

A Tough Nut for Mathematical Knowledge Management

Manfred Kerber¹ and Martin Pollet²

¹ School of Computer Science, University of Birmingham,
Birmingham B15 2TT, England

<http://www.cs.bham.ac.uk/~mmk>

² Fachbereich Informatik, Universität des Saarlandes,
66041 Saarbrücken, Germany

<http://www.ags.uni-sb.de/~pollet>

Abstract. In this contribution we address two related questions. Firstly, we want to shed light on the question how to use a representation formalism to represent a given problem. Secondly, we want to find out how different formalizations are related and in particular how it is possible to check that one formalization entails another. The latter question is a tough nut for mathematical knowledge management systems, since it amounts to the question, how a system can recognize that a solution to a problem is already available, although possibly in disguise. As our starting point we take McCarthy's 1964 mutilated checkerboard challenge problem for proof procedures and compare some of its different formalizations.

1 Introduction

Mathematical colloquial language as well as languages of formalized mathematics offer a large variety of ways to formalize a problem. If a problem is given in an informal way, the first question is, how to formalize it, that is, how to write it down. While this is already the first question in a mathematical vernacular as well, it is more acute in a formal system, be it a proof development environment, or a mathematical knowledge representation system. Even within one such formal system, it is typically possible to represent the same problem in a large variety of ways. The adequacy of a representation depends of course on its intended purpose (e.g., information retrieval, tutoring system, automated problem solving). In this contribution we focus mainly on the representation of a problem in order to find a proof. The obvious question is then, which representations are appropriate and which ones not. The choice of a good formalization depends on the formal language itself as well as the available tools and system support available for certain representations. In consequence, users of different proof development systems will use formalizations which are particularly adequate for their system. This leads to the question how we can know that two different formulations are equivalent, or in a weaker version how we can know that one formulation entails another.

We will use the so-called “mutilated checkerboard problem” to study the relationship between different problem formalizations. This problem was introduced by McCarthy as a challenge problem for proof procedures [1]. While the challenge was mastered by different proof procedures, it is still a challenge for mathematical knowledge management.

In McCarthy’s original paper [1] we can find two different formalizations. McCarthy [2] himself presented at the second QED workshop a different formalization which makes – compared to the original paper – use of set-theoretical notions and basic integer arithmetic. We will also look at the formalizations in Isabelle [3] and Coq [4].

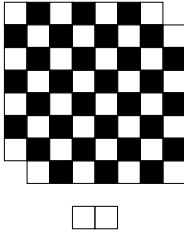
There are various proofs of the problem available. Some are close to one of McCarthy’s formalizations (e.g., [5]), others are using formalizations which are significantly different and need creative thought to understand that they are related to the original problem. Even the close formalizations of [2] and [5] need some adaptation in order to see that they can be mapped into each other. While McCarthy numbers the checkerboard from 0 to 7, Bancerek uses 1 to 8. A shift of an index in an array by 1 is a trivial re-representation to anybody with mathematical training, but it has to be recognized as an index shift. Neither do the strings directly match, nor will simple unification do, since 0 is not 1 and 7 not 8.

Why would not everybody just take McCarthy’s original formalization? Obviously, different formal systems have different strengths and limitations. The first formalization by McCarthy is a first order formalization which does not make use of equality and function symbols. While this is a very restricted formalism, some systems can deal with just this formalism only. McCarthy’s second formalism contains function symbols and equality. A reasoner which can deal with function symbols and equality will have special procedures how to do that and would probably not live up to its strengths if given the first formalization. Likewise Isabelle and Coq, which have very powerful representation languages, have strengths which these systems could not use if they had to stick to one of the original formalizations. For this reason it should be considered as legitimate that each system user chooses a representation which suits their system best. However, when we take a closer look at the formalizations, it is neither trivial to see that they solve the original problem nor to see that they can be identified or subsumed by one another.

While problems such as a mutilated checkerboard problem exhibit a big gap between their informal description in natural language and their various formal representations, the problem is universal in mathematics. Typically even in a single very strict formal system it is possible to say the same thing in a wide variety of ways and not for all of these possibilities it is obvious that they are equivalent on some level. The reason why one formalization entails another may involve a simple syntactic modification, or may rest on a deep semantic connection.

2 The Mutilated Checkerboard Problem(s)

Let us first introduce the problem and proof from McCarthy’s original memo [1, p.1]:



“It is impossible to cover the mutilated checkerboard shown in the figure with dominoes like the one in the figure. Namely, a domino covers a square of each color, but there are 30 black squares and 32 white squares to be covered.”

While the original challenge of the problem was the size of the search space in various formalizations – and still is for a naive usage – the much bigger challenge for proof procedures is that a good proof contains a creative invention, namely the colouring of the arrays of the board and the domino so that the squares with different colours can be counted. For the solution we can differentiate between the following phases:

1. Formalization of the problem.
2. Formalization of a concept representing the creative invention.
3. Realization of the proof on the basis of the creative invention.

The formulation of step 2 depends on the problem description given in step 1, and that the proof idea consists of both step 2 and step 3.

