INFORMATICA, 2014, Vol. 25, No. 4, 541–550

DOI: http://dx.doi.org/10.15388/Informatica.2014.28

Decidability of Logic of Correlated Knowledge

Haroldas GIEDRA*, Jūratė SAKALAUSKAITĖ, Romas ALONDERIS

Institute of Mathematics and Informatics, Vilnius University
Akademijos 4, LT-08663 Vilnius, Lithuania
e-mail: haroldas.giedra@gmail.com, jurate.sakalauskaite@mii.vu.lt, romas.alonderis@mii.vu.lt.

Received: January 2014; accepted: December 2014

Abstract. Terminating procedure GS-LCK-PROC of the proof search in the sequent calculus GS-LCK of logic of correlated knowledge is presented in this paper. Also decidability of logic of correlated knowledge is proved, where GS-LCK-PROC is a decision procedure.

Key words: logic of correlated knowledge, decidability, decision procedure, proof system, sequent calculus.

Introduction

Logic of correlated knowledge is an epistemic logic enriched by observational capabilities of agents. Traditionally, agents can make a logical inference, positive and negative introspection and their knowledge is truthful. Applications of the epistemic logic cover fields such as distributed systems, merging of knowledge bases, robotics or network security in computer science and artificial intelligence. By adding observational capabilities to agents, logic of correlated knowledge can be applied, in addition, to reason about multipartite quantum systems and quantum correlations.

Quantum entanglement posed a problem to the lattice-theoretical approach of traditional Quantum Logic (Aerts, 1981; Valckenborgh, 2001). Logic of correlated knowledge (LCK) abstracts away from Hilbert spaces and suggests to accomodate correlation models to quantum systems and quantum entanglement. Alexandru Baltag and Sonja Smets introduced logic of correlated knowledge and Hilbert style proof system in Baltag and Smets (2010). Our main focus is to present the automated terminating proof search procedure GS-LCK-PROC for logic of correlated knowledge and to prove decidability of LCK in this paper.

We start from defining syntax and semantics of logic of correlated knowledge in Section 1. In Section 2, we introduce Gentzen style sequent calculus GS-LCK. Terminating proof search procedure GS-LCK-PROC is presented and decidability of logic of correlated knowledge is proved in Section 3.

*Corresponding author.

1. Logic of Correlated Knowledge

1.1. *Syntax*

Consider a set $N = \{a_1, a_2, \ldots, a_n\}$ of agents. Each agent can perform its local observations. Given sets $O_{a_1}, \ldots, O_{a_n}$ of possible observations for each agent, a joint observation is a tuple of observations $o = (o_a)_{a \in N} \in O_{a_1} \times \cdots \times O_{a_n}$ or $o = (o_a)_{a \in I} \in O_I$, where $O_I := \times_{a \in I} O_a$ and $I \subseteq N$. Joint observations together with results $r \in R$ make new atomic formulas o^r.

Each agent can know some information, and it is written as $K_{a_1} A$ or $K_{\{a_1\}} A$, which means that the agent a_1 knows A. A group of agents can also know some information and it is written as $K_{\{a_1, a_2, a_3\}} A$ or $K_I A$, where $I = \{a_1, a_2, a_3\}$. A more detailed description about the knowledge operator K is given in Fagin *et al*. (1992), van der Hoek and Meyer (1997).

Syntax of logic of correlated knowledge is defined as follows:

Definition 1. (Syntax of logic of correlated knowledge) The language of logic of correlated knowledge has the following syntax:

$$F := p \mid o^r \mid \neg F \mid F \vee F \mid F \wedge F \mid F \to F \mid K_I F$$

where p is any atomic proposition, $o = (o_a)_{a \in I} \in O_I$, $r \in R$, and $I \subseteq N$.

1.2. *Semantics*

Consider a system, composed of N components or locations. Agents can be associated to locations, where they will perform observations. States (configurations) of the system are functions $s : O_{a_1} \times \cdots \times O_{a_n} \to R$ or $s_I : O_I \to R$, where $I \subseteq N$ and a set of results R is in the structure (R, Σ) together with an abstract operation $\Sigma : \mathcal{P}(R) \to R$ of composing results. $\mathcal{P}(R)$ is a power set of R. For every joint observation $e \in O_I$, the local state s_I is defined as:

$$s_I\big((e_a)_{a \in I}\big) := \Sigma\big\{s(o) : o \in O_{a_1} \times \cdots \times O_{a_n} \text{ such that } o_a = e_a \text{ for all } a \in I\big\}$$

If s and t are two possible states of the system and a group of agents I can make exactly the same observations in these two states, then these states are observationally equivalent to I, and it is written as $s \overset{I}{\sim} t$. Observational equivalence is defined as follows:

Definition 2. (Observational equivalence) Two states s and t are observationally equivalent $s \overset{I}{\sim} t$ iff $s_I = t_I$.

