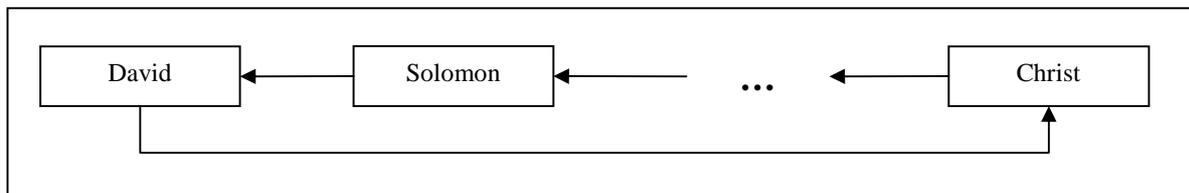


Figure 4. Lord-Servant relationship from David to Christ (who is a son of David)

This example deals with the extended “Father-Son” relationship which is compatible with “Lord/Master-Servant” relationship. A father (who is also a king) is the lord (master) of his own son (servant). This “Father-Son” relationship is extending to his own son’s sons. That is, the relationship is transitive as to an immediate “Ancestor-Descendant” relationship where “Father” is also referred to as one’s “Ancestor-Father” in a direct blood-lineage in the biblical context. Then the question is why David called Christ (who is his own descendent) “my lord” in Psalm 110:1. This illustrates clearly a circularity (a cycle or a loop) with an arrow-line here to signify “You-are-my-Lord” (that is, “Lord-Servant”) relationship, counterintuitive to the intended “Father-Son” relationship from David to Christ. As one may note, some of these circular relationships may not be so clear at a lexical or syntactic level. But one may require a further analysis to derive circular relationship semantically.

(5) In Matthew 22, one may find a stunning cluster of three paradoxes also viewed as “modal” reasoning (Matthew

22:15-22, 22:23-33, and 22:41-46 which has been discussed previously).

First (1), the paradox of Matthew 22:15-22 deals with the question of paying tax to Caesar or not. The question is formulated to be a decision problem demanding the answer “yes” or “no” with the excluded middle (of “either-or” with excluded-middle, but not “both-and” or “neither-nor”). It is designed and constructed logically (in the contemporary system of formal logic of the New Testament) to trap one into being a deadlocking dilemma (even worse toward a theological and political suicide) logically, politically, and theologically. The answer (Matthew 22:21) for this seemingly-impossible problem is solved so trivially after all. First (1), the solution is circular (that is, “A for A”, and “B for B”). Second (2), it is modal. As noted in the reply of Jesus, there is one model where “paying Tax to Caesar” is true and there is another model where “paying Tax to Caesar” is false. A deeper analysis may reveal that this is not just a problem of “either-or” but it could be a problem of “both-and” if one considers the theological mandate (for example, Romans 13:6-7). The breath-taking threat of the deadly plot (which is so beautifully orchestrated by the leading religious and political authority) is now taking a stunning pivotal point (which is again so beautifully and simply orchestrated) resolved by the reply of this young teacher Jesus (cf. Psalm 8:2, 1 Corinthians 1:25). As noted in biblical paradox, modal reasoning is very promising for handling some of the biblical and theological controversies dealing with the seemingly contradictory bipolar-thesis (for example, the problems of (1) the sovereignty of God and the free will of man, (2) the grace and faith in salvation, and (3) the justice of God and the existence of the evil and suffering).

Second (2), similarly the paradox of Matthew 22:23-33 is constructed to trap someone in an intellectual and theological deadlock and dilemma. The question is dealing with marriage and resurrection. Again the question appears to demand either true or false answer of a logical dichotomy (of “either-or” with excluded-middle, but not “both-and” or “neither-nor”). Moreover the answer seems so impossible to decide either this or that (that is, whose wife the woman is) in this world of the living. But after all it seems so trivial to say “neither-nor” in that world of the resurrected. The marital (single, married, or widowed) and spousal (whose wife) attribute of the woman changes time after time as well as the life-attribute (living or dead) of all the characters in the paradox changes. This aspect brings our attention to an additional and critical feature of this paradox for “nonmonotonic” reasoning. That is, the spousal attribute or the husband-identity of the woman is not monotonic (for one to be married and then to stay married) but nonmonotonic (from being married to being married again where the identity of her husband is changing each time). Nonmonotonic reasoning is one of the common motifs and themes in the Bible (for example, Ecclesiastes 3:1-10, 7:14), seemingly contradictory to the monotonic reasoning or principle. For example, if the righteous are blessed and the evil are cursed, then how can a righteous man be cursed or persecuted (cf. Job 1-2, Matthew 5:10-12). Another way to look at nonmonotonic reasoning is that it deals with not only what is true (that is, monotonic) but also what is false.

