

Two weak arithmetics

Peter Smith

November 1, 2009

Our last big theorem – Theorem 6 – tells us that *if* a theory meets certain conditions, then it must be negation incomplete. And we made some initial arm-waving remarks to the effect that it *seems* plausible that we should want theories which meet those conditions. Later, we announced that there actually *is* a consistent weak arithmetic with a first-order logic which meets the conditions (in which case, stronger arithmetics will also meet the conditions); but we didn't say anything about what *such* a weak theory really looks like.

In fact, we haven't looked at *any* detailed theory of arithmetic yet! It is high time, then, that we stop operating at the extreme level of abstraction of Episodes 1 and 2, and start getting our hands dirty. This Episode introduces a couple of weak arithmetics, before we tackle the canonical first-order arithmetic PA in the following instalment. By all means skip fairly lightly over some of the more boring proof details! But it is certainly worth getting just a flavour of how these two simple formal theories work.

9 Baby Arithmetic

We start by looking at a theory BA ('Baby Arithmetic') with a very limited language L_B : because this language is so limited, the theory can in fact prove or disprove every sentence constructible in L_B .

9.1 The language L_B

The language L_B has non-logical symbols

$0, S, +, \times$

The first of these is of course a constant (intended to denote zero). The next symbol is a one-place function symbol (intended to denote the successor function). The last two symbols in the list are two-place function symbols (with the obvious standard interpretations).

A *term* τ of L_B is a referring expression built up from occurrences of '0' and applications of the function expressions 'S', '+', ' \times '. The value of a term is the number it denotes when standardly interpreted. Note that, among its terms, L_B contains in particular the standard numerals in the sense of Defn. 11. And recall, we use 'n' to represent the standard numeral $SS \dots S0$ with n occurrences of 'S': the value of 'n' is of course n . [Recall our convention: **sans serif** expressions belong to L_B , or whatever formal language is under discussion: *italic* symbols are just part of everyday mathematical English.]

The logical symbols of L_B are

$=, \neg, \wedge, \vee$

Since L_B lacks non-logical predicates, its only way of forming atomic wffs is by taking terms constructed from the non-logical symbols and putting the identity sign between them. In other words, the atomic wffs of L_B are *equations* relating terms denoting particular numbers. But we also want to be able to express *inequations*, hence the negation sign. [Note, we will abbreviate wffs of the form $\neg \tau_1 = \tau_2$ by $\tau_1 \neq \tau_2$.] Further, we might as well round things out so as to give L_B some expressively complete set of propositional connectives.

We'll also of course need brackets for scoping both the two-place functions and the two-place connectives. The syntax for constructing wffs of L_B is then exactly as you'd expect, and the semantics equally obvious. [Exercise: write out the syntactic rules for L_B carefully!]

9.2 The axioms and logic of BA

The theory BA in the language L_B comes equipped with some classical propositional deductive system to deal with the connectives (choose your favourite system!) and the usual accompanying identity rules.

Next, we want non-logical axioms governing the successor function. We want to capture the ideas that, if we start from zero and repeatedly apply the successor function, we keep on getting further numbers – i.e. different numbers have different successors: contraposing, for any m, n , if $Sm = Sn$ then $m = n$. And further, we never cycle back to zero: for any n , $0 \neq Sn$.

However, there are no quantifiers in L_B . So we can't directly express those general facts about the successor function inside the object language L_B . Rather, we have to employ *schemata* (i.e. general templates) and use the generalizing apparatus in our English metalanguage to say: *any sentence that you get from one of the following schemata by substituting standard numerals for the place-holders 'ζ', 'ξ' is an axiom.*

Schema 1. $0 \neq S\zeta$

Schema 2. $S\zeta = S\xi \rightarrow \zeta = \xi$

Next, we want non-logical axioms for addition. This time we want to capture the idea that adding zero to a number makes no difference: for any m , $m + 0 = m$. And adding a larger number Sn to m is governed by the rule: for any m, n , $m + Sn = S(m + n)$. Those two principle together tell us how to add zero to a given number m ; and then adding one is defined as the successor of the result of adding zero; and then adding two is defined as the successor of the result of adding one; and so on up – thus defining adding n for any particular natural number n .

