LQMR 2015 Workshop pp. 35–40
DOI: 10.15439/2015F326
ACSIS, Vol. 7

On (in)Validity of Aristotle's Syllogisms Relying on Rough Sets

Tamás Kádek
University of Debrecen
Faculty of Informatics
Kassai street 26, 4028 Debrecen, Hungary
Email: kadek.tamas@inf.unideb.hu

Tamás Mihálydeák
University of Debrecen
Faculty of Informatics
Kassai street 26, 4028 Debrecen, Hungary
Email: mihalydeak.tamas@inf.unideb.hu

***Abstract*—The authors investigate the properties of first-order logic having its semantics based on a generalized (partial) approximation of sets. The goal of the investigation in this article is to compare the classical first-order semantics with a partial and lower approximation-based one. The idea is that lower approximation represents the reliable knowledge, so the reasoning used by the lower approximation may be valid or may be valid with some limitations. First, the authors show an experimental result which confute the previous supposition and the result of an algorithm which generates refutations for some well-known valid arguments: the 12 syllogisms of Aristotle. We think that these syllogisms represent the most common usage of categorical statements. A language with single-level quantification is constructed, as syllogisms can be formalized using this language. Based on the experimental results, the authors suggest some modifications of the semantics if the goal is to approximate the classical case.**

I. Introduction

The rough set theory gives the ability to construct different first-order logical systems (see [1], [2]). By the generalization[1] of rough set theory, the truth domain of a formula can be approximated using a partial approximation of sets. The authors introduced earlier a tool-based system as the semantical basis of a generalized first-order logic [5]. The introduced language let us use more than one kind of approximation and allowed it in the language level — with approximative sentence functors — to mix the crisp and rough evaluation. This rich language was very different from the classical case; furthermore, many of the classical rules — such as modus-ponens — failed when they were combined with the approximative functors. In this work, we focus only to the lower approximation of sets (later the lower approximation of truth domains) because of the naïve idea that while the upper approximation represents possibility, the lower approximation represents certainty.

We are eager to know whether what we conclude using the approximation is equal to what we conclude using the classical first-order reasoning. Whether it is possible to formulate some conditions which guarantee the validity of the results made by the approximation.

[1]Different generalizations of rough set theory (see [3]) and granular computing play a crucial role in computer sciences (see, e.g. in [4]).

The investigation starts with an experiment, testing some well-known valid arguments, the 12 syllogisms of Aristotle. These syllogisms were chosen in order to represent the most common usage of categorical statements. In the experiment, we use a simplified language which gives us the ability to formalize the syllogisms and test their validity using a lower approximation-based semantics. The language is restricted to single-level quantification only, but it is still expressive enough to formalize categorical statements. Initially, this semantics is defined in the same way as in the tool-based first-order case, but restricted only to the lower approximation. Later, we suggest some modifications in the semantical level to ensure the validity of the classical arguments.

II. Aristotle's syllogisms

Aristotle's syllogisms are valid reasoning, constructed from three sentences: two premises and one conclusion. The premises are usually categorized into four types: [6]

- a-type $\forall x(p_1(x) \supset p_2(x))$
- i-type $\exists x(p_1(x) \wedge p_2(x))$
- e-type $\neg\exists x(p_1(x) \wedge p_2(x))$
- o-type $\exists x(p_1(x) \wedge \neg p_2(x))$

Each statement contains two from three predicates — usually denoted by p, s, m — and each predicate appears in exactly two statements.

1st figure	2nd figure	3rd figure	
$m-p$	$p-m$	$m-p$	premise
$s-m$	$s-m$	$m-s$	premise
$s-p$	$s-p$	$s-p$	conclusion

For example, the syllogism called *Barbara* contains only a-type premises and a-type conclusion:

$$\forall x(m(x) \supset p(x)), \forall x(s(x) \supset m(x)) \models \forall x(s(x) \supset p(x))$$

During the investigation, we focus on the syllogisms from the first 3 figures, and we do not take care of those which require an existential pre-supposition, only the remaining 12:

- 1st figure: *Barbara*, *Celarent*, *Darii*, *Ferio*
- 2nd figure: *Cesare*, *Camestres*, *Festion*, *Baroco*
- 3rd figure: *Disamnis*, *Datisi*, *Bocardo*, *Ferison*

There were several similarities in the results, that is why this article presents only those which belong to the first figure. All of them are valid in the classical case, and now the question is: could we create any refutations using lower approximations only?

