

# Using Domain-Independent Problems for Introducing Formal Methods

Raymond Boute — INTEC, Ghent University

2006/08/24

## Overview

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements

Bonus: *Micro-Semantics*: a 3-line basic program semantics  
(the “crystal radio” of formal programming language semantics)

4. Using wide-scope domain-independent problems
5. Final Remarks

Bonus: language expressiveness and specification clarity: an example

## Starting topic

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements
4. Using wide-scope domain-independent problems
5. Final Remarks

# 1. Introduction: context, domain-independent problems

## 1.1 General: a gap in engineering professionalism

- **The gap** (see also: David Parnas, C. Michael Holloway etc.)
  - Classical engineering: use of mathematical models in practice is routine
  - Software “engineering” (HAH!): 3T practice prevails  
3T = Type code, Try it out, Tinker until you think it works
- **The cause of the deficiency in mathematical modelling**
  - NOT the usual conjectures (“too difficult for the average engineer” etc.)
  - BUT: insufficient coverage in curricula to provide the massive injection of qualified people in industry necessary to make a difference
- **Main impeding factor:** university staff often more traditionalist than industry (issue not further discussed here, but to be taken into account)

## 1.2 Educational aspects

- On using tools at introductory levels at universities (freshman, junior)
  - No problem for classical math (due to previous high school math basis)
  - Logic-oriented tools: “sorcerer’s apprentices” (no previous logic basis)

⇒ *The best preparation for using tools is a solid math & logic education*
- Curriculum design: structure that makes classical engineering strong
  - Provide a solid mathematical basis that is usable in all other courses
  - Use mathematical modelling intensively in all other courses  
(for enhancing *motivation* and *proficiency*)

⇒ Mathematical basis must be provided *at the start of the curriculum*

## 1.3 The role of domain-independent problems

- Providing the mathematical basis at the start of the curriculum
  - Yet: illustrations and example problems needed (facilitating abstraction)
  - No prior domain-dependent knowledge is available at that stage

⇒ necessity of domain-independent examples and illustrations
- Additional opportunities offered by domain-independent examples
  - Easily accessible (professional & recreational literature), fun to solve
  - Makes the basic courses useful to disciplines outside CS and EE
- Important new dimension emphasized in our approach
  - Usual solutions are “cleverness-oriented” (showing off end result)
  - Here: emphasis on exploratory aspects of problem solving
    - \* Translating informal statements into formal statements
    - \* Going through the intermediate steps *assisted by formality*

## Next topic

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements
4. Using wide-scope domain-independent problems
5. Final Remarks

## 2. On logic and properly formalizing informal statements

### 2.1 Starting example: two experiments as reported by Johnson-Laird

One of the following assertions is true about a particular hand of cards, and one of them is false about the same hand of cards:

If there is a king in the hand, then there is an ace in the hand.

If there isn't a king in the hand, then there is an ace in the hand.

Q: What follows?

Subjects overwhelmingly infer that there is an ace in the hand.

---

Only one of the following assertions is true about a particular hand of cards:

There is a king in the hand, or an ace, or both.

There is a queen in the hand, or an ace, or both.

There is a jack in the hand, or a ten, or both.

Q: Is it possible that there is an ace in the hand?

Nearly every participant in our experiment responded: 'yes'.

## 2.2 Explanation of the results

- Johnson-Laird's explanation (in slightly different wording)

*Insufficient awareness of logic and inadequacy of intuition*

- Alternative explanation (based on deeper probing own students)

- Inadequacy of intuition is indeed a major factor, but:  
logic awareness, even previous course in (traditional) logic, is insufficient
- “Reverse engineering” of typical interpretation of first informal statement

$$(k \Rightarrow a) \wedge (\neg k \Rightarrow a)$$

- Major cause: jumping to conclusions without reading the “specs” properly
- Remedy: discipline taking every part of informal statement into account  
Strongly supported by writing down formalization of every sentence

$$\begin{aligned}\alpha \oplus \beta &\equiv 1 \\ \alpha &\equiv k \Rightarrow a \\ \beta &\equiv \neg k \Rightarrow a\end{aligned}$$

## 2.3 Additional precision in formalizing informal statements

Recall: One of the following assertions is true about a particular hand of cards, and one of them is false about the same hand of cards

Observations

- Translation into  $\alpha \oplus \beta \equiv 1$  seems “obvious” ...  
... if one thinks in terms of propositional logic only!
- However: “One of the following ...” involves counting (i.e., beyond P.L.)  
Moreover, above translation into propositional logic is not generalizable
  - What if an informal statement says “Three of the following ...” ?
  - Also: does “One” mean “Exactly one” or “At least one” ?