Again, however, because of its lack of quantifiers, we can't express all that directly inside L_B . We have to resort to schemata again, and say that anything you get by substituting standard numerals for placeholders in the following is an axiom:

Schema 3. $\zeta + 0 = \zeta$

Schema 4. $\zeta + S\xi = S(\zeta + \xi)$

We can similarly pin down the multiplication function by requiring every numeral instance of the following to be an axiom too:

Schema 5. $\zeta \times 0 = 0$

Schema 6. $\zeta \times S\xi = (\zeta \times \xi) + \zeta$

Instances of Schema 5 tell us the result of multiplying by zero. Instances of Schema 6 with 'ξ' replaced by '0' define how to multiply by one in terms of multiplying by zero and then applying the already-defined addition function. Once we know about multiplying by one, we can use another instance of Schema 6 with 'ξ' replaced by 'S0' to tell us how to multiply by two (multiply by one and do some addition). And so on and so forth, thus defining multiplication for every number.

To summarize, then,

Defn. 18. BA is the theory whose language is L_B , logic is propositional logic plus standard identity rules, and whose non-logical axioms are every numerical instance of Schemas (1) to (6).

9.3 Two little proofs inside BA

We'll give two little examples of how arithmetic can be done inside BA. First, let's show that $BA \vdash 4 \neq 2$, i.e. $BA \vdash SSSS0 \neq SS0$

1.	$SSSS0 = SS0$	Supposition
2.	$SSSS0 = SS0 \rightarrow SSS0 = S0$	Axiom, instance of Schema 2
3.	$SSS0 = S0$	From 1, 2 by MP
4.	$SSS0 = SS0 \rightarrow SS0 = 0$	Axiom, instance of Schema 2
5.	$SS0 = 0$	From 3, 4 by MP
6.	$0 \neq SS0$	Axiom, instance of Schema 1
7.	Contradiction!	From 5, 6 and identity rules
8.	$SSSS0 \neq SS0$	From 1 to 7, by RAA.

And a little reflection on that illustrative proof should now convince you of this general claim:

Theorem 7. *If s and t are distinct standard numerals of L_B , then $BA \vdash s \neq t$.*

And for our second example, we'll show that $BA \vdash 2 \times 1 = 2$. In unabbreviated form, though dropping outermost brackets, we need to derive $SS0 \times S0 = SS0$.

- | | | |
|----|--|-----------------------------|
| 1. | $SS0 \times 0 = 0$ | Axiom, instance of Schema 5 |
| 2. | $SS0 \times S0 = (SS0 \times 0) + SS0$ | Axiom, instance of Schema 6 |
| 3. | $SS0 \times S0 = 0 + SS0$ | From 1, 2 by LL |

('LL' of course indicates the use of Leibniz's Law to intersubstitute identicals.) To proceed, we now need to show that $0 + SS0 = SS0$. For note, this *isn't* an instance of Schema 3. So:

- | | | |
|----|-----------------------|-----------------------------|
| 4. | $0 + 0 = 0$ | Axiom, instance of Schema 3 |
| 5. | $0 + S0 = S(0 + 0)$ | Axiom, instance of Schema 4 |
| 6. | $0 + S0 = S0$ | From 4, 5 by LL |
| 7. | $0 + SS0 = S(0 + S0)$ | Axiom, instance of Schema 4 |
| 8. | $0 + SS0 = SS0$ | From 6, 7 by LL |

Which gives us what we want:

- | | | |
|----|-----------------------|-----------------|
| 9. | $SS0 \times S0 = SS0$ | From 3, 8 by LL |
|----|-----------------------|-----------------|

That's pretty laborious, to be sure, but again it works. And inspection of BA 's axioms and a little reflection on our second illustrative proof should now convince you of a further general claim:

Theorem 8. *Suppose τ is a term of L_B and the value of τ on the intended interpretation of the symbols is t . Then $BA \vdash \tau = t$.*

In other words, BA can correctly evaluate all terms of its language.