III. Simplified Partial First-Order Logic Based on Set Approximation

In this section, we would like to introduce a simplified first-order language, expressive enough to formalize the syllogisms.

A. First-Order Language

Let $\langle LC, Var, Con, Tool, Form\rangle$ be a simplified first-order language, where:

- logical constant symbols $LC = \{\neg, \wedge, \supset, \exists, \forall, (,)\}$,
- variables $Var = \{x\}$, note that one variable is enough,
- nonlogical constant symbols $Con = \{p_1, p_2, \cdots, p_n\}$, where $n \geq 1$,
- set of tools $Tool = \{t_1, t_2, \cdots, t_k\}$, where $k \geq 1$.

The formulas of the language are given by the following definition:

1) Let QF be the set of quantification-free expressions, so that
   a) $p(x) \in QF$ — and it is atomic — if $p \in Con$,
   b) $\neg A \in QF$ if $A \in QF$,
   c) $(A \wedge B) \in QF$ and $(A \supset B) \in QF$ if both $A \in QF$ and $B \in QF$
2) The set $Form$ is given by the following inductive definition:
   a) $\forall x A \in Form$ and $\exists x A \in Form$ if $A \in QF$
   b) $\neg A \in Form$ if $A \in Form$,
   c) $(A \wedge B) \in Form$ and $(A \supset B) \in Form$ if both $A \in Form$ and $B \in Form$

Note that the language is restricted to unary predicates (that is, they are monadic) and single-level quantification. So we define a fragment of the first-order logic which is decidable. The disjunction symbol $\vee$ is also missing from the language, it is not necessary to formalize the syllogisms later, but this is not a real restriction.

B. Partial Approximation of Sets

The ordered 5–tuple $\langle U, \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, \mathsf{l}, \mathsf{u}\rangle$ is a general partial approximation space[2] if

1) U is a nonempty set;
2) $\mathfrak{B} \subseteq 2^U \setminus \emptyset$, $\mathfrak{B} \neq \emptyset$;
3) $\mathfrak{D}_{\mathfrak{B}}$ is an extension of $\mathfrak{B}$, i.e., $\mathfrak{B} \subseteq \mathfrak{D}_{\mathfrak{B}}$, such that $\emptyset \in \mathfrak{D}_{\mathfrak{B}}$; and $\bigcup B \in \mathfrak{D}_{\mathfrak{B}}$ for all $B \subseteq \mathfrak{B}$
4) the functions l and u form a Pawlakian approximation pair $\langle \mathsf{l}, \mathsf{u}\rangle$, i.e.,
   a) the lower approximation of an $S \in 2^U$ set is
   $$\mathsf{l}(S) \stackrel{def}{=} \cup\{B : B \in \mathfrak{B} \text{ and } B \subseteq S\};$$
   b) the upper approximation of an $S \in 2^U$ set is
   $$\mathsf{u}(S) \stackrel{def}{=} \cup\{B : B \in \mathfrak{B} \text{ and } B \cap S \neq \emptyset\}.$$

[2]One of the most general notion of weak and strong approximation pairs can be found in Düntsch and Gediga [7].

The Pawlakian approximation pair was chosen because it is very well-known and most widely used. For other solutions see [8].

C. Interpretation

The $\langle U, \varrho\rangle$ pair is an interpretation of the language $\langle LC, Var, Con, Tool, Form\rangle$ if

- U is a nonempty set of objects, and
- $\varrho : Con \cup Tool \to 2^U$ is a mapping, and
- $\varrho(t_i) \neq \emptyset$ for all $i \in \{1, ..., k\}$.

The ϱ mapping assigns a truth domain to each nonlogical constant symbol and a nonempty truth domain to the tools.

D. Semantic Rules

Let $\langle U, \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, \mathsf{l}, \mathsf{u}\rangle$ be a general partial approximation space generated by the $\langle U, \varrho\rangle$ interpretation of a given $\langle LC, Var, Con, Tool, Form\rangle$ simplified first-order language. The ϱ mapping and the $Tool$ set generate the approximation space:

$$\mathfrak{B} = \{\varrho(t) : t \in Tool\}$$

No sentence functors appears in the language level. But the semantic value of a formula $[\![F]\!]^{\langle U,\varrho\rangle}$ and the semantic value of the quantification-free expressions $[\![Q]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$ are defined based on the lower approximation.