At each step there are several possibilities, colouring the arrays is only one of several creative inventions. Most publications which have taken up the challenge start with the motivation that their problem formalization is adequate with respect to the informal description and concentrate on steps 2 and 3.

In this contribution, we want to investigate the different possibilities and choices which can be made in step 1. The mathematical problem formalization assumes certain background knowledge that is available to the reader. Only with this background knowledge it is possible to understand why the problem formalization actually covers the original problem. The main objects, notions, and properties used (explicitly or implicitly) in the proofs are the following.

- Board: a rectangular structure containing squares. The squares are oriented in vertical/horizontal lines.
- Domino: a domino can be associated with two adjacent squares, either vertically or horizontally.
- Adjacent: for a square the square to the left, to the right, below, or on top.
- Sets: for collecting dominoes.
- Covering: all squares of the board are associated with exactly one domino.
- Numbers: numbers can be used to identify the different squares of the board, and to express the structure in which the squares are related to each other.
- Pairs and Cartesian product: pairs which contain numbers as representation of squares, Cartesian product as board.

Later, the creative invention of colouring corresponds to mapping squares, represented by pairs of numbers, to the set $\{black, white\}$. Sets can be counted, mapped to numbers, so we have cardinality and relations for the comparison of cardinality.

2.1 Challenge Problems by McCarthy

McCarthy gives already two problem formalizations in his initial paper [1] and a set-theoretic description thirty years later [2]. Whereas the early formalizations describe the problem completely, the latter builds on existing concepts of set theory, including cardinality, and integer arithmetic.

M64a

The language is a predicate logic without function symbols and without equality. The signature consists of constants $1, \dots, 8$, and the binary relations given here with their intended meaning.

$S(x, y)$	$y = x + 1$
$L(x, y)$	$x < y$
$E(x, y)$	$x = y$
$G^1(x, y), \dots, G^4(x, y)$	square (x, y) and the top/right/bottom/left neighbour square are covered by a domino
$G^5(x, y)$	square (x, y) is uncovered

The problem is stated in form of unsatisfiable axioms.

- | | |
|---|---|
| <ol style="list-style-type: none"> 1. $S(1, 2) \wedge S(2, 3) \wedge S(3, 4) \wedge S(4, 5)$
 $\quad \wedge S(5, 6) \wedge S(6, 7) \wedge S(7, 8)$ 2. $S(x, y) \Rightarrow L(x, y)$ 3. $L(x, y) \wedge L(y, z) \Rightarrow L(x, z) \wedge \neg S(x, z)$ 4. $L(x, y) \Rightarrow \neg E(x, y)$ 5. $E(x, x)$ 6. $G^1(x, y) \vee G^2(x, y) \vee G^3(x, y) \vee G^4(x, y) \vee G^5(x, y)$ 7. $G^1(x, y) \Rightarrow \neg(G^2(x, y) \vee G^3(x, y) \vee G^4(x, y) \vee G^5(x, y))$ 8. $G^2(x, y) \Rightarrow \neg(G^3(x, y) \vee G^4(x, y) \vee G^5(x, y))$ 9. $G^3(x, y) \Rightarrow \neg(G^4(x, y) \vee G^5(x, y))$ 10. $G^4(x, y) \Rightarrow \neg G^5(x, y)$ 11. $G^5(1, 1) \wedge G^5(8, 8)$ 12. $G^5(x, y) \Rightarrow (E(1, x) \wedge E(1, y)) \vee (E(8, x) \wedge E(8, y))$ 13. $S(x_1, x_2) \Rightarrow (G^1(x_1, y) \Leftrightarrow G^3(x_2, y))$ 14. $S(y_1, y_2) \Rightarrow (G^2(x, y_1) \Leftrightarrow G^4(x, y_2))$ 15. $\neg G^3(1, y) \wedge \neg G^1(8, y) \wedge \neg G^2(x, 8) \wedge \neg G^4(x, 1)$ | $\left. \begin{array}{l} \right\} \text{properties of numbers} \\ \left. \begin{array}{l} \right\} \text{placement} \\ \left. \begin{array}{l} \right\} \text{of dominoes} \\ \left. \begin{array}{l} \right\} \text{uncovered squares} \\ \left. \begin{array}{l} \right\} \text{adjacency of dominoes} \\ \left. \begin{array}{l} \right\} \text{border of the board} \end{array} \right\} \end{array} \right\} \end{array} \right\}$ |
|---|---|

M64b

The second problem is formalized in a predicate logic with function symbols and equality. The five predicates G^1, \dots, G^5 are represented by the values of the function g . The problem is again given in form of axioms.