9.4 BA is a sound and complete theory of the truths of L_B

Our two little theorems enable us to prove the following:

Theorem 9. *Suppose σ and τ are terms of L_B . Then if $\sigma = \tau$ is true, then $BA \vdash \sigma = \tau$, and if $\sigma = \tau$ is false, then $BA \vdash \sigma \neq \tau$.*

Proof Let σ evaluate to s and τ evaluate to t . Then, by Theorem 8, (i) $BA \vdash \sigma = s$ and (ii) $BA \vdash \tau = t$.

Now, suppose $\sigma = \tau$ is true. Then $s = t$, and so s must be the very same numeral as t . We can therefore immediately conclude from (i) and (ii) that $BA \vdash \sigma = \tau$ by the logic of identity.

Suppose, on the other hand, that $\sigma = \tau$ is false, so $s \neq t$. Then by Theorem 7, $BA \vdash s \neq t$, and together with (i) and (ii) that implies $BA \vdash \sigma \neq \tau$, again by the logic of identity. \square

And from that last theorem, it more or less immediately follows that

Theorem 10. *BA is negation complete.*

Proof The only atomic claims expressible in BA are equations involving terms; all other sentences are truth-functional combinations of such equations. But we've just seen that we can (1) prove each true equation and (2) prove the true negation of each false equation. Then there's a theorem of propositional logic which tells us that we can derive any true truth-functional combination of the atoms (equations), i.e. prove any true sentence. Likewise, we can also derive the negation of any false truth-functional combination of atoms (equations), i.e. disprove any false sentence. Hence, for *any* sentence φ of BA , since either φ is true or false, either $BA \vdash \varphi$ or $BA \vdash \neg\varphi$. Hence BA is negation complete. \square

So the situation is this. BA is obviously a sound theory – all its axioms are trivial arithmetical truths, and its logic is truth-preserving, so all its theorems are true. BA is also, as we've just seen, a complete theory in the sense of entailing all the truths expressible in its language L_B . However, the language L_B only allows us to express a limited class of facts about adding and multiplying particular numbers (it can't express numerical generalizations). And, prescinding from practical issues about memory or display size, any pocket calculator can in effect tell us about all such facts: so it is no surprise that we can get a formalized theory to do the same!

10 Robinson Arithmetic

That's all very straightforward, but not very exciting. The reason that Baby Arithmetic manages to prove every correct claim *that it can express* – and is therefore negation complete by our definition – is that it can't express very much. In particular, as we stressed, it can't express any generalizations at all. BA's completeness comes at the high price of being expressively extremely impoverished. The obvious way to start beefing up BA into something more interesting is to restore the familiar apparatus of quantifiers and variables. So that's what we'll start doing.

10.1 The language L_A

We'll keep the same non-logical vocabulary as in L_B : so there is still just a single non-logical constant denoting zero, and the three built-in function-symbols, $S, +, \times$ expressing successor, addition and multiplication. But now we allow ourselves the full linguistic resources of first-order logic, with the usual supply of quantifiers and variables to express generality. We fix the domain of the quantifiers to be the natural numbers. The result is the interpreted language L_A .

L_A is the least ambitious language which 'contains the language of basic arithmetic' in the sense of Defn. 8. (For, of course, L_A has the the predicate expression ' $x = x$ ' which has the numbers as its extension, so fits our definition.)

10.2 The axioms and logic of Q

The theory Q is built in the formal language L_A , and is equipped with a full first-order classical logic. And as for the non-logical axioms, now we have the quantifiers available to express generality we can replace each of BA's metalinguistic schemata (specifying an infinite number of formal axioms governing particular numbers) by a single generalized Axiom expressed inside L_A itself. For example, we can replace the first two schemata governing the successor function by

Axiom 1. $\forall x(0 \neq Sx)$

Axiom 2. $\forall x\forall y(Sx = Sy \rightarrow x = y)$

Obviously, each instance of our earlier Schemata 1 and 2 can be deduced from the corresponding Axiom by instantiating the quantifiers.