Let $\stackrel{w}{\neg}$, $\stackrel{w}{\supset}$, $\stackrel{w}{\wedge}$ be weak Kleene connectives [9], such that

$\stackrel{w}{\neg}$	
0	1
1	0
2	2

$\stackrel{w}{\supset}$	0	1	2
0	1	1	2
1	0	1	2
2	2	2	2

$\stackrel{w}{\wedge}$	0	1	2
0	0	0	2
1	0	1	2
2	2	2	2

Our selection fell on Kleene's weak connectives because of the idea to keep the truth value gap. It was the basis of the semantics defined later for quantifiers too.

1) The semantic value of an atomic expression $p(x) \in QF$ using a given interpretation $\langle U, \varrho\rangle$ and a variable assignment $x \mapsto u$ where $u \in U$:
$$[\![p(x)]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{def}{=} \begin{cases} 1 & \text{if } u \in \mathsf{l}(\varrho(p)) \\ 0 & \text{if } u \in \mathsf{l}(U \setminus \mathsf{u}(\varrho(p))) \\ 2 & \text{otherwise} \end{cases} \quad (1)$$
where the $\langle \mathsf{l}, \mathsf{u}\rangle$ approximation pair belongs to the approximation space generated by the $Tool$ and $\langle U, \varrho\rangle$.
2) The semantic value of quantification-free expression is defined recursively
$$[\![\neg A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{def}{=} \stackrel{w}{\neg} [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$$
$$[\![(A \supset B)]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{def}{=} [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{w}{\supset} [\![B]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$$
$$[\![(A \wedge B)]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{def}{=} [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} \stackrel{w}{\wedge} [\![B]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$$
3) The semantic value of a formula from $Form$

TABLE I
ARISTOTLE'S SYLLOGISMS — THE FIRST FIGURE

Syllogism	first premise	second premise	conclusion
Barbara	a-type m, p	a-type s, m	a-type s, p
	$\forall x(m(x) \supset p(x))$	$\forall x(s(x) \supset m(x))$	$\forall x(s(x) \supset p(x))$
Celarent	e-type m, p	a-type s, m	e-type s, p
	$\neg\exists x(m(x) \wedge p(x))$	$\forall x(s(x) \supset m(x))$	$\neg\exists x(s(x) \wedge p(x))$
Darii	a-type m, p	i-type s, m	i-type s, p
	$\forall x(m(x) \supset p(x))$	$\exists x(s(x) \wedge m(x))$	$\exists x(s(x) \wedge p(x))$
Ferio	e-type m, p	i-type s, m	o-type s, p
	$\neg\exists x(m(x) \wedge p(x))$	$\exists x(s(x) \wedge m(x))$	$\exists x(s(x) \wedge \neg p(x))$

$$[\![\forall x A]\!]^{\langle U,\varrho\rangle} \stackrel{def}{=} \begin{cases} 2 & \text{if } [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 2 \text{ for all } u \in U, \\ 0 & \text{if there is an } u \in U, \\ & \quad \text{where } [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 0, \\ 1 & \text{otherwise.} \end{cases}$$

$$[\![\exists x A]\!]^{\langle U,\varrho\rangle} \stackrel{def}{=} \begin{cases} 2 & \text{if } [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 2 \text{ for all } u \in U, \\ 1 & \text{if there is an } u \in U, \\ & \quad \text{where } [\![A]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

where $A \in QF$.

$$\begin{aligned} [\![\neg A]\!]^{\langle U,\varrho\rangle} &\stackrel{def}{=} \stackrel{w}{\neg} [\![A]\!]^{\langle U,\varrho\rangle} \\ [\![(A \supset B)]\!]^{\langle U,\varrho\rangle} &\stackrel{def}{=} [\![A]\!]^{\langle U,\varrho\rangle} \stackrel{w}{\supset} [\![B]\!]^{\langle U,\varrho\rangle} \\ [\![(A \wedge B)]\!]^{\langle U,\varrho\rangle} &\stackrel{def}{=} [\![A]\!]^{\langle U,\varrho\rangle} \stackrel{w}{\wedge} [\![B]\!]^{\langle U,\varrho\rangle} \end{aligned}$$

where $A, B \in Form$.