- | | |
|---|---|
| <ol style="list-style-type: none"> 1'. $s(s(s(s(s(s(s(8))))))) = 8$ 2'. $\neg s(s(s(s(x)))) = x$ 3'. $g(x, y) = 5 \Leftrightarrow x = 8 \wedge y = 8 \vee x = 1 \wedge y = 1$ 4'. $g(x, y) = 1 \Leftrightarrow g(s(x), y) = 3$ 5'. $g(x, y) = 2 \Leftrightarrow g(x, s(y)) = 4$ | $\left. \begin{array}{l} \right\} \text{eight distinct numbers} \\ \left. \begin{array}{l} \right\} \text{uncovered squares} \\ \left. \begin{array}{l} \right\} \text{adjacency of dominoes} \end{array} \right\}$ |
|---|---|

- 6'. $g(1, y) \neq 3 \wedge g(8, y) \neq 1 \wedge g(x, 1) \neq 4 \wedge g(x, 8) \neq 2$ } border of the board
 7'. $1 = s(8) \wedge 2 = s(1) \wedge 3 = s(2) \wedge 4 = s(3) \wedge 5 = s(4)$ } names for numbers
 8'. $g(x, y) = 1 \vee g(x, y) = 2 \vee g(x, y) = 3 \vee g(x, y) = 4 \vee g(x, y) = 5$ } covering

M95

The language is predicate calculus and expects a formalization of the set theoretical concepts, like operations on sets, Cartesian product, ordered pairs, and cardinality. Furthermore the formalization uses operations on integers including the absolute value function. The additional concepts are introduced by definitions, the problem is stated as a formula to be proved.

Definitions

$$\begin{aligned}
 \text{Board} &= \mathbb{Z}_8 \times \mathbb{Z}_8 \\
 \text{mutilated-board} &= \text{Board} \setminus \{(0, 0), (7, 7)\} \\
 \text{domino-on-board}(x) &\Leftrightarrow (x \subset \text{Board}) \wedge \text{card}(x) = 2 \\
 &\quad \wedge (\forall x_1 x_2)(x_1 \neq x_2 \wedge x_1 \in x \wedge x_2 \in x \\
 &\quad \Rightarrow \text{adjacent}(x_1, x_2)) \\
 &\Leftrightarrow (x \subset \text{Board}) \wedge \text{card}(x) = 2 \\
 &\quad \wedge (\forall x_1 x_2)(x = \{x_1, x_2\} \Rightarrow \text{adjacent}(x_1, x_2)) \\
 \text{adjacent}(x_1, x_2) &\Leftrightarrow |c(x_1, 1) - c(x_2, 1)| = 1 \wedge c(x_1, 2) = c(x_2, 2) \\
 &\quad \vee |c(x_1, 2) - c(x_2, 2)| = 1 \wedge c(x_1, 1) = c(x_2, 1) \\
 &\Leftrightarrow |c(x_1, 1) - c(x_2, 1)| + |c(x_1, 2) - c(x_2, 2)| = 1 \\
 c((x, y), 1) &= x \\
 c((x, y), 2) &= y \\
 \text{partial-covering}(z) &\Leftrightarrow (\forall x)(x \in z \Rightarrow \text{domino-on-board}(x)) \\
 &\quad \wedge (\forall x y)(x \in z \wedge y \in z \Rightarrow x = y \vee x \cap y = \{\})
 \end{aligned}$$

Theorem

$$\neg(\exists z)(\text{partial-covering}(z) \wedge \bigcup z = \text{mutilated-board})$$

Note that McCarthy defines *domino-on-board* and *adjacent* in two equivalent ways. Which definition to prefer depends on the context of its usage.

2.2 Formalization with Inductive Definitions

Paulson presented his formalization and proof of the checkerboard problem in the Isabelle system [3].

P96

The language is Isabelle/HOL which allows inductive definitions and supports reasoning with them. The formalization uses notions from integer arithmetic and set theory.

Definitions

$$\begin{aligned}
 \text{lessThan}(m) &:= \{i \in \mathbb{N} \mid i < m\} \\
 \text{board} &:= \text{lessThan}(2 \cdot s(m)) \times \text{lessThan}(2 \cdot s(n))
 \end{aligned}$$

$$\begin{aligned} \text{tiling}(A) : \text{set}(\text{set}(\alpha)) &:= \{ \} \in \text{tiling}(A) \wedge (a \in A \wedge t \in \text{tiling}(a) \wedge a \cap t = \{ \}) \\ &\Rightarrow a \cup t \in \text{tiling}(a) \end{aligned}$$

$$\begin{aligned} \text{domino} : \text{set}(\text{set}(\mathbb{N} \times \mathbb{N})) &:= \{ (i, j), (i, s(j)) \} \in \text{domino} \wedge \\ &\{ (i, j), (s(i), j) \} \in \text{domino} \end{aligned}$$

Theorem

$$(\text{board} \setminus \{(0, 0)\}) \setminus \{(s(2 \cdot m), s(2 \cdot n))\} \notin \text{tiling}(\text{domino})$$

2.3 Formalization in Second Order Logic

Huet [4] formulates the problem on a level which is more abstract than all the other formalizations. The formalization is based on properties such as injectivity and surjectivity of functions, and finiteness (characterized on the basis of injectivity and surjectivity). The geometry of the board is not considered at all, numbers or cardinals are not necessary for the argument.