These Axioms tell us that zero isn't a successor, but they don't explicitly rule it out that there are other objects that aren't successors cluttering up the domain of quantification (i.e. there could be 'pseudo-zeros'). We didn't need to fuss about this before, because by construction BA can only talk about the numbers represented by standard numerals in the sequence ' $0, S0, SS0, \dots$ '. But now we have the quantifiers in play. And these quantifiers are intended to run over the natural numbers – we certainly don't intend them to be running over stray objects that aren't successors. So let's reflect that in our axioms by explicitly ruling out such strays:

Axiom 3. $\forall x(x \neq 0 \rightarrow \exists y(x = Sy))$

Next, we can similarly replace our previous schemata for addition and multiplication by universally quantified Axioms in the obvious way:

Axiom 4. $\forall x(x + 0 = x)$

Axiom 5. $\forall x\forall y(x + Sy = S(x + y))$

Axiom 6. $\forall x(x \times 0 = 0)$

Axiom 7. $\forall x\forall y(x \times Sy = (x \times y) + x)$

Again, each of these axioms entails all the instances of BA's corresponding schema.

In summary,

Defn. 19. *The formalized theory with language L_A , Axioms 1 to 7, plus a classical first-order logic, is standardly called Robinson Arithmetic, or simply Q.*

It is worth noting, for future reference, that it was first isolated as a weak system of arithmetic worthy of study in 1952 – i.e. long after Gödelian incompleteness was discovered.

10.3 Q is not complete

Q is assuredly a sound theory. Its axioms are all true; its logic is truth-preserving; so its derivations are proper proofs in the intuitive sense of demonstrations of truth and every theorem of Q is true. But just which truths of L_A are theorems?

Since any old BA axiom – i.e. any instance of one of our previous schemata – can be derived from one of our new Q Axioms, every L_B -sentence that can be proved in BA is equally a quantifier-free L_A -sentence which can be proved in Q. Hence,

Theorem 11. *Q correctly decides every quantifier-free L_A sentence (i.e. proves a quantifier free $Q \vdash \varphi$ if the quantifier-free φ is true, and $Q \vdash \neg\varphi$ if such a φ is false).*

So far, so good. However, there are very simple true quantified sentences that Q can't prove. For example, Q can of course prove any particular wff of the form $0 + n = n$. *But it can't prove the corresponding universal generalization:*

Theorem 12. $Q \not\vdash \forall x(0 + x = x)$.

Proof Since Q is a theory with a standard first-order theory, for any L_A -sentence φ , $Q \vdash \varphi$ if and only if $Q \models \varphi$ (that's just the completeness theorem for first-order logic). Hence one way of showing that $Q \not\vdash \varphi$ is to show that $Q \not\models \varphi$: and we can show *that* by producing a countermodel to the entailment – i.e. by finding an interpretation (a deviant, unintended, re-interpretation) for L_A 's wffs which makes Q's axioms true-on-that-interpretation but which makes φ false.

So here goes: take the domain of our deviant, unintended, re-interpretation to be the set N^* comprising the natural numbers but with two other 'rogue' elements a and b added (these could be e.g. Kurt Gödel and his friend Albert Einstein – but any other pair of distinct non-numbers will do). Let '0' still to refer to zero. And take 'S' now to pick out the successor* function S^* which is defined as follows: $S^*n = Sn$ for any natural number in the domain, while for our rogue elements $S^*a = a$, and $S^*b = b$. It is very easy to check that Axioms 1 to 3 are still true on this deviant interpretation. Zero is still not a successor. Different elements have different successors. And every non-zero element is a successor.

We now need to extend this interpretation to re-interpret the function-symbol '+'. Suppose we take this to pick out addition*, where $m +^* n = m + n$ for any natural numbers m, n in the domain, while $a +^* n = a$ and $b +^* n = b$. Further, for any x (whether number or rogue element), $x +^* a = b$ and $x +^* b = a$.