IV. EXPERIMENTAL RESULTS

If we have $|U| = 4$, then the number of different interpretations is:

$$\left(2^{|U|}\right)^3 \cdot \left(2^{\left(2^{|U|}-1\right)} - 1\right) = 4\,096 \cdot 32\,767 = 134\,213\,632,$$

where the number of different approximation spaces is 32 767 (where the members of the $Tool$ set has different $\varrho(t)$ nonempty truth domain), and there exists 4 096 different interpretation for $Con = \{p, s, m\}$. With such a small U, there is an efficient way to implement the formula evaluation. The idea is that if there is a given $\langle U, \varrho\rangle$ interpretation, we can generate the truth domain and falsity domain for each predicate before the evaluation.

Let us define the truth and falsity domain of an atomic formula in the classical case

$$[\,p\,]^+ = \varrho(p) \text{ and } [\,p\,]^- = U \setminus \varrho(p) \text{ if } p \in Con \cup Tool$$

and in case of the introduced semantics (with the lower approximation based on (1))

$$\begin{aligned} \left[{}^{\downarrow}p\right]^+ &\stackrel{def}{=} \mathsf{l}(\varrho(p)) \\ &= \left\{u \in U \;:\; [\![\,p(x)\,]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 1\right\} \end{aligned}$$

$$\begin{aligned} \left[{}^{\downarrow}p\right]^- &\stackrel{def}{=} \mathsf{l}(U \setminus \mathsf{u}(\varrho(p))) \\ &= \left\{u \in U \;:\; [\![\,p(x)\,]\!]^{\langle U,\varrho\rangle}_{x\mapsto u} = 0\right\} \end{aligned}$$

While the set $[\,p\,]^+$ denotes the truth domain of p in case of classical semantics, the $\left[{}^{\downarrow}p\right]^+$ represents the lower approximation of this truth domain. Note that $\left[{}^{\downarrow}p\right]^+ \subseteq [\,p\,]^+$ and $\left[{}^{\downarrow}p\right]^- \subseteq [\,p\,]^-$. Earlier — in [5] — we defined not only a lower approximation-based semantics but also a first-order language with lower and upper approximative sentence functors denoted by $\downarrow$ and $\uparrow$. Now we focus only on the lower approximation, supposing that is represents certainty.

The Java code sample shows the implementation of the semantics in the case of the $\forall$ quantifier and in the case of the $\wedge$ connective. The syllogisms are transformed into a postfix form. For example, in case of *Barbara*:

$$\forall x(m(x) \supset p(x)), \forall x(s(x) \supset m(x)) \models \forall x(s(x) \supset p(x))$$

is valid, and so

$$\forall x(m(x) \supset p(x)) \wedge \forall x(s(x) \supset m(x)) \wedge \neg\forall x(s(x) \supset p(x))$$

is unsatisfiable. The last formula is converted to a string representation as `"mp>Asm>A&sp>A-&"`, where $\supset$, $\wedge$, $\neg$, and $\forall$ is replaced with `>`, `&`, `-`, and `A`, respectively.

The algorithm uses a pair of stacks to evaluate the postfix expression, one (`sfd` integer array) for falsity domain, and another (`std` integer array) for truth domain. `sp` refers to the top of the stack. The $\left[{}^{\downarrow}p\right]^+$ and $\left[{}^{\downarrow}p\right]^-$ sets are represented by the `ptd['p']` and `pfd['p']` integers. Each set data structure is represented as a bit array stored in an integer.

A. Refutations for the First Figure

The table (cf. table II) summarizes the results showing the number of refutations for the syllogisms. It was not necessary

TABLE II
NUMBER OF REFUTATIONS FOR THE FIRST FIGURE

	Number of interpretations	Number of approximation spaces	Barbara	Celarent, Darii, Ferio
Total	134 213 632	32 767	121 536	227 232
$\lvert Tool\rvert \leq 3$	2 355 200	575	4 728	9 576
$\lvert Tool\rvert \leq 2$	491 520	120	912	912
covering	132 288 512	32 297	117 696	219 936
not covering	1 925 120	470	3 840	7 296
disjoint tools	208 896	51	1 104	1 104
$[p]^+ \neq \emptyset \quad \forall p \in Con$	64 139 967	32767	110448	193776
$[p]^- \neq \emptyset \quad \forall p \in Con$	689 831	32 767	12 168	25 716
$[p]^+ \neq \emptyset \wedge [p]^- \neq \emptyset \quad \forall p \in Con$	229 940	2 680	1368	10 608
$\varrho(p) \neq \varrho(t) \quad \forall t \in Tool, \forall p \in Con$	5 18 720	30 580	540	4 752