H96

The theorem is formalized in the Coq system, but needs only the expressiveness of second order logic. Given a signature of sets B and W , two functions $\text{Board} : B \rightarrow W$ and $\text{Domino} : W \rightarrow B$ representing the board and the dominoes, the existence of a tiling for the full checkerboard problem is stated as:

$$\text{injective}(\text{Board}) \wedge \text{injective}(\text{Domino}) \wedge \text{finite}(B) \Rightarrow \text{surjective}(\text{Domino}).$$

We can generate from the full checkerboard a mutilated one by taking B' as a proper subset of B . The theorem is then that for an injective function Board' with $\text{Board}' : B \rightarrow W$ and $\text{finite}(B')$ there is no function $\text{Domino} : W \rightarrow B'$ which is injective (i.e., a – possibly partial – covering) and surjective (i.e., total).

2.4 A Very Abstract Formalization

Let us add here another formalization which is even more abstract than Huet's and just reasons about the cardinality of sets (finite or infinite).

KP05

Two sets B and W cannot at the same time have the same cardinality and a different cardinality, that is, not $|B| = |W|$ and $|B| \neq |W|$.

More concretely we can say, if $|B| < |W|$ (that is, we have strictly fewer black than white elements) then we cannot have $|W| = |B|$ (that is, equal numbers which would follow from a covering with 2×1 -dominoes).

3 Relationships Between Formalizations

Different systems make use of different problem representations. Fig. 1 gives an overview (which is not complete with respect to problem representations as well as solutions). The initial formalization *M64a* was verified by the model generator Mace, only the symbols L and E together with the corresponding axioms were replaced by built-in concepts [6]. The proof in the TPS system uses exactly

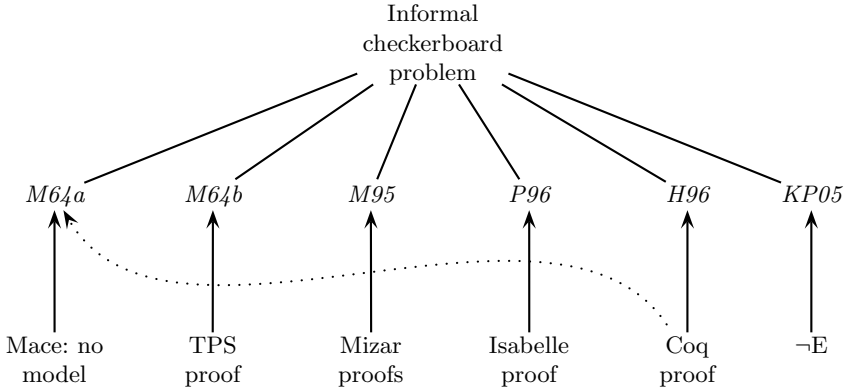


Fig. 1. Overview of formalizations of the checkerboard problem and proofs

the formalization given as $M64a$ [7]. There are two proofs in the Mizar system for $M95$ [5,8]. Paulson and Huet constructed proofs for their formalizations in Isabelle [3] and Coq [4], respectively. $KP05$ can be proved in almost any formal system by calling the corresponding calculus level rule.

Every given solution is a formal proof for the corresponding problem representation and thus for the informal representation of the checkerboard problem. While it is difficult to be more rigorous with respect to the informal statement – and probably impossible to be fully rigorous – we can ask whether the Coq proof is a solution for the formalization given as $M64a$ (represented as a dotted line in Fig. 1) and ask for a formal justification. In extreme cases (relationship between $M64a$ and $KP05$), such a formal justification may be as difficult as a proof of the one problem (here $M64a$) from first principles. We will come back to the implications of this observation for mathematical knowledge management, but first we look at concrete problem transformations.

3.1 Problem Formalizations and Their Generalizability

Each formalization can be looked at from different perspectives. $M64a$ is very concrete and its proof captures exactly the original problem and is difficult to generalize. $M64b$ is slightly more general (also capturing tori). $M95$ although concrete in the formulation, allows for an easy generalization of the argument.

$P96$ and $H96$ are much more general and go beyond the concrete 8×8 -mutilated checkerboard. In Paulson’s formulation the problem is shown for any $2m \times 2n$ rectangular structure in which two opposite corners are missing, while Huet’s formulation entails that arbitrary finite sets of black and white squares with an injective mapping from black to white sets, which is not surjective, cannot allow for an injective mapping from the white to the black squares. This proof is very abstract and many different aspects of the problem are irrelevant: the relative relationship of black and white squares, how many fewer black squares than white squares we have, and the shape of the dominoes. Important is only

that the set is finite and that a domino covers exactly one black and one white square (actually the problem representation does not speak of coverings, so this is yet another interpretation).

KP05 is even more general than *H96*. We speak only about cardinality of sets and argue that the cardinality cannot be the same and different at the same time. We do not require anymore that the sets we speak about are finite. If we cover, for instance, pairs (a, b) in which the a s are rational numbers and the b s real numbers so that each rational number is covered exactly once by an a then the b s cannot cover the real numbers since there are more real numbers than rational numbers.

Even more general than *KP05* is the statement *true*, which is logically equivalent to all true statements.

3.2 Proofs and Formalizations

If we revisit the different representations then the last one, *true*, does not require any proof. It is *trivially* satisfied, but it does not help us anything in understanding the problem.¹ Let us imagine the situation of a teacher and a student and the teacher wants to convince the student that the mutilated checkerboard cannot be covered by 2×1 -dominoes. She may say “This is trivial.” to which the student may answer either “Yes, it is trivial, I learnt to know this problem a while back already.” or “No, I don’t believe it.”