Third (3), the paradox of Matthew 22:41-46 (as we discussed earlier for its circular construct in detail) provides a classical case of the paradoxes of circularity. All of these three examples in Matthew 22:15-46 use a paradox as a literary and rhetoric device to generate a stunning awe, wonder, mystery, or even confusion as a reader response. In particular, the application of a paradox is meant to be a “deadly and forcefully” offensive rhetoric device for all three examples. All of these examples exhibit further the application and appreciation of paradox as a literary genre and rhetorical device by the contemporary people of Jesus but also throughout the Bible. Its intended goal (or its application as a means) is to demonstrate the wisdom and authority of the speaker over the audiences (including the spectators) or the readers (whether its application is initiated by a challenger or a responder). There is a stunning turning point of breathtaking suspension (climax, awe, joy, victory, or excitement). And these are achieved purely by means of a literary, rhetoric, and logical device in a discourse (or a plot). The author (of Matthew) is well aware of paradox as one of the literary genres and the means of rhetoric and discourse, and the dynamic of its narrative feature and structure. And he is keenly aware of and has a mastery of its rhetoric purpose and effect (for example, Matthew 22:15 and 22:46) on the audience and his readers, as a paradox is commonly used and constructed in his contemporary world, shared by the other contemporary authors of the gospels (for example, Luke 5:26).

Finally (4), the passage in Matthew 22:34-40 is placed between these offensive paradoxes. And thus it seems to be somewhat misplaced or seemingly diverting current plot-tempo toward the climax (Matthew 22:41-46) as similarly one noted in Mark 12:28-34. In contrast, a similar but more extensive narrative (question-answer) in Luke 10:25-37 seems to be a better and dynamic presentation than that of Matthew 22:34-40, in context. However, the theological question and thus its answer, for what is the greatest commandment in the Law, is indeed one of the most important and leading-edge (biblical-legal) research topics at the time of Jesus, and that of a common phenomenon (e.g.,

Matthew 7:12) among devoted biblical experts and teachers. The offenders in Matthew 22:15-46 seem to know the answer with their lips but ignorant of the answer in terms of their obedience (that is, a paradox). Thus the passage seems to enhance the underlying motif (Matthew 22:15) of the paradox and its surrounding context of this passage. In addition, in a legal system one crucial problem and challenge is to avoid a potential legal conflict or deadlock with competing but laws conflicting with each other, by providing a hierarchy into the legal system (determining which law is higher or superior than the other laws). Thus one possible solution to avoid a potential deadlock or conflict in a legal system is to arrange the laws in a hierarchical layer in precedence or in priority. In this light, the passage in Matthew 22:34-40 could be also viewed as a meta-rule (to set two laws in order, above all the other laws), as a safeguard. It is (1) to avoid the deadlock or conflict potentially implicated by the paradoxical modal complications in the biblical legal system, (2) to clarify any potential misunderstanding or confusion caused by these modal implications in practice for a due-care, and (3) to continue and enhance the underlying narrative momentum of the paradoxical debates. The passage does not undermine the inertia and the momentum of the paradoxical discourse but provide further a clear perspective of the biblical legal system (for its highest purpose and goal) in the midst of a confusing discourse of biblical paradox and its modal and nonmonotonic reasoning.