Algorithm 1 Calculating the truth value of a formula in Java

```
switch (next) { // next char of postfix formula
case 'A': // forall
    sp--; // pop an argument
    afd = sfd[sp];
    atd = std[sp];
    // push the result
    sfd[sp] = (afd & mask) != 0 ? mask : 0;
    std[sp] = (atd & mask) != 0
          && (afd & mask) == 0 ? mask : 0;
    sp++;
    break;

case '&': // conjunction
    sp--; // pop an argument
    int rfd = sfd[sp];
    int rtd = std[sp];
    sp--; // pop an argument
    int lfd = sfd[sp];
    int ltd = std[sp];
    // push the result
    sfd[sp] = (lfd & (rfd|rtd))
          | (rfd & (lfd|ltd));
    std[sp] = ltd & rtd;
    sp++;
    break;

// ... other connectives ...

default: // an atom
    sfd[sp] = pfd[next];
    std[sp] = ptd[next];
    sp++;
    break;
}
```

to show all the 12 (only those which belong to the first figure), because the same number of refutations appears in the cases of *Barbara*, *Baroco*, and *Bocardo*, as well as in all of the other cases (*Celarent*, *Darii*, *Ferio*, *Cesare*, *Camestres*, *Festion*, *Disamnis*, *Datisi*, and *Ferison*). The upper half of the table summarizes the tested conditions defined on the approximation space. The restrictions on the $Tool$ set take their effect on the generated approximation space:

- by restricting the number of different tools to at most 3,
- or at most 2,
- using a covering approximation space $U = \bigcup_{t \in Tool} t$,
- or even a noncovering one,
- or by using tools with disjoint truth domain only.

The outcome achieved is not exactly we had hoped for. The number of approximation spaces decreases, but there still exist some interpretations where the syllogisms do not hold.

In the lower half of the table, there are some restrictions on the interpretation of the predicates (the members of the Con set). Here, the number of approximation spaces can be lower than 32 767 only if there are some approximation spaces where none of the interpretations satisfy the condition:

- the truth domain of the lower approximated predicates must not be empty,
- the falsity domain of the lower approximated predicates must not be empty,
- neither the truth domain nor the falsity domain of the lower approximation of the predicates are empty,
- there is no predicate which is also a tool.

	p	s	m	t_1	t_2	t_3	$^{\downarrow}p$	$^{\downarrow}s$	$^{\downarrow}m$
u_1	0	0	0	1	0	0	0	0	0
u_2	1	0	0	0	1	0	1	0	0
u_3	0	1	1	0	0	1	2	1	1
u_4	1	1	1	0	0	1	2	1	1

The above example shows an interpretation which is a refutation for all of the followings: *Darii*, *Ferio*, *Festion*, *Datisi*, and *Ferison*. Note that the approximation space is covering, each approximated predicate has nonempty truth domain and nonempty falsity domain. The column title p is an abbreviation for $|p(x)|^{\langle U,\varrho\rangle}_{x\mapsto u}$, and $^{\downarrow}p$ is for $[\![p]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$.

B. How to Create Refutations?

The presented experimental results show that the lower approximation and the three-valued — two-valued but with truth value gap — semantics could not represent irrefutable

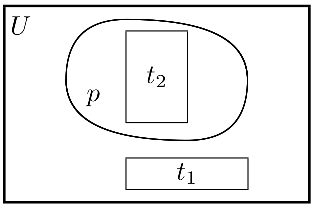

Fig. 1. Illustration of an approximation space

knowledge. In this section, we show a simple way to construct a formula which is easy to refute.

Now our goal is to construct a formula A such that $[\![A]\!] = 1$ but $[A] = 0$. It is enough to create a sentence which is about objects that exist but are outside of the lower approximation.