In the latter case she would need to give him a better argument (based on *KP05*) such as “Look at a partition of the checkerboard into two disjoint sets so that there are arguments that the two sets must have the same number of elements and at the same time must not have the same number of elements.” Again the student may now believe the statement and accept this as a proof, or not.

If the previous argument is not accepted *H96* can further refine it, and so on. Each formulation will get closer to the representation of a concrete checkerboard on the one hand. On the other hand, the proof of why each statement holds becomes increasingly more difficult.

3.3 Translations Between Different Formalizations

A more formal way to look at the relationship between the different formalizations is to see how one can be translated into another. We will discuss now some of these translations to a varying degree of depth.

M64a → *M64b*

Let us assume now that the predicative formulation *M64a* is given and we want to generate from this a functional formulation of the type *M64b*. To this end we would need to map different objects of the first signature to the second.

¹ Richard Feynman states only half jokingly that “mathematicians can prove only trivial theorems, because every theorem that’s proved is trivial.” [9, p.84].

$M64a$	\mapsto	$M64b$	
$S(x, y)$	\mapsto	$s(x) = y$	(1)
$1, 2, \dots, 8$	\mapsto	$1, s(1), \dots, s(s(s(s(s(s(1))))))$	(2)
$G^i(x, y)$	\mapsto	$g(x, y) = i$ for $i \in \{1, \dots, 5\}$	(3)
$E(x, y)$	\mapsto	$x = y$	(4)

Why is this translation justified? (1) means that we can translate the binary predicate symbol S into a unary function symbol s . In order to justify such a translation we need to show that S not only represents a general relation, but a left total and right unique one. Let us first observe that it suffices to look at the numbers 1 through 8. S is right unique for these numbers since from the axioms 1.–5. follows that the usual relationships between these numbers hold, for instance, $1 < 3$ from $S(1, 2)$ and $S(2, 3)$ with the second and third axiom. With the third axiom follows also $\neg S(1, 3)$. Likewise $\neg S(1, 4)$ and so on, that is, S is right unique.

The left totality is a more subtle issue. While it is intentionally clear that we have to consider only the eight different numbers $1, 2, \dots, 8$, the formalization $M64a$ does not warrant that only eight different numbers exist. More seriously, we get from the axioms 1.–5. as a consequence that for all $x \in \{1, 2, \dots, 8\}$ holds $\neg S(8, x)$.² The argument why S can be translated to a function requires either to extend the range to all natural numbers or to give up the idea of a strict order and to consider a cyclic structure in which $s(8) = 1$.

Both approaches are formally possible and amount to the same covering problem, assumed we never make use of $s(8)$. Axiom $6'$ states that dominoes do not stick out, that is, that only the dominoes in a range from 1 through 8 are to be considered. The first approach means we view the mutilated checkerboard as a subset of the infinite plane. When putting dominoes down, we never cross the boundaries. The second approach means that we consider the checkerboard as a torus, axiom $6'$ puts up boundaries between the 8th row and the 1st row (and the columns likewise) which we may not cross in covering the checkerboard. Strictly speaking, axiom $6'$ is not necessary to prove the theorem, since the mutilated torus, that is, the full torus without the images of the two deleted corners of the checkerboard, cannot be covered by dominoes either, even if it is possible to cross the invisible boundaries, since the mutilated torus contains two white squares more than black ones.

In $M64b$ McCarthy chooses the second view (of a torus). The transition from $M64a$ to $M64b$ cannot be syntactically established at a low level (but only almost), since the original binary predicate symbol S is not left-total. This does not matter, however, since $s(8)$ has never to be used. Formally the second approach has the advantage that we have only finitely many (eight) different elements in the Herbrand universe, and hence has a finite search space, while the other formalization has a potentially infinite search space. Depending on which procedure one uses, it is advantageous to be more restrictive and to include $6'$ (for instance, when using Mace to show that no model exists), or to allow coverings of the torus which cross the magic boundary and have no correspondence of cov-

² If we had $S(8, x)$ for some x we would get with the second axiom $L(8, x)$. Together with $L(x, 8)$ (which follows from axioms 1.–3. for $x \neq 8$, and from $L(8, x)$ for $x = 8$) we get $L(8, 8)$, hence $\neg E(8, 8)$, which contradicts $E(x, x)$.

erings in a real checkerboard with 2×1 -dominoes and exclude $6'$ (for instance, when proving the theorem with TPS).

As replacing the predicate symbol S by a function symbol s , it is possible to replace the predicate symbols G^i (with $i = 1, \dots, 5$) by a function symbol g so that $G^i(x, y)$ goes to $g(x, y) = i$. Note that initially the G^i are different names for binary predicate symbols (and could have been called P, Q, R, S, T instead). Since the names do not matter we can use a single ternary predicate symbol G instead which takes the index i as an argument, that is, $G(x, y, i)$ instead³ of $G^i(x, y)$. $G(x, y, i)$ corresponds for a left-total and right-unique G to $g(x, y) = i$. The left-totality and right-uniqueness of the $G(x, y, i)$ follows from axioms 6.–10.

