2014

# TR-2014002: Proof Complexity and Quantitative Epistemology

Natalia Novak

# Proof Complexity and Quantitative Epistemology

Natalia Novak
Department of Mathematics and Computer Science
Bronx Community College,
City University of New York
Bronx, NY 10043
natalia.novak@bcc.cuny.edu

January 9, 2014

**Abstract**
There is a famous epistemic Surprise Examination Paradox (SEP) which was studied for at least half a century. We look at its formalization in MetaPRL logical framework in order to see how the change in its dimensions affects the automatic proof search.

## 1 Introduction

There is a famous epistemic Surprise Examination Paradox (SEP): A professor tells students in his class that there will be a surprise in-class exam during the next week. There are 5 weekdays and class meets every day. Can the professor give an exam? We assume that professor and students are truthful and smart. By the usual backward induction argument, students figure that the exam cannot be given at all. Indeed, the exam cannot be given at the last day, since it would not be a surprise. Therefore, the exam cannot be given the day before, etc. The paradox occurs when the professor gives an exam on day two, and it is a complete surprise for the students!
In the beginning of this article we present our own take on resolving the paradox. We do not know if our suggestion was already considered by anybody, e.g. [Chow, 1998] has 12 pages of references on the subject, and it was not our goal to become experts in this area. Further on, we provide the promised formalization of the paradox in MetaPRL. In this formalization we take a conservative approach and simply prove that full set of assumptions is contradictory or that the exam cannot happen on the last day of the week.

## 2 Our take on resolving the paradox

Consider a very trustful student who despite his or her intelligence is very confused on the last day, if the test did not happen yet. He/she "knows" that professor never lies or makes mistakes, and at the same time it looks like there is no alternative option and exam has to happen on this last day. Any certain, positive or negative, answer to the question "Do you think, that the test will happen today?" will lead to a contradiction. Therefore, the natural answer is "I do not know", which implies (just as negative answer) that the student considers a possibility, that there will be no test. This implicit additional outcome resolves the paradox.

## 3 Paradox formalization in MetaPRL

We work in $\mathsf{S4}_n^{\mathsf{J}}$, its implementation and automatic prover were described in [Bryukhov, 2006]. We use different agent indices to represent different time, and different students, if needed. Knowledge relation between different moments of time for the same student have to be explicitly stated for individual formulae, as $\mathsf{S4}_n^{\mathsf{J}}$ has no built-in support for time.

$d_i$ means that the exam took place on $i$th day. For the case with one student, $\mathtt{K}_i A$ means that the student will know $A$ just before day $i$. With this setup $\mathtt{K}_1$ has sligtly special meaning, because it describes what we know up front about this puzzle.

### 3.1 2-day week

$$\begin{array}{ll} 1. & \mathsf{J}\left((d_1 \wedge \neg d_2) \vee (\neg d_1 \wedge d_2)\right) \\ 2. & \mathtt{K}_1(\mathtt{K}_2 d_1 \vee \mathtt{K}_2 \neg d_1) \\ 3. & \mathtt{K}_1 \neg \mathtt{K}_2 d_2 \\ 4. & \neg \mathtt{K}_1 d_1 \\ \hline & \bot \end{array}$$

First assumption states that exam will happen on either of the days. Second assumption states that after first day, it will be known if the exam took place already. Third assumption is effectively our formalization of "surprise" - before second day we will not know if the exam will happen on second day. And the last assumption is also "surprise", for the first day, this time. MetaPRL finds a proof for this theorem in a few seconds.