A model of logic of correlated knowledge is a multi-modal Kripke model (Kripke, 1963), where the relations between states mean observational equivalence. It is defined as:

DEFINITION 3. (Model of logic of correlated knowledge) For a set of states S, a family of binary relations $\{\overset{I}{\sim}\}_{I \subseteq N} \subseteq S \times S$ and a function of interpretations $V : S \to (P \to \{true, false\})$, where P is a set of atomic propositions, the model of logic of correlated knowledge is a multi-modal Kripke model $(S, \{\overset{I}{\sim}\}_{I \subseteq N}, V)$ that satisfies the following conditions:

1. for each $I \subseteq N$, $\overset{I}{\sim}$ is a multi-modal equivalence relation;
2. information is monotonic: if $I \subseteq J$, then $\overset{J}{\sim} \subseteq \overset{I}{\sim}$;
3. observability principle: if $s \overset{N}{\sim} s'$, then $s = s'$;
4. vacuous information: $s \overset{\emptyset}{\sim} s'$ for all $s, s' \in S$.

The satisfaction relation $\models$ for model M, state s and formulas o^r and $K_I A$ is defined as follows:

- $M, s \models K_I A$ iff $M, t \models A$ for all states $t \overset{I}{\sim} s$.
- $M, s \models o^r$ iff $s_I(o) = r$.

The formula $K_I A$ means that the group of agents I carries the information that A is the case, and o^r means that r is the result of the joint observation o.

If formula A is true in any state of any model, then it is named as a valid formula.

2. Gentzen Style Sequent Calculus GS-LCK

Gerhard Gentzen introduced sequent calculus in Gentzen (1934). Sequents in the system GS-LCK are statements of the form $\Gamma \Rightarrow \Delta$, where Γ and Δ are finite, possibly empty multisets of relational atoms $s \overset{I}{\sim} t$ and labeled formulas $s : A$, where $s, t \in S$, $I \subseteq N$ and A is any formula in the language of logic of correlated knowledge. The formula $s : A$ means $s \models A$, and $s \overset{I}{\sim} t$ is an observational equivalence or relation between the states in the model of logic of correlated knowledge.

The sequent calculus consists of axioms and rules. Applying rules to the sequents, a proof-search tree for the root sequent is constructed. If axioms are in all the leaves of the proof-search tree, then the root sequent is called as a provable sequent and Δ follows from Γ of the root sequent.

Fixing a finite set $N = \{a_1, \dots, a_n\}$ of agents, a finite result structure (R, Σ) and a tuple of finite sets $\boldsymbol{O} = (O_{a_1}, \dots, O_{a_n})$ of observations, for every set $I, J \subseteq N$, every joint observation $o \in O_I$, $O_I = \times_{a \in I} O_a$, and results $r, p \in R$, the Gentzen style sequent calculus GS-LCK for logic of correlated knowledge over $(R, \Sigma, \boldsymbol{O})$ is as follows:

- Axioms:
  - $s : p, \Gamma \Rightarrow \Delta, s : p$.
  - $s : o^r, \Gamma \Rightarrow \Delta, s : o^r$.
  - $s : o^{r_1}, s : o^{r_2}, \Gamma \Rightarrow \Delta$, where $r_1 \neq r_2$.

- Propositional rules:

$$\frac{\Gamma \Rightarrow \Delta, s:A}{s:\neg A, \Gamma \Rightarrow \Delta}\ (\neg \Rightarrow) \qquad \frac{s:A, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, s:\neg A}\ (\Rightarrow \neg)$$

$$\frac{s:A, \Gamma \Rightarrow \Delta \quad s:B, \Gamma \Rightarrow \Delta}{s:A \vee B, \Gamma \Rightarrow \Delta}\ (\vee \Rightarrow) \qquad \frac{\Gamma \Rightarrow \Delta, s:A, s:B}{\Gamma \Rightarrow \Delta, s:A \vee B}\ (\Rightarrow \vee)$$