Axioms 11 and 12 mean that precisely the squares (1,1) and (8,8) are missing. They are translated as $g(1, 1) = 5 \wedge g(8, 8) = 5$ and $g(x, y) = 5 \rightarrow (E(1, x) \wedge E(1, y)) \vee (E(8, x) \wedge E(8, y))$ which is with the translation of $E(x, y)$ to $x = y$ logically equivalent to axiom $3'$.

Axioms 13 and 14 can be translated directly to $4'$ and $5'$ (with the simplification to rename x_1 to x , and get rid of x_2 by replacing it by $s(x)$). Axiom 6 goes to $8'$, and 15 to $6'$.

Note that the translation from S to s and G to g leads to proof obligations, namely that the functions s and g , used to represent the relations S and G are well-defined. While this should be a straightforward syntactical proof and actually is for G , matters are more subtle with S , and intuitively need an argument why a mutilated checkerboard cannot be covered when a mutilated toroid checkerboard cannot. The translation of E to equality strictly also leads to proof obligations that the usual properties of reflexivity, symmetry, transitivity, and substitutivity are satisfied. Only the first of the four is actually formally given (in axiom 5), since only this one is needed.

M64b \longrightarrow M64a

The reverse transformation is much simpler. Functional expressions such as $f(x_1, \dots, x_n) = y$ are translated to predicative ones such as $F(x_1, \dots, x_n, y)$. The additional axioms for the left-totality and right-uniqueness of F may be added. Nested functional expressions such as $g(s(x), y) = 3$ require the introduction of new variables as in $S(x_1, x_2) \rightarrow G(x_2, y, 3)$.

M95 \longrightarrow H96

Let us see now how the solution of Huet formalized as $H96$ can be applied to the problem formalization given by McCarthy as $M95$. We start again to relate the objects in the different formalizations.

<u>M95</u>	\mapsto	<u>H96</u>	
$\{(x, y) \mid a \in \text{Board} \wedge 0 = (x + y) \pmod 2\}$	\mapsto	B	(1)
$\{(x, y) \mid a \in \text{Board} \wedge 1 = (x + y) \pmod 2\}$	\mapsto	W	(2)
Board	\mapsto	$\text{Board}_H : B \rightarrow W$	(3)
set D with	\mapsto	$\text{Domino} : W \rightarrow B$	(4)
<u>$\text{partial-covering}(D) \wedge W \subset \bigcup D$</u>			

³ Note also that the order of the arguments does not matter, provided any re-ordering is done consistently throughout the whole problem description.

None of the objects in the formalization in *H96* exists in *M95*. The sets *B* and *W* have to be constructed. The mapping representing the board $Board_H$ has to be defined using the set *Board* and the sets corresponding to *B* and *W*. The same holds for the mapping *Domino*, which additionally needs the set of dominoes covering all white squares.

In terms of the three steps describing the process of solving the problem given in Sec. 2, the formalization *M95* is the problem formalization of step 1, the proof given for *H96* corresponds to the realization of the proof in step 3, and the transformation which identifies objects in *M95* with objects in *H96* is the creative invention of step 2.

For the mutilated board of *M95* the set $B' = B \setminus \{(0, 0), (7, 7)\}$ corresponds to the restriction of $Board_H : B \rightarrow W$ to $Board'_H : B' \rightarrow W$, and the theorem of *M95* expressed with respect to the notions in *H96* translates then to $\neg(\exists Domino : W \rightarrow B' \text{ injective}(Domino))$.

M95 \longleftrightarrow *P96*

The correspondence between the objects in *M95* and *P95* is more direct. In both formalizations the board is represented by the Cartesian product of the integer interval $[0, 2n - 1]$, the squares of the board are elements of the Cartesian product, and dominoes are sets containing exactly two squares which are adjacent.⁴

We see here that even when there is an agreement how to represent the objects, there can be a difference how properties of the objects are represented. For example, the predicate *partial-covering* in *M95* is defined for the given board, and includes the definition of adjacent dominoes. The predicate *tiling* in *P96* defines the covering independently from dominoes. Here we see one dimension in the choice for a formalization: the generality of the introduced concepts.

In *M95* the general concept of tiling is expressed by the formula $tiling'(z) \Leftrightarrow (\forall x y)(x \in z \wedge y \in z \Rightarrow x = y \vee x \cap y = \{\})$ which is equivalent to *tiling* in *P96*. Even for equivalent concepts there is a choice in the formalization. This choice may depend on the assistance provided by the different proof systems. McCarthy supposes a ‘heavy-duty’ set theory prover, whereas the choice of Paulson is motivated by the support for inductive definitions available in Isabelle.

M64b \longrightarrow *M95*

When we compare the problem formalizations *M64b* and *M95*, then we find that the latter contains explicit objects to model the board, the squares of the board as pairs of coordinates, and the dominoes as sets with exactly two pairs. In *M64b* we find a representation which formalizes the different situations for each square (x, y) of the board as values of the function $g(x, y)$. Abstract concepts, like dominoes, size of the board, and covering are expressed as restrictions on the values of g and thus are given only indirectly.