5. Conclusion

In this paper, the new perspective and paradigm of coinductive reasoning and its application to biblical texts are presented and analyzed. Current approach and methodology presented in this paper is distinctively computational. As noted in John 8:13-18, two methods of inductive and coinductive reasoning are examined and compared. This solid landmark example provides not only clear evidence of inductive and coinductive reasoning, side by side in usage and action, as the proof methods in the Bible. The discovery and presence of coinductive reasoning in John 8:13-18 creates a concrete basis for the critical method of reasoning, further to extend and understand the biblical system of reasoning, its complexity and dynamics, and its related theological concepts and motifs in John. One of the unifying concepts is the concept of secure message and communication in the biblical system, closely related to revelation, prophecy, sign, and other communication devices such as parable and dream-vision.

The current working definition of the term “paradox” is “contrary to opinion”. As noted, this definition is both satisfactory and flexible enough for the purpose and scope of this paper, conservatively following the common usage and meaning of the early Greek and New Testament. Other potential candidate meanings of paradox in the contemporary discussions (such as mystery, apparent or actual self-contradiction, etc.) have been examined for understanding and clarification of the reader. Further the landmark and well-known examples of the selected biblical paradoxes are presented, examined, and analyzed for their inherent circular, paradoxical, modal, and nonmonotonic features. These examples of circular paradoxes are the meaning of the name of God in Exodus 3:14, the Father-Son relationship in John 14:10, the Liar’s Paradox in Titus 1:12, three paradoxes from Matthew 22:15-46.

As noted, the scope of this paper is restricted to the class of biblical paradoxes of circularity. Current approach with coinductive reasoning provides a promising prospective and results, for many of the classical problems, in biblical paradox. Even though current approach does not solve all the problems, the new paradigm is very promising. Many of the confusions, the failures, the misunderstandings, and the negligence dealing with biblical paradox in the past are now well-understood and resolved through coinductive reasoning. This is the authors’ hope through this study to bring a renewed interest, understanding, and excitement toward the study of biblical paradox in the dawn of the 21st century.

Acknowledgments: we acknowledge Luke Simon, Ajay Mallya, and Ajay Bansal for their valuable contribution to co-LP research at ALPS Lab at the University of Texas at Dallas, and Vítor Santos Costa and Ricardo Rocha for their assistance for implementing CoLP on top of YAP, Peter Stuckey for his valuable advice and critiques have helped this research of co-LP standing on a solid theoretical foundation, George Kelly for his readership, Dave Basinger for readership and valuable critique for this paper, Se June Hong at IBM Thomas J. Research Center, Katy Barnwell at Wycliffe Bible Translators, Paul Miller at Gramcord, and Sam Underwood at First Baptist Church - Farmers Branch, for his friendship, mentorship, support, and encouragement for many years.