Conclusion: precise translation even of seemingly “logic” problem statements requires a more expressive language than pure logic. Example (Funmath):

$$\sum (\alpha, \beta) \geq 1 \quad \wedge \quad \sum (\neg \alpha, \neg \beta) \geq 1 .$$

Since publication of the FM06 Proceedings, Johnson-Laird's web pages moved to:

```
https://weblamp.princeton.edu/~psych/psychology/research/johnson_laird
```

In particular, the location mentioned in the Proceedings

```
http://www.princeton.edu/~psych/PsychSite/fac_phil.html
```

is now

```
http://weblamp.princeton.edu/~psych/PsychSite/fac_phil.html
```

Simply replace `www` by `weblamp` .

## Next topic

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements
4. Using wide-scope domain-independent problems
5. Final Remarks

## 3. Covering intermediate phases in formalization

### 3.1 Example: problem statement far beyond pure logic

A school has 1000 students and 1000 lockers, all in a row. They all start out closed. The first student walks down the line and opens each one. The second student closes the even numbered lockers. The third student approaches every third locker and changes its state. If it was open he closes it; if it was closed he opens it. The fourth student does the same to every fourth locker, and so on through 1000 students. To illustrate, the tenth locker is opened by the first student, closed by the second, reopened by the fifth, and then closed by the tenth. All the other students pass by the tenth locker, so it winds up being closed. How many lockers are open?

Solution given alongside (Karl Dahlke, <http://www.eklhad.net/funmath.html>)

The  $n^{\text{th}}$  locker is opened or closed by student number  $k$  precisely when  $k$  divides  $n$ . So if student  $k$  changes locker  $n$ , so does student  $n/k$ . They cancel each other out. This always holds unless students  $k$  and  $n/k$  are precisely the same person. That is,  $k = n/k$ . The lockers that are exact squares will remain open. These are lockers 1, 4, 9, 16, 25, etc. How many of these are there in a row of 1000? You can go all the way up to  $31 \times 31 = 961$ , hence there are 31 lockers open.

## 3.2 Various formalizations of the informal statement and solutions

- Common initial step: tightening the informal statement (not always evident!)

A school has 1000 students and 1000 lockers in a row, all initially closed. All students walk successively along the row, and the  $k^{\text{th}}$  student inverts the state of every  $k^{\text{th}}$  locker, that is: opens it if it was closed and vice versa. How many lockers are open in the end?

- **Example:** formalization of the first step in the preceding informal reasoning

$$\begin{aligned} \textit{Answer} &= |\{n:1..N \mid \textit{Open } n\}| && (N = \text{number of lockers}) \\ \textit{Open } n &\equiv \text{Odd } |\{k:1..K \mid k \text{ divides } n\}| && (K = \text{number of students}) \end{aligned}$$

Solution amounts to calculating a more elegant expression for *Answer*.

- More essential issue: problem is stated in terms of a *procedure*  
⇒ precise formalization requires describing this procedure
- **Example:** a possible procedural description

```
for k in 1..K do
  (for n in 1..N do if (k divides n) then inv (L n) fi od) od .
```

### 3.3 Some of the many issues raised by such refinements

Recall 

```
for k in 1..K do
  (for n in 1..N do if (k divides n) then inv (L n) fi od) od .
```

#### Issues raised

- Data representation: formalizing state of lockers, definition of “inversion”  
⇒ many ways for writing `inv (L n)`, for instance, `L n := ¬(L n)`.
- Mathematical derivation of a solution
  - Formal semantics: from procedure description to equations  
⇒ formal program semantics ( `Micro-Semantics` , see next image)
  - Prior program transformation: e.g., interchanging loops in the example
  - Various possible interpretations of the informal procedure description  
Example: “every  $k$ th locker” — how does a student proceed?  
⇒ alternative for the inner loop (nicer, but destroys interchangeability)

```
for k in 1..K do (n := 0; while n + k ≤ N do n := n + k; inv (L n) od) od
```

### 3.4 Micro-Semantics: from procedure descriptions to state equations

- Purpose: usable already from the second week in a freshman-level course
- Chosen basis: formal substitution as described in the first chapter of David Gries and Fred Schneider, *A Logical Approach to Discrete Math*

Substituting expression  $d$  for variable  $v$  in expression  $e$  is written  $e[d^v]$ .