### 3.2 2-day week, many students

The same problem with 2 students is formulated trivially by duplicating relevant formulas and replacing modality indices:

$$
\begin{array}{ll}
1. & \mathsf{J}\,((d_1 \wedge \neg d_2) \vee (\neg d_1 \wedge d_2)) \\
2. & \mathtt{K}_1(\mathtt{K}_2 d_1 or \mathtt{K}_2 \neg d_1) \\
3. & \mathtt{K}_1 \neg \mathtt{K}_2 d_2 \\
4. & \neg \mathtt{K}_1 d_1 \\
5. & \mathtt{K}_2(\mathtt{K}_3 d_1 or \mathtt{K}_3 \neg d_1) \\
6. & \mathtt{K}_2 \neg \mathtt{K}_3 d_2 \\
7. & \neg \mathtt{K}_3 d_1 \\
\hline
 & \bot
\end{array}
$$

Extending this theorem for 40 students has some, but little effect on proof search timing, still keeping it under one second; it seems to be linear or polynomial of low degree. This is expected, as conclusion of the theorem is independent from the number of students, and first 4 assumptions are sufficient anyway.

### 3.3 3-day week

$$
\begin{array}{ll}
1. & \mathsf{J}\,((d_1 \wedge \neg d_2 \wedge \neg d_3) \vee (\neg d_1 \wedge d_2 \wedge \neg d_3) \vee (\neg d_1 \wedge \neg d_2 \wedge d_3)) \\
2. & \mathsf{J}(\mathtt{K}_2 d_1 \vee \mathtt{K}_2 \neg d_1) \\
3. & \mathsf{J}(\mathtt{K}_3 d_1 \vee \mathtt{K}_3 \neg d_1) \\
4. & \mathsf{J}(\mathtt{K}_3 d_2 \vee \mathtt{K}_3 \neg d_2) \\
5. & \mathtt{K}_1 \neg \mathtt{K}_2 d_2 \\
6. & \mathtt{K}_1 \neg \mathtt{K}_3 d_3 \\
7. & \mathtt{K}_2 \neg \mathtt{K}_3 d_3 \\
8. & \neg \mathtt{K}_1 d_1 \\
\hline
 & \neg d_3
\end{array}
$$

First assumption states that exam will happen on either of the days. 2nd, 3rd and 4th assumptions say that we know results of previous days. 5th to 8th assumptions are "surprise" conditions.
Note that this theorem only states the impossibility of the exam on the last day of the week and does not involve backward induction. So this is a simplified version, basically first step in establishing the paradox.
If we unfold all disjunctions, MetaPRL has no problems completing the proof. But it could not find the proof in fully automatic mode in an hour. In this sense, this theorem is harder than Muddy Children puzzle for 4 children, which can be solved in 150-250sec. The reason why fully automatic mode fails is because all J-boxed assumptions have to be used twice in the reasoning and the way prover works is that it first exhaust all proof matrices with modalities used at most once, then expands the search space by allowing each modality to have two prefix interpretations; the resulting search space is big and solution does not show up quickly. Muddy children puzzle also needs this search space expansion, but we get more luck there, it is not clear why exactly.

Muddy Children puzzle for 4 children can be sped up by an order of magnitude by manually duplicating assumptions that will be used twice in the reasoning and avoiding automatic expansion of all modalities. We tried to apply the same technique to our problem at hand but it produced too big of a search space, and the search simply did not finish within a reasonable time.
Originally, we wanted to see how $\mathsf{S4}_n^{\mathsf{J}}$-to-$\mathsf{S4}_n\mathsf{LP}$ realization procedure [Novak, 2009] performs on proofs of Surprise Exam paradox for different number of days, but the current implementation of the realization procedure supports only proofs generated fully automatically. Extention of the current implementation of this procedure to manually constructed proofs is not exactly trivial. $\mathsf{S4}_n^{\mathsf{J}}$ prover and $\mathsf{S4}_n\mathsf{LP}$ realizer communicate via a custom proof format, independent of MetaPRL proof format, hence convenient support for manually provided proofs will amount to implementing a small proof assistant just for $\mathsf{S4}_n^{\mathsf{J}}$, which falls outside of the scope of this effort.

