Pronunciation of the integers with full use of the place value system

Thomas Colignatus
http://thomascool.eu
May 9 2018

Abstract

Kids in kindergarten live in a world of sounds, so that the pronunciation of numbers is important. When they start learning to read and write, the co-ordination of (i) sounds, (ii) words and (iii) numerals is important. We already have the place value system fully in the numerals but not yet in written words and pronunciation. It appears that we can provide for a pronunciation of the integers with the full use of the place value system. The definitions should become an ISO standard, though the notebook and package still are quite simple. This notebook and package provide an implementation for English, German, French, Dutch and Danish.

Keywords

mathematics education, place value system, pronunciation, Common Core, Mathematica, Wolfram language, programming, package

MSC2010

97M70 Mathematics education. Behavioral and social sciences

Cloud

This notebook with package is also available at:

(1) http://community.wolfram.com/web/cool

(2) https://www.wolframcloud.com/objects/thomas-cool/MathEd/2018-05-09-Pronunciation-of-integers.nb

(3) https://zenodo.org/record/1244009 or DOI 10.5281/zenodo.1244009 for this notebook and package and https://zenodo.org/record/1244064 or DOI 10.5281/zenodo.1244064 for the PDF

I have not seen this implementation of pronunciation elsewhere (except in Chinese), so please refer to these locations so that others can find the full documentation.

Contents

Start (subsection for the initialisation cell with the package)

1. Introduction

A picture says more than a thousand words:

$$25 = 2 \times 10 + 5 = \text{two·ten five}$$

```
PronounceIntegers[25, Speak → False]
```

two·ten five·

The trailing center dot is deliberate and indicates the ghost of departed one. The strict use of the place value system is that 1 is actually 1 of 1.

```
PronounceIntegers[25, Speak → False, Simplify → False]
```

two·ten five·one

Kids in kindergarten live in a world of sounds, so that the pronuncation of numbers is important. When they start learning to read and write, the co-ordination of (i) sounds, (ii) words and (iii) numerals is important. We already have the place value system fully in the numerals but not yet in written words and pronunciation. This notebook and package provide an implementation for English, German, French, Dutch and Danish. The user can supply other languages. Only needed are 17 terms, for sign, ten digits, and the six names for the powers of 10 to a million.

The system proposed here is simple but still supports a *full use of the place value system* for education in elementary school. It hould become an ISO standard, so that researchers, educators and textbook publishers have stability of their enviroment. Even when schools would not implement the system, researchers require a standard to correct for confounding by the natural languages.

The suggestion is that schools indeed embrace this *full use of the place value system*, so that the common language used at home can be regarded as a *dialect* of the mathematical language only. Kids can deal with such differences in language.

I discussed the issue in:

- (2015a), *A child wants nice and no mean numbers*
- (2015b), *The need for a standard for the mathematical pronunciation of the natural numbers. Suggested principles of design. Implementation for English, German, French, Dutch and Danish.*

The documentation and argumentation in this present notebook is rather brief, and the reader is referred to (2015ab) for the more involved discussion. The current notebook and package only provide for a software implementation in *Mathematica.* The reason for this supplement is that I am currently looking at the US Common Core State Standards (2018) and applications of *Mathematica.*

Though the package contains the full system, with definitions for an ISO standard, it has practical limitations, like the million maximum and no relation to current pronunciation in the natural languages (already provided by *Mathematica* though). The main purpose of this notebook is to set some first steps towards implementation and to circulate the idea. Programming on language is quite involved, and implementing change even more.

The discussion on content can be short: see (2015ab). Below, I briefly mention some history and other researchers. The main section explains the working of the package. It will be most instructive though to start with examples that highlight the properties. The main body of the text will use

English and German while French, Dutch and Danish are in the Appendix.

PM 1. The center dots are not pronounced.

PM 2. The West reads and writes from left to right, while the numbers came from a region in India that read and wrote from right to left. We can leave this issue as it is, for there is advantage in starting with the biggest value.

2. Discussion

2.1. History and others

Jane Austen (1775-1817) apparently still wrote “three and twenty” insteady of “twenty-three”. The English speaking people are lucky that they managed the change.

