CSC 530 Lecture Notes Week 6 -- Addendum

Correction to the Denotational Definition of Binary Numbers; Discussion of Alternative Denotational Definitions

I. There's a subtle flaw in the definition of binary numbers in the initial version of Lecture Notes Week 6.

A. Since the fix leads to some instructive discussion, I'd like to say that I put in the original flaw on purpose, just to see if anyone was paying attention.

B. Not.

C. That said, the discussion of this problem is a very good example of how a small-*looking* flaw in a dense semantic definition can in fact be a major semantic problem.

II. So what was wrong with the original definition?

A. The parentheses in the fourth semantic function definition

$$\mathcal{F}[\![B\ F]\!] = (\mathcal{F}[\![B]\!] + \mathcal{F}[\![F]\!])/2$$

must be omitted, i.e.,

$$\mathcal{F}[\![B\ F]\!] = \mathcal{F}[\![B]\!] + \mathcal{F}[\![F]\!]/2$$

B. The same correction needs to be made to the comparable rule in the attribute grammar definition.

C. (The corrections have been made to the latest online versions of the notes.)

III. For the purposes of fully exploring denotational details, here's a simpler-looking denotation definition of binary numbers that at first glance may look OK:

Abstract syntax: N $\in$ **Nml** = binary numerals

N ::= I . F
I ::= B | I B
F ::= B | B F
B ::= 0 | 1

Semantic domain: **Z** = real numbers

Semantic function: $\mathcal{N}$: **Nml** $\rightarrow$ **Z**

$\mathcal{N}[\![I\ .\ F]\!] = \mathcal{N}[\![I]\!] + \mathcal{N}[\![F]\!]$
$\mathcal{N}[\![I\ B]\!] = 2 * \mathcal{N}[\![I]\!] + \mathcal{N}[\![B]\!]$
$\mathcal{N}[\![B\ F]\!] = (\mathcal{N}[\![B]\!] + \mathcal{N}[\![F]\!]) / 2$
$\mathcal{N}[\![0]\!] = 0$
$\mathcal{N}[\![1]\!] = 1$

A. Upon close inspection, one should recognize what's wrong here -- it's the lack of distinction between how binary digits are interpreted in the integer versus fractional parts of the number.

B. There are a number approaches to fixing this problem, as outlined below.

C. The approaches illustrate that in order to produce a sound denotational definition, there must be sufficient grammar context and/or a sufficient number of distinct semantic functions such that each semantically distinct construct of a language can be processed distinctly.

IV. Fix by adding new semantic functions

A. The (corrected) definition in the lecture notes is probably the most direct way to overcome the problem with the preceding definition.

B. It leaves the syntax as is and adds two new semantic functions that compute the integer and fractional parts of the number separately.

V. Fix by reorganizing the grammar.

A. An alternative to the preceding solution is leave the single semantic function, and to change the grammar so that it carries the distinction between the integer and fractional parts of the number.

B. Here's the definition

Abstract syntax: N $\in$ **Nml** = binary numerals

N ::= I F
I ::= B | I B
F ::= .B | .B F
B ::= 0 | 1

Semantic domain: **Z** = real numbers

Semantic function: $\mathcal{N}$: **Nml** $\rightarrow$ **Z**

$\mathcal{N}[[\text{I F}]] = \mathcal{N}[[\text{I}]] + \mathcal{N}[[.\text{F}]]$
$\mathcal{N}[[\text{I B}]] = 2*\mathcal{N}[\text{I}] + \mathcal{N}[[\text{B}]]$
$\mathcal{N}[[0]] = 0$
$\mathcal{N}[[1]] = 1$
$\mathcal{N}[[.\text{B F}]] = (\mathcal{N}[[.\text{B}]] + \mathcal{N}[[.\text{F}]]) / 2$
$\mathcal{N}[[.0]] = 0$
$\mathcal{N}[[.1]] = 1/2$

Test case: 1101.01

$\mathcal{N}[[1101.01]] = \mathcal{N}[[1101]] + \mathcal{N}[[.01]]$
$= (2*\mathcal{N}[[110]] + \mathcal{N}[[1]]) + ((\mathcal{N}[[.0]] + \mathcal{N}[[.1]])/2)$
$= (2*(2*\mathcal{N}[[11]] + \mathcal{N}[[0]]) + \mathcal{N}[[1]]) + ((\mathcal{N}[[.0]] + \mathcal{N}[[.1]])/2)$
$= (2*(2*(2*\mathcal{N}[[1]] + \mathcal{N}[[1]]) + \mathcal{N}[[0]]) + \mathcal{N}[[1]]) + ((\mathcal{N}[[.0]] + \mathcal{N}[[.1]])/2)$
$= (2*(2*(2*1 + 1) + 0) + 1) + (0 + 1/2)/2$

C. This syntactic reorganization uses a well-known technique called "left factoring".

1. Left factoring involves changing grammar rules so that RHSs have unique leading prefixes.
2. It is used commonly to render grammars suitable for top-down recursive descent parsing.
3. Interestingly, a natural way to apply a denotationally-defined semantic function is to follow the top-down strategy used in a recursive descent parser.