Let $P = \{p_1, \cdots, p_n\}$ and $N = \{p_{n+1}, \cdots, p_{n+m}\}$ be disjoint sets of predicates, such that $n + m \geq 1$. Let $S \subseteq U$ be a set, where

1) $S = \left(\bigcap_{i=1}^{n} [p_i]^+\right) \cap \left(\bigcap_{i=n+1}^{m} [p_i]^-\right) \neq \emptyset$
2) $S \cap \left(\bigcap_{i=1}^{n} [{}^{\downarrow}p_i]^+\right) \cap \left(\bigcap_{i=n+1}^{m} [{}^{\downarrow}p_i]^-\right) = \emptyset$
3) $\bigcap_{i=1}^{n+m} \left([{}^{\downarrow}p_i]^+ \cup [{}^{\downarrow}p_i]^-\right) \neq \emptyset$

Since the set S is not empty,

$$|\neg\exists x(p_1(x) \land \cdots \land p_n(x) \land \land \neg p_{n+1}(x) \land \cdots \land \neg p_{n+m}(x))| = 0,$$

but — because of the second criterion — the set is hidden for the lower approximation. It causes

$$[\![\neg\exists x(p_1(x) \land \cdots \land p_n(x) \land \land \neg p_{n+1}(x) \land \cdots \land \neg p_{n+m}(x))]\!] \neq 0.$$

The third criterion ensures computability. In other words, it avoids the truth value gap, so the existentially quantified formula must have a truth value other than 2. As a result,

$$[\![\neg\exists x(p_1(x) \land \cdots \land p_n(x) \land \land \neg p_{n+1}(x) \land \cdots \land \neg p_{n+m}(x))]\!] = 1.$$

C. Example

Let $\langle U, \varrho\rangle$ be an interpretation — illustrated by Fig. 1 — for a language with $Con = \{p, t_1, t_2\}$ and $Tool = \{t_1, t_2\}$.

Let S be a set in connection with the approximation space such that $P = \{p\}$ and $N = \{t_2\}$.

1) $S = [p]^+ \cap [t_2]^-$ and $S \neq \emptyset$
2) $S \cap [{}^{\downarrow}p]^+ \cap [{}^{\downarrow}t_2]^- = S \cap [t_2]^+ \cap [t_1]^+ = \emptyset$
3) $([{}^{\downarrow}p]^+ \cup [{}^{\downarrow}p]^-) \cap ([{}^{\downarrow}t_2]^+ \cup [{}^{\downarrow}t_2]^-) =$ $([t_2]^+ \cup [t_1]^+) \cap ([t_2]^+ \cup [t_1]^+) \neq \emptyset$.

The formula created from the sets P and N is

$$\neg\exists x\,(p(x) \land \neg t_2(x)),$$

which is false in the classical case. There are some objects in $S = \varrho(p) \cap (U \setminus \varrho(t2)) \neq \emptyset$, but the lower approximation hides them. Even in the very simple case, which is illustrated by 1, it was easy to show that $|A| = 1$ is not the logical consequence of $[\![A]\!] = 1$.

A sample $\langle U, \varrho\rangle$ interpretation in a connection with Fig. 1. can be

$$\varrho(p) = \{u_2, u_3\}, \quad \varrho(t_1) = \{u_4\}, \quad \varrho(t_2) = \{u_3\}$$

	p	t_1	t_2	${}^{\downarrow}p$	${}^{\downarrow}t_1$	${}^{\downarrow}t_2$
u_1	0	0	0	2	2	2
u_2	1	0	0	2	2	2
u_3	1	0	1	1	0	1
u_4	0	1	0	0	1	0

in case of $U = \{u_1, u_2, u_3, u_4\}$.

V. Relevance of the Lower Approximation

In this section, we summarize the conditions of stating that $|A| = 1$ is a logical consequence of $[\![A]\!] = 1$. The conditions are formalized in a form of fractions, like,

$$\frac{[\![A]\!] = 1 - \tau \Rightarrow |A| = 1 - \tau}{[\![\neg A]\!] = \tau \Rightarrow |\neg A| = \tau}$$

where $\tau \in \{0, 1\}$. Note that τ refers to the classical semantical value of a formula A, denoted by $|A|^{\langle U,\varrho\rangle}$ or for the sake of simplicity: $|A|$. ($|A|^{\langle U,\varrho\rangle} \in \{0, 1\}$.) To satisfy the condition below the line, it is enough to satisfy the condition above the line. As an example, $[\![\neg p(x)]\!] = 1 \Rightarrow [\neg p(x)] = 1$ holds if $[\![p(x)]\!] = 0 \Rightarrow [p(x)] = 0$ holds as well. The $\emptyset$ represents no conditions.