$$\frac{s:A, s:B, \Gamma \Rightarrow \Delta}{s:A \wedge B, \Gamma \Rightarrow \Delta}\ (\wedge \Rightarrow) \qquad \frac{\Gamma \Rightarrow \Delta, s:A \quad \Gamma \Rightarrow \Delta, s:B}{\Gamma \Rightarrow \Delta, s:A \wedge B}\ (\Rightarrow \wedge)$$

$$\frac{\Gamma \Rightarrow \Delta, s:A \quad s:B, \Gamma \Rightarrow \Delta}{s:A \to B, \Gamma \Rightarrow \Delta}\ (\to\Rightarrow) \qquad \frac{s:A, \Gamma \Rightarrow \Delta, s:B}{\Gamma \Rightarrow \Delta, s:A \to B}\ (\Rightarrow\to)$$

- Knowledge rules:

$$\frac{t:A, s:K_I A, s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}{s:K_I A, s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}\ (K_I \Rightarrow) \qquad \frac{s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta, t:A}{\Gamma \Rightarrow \Delta, s:K_I A}\ (\Rightarrow K_I)$$

The rule $(K_I \Rightarrow)$ requires that $I \neq N$ and $t:A$ be not in Γ. The rule $(\Rightarrow K_I)$ requires that $I \neq N$ and t be not in the conclusion. Set I maybe an empty set in both rules.

$$\frac{s:A, s:K_N A, s \overset{N}{\sim} s, \Gamma \Rightarrow \Delta}{s:K_N A, s \overset{N}{\sim} s, \Gamma \Rightarrow \Delta}\ (K_N \Rightarrow) \qquad \frac{s \overset{N}{\sim} s, \Gamma \Rightarrow \Delta, s:A}{\Gamma \Rightarrow \Delta, s:K_N A}\ (\Rightarrow\ K_N)$$

The rule $(K_N \Rightarrow)$ requires that $s:A$ be not in Γ. The rule $(\Rightarrow K_N)$ requires that $s:A$ be not in Δ.

- Observational rules:

$$\frac{s \overset{I}{\sim} t, \{s:o^{r_o}\}_{o \in O_I}, \{t:o^{r_o}\}_{o \in O_I}, \Gamma \Rightarrow \Delta}{\{s:o^{r_o}\}_{o \in O_I}, \{t:o^{r_o}\}_{o \in O_I}, \Gamma \Rightarrow \Delta}\ (OE)$$

The rule (OE) requires that $I \neq \emptyset$ and formulas $s \overset{I}{\sim} t$, $s:o^{r_o}$ and $t:o^{r_o}$ be not in Γ, where $o \in O_I$.

$$\frac{\{s:o_I^r, \Gamma \Rightarrow \Delta\}_{r \in R}}{\Gamma \Rightarrow \Delta}\ (OYR)$$

The rule (OYR) requires:

1. $s:o_I^r$ be not in Γ for all $r \in R$ and $s:o_I^{r_1}$ be in Δ for some $r_1 \in R$.
2. $I \neq \emptyset$.

$$\frac{s:e_I^{\Sigma\{r_{o_N}:o_N \in \bar{e}\}}, \{s:o_N^{r_{o_N}}\}_{o_N \in \bar{e}}, \Gamma \Rightarrow \Delta}{\{s:o_N^{r_{o_N}}\}_{o_N \in \bar{e}}, \Gamma \Rightarrow \Delta}\ (CR)$$

The rule (CR) requires that $s:e_I^{\Sigma\{r_{o_N}:o_N \in \bar{e}\}}$ be not in Γ.

- Substitution rules:

$$\frac{s : p, t : p, s \overset{N}{\sim} t, \Gamma \Rightarrow \Delta}{t : p, s \overset{N}{\sim} t, \Gamma \Rightarrow \Delta}\ (Sub(p) \Rightarrow)$$

$$\frac{s : o^r, t : o^r, s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}{t : o^r, s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}\ (Sub(o^r) \Rightarrow)$$

The rules $(Sub(p) \Rightarrow)$ and $(Sub(o^r) \Rightarrow)$ require that $s : p$ and $s : o^r$ be not in Γ, accordingly.