It is again easily checked that interpreting '+' as addition* still makes Axioms 4 and 5 true. (In headline terms: For Axiom 4, we note that adding* zero on the right always has no effect. For Axiom 5, just consider cases. (i) $m +^* S^*n = m + Sn = S(m + n) = S^*(m +^* n)$ for 'ordinary' numbers m, n in the domain. (ii) $a + S^*n = a = S^*a = S^*(a +^* n)$, for 'ordinary' n . Likewise, (iii) $b + S^*n = S^*(b +^* n)$. (iv) $x +^* S^*a = x + a = b = S^*b = S^*(x +^* a)$, for any x in the domain. (v) Finally, $x +^* S^*b = S^*(x +^* b)$. Which covers every possibility.)

We are not quite done, however, as we still need to show that we can give a co-ordinate re-interpretation of 'x' in Q by some deviant multiplication* function. We can leave it as an exercise to fill in suitable details. Then, with the details filled in, we will have an overall interpretation which makes the axioms of Q true and $\forall x(0 + x = x)$ false. So $Q \not\vdash \forall x(0 + x = x)$ \square

Theorem 13. *Q is negation-incomplete.*

Proof Put $\varphi = \forall x(0 + x = x)$. We've just shown that $Q \not\vdash \varphi$. But obviously, Q can't prove $\neg\varphi$ either. Just revert to the standard interpretation built into L_A . Q certainly has true axioms on this interpretation. So all theorems are true on that interpretation, but $\neg\varphi$ is false on that interpretation, so it can't be a theorem. Hence φ is formally undecidable in Q.

Of course, we've already announced that Gödel's incompleteness theorem is going to prove that *no* sound axiomatized theory whose language is at least as rich as L_A can be negation complete – that was Theorem 1. But what we've just shown is that we don't need to invoke anything as elaborate as Gödel's arguments to see that Q is incomplete. Q is, so to speak, *boringly* incomplete.

10.4 Q can capture *less-than-or-equals*

We've just seen something that Q can't do: now for something it *can* do. We'll establish

Theorem 14. *In Q, the less-than-or-equal-to relation is captured by the wff $\exists v(v + x = y)$.*

Given the definition of capturing, Defn 15, that means we need to show that, for any particular pair of numbers, m, n , if $m \leq n$, then $Q \vdash \exists v(v + m = n)$, and otherwise $Q \vdash \neg \exists v(v + m = n)$.

Proof Suppose $m \leq n$, so for some $k \geq 0$, $k + m = n$. Q can prove everything BA proves and hence, in particular, can prove every true addition sum. So we have $Q \vdash k + m = n$. But logic gives us $k + m = n \vdash \exists v(v + m = n)$ by existential quantifier introduction. Therefore $Q \vdash \exists v(v + m = n)$, as was to be shown.

Suppose alternatively $m > n$. We need to show $Q \vdash \neg \exists v(v + m = n)$. We'll first demonstrate this in the case where $m = 2, n = 1$, using a Fitch-style proof-system. For brevity we will omit statements of Q's axioms and some other trivial steps; we drop unnecessary brackets

1.	$\exists v(v + SS0 = S0)$	Supposition
2.	$a + SS0 = S0$	Supposition
3.	$a + SS0 = S(a + S0)$	From Axiom 5
4.	$S(a + S0) = S0$	From 2, 3 by LL
5.	$a + S0 = S(a + 0)$	From Axiom 5
6.	$SS(a + 0) = S0$	From 4, 5 by LL
7.	$a + 0 = a$	From Axiom 4
8.	$SSa = S0$	From 6, 7 by LL
9.	$SSa = S0 \rightarrow Sa = 0$	From Axiom 2
10.	$Sa = 0$	From 8, 9 by MP
11.	$0 = Sa$	From 10
12.	$0 \neq Sa$	From Axiom 1
13.	Contradiction!	From 11, 12
14.	Contradiction!	$\exists E$ 1, 2–13
15.	$\neg \exists v(v + SS0 = S0)$	RAA 1–14

The only step to explain is at line (14) where we use a version of the Existential Elimination rule: if the temporary supposition $\varphi(a)$ leads to contradiction, for arbitrary a , then $\exists v\varphi(v)$ must lead to contradiction. And having done the proof for the case $m = 2, n = 1$, inspection reveals that we can use the same general pattern of argument to show $Q \vdash \neg \exists v(v + m = n)$ whenever $m > n$. [Exercise: convince yourself that this claim is true!] So we are done. \square