$$\frac{\emptyset}{[\![p(x)]\!] = \tau \Rightarrow |p(x)| = \tau}$$

$$\frac{[\![A]\!] = \tau \Rightarrow |A| = \tau; \qquad [\![B]\!] = \tau \Rightarrow |B| = \tau}{[\![(A \land B)]\!] = \tau \Rightarrow |(A \land B)| = \tau}$$

$$\frac{[\![A]\!] = 1 - \tau \Rightarrow |A| = 1 - \tau; \qquad [\![B]\!] = \tau \Rightarrow |B| = \tau}{[\![(A \supset B)]\!] = \tau \Rightarrow |(A \supset B)| = \tau}$$

The rules above are promising in the case of the zero-order logical connectives but not in the case of the quantifiers.

$$\frac{[\![A]\!] = 0 \Rightarrow |A| = 0}{[\![\forall x A]\!] = 0 \Rightarrow |\forall x A| = 0}$$

$$\frac{[\![A]\!] = 1 \Rightarrow |A| = 1}{[\![\exists x A]\!] = 1 \Rightarrow |\exists x A| = 1}$$

Using the semantics which gives us the ability to assign 0 or 1 truth value to quantified formulas if we have at least partial information causes that we lost the guarantees to have the classical results.

Now we suggest a modification on the first-order semantics, in a case where only the lower approximation is used. The *pessimistic*[3] semantics of quantifiers is based on the idea that the missing knowledge (represented by the value 2) could be relevant.

The *pessimistic* semantic value of a formula $[\![F]\!]^{\langle U,\varrho\rangle}$ and the *pessimistic* semantic value of the quantification-free expressions $[\![Q]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$ are defined recursively:

1) The semantic value of an atomic expression $p(x) \in QF$ using a given interpretation $\langle U, \varrho\rangle$ and a variable substitution $x \mapsto u$ where $u \in U$:

$$\lfloor\, p(x)\, \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} = [\![\, p(x)\,]\!]^{\langle U,\varrho\rangle}_{x\mapsto u}$$

where the $\langle \mathsf{l}, \mathsf{u}\rangle$ approximation pairs belongs to the approximation space generated by the $Tool$ and $\langle U, \varrho\rangle$.

2) The semantic value of a quantified expression is defined as

$$\lfloor \forall x A \rfloor^{\langle U,\varrho\rangle} \overset{def}{=} \begin{cases} 0 & \text{if there is an } u \in U, \\ & \quad \text{where } \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} = 0, \\ 1 & \text{if } \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} = 1 \text{ for all } u \in U, \\ 2 & \text{otherwise.} \end{cases}$$

$$\lfloor \exists x A \rfloor^{\langle U,\varrho\rangle} \overset{def}{=} \begin{cases} 0 & \text{if } \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} = 0 \text{ for all } u \in U, \\ 1 & \text{if there is an } u \in U, \\ & \quad \text{where } \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} = 1, \\ 2 & \text{otherwise.} \end{cases}$$

3) The semantic value of a quantification-free expression or a formula based on zero-order connectives
Let $\overset{s}{\neg}$, $\overset{s}{\supset}$, $\overset{s}{\wedge}$ be strong Kleene connectives, such that

$\overset{s}{\neg}$	
0	1
1	0
2	2

$\overset{s}{\supset}$	0	1	2
0	1	1	1
1	0	1	2
2	2	1	2

$\overset{s}{\wedge}$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	2

$$\lfloor \neg A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} \overset{def}{=} \overset{s}{\neg} \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u}$$
$$\lfloor (A \supset B) \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} \overset{def}{=} \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} \overset{s}{\supset} \lfloor B \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u}$$
$$\lfloor (A \wedge B) \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} \overset{def}{=} \lfloor A \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u} \overset{s}{\wedge} \lfloor B \rfloor^{\langle U,\varrho\rangle}_{x\mapsto u}$$

where $A, B \in QF$, and

$$\lfloor \neg A \rfloor^{\langle U,\varrho\rangle} \overset{def}{=} \overset{s}{\neg} \lfloor A \rfloor^{\langle U,\varrho\rangle}$$
$$\lfloor (A \supset B) \rfloor^{\langle U,\varrho\rangle} \overset{def}{=} \lfloor A \rfloor^{\langle U,\varrho\rangle} \overset{s}{\supset} \lfloor B \rfloor^{\langle U,\varrho\rangle}$$
$$\lfloor (A \wedge B) \rfloor^{\langle U,\varrho\rangle} \overset{def}{=} \lfloor A \rfloor^{\langle U,\varrho\rangle} \overset{s}{\wedge} \lfloor B \rfloor^{\langle U,\varrho\rangle}$$

where $A, B \in Form$.