- Micro-Semantics

State: *tuple of program variables*    Command: *function on the state*

Basic commands defined directly as functions: writing  $s$  for the state,

$$\begin{aligned}(v := e) s &= s[d^v] \\ (c ; c') s &= c'(c s) \\ (\text{if } b \text{ then } c \text{ else } c' \text{ fi}) s &= b ? c s \uparrow c' s\end{aligned}$$

Other usual commands defined in terms of the basic ones, e.g.,

$\text{while } b \text{ do } c \text{ od} = \text{if } b \text{ then } (c ; \text{while } b \text{ do } c \text{ od}) \text{ fi}$

## Next topic

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements
4. Using wide-scope domain-independent problems
5. Final Remarks

## 4. Using wide-scope domain-independent problems

### 4.1 Basic observations

- The preceding example shows that even a very small problem can provide “handles” for motivating, introducing and exercising *nearly* every concept of interest in a serious introductory course on the basic mathematics underlying formal methods — even if the problem is not designed *a priori* to do so.
- Question: are there similar problems, perhaps somewhat larger, that can provide the handles for *all* concepts of interest in such a course?  
The answer is “yes”.
- Illustration: the following puzzle by Jim Propp, found in *Math Horizons*  
<http://www.maa.org/mathhorizons/volume/volume12.html> (Feb. 2005)
- Note: this puzzle is evidently *not* designed *a priori* for the stated purpose, yet one might see it initially as “contrived” due to the self-referential aspects.  
Reply: self-referential aspects are important in a serious mathematical basis.

## 4.2 Statement of Jim Propp's puzzle (excerpt)

0. The first question whose answer is B is question  
(A) 0 (B) 1 (C) 2 (D) 3 (E) 4
  1. The only two consecutive questions with identical answers are questions  
(A) 5 and 6 (B) 6 and 7 (C) 7 and 8 (D) 8 and 9 (E) 9 and 10  
...
  5. The answer to question 16 is  
(A) C (B) D (C) E (D) none of the above (E) all of the above
  6. Alphabetically, the answer to this question and the answer to the following one are  
(A) 4 apart (B) 3 apart (C) 2 apart (D) 1 apart (E) the same
  7. The number of questions whose answers are vowels is  
(A) 4 (B) 5 (C) 6 (D) 7 (E)
  8. The next question with the same answer as this one is question  
(A) 9 (B) 10 (C) 11 (D) 12 (E) 13  
...
  18. The answer to this question is:  
(A) A (B) B (C) C (D) D (E) E
  19. Standardized test is to intelligence as barometer is to  
(A) temperature (B) wind-velocity (C) latitude (D) longitude (E) all of the above
- And also the fact that the puzzle has a solution.

### 4.3 Formalization (in Funmath, largely similar to standard math)

$$a_0 = \bigwedge i: \square 5 \mid a_i = 1$$

$$a_1 = \prod (i: 5..9 \mid P i) - 5 \text{ and } \exists P_{\in 5..9} \text{ and } !P \text{ where } P := i: \square 19 . a(i+1) = a_i$$

$$a_2 = 4 \$ a$$

$$a_3 = 0 \$ a - 4$$

$$a_4 = (a_{<5})^- (a_4)$$

$$a_5 = (3, 3, 0, 1, 2) (a_{16})$$

$$a_6 = 4 - \text{abs}(a_7 - a_6)$$

$$a_7 = (0 \$ a + 4 \$ a) - 4$$

$$a_8 = \bigwedge i: \square 5 \mid a(i+9) = a_8$$

$$a_9 = (3, 0, 4, 1, 2)^- (a_{15})$$

$$a_{10} = 1 \$ a_{<10}$$

$$a_{11} = ((\text{Evn}, \text{Odd}, \text{Sqr}, \text{Prm}, \text{Mof})^\top (1 \$ a + 2 \$ a + 3 \$ a))^- 1$$

$$a_{12} = ((a_{\text{Evn}})^- 0 - 8) / 2$$

$$a_{13} = 3 \$ a - 6$$

$$a_{14} = a_{11}$$

$$a_{15} = (3, 2, 1, 0, 4)^- (a_9)$$

$$a_{16} = (3, 3, 0, 1, 2) (a_5)$$

$$a_{17} = \forall (i: 1..4. 0 \$ a \neq i \$ a) ? 4 \uparrow ((\$ a) \uparrow (1..4))^- (0 \$ a) - 1$$

$$a_{18} = a_{18} \quad \text{Question 19 is not mathematical, but asks for an opinion (say, } a_{19} = 4 \text{).}$$

## 4.4 Formal calculational derivation of the solution (excerpt)

- Reason for choosing this particular excerpt: (a) part with nonfamiliar math, (b) formalization refined since publication of proceedings.

1. The only two consecutive questions with identical answers are questions (A) 5 and 6 (B) 6 and 7 (C) 7 and 8 (D) 8 and 9 (E) 9 and 10

$a \quad 1 = \boxed{\exists (i:5..9 \mid P i) - 5}$  and  $\exists P_{\in 5..9}$  and  $!P$  where  $P := i: \square 19 . a(i+1) = a i$

- Axioms for the nonfamiliar symbols (notions unusual in “standard” math)

Choice operator  $\boxed{\phantom{x}}$  with definition  $\mathcal{R} f \neq \emptyset \Rightarrow \boxed{f \in \mathcal{R} f}$ .