Norway managed the change in the 1950s.

Fred Schuh (1875-1966) of TU Delft proposed this in the 1950s in Holland but didn’t convince the minister of education. I am not aware of others who think alike nowadays.

In Germany there is a small movement with Lothar Gerritzen to change their "ein und zwanzig" into "zwanzig eins": https://zwanzigeins.jetzt/

In Denmark there are Lisser Rye Ejersbo and Morten Misfeldt at Aarhus: http://vbn.aau.dk/da/publications/danish-number-names-and-number-concepts(7b79a70d-d42b-49dc-af1f-75775c9292f6)/export.html

China uses the full place value system.

Fateman (2013) discusses computerised speaking of math, with a proper distinction between how we currently pronounce the numbers and how we “should” do this. It is much wider and deeper than my present purposes.

2.2. Main advantages

To teach the place value system, it is advisable *to actually use it*.

- Words and pronuncation would be co-ordinated with the numerals. (Number: numeral, words, pronunciation.)
- The workflow is into a single direction, not jumping around.
- Merely speaking a number aloud can already solve calculation questions.
- Kids do not have to grope around in the dark. Current co-ordination failures often are not discussed so that kids simply do not know why they find issues complicated.
- There will be a sizeable gain in teaching and learning time, that can be spent on real issues.
- Research on mathematical skills and number sense is highly contaminated by this pronunciation issue. Much research draws invalid conclusions. Doing good research is quite impossible because you cannot simply experiment on kids by first training them on your own theory on the place value method. We neither can compare international results when there is such variety. (In studies in Holland and Denmark, kids were observed to use English, and telling others to do so too, because they better understood the numbers in that way.)

3. Example in English

The following describes the notebook and package (and not the English language as it is). There could be a difference between the output that the package generates, intended for interactive learning, and texts that a textbook could show. For example, a textbook could have two·ten five but the package puts in some more blanks.

3.1. Writing and speaking

There is a difference between speaking and writing (default).

In both cases, a center dot indicates a combination of the *weight* and the *value of the place*, which dot is not pronounced.

Writing uses a wider space to separate the place value positions, and speaking uses a pause. The present pause might be a bit long, but this length was pleasant for the transliteration of other languages.

```
PronounceIntegers[21, Speak → False]
```

two·ten ·one

```
PronounceIntegers[21]  (*default Speak → True*)
```

two·ten <silence msec="300"/> ·one

3.2. Simplify

For the place value, it is important to recognise that 1 is actualy 1 of 1. But after a while this meticulous accuracy becomes annoying. It is advisable to use the place value system, but it is also acceptable to simplify “one hundred” into “hundred” and not to mention “zero hundred”. Nevertheless, for teaching it is important to show both the whole system and its simplification.

```
PronounceIntegers[1, Speak → False]
```

one

```
PronounceIntegers[1, Speak → False, Simplify → False]
```

one·one

Above, there has been simplification of 21, indicated by a remaining center dot. If we do not simplify than we see the results of strictly follow the rules.

```
PronounceIntegers[21, Speak → False, Simplify → False]
```

two·ten one·one

3.3. The sign

We only write and speak the minus sign explicitly, and that a number is positive. The proper pronunciation is “negative”, because “minus” is used for the binary operation. Unfortunately, *Mathematica* uses Minus[x] for *-x,* and also pronounces it that way. Wikipedia (a portal and no source) regards

"negative" as American English, and this fits that Stephen Wolfram came from the UK, but the better diagnosis is that British English is imprecise.

```
Minus[10]
```

−10

```
Speak[%]
```

```
PronounceIntegers[%%, Speak → False, Simplify → False]
```

negative one·ten zero·one

```
PronounceIntegers[%%%, Speak → False]
```

negative ten

3.4. Myriad and lakh

Remarkably, there are no specific terms for 10^4 and 10^5 in all the languages in the package. The figure 12,345 is pronounced as 12 thousand etcetera. This is another breach upon the place value system, different from the ones that we mentioned earlier. Wikipedia (a portal and no source) observes the lack too, and mentions Greek myriad and Indian lakh. I have adopted these words for the languages mentioned in the packages too. It takes some adjustment to hear 500,000 being pronounced as 5 lakh, but such adjustment is the very purpose of the exercise.

```
PronounceIntegers[500 000, Speak → False]
```

five·lakh

4. Zig in German and tig in Dutch

In English there is no conflict between 19 = ten nine and 90 = nine·ten, and with the current use of nineteen and ninety.