10.5 Adding ' \leq ' to Q

Given the result we've just proved, we can sensibly add the standard symbol ' \leq ' to L_A , the language of Q, defined so that whatever we put for ' ξ ' and ' ζ ', $\xi \leq \zeta$ is just short for $\exists v(v + \xi = \zeta)$, and then Q will be able to prove at least the expected facts about the less-than-or-equals relations among quantifier-free terms. (Well, we really need to be a bit more careful than that in stating the rule for unpacking the abbreviation, if we are to avoid any possible 'clash of variables'. But we're not going to fuss about the details.)

Note, by the way, that some presentations in fact treat ' \leq ' as a primitive symbol built into our formal theories like Q from the start, governed by its own additional axiom(s). But nothing important hangs on the difference between that approach and our policy of introducing the symbol by definition. (And of course, nothing hangs either on our policy of introducing ' \leq ' as our basic symbol rather than '<', which could have been defined by $\xi < \zeta =_{\text{def}} \exists v(Sv + \xi = \zeta)$.)

Since it so greatly helps readability, we'll henceforth make very free use of ' \leq ' as an abbreviatory symbol inside formal arithmetics. We will also adopt a second, closely related, convention. In informal mathematics we often want to say that all/some numbers less than or equal to a given number have some particular property. We can now express such claims in formal arithmetics by wffs of the shape $\forall \xi(\xi \leq \kappa \rightarrow \varphi(\xi))$ and $\exists \xi(\xi \leq \kappa \wedge \varphi(\xi))$, where ' \leq ' is to be unpacked as we've just explained. And it is standard to further abbreviate such wffs by $(\forall \xi \leq \kappa)\varphi(\xi)$ and $(\exists \xi \leq \kappa)\varphi(\xi)$ respectively.

10.6 Why Q is interesting

Given it can't even prove $\forall x(0 + x = x)$, Q is evidently a *very* weak theory of arithmetic. Which is probably no surprise as (apart from Axiom 3) we've added little *axiomatic* proof-power to BA while adding a lot of *expressive power* to its language by adding quantifiers. So it's only to be expected that there will be lots of newly expressible truths that Q can't prove (and since Q is sound, it won't be able to disprove these truths either).

Even so, despite its great shortcomings, Q does have some nice properties. As we saw, it can capture the decidable relation that obtains when one number is at least as big as another. Moreover, we can show that Q *can capture* all *decidable numerical properties and relations* – *i.e. it is sufficiently strong in the sense of Defn 16*. That might initially seem surprising, given Q's weakness. But remember, 'sufficient strength' was defined as a matter of being able to *case-by-case* prove enough wffs about decidable properties of individual numbers. And it turns out that Q's hopeless weakness at proving generalizations doesn't stop it doing that.

So that's why Q is particularly interesting – it is about the weakest arithmetic which is sufficiently strong (and it was isolated by Robinson for that reason), and for which Gödelian proofs of incompleteness can be run. Suppose, then, that a theory is formally axiomatized, consistent and can prove everything Q can prove (those do indeed seem very modest requirements). Then what we've just announced and promised can be proved is that any such theory will be 'sufficiently strong'. And therefore e.g. Theorem 6 will apply – any such theory will be incomplete.

However, we can only establish that Q *does* have sufficient strength to capture all decidable properties if and when we have a quite general theory of decidability to hand. And we don't want to get embroiled in that (at least yet). So what we *will* be proving quite soon (in Episode 6) is a rather weaker claim about Q. We'll show that it can capture all so-called 'primitive recursive' properties, where these form a very important subclass of the decidable properties. This major theorem will be a crucial load-bearing part of our proofs of various Gödel style incompleteness theorems: it means that Q gives us 'the modest amount of arithmetic' need for Theorem 2.

But before we get round to showing all this, we are first going to take a look at a *much* richer arithmetic than Q, namely PA.

For parallel reading to this Episode, see *IGT*, Ch. 8, and Ch. 9, §§9.1–9.4.