Uniqueness operator  $!$  with  $!P \equiv \forall (x, y) : (\mathcal{D} P)^2 . P x \wedge P y \Rightarrow x = y$

- Calculation excerpt (out of context, just showing style and supporting claim)

[1]  $\Rightarrow$   $\langle \text{Weaken} \rangle \exists (i:5..9 . a(i+1) = a i) \wedge !i: \square 19 . a(i+1) = a i$

$\Rightarrow$   $\langle \text{Lemma} \rangle \forall i: \square 19 . i \notin 5..9 \Rightarrow a(i+1) \neq a i$

$\Rightarrow$   $\langle \text{Instantiate } i := 15 \rangle a 16 \neq a 15$

$\Rightarrow$   $\langle \text{Leibniz, } a 15 = 3 \rangle a 16 \neq 3$

$\Rightarrow$   $\langle a 16 = 1 \vee a 16 = 3 \rangle \mathbf{a 16 = 1}$

Lemma:  $!P \Rightarrow X \subseteq \mathcal{D} P \Rightarrow \exists P_{\in X} \Rightarrow \forall x: \mathcal{D} P . x \notin X \Rightarrow \neg(P x)$

## Next topic

1. Introduction: context, domain-independent problems
2. On logic and properly formalizing informal statements
3. Covering intermediate phases in formalizing informal statements
4. Using wide-scope domain-independent problems
5. Final Remarks

## 5. Final Remarks

### 5.1 General observations

- We have demonstrated and illustrated
  - The wide scope and opportunities of domain-independent problems
  - The importance of refined formalizations
  - The role of formality in making intermediate steps explicit
- Not (much) illustrated here but *extremely important* is the following  
Popular prejudice: formal approaches are not useful (e.g., cannot protect against errors) in the translation process from informal to formal statements.  
This is *wrong*; here is how formality can support this process very effectively.
  - Derive several (at least 2) formalizations, different in style and viewpoint (advantageously by different people)
  - Find calculationaly the relationship between the formalizations (equivalence, refinement, contradiction etc.)

## 5.2 Importance of formal language expressiveness

- Principle
  - Faithful (complete, refined) translation reduces risk of making errors
  - Expressive formal languages support faithful translation
- An illustration: various styles of formalization
  - Informal spec: removing adjacent duplications of symbols in a string
  - Formalization in TLA<sup>+</sup>: (Lamport) for any infinite sequence  $\sigma$ ,

$$\begin{aligned} \llbracket \sigma \rrbracket &\triangleq \text{LET } f[n \in \text{Nat}] \triangleq \text{IF } n = 0 \text{ THEN } 0 \\ &\quad \text{ELSE IF } \sigma[n] = \sigma[n - 1] \\ &\quad \quad \text{THEN } f[n - 1] \\ &\quad \quad \text{ELSE } f[n - 1] + 1 \\ S &\triangleq \{f[n] : n \in \text{Nat}\} \\ \text{IN } [n \in S \mapsto \sigma[\text{CHOOSE } i \in \text{Nat} : f[i] = n]] \end{aligned}$$

- Formalization in Funmath: for any finite or infinite sequence  $\beta$ ,  
Idea: decomposition ( $\beta = \bigcup n : \mathcal{D} \beta . n \mapsto \beta n$ ), domain point change

Result:  $\llbracket \beta \rrbracket = \bigcup n : \mathcal{D} \beta . \sum (k : \square n . \beta(k + 1) \neq \beta k) \mapsto \beta n$

## 5.3 Educational experience with this approach

- Reception by students not uniform
  - E.g., “I am not interested in puzzles, only in *real* problems and programming”
- Analysis (based on deeper probing, performance in tests)
  - When offered a choice between ‘theoretical’ and ‘application’ problems in tests, *most choose the former and performance is poorer on the latter*
    - ⇒ Expressed preference for ‘real’ problems is often a façade
  - ‘Reality content’ of problems/projects in most CS courses is illusory (problems/projects seemingly practical, but intellectually insignificant)
    - See also David Parnas, “Education for Computing Professionals”
      - ⇒ Gives students illusions and reduces motivation for real *content*

The illusion provided by such courses:

Lost weak I cud'n spel "injenear' an naw I or wun!

(See the critique/recommendations by Holloway, Page, Parnas and many others)

## [Educational experience] (continuation)

- Impact on curriculum design
    - Many curricula degrade the students' ability to cope with the 'delayed gratification' when doing math (favoring 3T of programming assignments)
    - Often 'evaluation' by students makes (young) lecturers follow demands of a small minority rather than the educational needs of the majority
- ⇒ Shall we serve the needs of the students or the rules of the bureaucrats?

THE END

Questions?