For German there is this conflict: 19 is currently pronounced as neun-zehn, which conflicts with the place value pronunciation of 90. Dutch has the same problem.

Comparable to English "ty", German uses "zig" at the end of 40, 50, 60, 70, 80 and 90. Dutch has "tig". We thus may use *zig* for 10 in German and *tig* for 10 in Dutch. For me, this flash of insight was a major step for accepting that there is scope for a full use of the place value system (in Holland). Thus the words above 30 remain the same but we only require changes for 10-30.

Germans or Dutch who have a problem with zig or tig but who still want to make full use of the place value system, could also consider borrowing "ten" from English, or use scientific "deca" (though two syllables).

```
PronounceIntegers[19, Speak → False, Language → "German"]
```

·zig neun·

```
PronounceIntegers[90, Speak → False, Language → "German"]
```

neun·zig

```
PronounceIntegers[19, Speak → False, Language → "Dutch"]
```

·tig negen·

```
PronounceIntegers[90, Speak → False, Language → "Dutch"]
```

negen·tig

The German association “Zwanzig-eins” are not fundamentalists.

```
PronounceIntegers[21, Speak → False, Simplify → False, Language → "German"]
```

zwei·zig ein·ein

```
PronounceIntegers[21, Speak → False,  Language → "German"]
```

zwei·zig ·ein

Current German makes a distinction between “eins” for 1 separately and “ein” for further use. Current German uses “sieben” for 7 separately, and further has combinations with “sieb”. Such deviations happen in natural language but a systematic use has only one application, and I took what currently are most frequent: ein and sieb. (Danish has “syv”.)

```
PronounceIntegers[77, Speak → False, Language → "German"]
```

sieb·zig sieb·

```
PronounceIntegers[77,  Language → "German"]
```

sieb·tzeeg <silence msec="300"/> sieb·

The present proposal clashes with the “Zwanzig-eins” movement on the numbers 0-20 that are important for education: zig for zehn, and thus also 11-19 using zig, and zwei·zig, to show the working of the place value system. Arbitrary in my implementation are: ein for eins, sieb for sieben, dreißig would also become drei·zig. See Colignatus (2015b). Remember also my distinction between the math language in school and the dialect of current language. My suggestion is that the movement could become stronger if it includes the full place value system alongside their original views.

5. Translate versus transliterate

The user can adapt the $Language in the Preferences of *Mathematica,* and then the provided words will be pronounced properly in the specified language. The package indicates the proper values by the option True.

However, there will be users who have *Mathematica* running under one specific language (e.g. $Language = “English”) and who still have interest in the pronunciation of the different languages. In that case, transliteration may help.

Above, we saw “zig” in German being transliterated by “tzeeg” in English, for the simple reason that the current setting of $Language is “English”.

The package can be simple since we only require 1+10+6 = 17 transliterations per language, to pronounce the integers from -9,999,999 to 9,999,999.

6. Key properties of the package

Key properties of the package are:

