Global Journal of Engineering and Technology [GJET] ISSN: 2583-3359
1(1) Page 18-19

Complexity Classes Modal Logic (CCML)

Received : 19/03/2022 Accepted : 22/03/2022 Published : 24/03/2022

Iaroslav Petik

senior scientific researcher in museum Of Outstanding Figures in Ukrainian Culture

Summary
This paper proposes and provides a technical description of a new modal logic used for the description of the relation between classes of algorithmic complexity in computer science - complexity classes modal logic (CCML).

Keywords: algorithm, complexity problem, possible worlds, Turing Machine.

Introduction. Syntax and semantics.

The purpose of this short paper is to introduce a new propositional modal logic – complexity classes' modal logic (CCML). CCML mostly follows the structure of the classical propositional logic and K4 for modal modifications.
Next axioms within CCML are assumed:
Basic axioms of simple propositional logic (presented as by Jan Łukasiewicz) [2]:
$p \to (q \to p)$
$p \to (q \to r) \to ((p \to q) \to (p \to r))$
$(\neg p \to \neg q) \to (q \to p)$

However, the truth functions for semantics are different. CCML must provide the formalization of statements about relations between classes of algorithmic complexity. So we have the next truth functions for the semantics:
V(p) = 1, iff set of symbols A is computable;
V(¬p) = 1, iff set of symbols A is not computable;
V (p (and) q) = 1 iff sets of symbols A and B are computable;
V (p v q) = 1 iff either the set of symbols A or set of symbols B is computable;
V (p -> q) = 1 iff computability of set of symbols A entails computability of set of symbols B.
V (□ p) = 1, iff V(p) = 1 for all v, for which the relation wRv is true (where w – central world, and v – any other world, R – accessibility relations that relates worlds).

Modality operator brings all sorts of questions but they will be answered in the modality chapter.

The most interesting part here is the ruler for the implication and its consequences. It is evident that different computable objects and facts about their properties entail other facts about these objects and their properties. So the implication is needed. But is it adequate for all the cases of facts entailing facts? You need to check every single special case to answer that which is impossible.

This problem is about the adequacy of the logical model for this particular domain of application. In other words, it is a question of whether CCML is consistent. Simple propositional logic is consistent, it is a well-known proven result. CCML's semantics is a more narrow language so its consistency is entailed by the previous proving for the classical system.

There may be questions about the definition of sets A, B, and so on. Particular propositions refer to different particular computable objects as classical propositional logic refers to statements of the natural language.

Definition for formal languages

We need to define formally, what computable structures we are talking about actually are. It does not matter for the logic as it operates with statements about facts but it is useful for understanding the bigger picture. Definitions are taken from a textbook on computability [1].
Turing machine M is a tuple hΣ, Γ, Q, δi, where Σ, Γ, Q are finite nonempty sets with $\Sigma \subseteq \Gamma$ and $b \in \Gamma - \Sigma$. The state set Q contains three special states q0, q accept, q reject. The transition function δ satisfies:
$\delta : (Q - \{qaccept, qreject\}) \times \Gamma \to Q \times \Gamma \times \{-1, 1\}$

∑ is a finite alphabet with at least two symbols. ∑* is a set of finite strings over ∑. Language *L* is a subset of ∑*. M is a Turing machine. For each string *w* in *L,* there is a computation on M with input *w*. M accepts *w* if the computation ends in an accepting state and rejects it if the

computation ends in a rejecting state. The language L accepted by M is denoted as:
L(M) = {*w* ∈ ∑* | M accepts w}
The definition of checking relation R is:
L(R) = {w#y | R(w, y)}
Where R is a binary checking relation and # is not in the alphabet.

Modality

The special modal logic CCML (complexity classes modal logic) is defined:
V(p) = 1, iff w ∈ L(M) ;
V(¬p) = 1, iff not w ∈ L(M);
V (p (and) q) = 1 iff w ∈ L(M) and w' ∈ L(M);
V (p v q) = 1 iff w ∈ L(M) or w' ∈ L(M);

V (p -> q) = 1 iff w ∈ L(M) entails w' ∈ L(M);

Special rules for modality. Proposition p is in scenario (possible world) *v* if *w* ∈ L(M) and L(M) ∈ complexity class A. According to the definition of the modal operator:

V (□ p) = 1, iff V(p) = 1 for all v, for which vRv' (where v – central world, v' – any other world, R – accessibility relation, which bounds the worlds).

R is a binary relation W x W where W is a set of possible worlds. If there is an R-relation between v and v', v' is accessible from world v.

CCML corresponds to the classical modal logic K4 as was mentioned in the beginning. The proof for its completeness and consistency are the same.

Modal axioms for the K4 system [3]:
N, Necessitation Rule: If p is a theorem then □p is likewise a theorem.
K, Distribution Axiom: □(p → q) → (□p → □q).
4, □p → □□p

Completeness and consistence.

The completeness of the system was stated earlier. It is evident that different logical systems can be used for the same domain. It does not mean some of them are worth more than others, though of course there may be advantages and disadvantages for every particular case. Adequacy issues can be reduced to these advantages and disadvantages.

Consistence results are the same as in the classical logic. A little bit more interesting is to talk about decidability. If the statement is undecidable it is probably a potential interesting statement not only for logic but first of all for the computational complexity theory.

Conclusion

CCML is interesting both for logic and complexity theory. It is a well-known fact some of the most complicated problems like "P vs. NP" cannot be solved this way but maybe some other minor problems can be. Apart from that this system is a development of formal languages and models in general

References

1. Arora Sanjeev, Barak Boaz. (2007). *Computational Complexity: A modern approach*. Internet-draft. 473 p.
2. Łukasiewicz Jan. (1970). *Selected works*. North-Holland Pub. Co. 405 p.
3. Sider Theodore. (2010). *Logic for Philosophy*. Oxford University Press. 304 p.