⁴ Although *M95* speaks about a concrete 8×8 checkerboard, the argument is as general as for the more general formalization of *P96*.

KP05 \longrightarrow H96

The most basic argument in *KP05* is very simple. If we have two sets, then we cannot have $|B| = |W|$ and $|B| \neq |W|$. The slightly more expanded argument is: If we have two sets with $|B| < |W|$ then we cannot have $|W| \leq |B|$ (which is a slightly stronger version than the original formulation, but maps much better to Huet’s argument).

How can we apply this statement to the case of *H96*? We need to know that the conditions are given, that is, we assume an injective but not surjective mapping $Board : B \rightarrow W$, and the finiteness of B . The injectivity entails $|B| \leq |W|$. The non-surjectivity entails together with the finiteness of B , $|B| \neq |W|$. The two facts together give us $|B| < |W|$. Hence *KP05* is applicable and we can conclude $\neg |W| \leq |B|$. Hence there is no injective mapping $Domino : W \rightarrow B$.

3.4 Soundness Considerations

The relationship between different formulations can in some cases be easily seen on the basis of a syntactic transformation. In other cases it may be necessary to make use of more complicated reasoning. Consider, for instance the relationship between *M64a* and *M64b*. The first describes the mutilated checkerboard as a subset of the plane, while the second describes it as a mutilated toroid. The reasoning why the one relationship entails the other is subtle.

In general we can compare the process of formalization of a problem to a projection from the *problem per se* to different formalizations as in Fig. 2. Since the *problem per se* is not given in a formal way, it is difficult to reason about the correctness of a formalization. However, it should always be possible the establish the relationship between different formalizations such as **F1** and **F2** in Fig. 2.

The relationship between different formalizations should follow the commutative diagram in Fig. 3. Note that the relationship may be non-trivial since the different formalizations may use different formal systems. A proof for the original problem $\Gamma \models \varphi$ will be obtained by transforming the assumptions Γ to Γ' from which a tranformed theorem φ' is derived in a potentially different calculus. Then φ' is translated back to the original theorem φ . Assumed that the translations and the calculus π are sound, then the overall construction is a sound argument why φ follows from Γ .

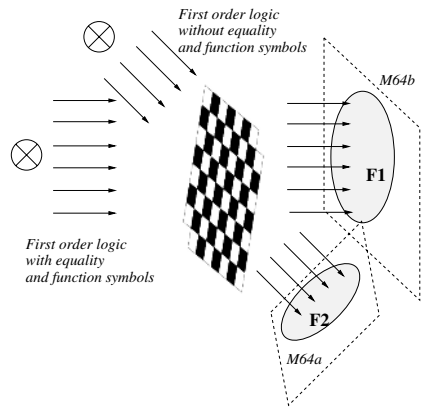


Fig. 2. Representation as projection

$$\begin{array}{ccc}
 \Gamma & \vdash_{\pi_{\varphi \circ \pi \circ \pi_{\Gamma}}} & \varphi \\
 \Downarrow \pi & & \downarrow \pi_{\varphi} \\
 \Gamma' & \vdash_{\pi} & \varphi'
 \end{array}$$

Fig. 3. Commutative diagram for proving by reformulation

Note that typically one formalization represents not only a single problem, but as discussed in Sec. 3.1 it represents a whole class which can be considered as the inverse image of different projections as displayed in Fig. 4.⁵

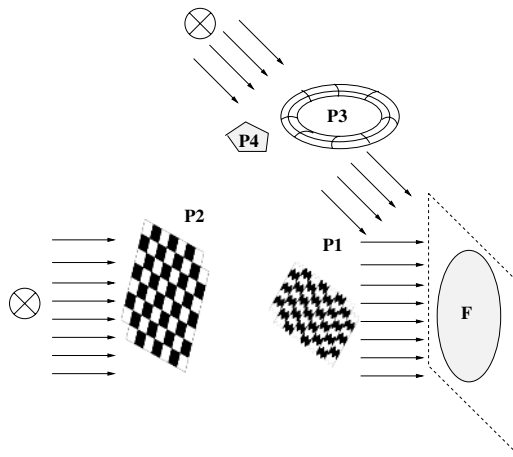


Fig. 4. Generalizability as inverse projection

4 Conclusion

It is a standard task when applying mathematical or (more stringently) formal methods to problems that we have to formalize them within a system. It is indeed an important aspect in the application of formal systems in safety critical areas, since any proof can start only once the formalization is given. If different people formalize the same problem they will typically come up with different formulations. Assumed the two formulations cannot be mapped to each other in a sound way, then at least one does not capture the original problem adequately.

⁵ One of the reviewers pointed out the relationship to Benjamin Whorf’s hypothesis that languages constrain thought. We do not want to go into the deep philosophical discussion surrounding this hypothesis in general. However, it seems suggestive that not only languages, but more so formulations within languages may facilitate creative reasoning and generalizations, or not. While the same problem can be formalized in various ways, it matters significantly for problem solving which formalization is used.