- Relational rules:

$$\frac{s \overset{I}{\sim} s, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\ (Ref) \qquad \frac{s \overset{I}{\sim} t, s \overset{I}{\sim} s', s' \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}{s \overset{I}{\sim} s', s' \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}\ (Trans)$$

The rule (*Ref*) requires that s be in the conclusion and $s \overset{I}{\sim} s$ be not in Γ. The rule (*Trans*) requires that $s \overset{I}{\sim} t$ be not in Γ.

$$\frac{s' \overset{I}{\sim} t, s \overset{I}{\sim} s', s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}{s \overset{I}{\sim} s', s \overset{I}{\sim} t, \Gamma \Rightarrow \Delta}\ (Eucl) \qquad \frac{s \overset{I}{\sim} t, s \overset{J}{\sim} t, \Gamma \Rightarrow \Delta}{s \overset{J}{\sim} t, \Gamma \Rightarrow \Delta}\ (Mon)$$

The rule (*Mon*) stands for monotonicity and requires that $I \subseteq J$. Sets I, J may be empty. The rules (*Eucl*) and (*Mon*) require that $s' \overset{I}{\sim} t$ and $s \overset{I}{\sim} t$ be not in Γ, accordingly.

The sequent calculus GS-LCK is sound and complete with respect to correlation models over $(R, \Sigma, \boldsymbol{O})$ (Giedra and Sakalauskaitė, 2011). If a sequent is provable in GS-LCK, then the formula of a sequent is valid. Also, all valid formulas are provable in GS-LCK, which expresses the completeness of the system.

The sequent calculus GS-LCK also allows to build a procedure, which is a decision procedure for LCK. Decidability of logic of correlated knowledge is proved in the next section.

3. Decidability of Logic of Correlated Knowledge

Decidability of logic of correlated knowledge is showed by first defining the terminating proof search procedure for LCK. Procedure uses tables TableLK and *TableRK* to save principal formulas of the applications of the rules $(K_I \Rightarrow)$, $(K_N \Rightarrow)$ and $(\Rightarrow K_I)$. Also chains of new appeared relational atoms of applications of the rule $(\Rightarrow K_I)$ are saved in table *TableRK*.

Definition 4. (Table TableLK) Table TableLK of the principal pairs of the applications of the rules $(K_I \Rightarrow)$ and $(K_N \Rightarrow)$:

TableLK	
Main formula	Relational atom

Example 1. Example of TableLK:

TableLK	
Main formula	Relational atom
$s : K_I A$	$s \overset{I}{\sim} t$
$l : K_I B$	$l \overset{I}{\sim} z$

Definition 5. (Table TableRK) Table TableRK of the principal formulas and chains of new appeared relational atoms of the applications of the rule ($\Rightarrow K_I$):

TableRK			
Main formula	Chain of the relational atoms	Length of chain	Max

where *Max* is the maximum length of the chain, defined by $n(K_I) + 1$. Formula $n(K_I)$ denotes the number of negative occurrences of knowledge operator K_I in a sequent.

Example 2. Example of TableRK:

TableRK			
Main formula	Chain of the relational atoms	Length of chain	Max
$s, s_1, s_2, w_1 : K_I A$	$s \overset{I}{\sim} s_1, s_1 \overset{I}{\sim} s_2, s_2 \overset{I}{\sim} s_3$	3	5
	$s \overset{I}{\sim} t_1$	1	5
	$s \overset{I}{\sim} w_1, w_1 \overset{I}{\sim} w_2$	2	5
$z, z_1 : K_J B$	$z \overset{J}{\sim} z_1, z_1 \overset{J}{\sim} z_2$	2	7

Definition 6. (Procedure of the proof search) Procedure GS-LCK-PROC of the proof search in the sequent calculus GS-LCK:
Initialization:

– Define set N of agents, tuple of sets $\boldsymbol{O} = (O_{a_1}, \dots, O_{a_n})$ of possible observations and result structure (R, Σ).

– Initialize the tables *TableLK* and *TableRK* by setting Max values to $(n(K_I) + 1)$, the length of the chain to 0 and the other cells leaving empty.
– Set Output = False.

PROCEDURE GS-LCK-PROC(Sequent, TableLK, TableRK, Output)
BEGIN

1. Check if the sequent is the axiom. If the sequent is the axiom, set *Output* = *True* and go to step Finish.
2. If possible, apply any of the rules $(\neg \Rightarrow)$, $(\Rightarrow \neg)$, $(\Rightarrow \vee)$, $(\wedge \Rightarrow)$, $(\Rightarrow\rightarrow)$ and go to step 1.
3. If possible, apply any of the rules $(\vee \Rightarrow)$, $(\Rightarrow \wedge)$ or $(\rightarrow\Rightarrow)$ and call procedure GS-LCK-PROC() for the premises of the application:

   Output1 = False;
   Output2 = False;