- Written output is always in the words selected by the Language → "language" option. The words themselves are user-supplied in PronounceIntegersRules["language"], with the option True → {settings}.
- Spoken output will be a transliteration of those words using Speak. This transliteration is user-supplied in PronounceIntegersRules["language"], option "transliteration language" → {settings}. The transliteration language would generally be $Language.
- Implemented are English, German, French, Dutch and Danish, assuming that $Language (the transliterator) is English.
- $Language has been set in the Preferences of *Mathematica.* The user can provide other languages.
- For each language the user provides: (i) option True → words for the minus sign, digits 0 ... 9 and values 10^1, ..., 10^6, in the relevant language, and (ii) option "transliterator language" → the same input as above but now as a *transliteration* in terms of that transliterator language (commonly $Language). Thus it is possible to have Language → Korean and a transliteration in "SpeakTransliteration" → "Japanese". Currently though, the latter only works when $Language = "Japanese" too. The current programme structure allows for future changes on this.

```
PronounceIntegersRules["German"]
```

```
{True → {Negative → negativ, N → {null, ein, zwei, drei, vier, fuenf, sechs, sieb, acht, neun},
      Power → {zig, hundert, tausend, myriad, lakh, million}},
  English → {Negative → nayguhteef, N → {nuhl, ayen, tsvwaayie, dry, fear, fuihnf, sex, sieb, akht, noin},
      Power → {tzeeg, hoondert, tausand, myriad, lakh, meelyohn}}}
```

7. The package

The package has:

```
?Cool`MathEd`PronounceIntegers`*
```

▼ Cool`MathEd`PronounceIntegers`

PronounceIntegers	PronounceIntegersRules	SpeakTransliteration

? PronounceIntegers

PronounceIntegers[n] gives n up to place value 10^6 as text or
in Speak, and the latter with pauses for hearing and possible transliteration
in the system default language (e.g. English). Options with defaults are:
Language –> "English" uses PronounceIntegersRules["English"], see there
SpeakTransliteration :> $Language (often "English"), see there
Speak –> True,
Length –> a string that contains an instruction
for the length of pause, that is used between the positions,
Join –> "·" to join the texts of the weight and place value, default the (unpronounced) center dot
Symplify –> True eliminates terms with 0 weights and replaces 1 P by P, for P = 1, 10, ..., 10^6

? SpeakTransliteration

SpeakTransliteration is an option of PronounceIntegers with default $Language. It selects the relevant
setting from the PronounceIntegersRules. If the Language option is equal to the transliterator
($Language) then writing and speaking will be done from the True rule. When the option Language
is set to another value, then the integers are written from True but spoken by the transliterator

? PronounceIntegersRules

PronounceIntegersRules["Language"] = {True –> t, "OtherLanguage" –> f}, with t =
{Negative –> string, N –> {strings for 0...9}, Power –> {strings for 10, ..., 10^6}}, for the
true way of writing and pronunciation (when this original language = $Language), and
f = {... similar ...} for the transliteration into the actual $Language. See the examples
for "English" (f is not required given $Language = "English") and "Dutch" (t <> f).
Negative –> how the minus sign is written or pronounced (better as negative)
N –> {names how the digits 0 ... 9 are written or pronounced}
Power –> {names how 10 ... 10^6 are written or pronounced}. See SpeakTransliteration for
checking on $Language. Implemented are English, German, French, Dutch and Danish (with
some guessing). The languages have (Greek) myriad for 10^4 and (Indian) lakh for 10^5

8. ISO standard

8.1. ISO and NEN

The package contains the definitions for the full use of the place value system in a way that fits education: both the full structure and the simplification on 0 and 1. It should become an ISO standard, so that educators, researchs and textbook and software publishers have a persistent environment. There are some limitations on maximum million and language particulars, but those are not material to the standard.

In Holland, NEN is the portal for International ISO: https://www.nen.nl/About-NEN.htm. They are no public utility so that the development of norms and standards must be paid for by users. They also

have a crowd-sourcing option for such initiatives: https://nencrowd.nl. Before they allow the use of this option, they want some guarantee that there is a community with an interest in the project. In this case, it would be researchers, the Ministry of Education and publishers. To my regret, I have met no interest yet.

8.2. Open source journals

Publication could be done in an open access journal, and the proper way to get open access journals is by the universities, see Gowers (2017) and my letter to VSNU, Colignatus (2016). Present (paywall) journals have also specialisations on approaches to education and psychology, may require field testing on actual implementation, and apparently are not open to the particular approach of *redesign of mathematics education* that this notebook and package present. Thus I observed that such journals can publish invalid research but will not publish or even report about this present proposal for improvement. Quis custodiet ipsos custodes.

9. Conclusions

It is obviously advantageous when not only the numeral but also the words and pronunciation fully use the place value system. It is only a question whether the educational system would be willing to adapt.