The goal of using the *pessimistic* semantic in case of the first-order connectives (quantifiers) are the validity of the rules

$$\frac{\lfloor A \rfloor = 1 \Rightarrow |A| = 1}{\lfloor \forall x A \rfloor = 1 \Rightarrow |\forall x A| = 1}$$

$$\frac{\lfloor A \rfloor = 0 \Rightarrow |A| = 0}{\lfloor \exists x A \rfloor = 0 \Rightarrow |\exists x A| = 0}$$

Unfortunately, as another effect, it increases the number of interpretations, where a quantified formula has truth value gap. Kleene's strong connectives have opposite effect. It is not necessary to change the weak connectives to strong, the goal is reached anyway:

$$\lfloor A \rfloor = \tau \Rightarrow |A| = \tau.$$

VI. Conclusion

As a general observation, we can conclude that if we change the semantics to lower approximation-based, then the syllogisms of Aristotle are not valid. There are several ways for further investigation by creating restrictions on the approximation space or by changing the semantical meaning of the logical connectives or the first-order quantifiers, as it was demonstrated in the second half of the article.

Acknowledgements

Many thanks to the anonymous referees for their valuable suggestions.

References

[1] Chakraborty, M.K., Banerjee, M., "Rough logic with rough quantifiers," Bull. Polish Acad. Sc. (Math.), **41**, 4, pp. 305–315, 1993
[2] Banerjee, M., Chakraborty, M.K.: "Rough logics: a survey with further directions," in: *Incomplete Information: Rough Set Analysis*, Orşowska, E. (ed.), Physica, Springer-Verlag, Heidelberg, pp. 579–600, 1998
[3] Yao, Y.Y.: "On generalizing rough set theory," *Proceedings of the 9th International Conference Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing* (RSFDGrC 2003). LNAI 2639, Springer-Verlag, pp. 44–51, 2003
[4] Yao, Y.Y.: "Information granulation and rough set approximation," *International Journal of Intelligent Systems* **16** (1), pp. 87–104, 2001
[5] Kádek, T., Mihálydeák, T.: "Some Fundamental Laws of Partial First-Order Logic Based on Set Approximations," in *Rough Sets and Current Trends in Computing*, Lecture Notes in Computer Science Volume 8536, Cornelis, C., Kryszkiewicz, M., Slezak, D., Menasalvas Ruiz, E., Bello, R., Shang, L. (eds.) 9th International Conference, RSCTC 2014, Granada and Madrid, Spain, pp. 47–58, 2014
[6] van Rooij, R.: "The propositional and relational syllogistic," *Logique et Analyse*, **55**, pp. 85–108, 2012
[7] Düntsch, I., Gediga, G.: "Approximation operators in qualitative data analysis," in *Theory and Applications of Relational Structures as Knowledge Instruments*. Volume 2929 of Lecture Notes in Computer Science, de Swart, H.C.M., Orlowska, E., Schmidt, G., Roubens, M., (eds.) Springer, pp. 214–230, 2003
[8] Csajbók, Z., Mihálydeák, T.: "Fuzziness in Partial Approximation Framework," in *Annals of Computer Science and Information Systems*, Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki (eds.) Volume 1: Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS). pp. 35–41, 2013
[9] Maienborn C., Heusinger K., Portner P. (eds.): *Semantics*, **3**, 2012, Walter de Gruyter GmbH, Berlin/Boston, ISBN 978-3-11-025337-5
[10] Mihálydeák, T.: "Partial first-order logic relying on optimistic, pessimistic and average partial membership functions," in *Proceedings of the 8th conference of the European Society for Fuzzy Logic and Technology* Pasi, G., Montero, J., Ciucci, D. (eds.) (EUSFLAT 2013), 2013 ISBN (on-line): 978-90786-77-78-9 doi:10.2991/eusflat.2013.53

[3]Our approach differs from the idea presented in [10] which talks about pessimistic, optimistic, and average membership functions.