   GS-LCK-PROC(Premise1, TableLK, TableRK, Output1);
   GS-LCK-PROC(Premise2, TableLK, TableRK, Output2);

   IF (Output1 == True) AND (Output2 == True)
   THEN Set Output = True and go to Finish;
   ELSE Set Output = False and go to Finish;

4. If possible to apply any of the rules $(K_I \Rightarrow)$ or $(K_N \Rightarrow)$, check if the principal pair is absent in the table TableLK. If it is absent, apply rule $(K_I \Rightarrow)$ or $(K_N \Rightarrow)$, add principal pair to TableLK and go to step 1.
5. If possible to apply rule $(\Rightarrow K_I)$, check if the principal formula is absent in the table TableRK and the length of the chain is lower than Max. If the principal formula is absent and the length of the chain is lower than Max, apply rule $(\Rightarrow K_I)$, add principal formula and new relational atom to TableRK, increment the length of the chain by 1, and go to step 1.
6. If possible, apply rule (*OYR*) and call procedure GS-LCK-PROC() for the premises of the application:

   For each k set Output(k) = False and call GS-LCK-PROC(Premise(k), TableLK, TableRK, Output(k)), where k is the index of the premise;

   IF (for each k Output(k) == True)
   THEN Set Output = True and go to Finish;
   ELSE Set Output = False and go to Finish;

7. If possible, apply any of the rules $(\Rightarrow K_N)$, (OE), (CR), $(Sub(p) \Rightarrow)$, $(Sub(o^r) \Rightarrow)$, (Ref), $(Trans)$, $(Eucl)$ or (Mon) and go to step 1.

8. Finish.

END

Procedure GS-LCK-PROC gets the sequent, *TableLK*, *TableRK*, starting *Output* and returns “True”, if the sequent is provable. Otherwise – “False”, if it is not provable. Procedure is constructed in such a way, that it produces proofs, where number of applications of the knowledge rules of sequent calculus GS-LCK is finite. Also number of applications of other rules are bounded by requirements to rules and finite initial sets of agents, observations and results, which allows procedure to perform terminating proof search.

Lemma 1. (Permutation of the rule $(K_I \Rightarrow)$) *Rule* $(K_I \Rightarrow)$ *permutes down with respect to all rules of GS-LCK, except rules* $(\Rightarrow K_I)$ *and* (OE). *Rule* $(K_I \Rightarrow)$ *permutes down with rules* $(\Rightarrow K_I)$ *and* (OE) *in case the principal atom of* $(K_I \Rightarrow)$ *is not active in it.*

Proof. The Lemma 1 is proved in the same way as the Lemma 6.3. in Negri (2005). □

Lemma 2. (Number of applications of the rule $(K_I \Rightarrow)$) *If a sequent* S *is provable in GS-LCK, then there exists the proof of* S *such that rule* $(K_I \Rightarrow)$ *is applied no more than once on the same pair of principal formulas on any branch.*

Proof. The Lemma 2 is proved by induction on the number N of pairs of applications of rule $(K_I \Rightarrow)$ on the same branch with the same principal pair.

$\langle N = 0\rangle$ The proof of the lemma is obtained.

$\langle N > 0\rangle$.

We diminish the inductive parameter in the same way as in the proof of Corollary 6.5. in Negri (2005), using Lemma 1. □

Lemma 3. (Number of applications of the rule $(\Rightarrow K_I)$) *If a sequent* S *is provable in GS-LCK, then there exists the proof of* S *such that for each formula* $s : K_I A$ *in its positive part there are at most* $n(K_I)$ *applications of* $(\Rightarrow K_I)$ *iterated on a chain of accessible worlds* $s \overset{I}{\sim} s_1, s_1 \overset{I}{\sim} s_2, \ldots$, *with principal formula* $s_i : K_I A$. *The latter proof is called regular.*

Proof. The Lemma 3 is proved by induction on the number N of series of applications of rule $(\Rightarrow K_I)$, which make the initial proof non-regular.

$\langle N = 0\rangle$ The proof of the lemma is obtained.

$\langle N > 0\rangle$

We diminish the inductive parameter in the same way as in the proof of Proposition 6.9 in Negri (2005). □

Theorem 1. (Termination of GS-LCK-PROC) *The procedure GS-LCK-PROC performs terminating proof search for each formula over* $(R, \Sigma, \boldsymbol{O})$.

Proof. From construction of the procedure GS-LCK-PROC follows that the number of applications of the rules $(K_I \Rightarrow)$ and $(\Rightarrow K_I)$ is finite.