Reference

- Aczel, Peter (1977). An introduction to inductive definitions. In K. Jon Barwise, editor, *Handbook of Mathematical Logic*, pages 739–782. North-Holland, Amsterdam.
- . (1988). *Non-well-founded Sets*, volume 14 of *CSLI Lecture Notes*. CSLI Publications, Stanford, CA.
- Baader, F., D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider (2003). *The Description Logic Handbook: Theory, Implementation, Applications*. Cambridge University Press, Cambridge, UK.
- Barwise, J., Moss, L. (1996). Vicious Circles: On the Mathematics of Non-Wellfounded Phenomena. CSLI Pub.
- Basinger, David (1987). Biblical Paradox: Does Revelation challenge logic?. *JETS* 30/2 (June 1987) 205-213.
- Bauer, Walter (1957). *A Greek-English Lexicon of the New Testament and Other Early Christian Literature*, trans. William F. Arndt and F. Wilbur Gingrich, rev. Frederick W. Danker, 2d ed. (Chicago: University of Chicago Press, 1957), 609.
- Brewka, Gerhard (1991). *Nonmonotonic reasoning: Logical foundations of commonsense*. Cambridge Univ. Press.
- Brewka, Gerhard, Jürgen Dix, and Kurt Konolige (1997). *Nonmonotonic reasoning: An overview*. CSLI Lecture Notes Number 73, CSLI Publications. Stanford, CA.
- Clark, K. L. (1978). Negation as Failure. In: Gallaire, H., Minker, J. (eds.) *Logic and Data Bases*. 293–322. Plenum Press, New York.
- Colmerauer, Alain (1990). An introduction to Prolog III. *Communications of the ACM*, 33(7):69–90.
- Colmerauer, Alain, and P. Roussel (1996). The birth of PROLOG. In *History of Programming Languages*. ACM Press/Addison-Wesley.
- Creech, Richard (1984). *Christology and Conflict: A comparative study of two central themes in the Johannine literature and the apocalypse*. Ph.D. dissertation, Baylor Univ., 1984. pp.30-114.
- Cryder, Christian (2001). "Reymond's rejection of paradox," *Trinity Journal* ns 22.1 (Spring 2001): 99-112.
http://www.biblicalstudies.org.uk/articles_trinity-journal_02.php
and <http://granitepeaks.com/sld/christian/paradox.pdf>
- Fitting, M. C. (1983). Notes on the Mathematical Aspects of Kripke's Theory of Truth. *Notre Dame Journal of Formal Logic* 27(1): 75–88.
- . (1985). A Kripke/Kleene Semantics for Logic Programs. *Journal of Logic Programming* 2, 295–312.
- Fitting, M. C., and R. L. Mendelsohn (1998). *First-order modal logic*. Synthese Library vol. 277. Kluwer Academic publishers.
- Franck, Sebastian (1986). Paradox or Wondrous Sayings, *Texts and Studies in Religion* 26, trans. E. J. Furcha (Lewiston: Edwin Mellen Press, 1986), 1.
- Goldin, D., and Keil, D. (2001). Interaction, Evolution and Intelligence. *Proc. Congress on Evolutionary Computing*
- Gramling, Gary Lee (1993). *The Metaphorical ego eimi sayings in the fourth Gospel: Their origin and significance*. Ph.D. dissertation, Golden Gate Baptist Theological Seminary, 1993.
- Gupta, G., A. Bansal, R. Min, L. Simon, and A. Mallya (2007). Coinductive Logic Programming and Its Applications. (Tutorial Paper). In: *Proc. of ICLP07*, 27–44. LNCS, Springer.

- Gupta, G., Simon, L., Bansal, A., Mallya, A., Min, R. (2009). Coinductive Logic Programming and its Applications. PADL'09. (Invited Paper). LNCS 5413, Springer.
<http://www.utdallas.edu/~rkm010300/research/PADL2009Gupta.pdf>
- Kant, I. (1781). Immanuel Kant's Critique of Pure Reason. London: Macmillan & Co. trans. Norman Kemp Smith, 1964. (Kant (1781) in his *Table of Judgments* in Book I of the *Transcendental Analytic*.)
- Kittel, Gerhard (1964). "παραδοξος" in TDNT, ed. G. Kittel, trans. Geoffrey W. Bormiley (Grand Rapids: Eerdmans, 1964) 2:255.
- . (1964). "δοξα" in TDNT, ed. G. Kittel, trans. Geoffrey W. Bormiley (Grand Rapids: Eerdmans, 1964) 2:233-236.
- Kowaiski, Robert A. (1974). Predicate logic as programming language. In Jack L. Rosenfeld, editor, *Proceedings of International Federation of Information Processing Conference*, pages 569–574, Stockholm, Sweden. North-Holland.
- Kowaiski, Robert A., and Donald Kuehner (1971). Linear resolution with selection function. *Journal of Artificial Intelligence*, 2:22–260.
- Kripke, S. (1975). Outline of a Theory of Truth. *Journal of Philosophy* 72. 690-716.
- Lloyd, J. W. (1987). *Foundations of Logic Programming*. 2nd Edition. Springer.
- Min, R. (2009). Predicate Answer Set Programming with Coinduction. Ph.D. Dissertation. Department of Computer Science, The University of Texas at Dallas. June 2009.
- Min, R. (2010). Predicate Answer Set Programming with Coinduction (Book ISBN: 978-3-8433-6582-6), Lamport Academic Publishing. Germany, 2011.
- Min, R. (2011). Biblical Paradox and "I-am" sayings in John. Society of Biblical Literature - International Meeting 2011, London, United Kingdom
- Min, R., Gupta, G. (2009). Coinductive Logic Programming and its Application to Boolean SAT. FLAIRS 2009.
- Min, R., Gupta, G. (2009). Coinductive Logic Programming with Negation. LOPSTR'09. LNCS 6037, Springer.
- Min, R., Gupta, G. (2010). Biblical Paradox and Coinductive Reasoning. Society of Biblical Literature - International Meeting 2010, Tartu, Estonia.
- Min, R., Bansal, A., Gupta, G. (2009). Towards Predicate Answer Set Programming via Coinductive Logic Programming. AIAI 2009. Springer. <http://www.utdallas.edu/~rkm010300/research/AIAI2009Min.pdf>
- Min, R., Bansal, A., Gupta, G. (2010). Predicate Answer Set Programming via Coinductive Logic Programming. *Journal Engineering Intelligent Systems (JEIS)*. Special Issue: Developments in the field of Intelligent Systems Applications. 2010. CRL Publishing.
- Reymond, Robert (1998). *A New Systematic Theology of the Christian Faith*, 50, 103. 2nd edition, Thomas Nelson
- Robinson, J. A. (1965). A machine-oriented logic based on the resolution principle. *Journal of the ACM*. 12:23–41.
- . (1971). Computational logic: The unification computation. In B. Meltzer and D. Michie, editors, *Machine Intelligence 6*, (1971). pages 63–72. Edinburgh University Press, Edinburgh.
- Russell's Paradox <http://plato.stanford.edu/entries/russell-paradox/>