# 4 Conclusion

Initially, our intention was to apply the $\mathsf{S4}_n\mathsf{LP}$ realization procedure to proofs of SEP with different number of days and different number of students, to estimate the empirical complexity of our implementation of the procedure.
It turned out, that it is possible to get an automatically generated proof of SEP for 2-day week, for one student and for many students cases. Changing the number of students does not affect the proof search time too much. Application of the realization algorithm took a few seconds, and proof term size does not change with the number of students. 3-day week with one student case changes the situation dramatically. Even after an hour the prover could not find a proof, and a few technical tricks did not help. At the same time, it did not take much time to find the proof interactively, by eliminating all boxed disjunctions and applying automatic prover to resulting subgoals. This does not allow us to compare the conversion time and the proof-terms length, since the current implementation of the relization procedure supports only automatically generated proofs.

# 5 Future Work

Considering the complications that were met in this project, it will be very instumental to modify or extend the current implementation of $\mathsf{S4}_n\mathsf{LP}$ realizer to accept manually generated proofs, which will allow to do the comparisons we are eager to get.

# References

[Artemov and Kuznets, 2006] Sergei Artemov and Roman Kuznets. Logical omniscience via proof complexity. Technical Report TR-2006005, Graduate Center, CUNY, 2006.

[Artemov and Nogina, 2005] Sergei Artemov and Elena Nogina. Introducing justification into epistemic logic. *J. Log. and Comput.*, 15(6):1059–1073, 2005.

[Artemov, 1995] Sergei Artemov. Operational modal logic. Technical Report MSI 95-29, Cornell University, 1995.

[Artemov, 2001] Sergei Artemov. Explicit provability and constructive semantics. *The Bulletin for Symbolic Logic*, 6(1):1–36, 2001.

[Artemov, 2004] Sergei Artemov. Evidence-based common knowledge. Technical Report TR-2004018, CUNY Ph.D. Program in Computer Science Technical Reports, November 2004.

[Artemov, 2006] Sergei Artemov. Justified common knowledge. *Theoretical Computer Science*, 357(1):4–22, 2006.

[Artemov, 2008] Sergei Artemov. Symmetric logic of proofs. 4800:58–71, 2008.

[Brezhnev and Kuznets, 2006] Vladimir Brezhnev and Roman Kuznets. Making knowledge explicit: how hard it is. *Theoretical Computer Science*, 357(1):23–34, 2006.

[Bryukhov, 2006] Yegor Bryukhov. *Integration of Decision Procedures into High-Order Interactive Provers.* PhD thesis, The Graduate School and University Center, CUNY, 2006.

[Chow, 1998] Timothy Y. Chow. The surprise examination or unexpected hanging paradox. *The American Mathematical Monthly*, 105:41–51, 1998.

[Fagin *et al.*, 1995] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge.* Massachusetts Institute of Technology, 1995.

[Fitting, 2005] Melvin Fitting. The logic of proofs, semantically. *Annals of Pure and Applied Logic*, 132(1):1–25, 2005.

[Hintikka, 1975] Jaakko Hintikka. Impossible possible worlds vindicated. *Journal of Philosophical Logic*, 4(3):475–484, 1975.

[Kuznets, 2008] Roman Kuznets. *Complexity Issues in Justificaion Logic.* PhD thesis, The Graduate School and University Center, CUNY, 2008.

[Meyer and Hoek, 1995] John-Jules Ch Meyer and Wiebe Van Der Hoek. *Epistemic Logic for AI and Computer Science*. Cambridge University Press, New York, NY, USA, 1995.

[Novak, 2009] Natalia Novak. *Computer-Aided Reasoning about Knowledge and Justifications.* PhD thesis, The Graduate School and University Center, CUNY, 2009.

[Parikh, 1987] Rohit Parikh. Knowledge and the problem of logical omniscience. In *In Methodologies for Intelligent Systems, Proceedings of the Second International Symposium*, pages 432–439. North-Holland, 1987.