In order to study the process of relating different formulations to each other and to understand how to represent one example we have looked at an old problem, McCarthy's mutilated checkerboard problem. This problem is interesting since it has been represented and proved in a large variety of systems. Of course, any representation has to abstract from unimportant details, such as the size of a square on the checkerboard and its material. Other details may or may not be represented, for instance, whether we speak of a checkerboard of size 8×8 , or arbitrary size $2n \times 2n$, or arbitrary finite size, or arbitrary size. Perhaps astonishingly, this easy problem exemplifies a wide range of problems involved.

Different formalizations are used all the time, and certain standard translations, such as different versions of relativization in sorted logic, representations of equality in a logic without equality, reification of higher order expressions in first order logic, or functional versus predicative formalizations, are used quite regularly when representing problems formally. Often certain relationships between different formalizations are proved by a meta-logical argument (e.g., the sort theorem for the relativization).

There is also a very general mechanism how different representations can be linked to each other. Farmer et al. [10,11] have introduced a theory mapping in their little theories approach. One representation can be mapped faithfully to another if the axioms of the one become theorems in the other in a systematic translation.

Two major challenges follow from the mutilated checkerboard problem for mathematical knowledge management. Obviously the problem is not restricted to this particular problem, but a general one.

Firstly, if a system is told that *H96* represents a proof for the mutilated checkerboard problem, how can it automatically check that this is so indeed. Secondly – even more difficult – if a system is given a problem such as the mutilated checkerboard problem, how can it find relevant information which may be given in forms so diverse as *M64a*, *M64b*, *M95*, *P96*, *H96*, or *KP05*.

The second problem may be eased by proper annotations. The first in its full generality can be arbitrarily hard. While retrieving and relating information given in different shapes seems a core activity not only in mathematical knowledge processing, the ways in which mathematical information is transformed is very rich. A good representation is one that is simple and at the appropriate level so that it captures the main ideas of the proof.

What is appropriate is, however, not independent of the actual user. An argument about cardinalities, for instance, requires at least some basic knowledge of the concepts by the user. Huet's elegant proof remains obscure to somebody who does not know anything about the relationships between injective and surjective mappings on finite sets of the same cardinality, and so on.

As Ayer [12, p.85f] puts it "A being whose intellect was infinitely powerful would take no interest in logic and mathematics. For he would be able to see at a glance everything that his definitions implied, and, accordingly could never learn anything from logical inference which he was not fully conscious of already." Everything is trivial for such a being. Experts may do with some easy arguments

on a high level, while beginners need low-level lengthy arguments at a low level. The relationships between the different levels are often syntactic, but can be complicated in detail.

A truly helpful system would not only find relevant information but also present it to a user on an appropriate level. A good understanding of typical transformations is an important step toward such a system.

References

1. McCarthy, J.: A tough nut for proof procedures (1964) Stanford Artificial Intelligence Project Memo No. 16, 1964. Available from <http://www-formal.stanford.edu/jmc/>.
2. McCarthy, J.: The mutilated checkerboard in set theory. [13] 25–26. Available from <http://www.mcs.anl.gov/qed/index.html>.
3. Paulson, L.C.: A simple formalization and proof for the mutilated chess board. *Logic Journal of the IGPL* **9** (2001) 475–485. Also published as Technical Report Computer Laboratory, University of Cambridge, 394, May 1996.
4. Huet, G.: The mutilated checkerboard (Coq library) (1996) <http://coq.inria.fr/contribs/checker.html>
5. Bancerek, G.: The mutilated chessboard problem – checked by Mizar. [13] 37–38 Available from <http://www.mcs.anl.gov/qed/index.html>.
6. McCune, W.: Another crack in a tough nut. *Association for Automated Reasoning Newsletter* **31** (1995) 1–3
7. Andrews, P.B., Bishop, M.: On sets, types, fixed points, and checkerboards. In Miglioli, P., Moscato, U., Mundici, D., Ornaghi, M., eds.: *Theorem Proving with Analytic Tableaux and Related Methods*. 5th International Workshop. (TABLEAUX '96), Terrasini, Italy, Springer Verlag, LNAI 1071 (1996) 1–15
8. Rudnicki, P.: The mutilated checkerboard problem in the lightweight set theory of Mizar. Technical Report TR96-09, Department of Computing Science, University of Alberta (1996)
9. Feynman, R.P.: *Surely you're joking Mr. Feynman*. Vintage, London, UK (1985)
10. Farmer, W.M., Guttman, J.D., Thayer, F.J.: *Little theories*. In Kapur, D., ed.: *Proceedings of the 11th CADE*, Saratoga Springs, New York, USA, Springer Verlag, LNAI 607 (1992) 567–581
11. Farmer, W.M.: An infrastructure for intertheory reasoning. In McAllester, D., ed.: *Proceedings of the 17th CADE*, Pittsburgh, Pennsylvania, USA, Springer Verlag, Berlin, LNAI 1831 (2000) 115–131
12. Ayer, A.J.: *Language, Truth and Logic*. 2nd edition, 1951 edn. Victor Gollancz Ltd, London, UK (1936)
13. Matuszewski, R., ed.: *The QED Workshop II*. (1995) Available from <http://www.mcs.anl.gov/qed/index.html>