The package gives the full definitions for a full use of the place value system for kindergarten and elementary school. Still it is simple and has some design features that must be arbitrary. The transliteration is only provided for some quick indications and not the key part. The notebook and package look meagre compared to the wealth and subtlety that already exist for the pronunciation of the natural languages and current conventions on pronouncing the numbers. The awkward ways of the past are being hardcoded now. Hopefully some of those resources can be used to support the full use of the place value system.

I have not seen this implementation of pronunciation elsewhere (except in Chinese), so please refer to these locations so that others can find the full documentation.

10. Appendix. French, Dutch and Danish

For Dutch, the design questions are similar like German.

```
PronounceIntegers[9119, Speak → False, Language → "Dutch"]
```

negen·duizend ·honderd ·tig negen·

```
PronounceIntegersRules["Dutch"]
```

{True → {Negative → min, N → {nul, een, twee, drie, vier, vijf, zes, zeven, acht, negen},
Power → {tig, honderd, duizend, myriad, lakh, miljoen}},
English → {Negative → min, N → {nuhl, ayn, tway, dree, fear, vaif, zes, zayven, akht, naigen},
Power → {tikgh, hon dairt, duuzand, myriad, lakh, milyoon}}}

For French and Danish I again refer to Colignatus (2015b) for their current complexity.

Danish can use *ti* and French can use *dix*, so we do not need a new word for 10.

```
PronounceIntegers[9189, Speak → False, Language → "French"]
```

neuf·mille ·cent huit·dix neuf·

```
PronounceIntegersRules["French"]
```

{True → {Negative → negative, N → {zero, un, deux, trois, quatre, cinq, six, sept, huit, neuf},
Power → {dix, cent, mille, myriad, lakh, million}},
English → {Negative → nayguhteef, N → {zeyroh, uhnn, duh, trouwah, kahtruh, saink, sees, seht, weet, neuhf},
Power → {dees, sahn, meel, myriad, lakh, meelyon}}}

```
PronounceIntegers[9189, Speak → False, Language → "Danish"]
```

ni·tusind ·hundrede otte·ti ni·

For Danish, I transliterated listening to Google Translate. I might not hear subtleties that native speakers would hear.

```
PronounceIntegersRules["Danish"]
```

{True → {Negative → negativ, N → {nul, en, to, tre, fire, fem, seks, syv, otte, ni},
Power → {ti, hundrede, tusind, myriad, lakh, million}},
English → {Negative → nayguhtew, N → {nul, ayn, toh, trya, fear, fem, sex, suwe, ude, knee},
Power → {tee, hoonah, toosind, myriad, lakh, meelyohn}}}

11. Literature

Thomas Colignatus is the scientific name of Thomas Cool, econometrician and teacher of mathematics, Scheveningen, Holland.

Colignatus, Th. (2015a), *A child wants nice and no mean numbers*,
(1) website: http://thomascool.eu/Papers/NiceNumbers/Index.html
(2) PDF on Zenodo: https://zenodo.org/record/291979
(3) Publisher for a print: https://www.mijnbestseller.nl/shop/index.php/catalog/product/view/id/118082/s/a-child-wants-nice-and-no-mean-numbers-79074-www-mijnbestseller-nl/

Colignatus, Th. (2015b), *The need for a standard for the mathematical pronunciation of the natural numbers. Suggested principles of design. Implementation for English, German, French, Dutch and Danish*, https://zenodo.org/record/292984

Colignatus, Th. (2016), *Letter to VSNU and others on membership dues and open access publishing,* https://boycottholland.wordpress.com/2016/10/12/letter-to-vsnu-and-others-on-membership-dues-and-open-access-publishing/

Common Core Standards (2018), *Number & Operations in Base Ten,* http://www.corestandards.org/Math/Content/NBT/ with also *Understand place value*

Fateman, R. (2013), *How can we speak math?,* https://people.eecs.berkeley.edu/~fateman/papers/speakmath.pdf

Gowers, T. (2017), *The end of an error?*, https://www.the-tls.co.uk/articles/public/the-end-of-an-error-peer-review/