- Santos, Narry Fajardo (1994). *The Paradox of Authority and Servanthood in the Gospel of Mark*. Ph.D. Thesis. Dallas Theological Seminary.
- . (1997). “The paradox of authority and servanthood in the gospel of Mark,” *Bibliotheca Sacra* 154, pp. 452-460.
- Scharlemann, Martin H. (1957). “The Paradox in Perspective” *ConcTM* 28 (May 1957), pages 349-359.
- Seat, L. K. (1967). “‘The Meaning of Paradox’: A Study of the Use of the Word ‘Paradox’ in Contemporary Theological and Philosophical Writings with Special Reference to Soren Kierkegaard.” Th.D. diss., Southern Baptist Theological Seminary, 1967. pages 2, 8-18.
- Shannon, Claude E. and Warren Weaver (1998). *The Mathematical Theory of Communication*. The University of Illinois Press, Urbana, Illinois, 1949. Reprinted: University of Illinois Press; First Edition.
- Shannon, Claude E. (1948). *A Mathematical Theory of Communication*, Bell System Technical Journal, Vol. 27, pp. 379–423, 623–656. <http://cm.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf>
- Simon, Luke, , Ajay Bansal, Ajay Mallya, and Gopal Gupta. (2006). Coinductive Logic Programming. In: ICLP'06. LNCS 4079, pp. 330–344. Springer.
- Simon, Luke, Ajay Bansal, Ajay Mallya, and Gopal Gupta. (2007). Co-Logic Programming. In: Proc. ICALP'07, pp. 472–483. LNCS. Springer.
- Stanford1. Stanford Encyclopedia of Philosophy. “Paradoxes and Contemporary Logic ” <http://plato.stanford.edu/entries/paradoxes-contemporary-logic/>
- Stanford2. Stanford Encyclopedia of Philosophy. Russell's Logical Atomism. <http://plato.stanford.edu/entries/logical-atomism/>
- Stanford3. Stanford Encyclopedia of Philosophy. “Tarski's Truth Definitions”. <http://plato.stanford.edu/entries/tarski-truth/>
- Stanford4. Stanford Encyclopedia of Philosophy. Wittgenstein's Logical Atomism, <http://plato.stanford.edu/entries/wittgenstein-atomism/>
- Stanford5. Stanford Encyclopedia of Philosophy. Self-Reference. <http://plato.stanford.edu/entries/self-reference/>
- Stewart, William K. (1928). “The study of paradox”, *HibbJ* 27 [October 1928]: 2. pages 1-14.
- Tarski, Alfred (1955). A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–309.
- . (1983). *Logic, semantics, meta-mathematics*. 2nd ed. edited and translated by Corcoran, John. Indianapolis: Hackett Publishing Company.
- YAP. YAP Prolog. <http://www.dcc.fc.up.pt/~vsc/Yap/>