All the propositional rules reduce the complexity of the root sequent. Since the sets N, (R, Σ), $\boldsymbol{O}$ and the number of applications of the rules $(K_I \Rightarrow)$, $(\Rightarrow K_I)$ are finite, and the requirements are imposed on the rules, the number of applications of the rules $(K_N \Rightarrow)$, $(\Rightarrow K_N)$, (OE), (OYR), (CR), $(Sub(p) \Rightarrow)$, $(Sub(o^r) \Rightarrow)$, (Ref), $(Trans)$, $(Eucl)$ and (Mon) is also finite.

According to finite number of applications of all rules, the procedure GS-LCK-PROC performs the terminating proof search for any sequent. □

Theorem 2. (Soundness and completeness of GS-LCK-PROC) *The procedure GS-LCK-PROC is sound and complete over* $(R, \Sigma, \boldsymbol{O})$.

Proof. From construction of the procedure GS-LCK-PROC follows that if procedure returns "True" for a sequent S, then S is provable in GS-LCK. If procedure returns "False", then sequent S is not provable in GS-LCK, according to Lemma 2 and Lemma 3. □

Theorem 3. (Decidibility of LCK) *Logic LCK is decidable.*

Proof. From Theorem 2 and Theorem 1 follows that GS-LCK-PROC is a decision procedure for logic LCK. □

Conclusions

Procedure GS-LCK-PROC performs terminating proof search for logic of correlated knowledge. Also it is a decision procedure for LCK, which allows us always to determine if the sequent is provable or not provable. If the sequent is provable, we get that the formula of the sequent is valid. Using this tool, knowledge can be analyzed and inferences can be checked if they follow from some knowledge base.

References

Aerts, D. 1981. Description of compound physical systems and logical interaction of physical systems. *Current Issues on Quantum Logic*, 8, 381–405.

Baltag, A., Smets, S. 2010. Correlated knowledge: an epistemic-logic view on quantum entanglement. *International Journal of Theoretical Physics*, 49(12), 3005–3021.

Gentzen, G. 1934. Untersuchungen uber das logische Schliesen, I. *Mathematische Zeitschrift*, 39(2), 176–210.

Giedra, H., Sakalauskaitė, J. 2011. Sequent calculus for logic of correlated knowledge. *Lithuanian Mathematical Journal*, 52 (spec. issue), 243–248.

Fagin, R., Halpern, J.Y., Vardi, M.Y. 1992. What can machines know? On the properties of knowledge in distributed systems. *Journal of the ACM*, 39(2), 328–376.

Kripke, S. 1963. Semantical analysis of modal logic, I. Normal propositional calculi. Zeitschrift fur mathematische Logik und Grundlagen der Mathematik, 9, 67–96.

Negri, S. 2005. Proof analysis in modal logic. *Journal of Philosophical Logic*, 34(5), 507–544.

van der Hoek, W., Meyer, J.-J.Ch. 1997. A complete epistemic logic for multiple agents-combining distributed and common knowledge. *Epistemic Logic and the Theory of Games and Decisions*, 35–68.

Valckenborgh, F. 2001. *Compound systems in quantum axiomatics*. PhD thesis, Vrije Universiteit Brussel.

H. Giedra is a PhD student in informatics at the Institute of Mathematics and Informatics of Vilnius University. His research interest include quantum modal logics and proof theory.

J. Sakalauskaitė received her PhD degree from Institute of Mathematics of Sanct Petersburg branch, Russia in 1986. She is a researcher of Institute of Mathematics and Informatics of Vilnius University.Her research interests include proof theory of non-classical logics such as modal, temporal and dynamic ones.

R. Alonderis is a researcher at Mathematical Logic Sector at Software Engineering Department of Institute of Mathematics and Informatics of Vilnius University. His research interest is proof theory of non-classical logics.

Koreliatyvių žinių logikos išsprendžiamumas

Haroldas GIEDRA, Jūratė SAKALAUSKAITĖ, Romas ALONDERIS

Straipsnyje pateikiama koreliatyvių žinių logikos įrodymų paieškos sekvenciniame skaičiavime GS-LCK baigtinė procedūra GS-LCK-PROC. Taip pat įrodomas koreliatyvių žinių logikos išsprendžiamumas. Naudojant GS-LCK-PROC procedūrą, visoms koreliatyvių žinių logikos formulėms galima patikrinti, ar formulė yra tapačiai teisinga.