4. Hence, it is not surprising that left factoring can be used to alter a grammar to make it suitable for the definition of denotational functions.

D. A short hand used in this definition is a form of semantic function polymorphism that allows the single semantic function $\mathcal{N}$ to operate on any syntactic form derived from the head grammar symbol N.

E. Without this polymorphism, the definition is bulkier, as shown in the following non-polymorphic equivalent of the immediately preceding definition:

Abstract syntax: N ∈ **Nml** = binary numerals
I ∈ **Int** = binary integers
F ∈ **Frac** = binary fractions
B ∈ **Bit** = binary bits

N ::= I F
I ::= B | I B
F ::= .B | .B F
B ::= 0 | 1

Semantic domain: **Z** = real numbers

Semantic functions: $\mathcal{N}$: **Nml** → **Z**
$\mathcal{I}$: **Int** → **Z**
$\mathcal{F}$: **Frac** → **Z**
$\mathcal{B}$: **Bit** → **Z**

$\mathcal{N}$[[I F]] = $\mathcal{N}$[[I]] + $\mathcal{N}$[[.F]]
$\mathcal{I}$[[I B]] = 2*$\mathcal{I}$[I] + $\mathcal{I}$[[B]]
$\mathcal{I}$[[B]] = $\mathcal{B}$[[B]]
$\mathcal{B}$[[0]] = 0
$\mathcal{B}$[[1]] = 1
$\mathcal{F}$[[.B F]] = ($\mathcal{F}$[[.B]] + $\mathcal{F}$[[.F]]) / 2
$\mathcal{F}$[[.B]] = $\mathcal{B}$[[.B]]
$\mathcal{B}$[[.0]] = 0
$\mathcal{B}$[[.1]] = 1/2

Test case: 1101.01
$\mathcal{N}$[[1101.01]] = $\mathcal{I}$[[1101]] + $\mathcal{F}$[[.01]]
= (2*$\mathcal{I}$[[110]] + $\mathcal{I}$[[1]]) + (($\mathcal{F}$[[.0]] + $\mathcal{F}$[[.1]])/2)
= (2*(2*$\mathcal{I}$[[11]] + $\mathcal{I}$[[0]]) + $\mathcal{I}$[[1]]) + (($\mathcal{F}$[[.0]] + $\mathcal{F}$[[.1]])/2)
= (2*(2*(2*$\mathcal{I}$[[1]] + $\mathcal{I}$[[1]]) + $\mathcal{I}$[[0]]) + $\mathcal{I}$[[1]]) + (($\mathcal{F}$[[.0]] + $\mathcal{F}$[[.1]])/2)
= (2*(2*(2*$\mathcal{B}$[[1]] + $\mathcal{B}$[[1]]) + $\mathcal{B}$[[0]]) + $\mathcal{B}$[[1]]) + (($\mathcal{B}$[[.0]] + $\mathcal{B}$[[.1]])/2)
= (2*(2*(2*1 + 1) + 0) + 1) + (0 + 1/2)/2

VI. Fix by adding new semantic functions and reorganizing the grammar

A. A bulkier solution than either of the preceding is to both reorganize the grammar and add new semantic functions.

B. Here's this third solution:

Abstract syntax: N $\in$ **Nml** = binary numerals
I $\in$ **Int** = binary integers
F $\in$ **Frac** = binary fractions

N ::= I . F
I ::= 0 | 1 | I 0 | I 1
F ::= 0 | 1 | 0 F | 1 F

Semantic domain: **Z** = real numbers

Semantic functions: $\mathcal{N}$: **Nml** $\rightarrow$ **Z**
$\mathcal{I}$: **Int** $\rightarrow$ **Z**
$\mathcal{F}$: **Frac** $\rightarrow$ **Z**

$\mathcal{N}[\![\text{I . F}]\!] = \mathcal{I}[\![\text{I}]\!] + \mathcal{F}[\![\text{F}]\!]$
$\mathcal{I}[\![\text{I 1}]\!] = 2 * \mathcal{I}[\![\text{I}]\!] + 1$
$\mathcal{I}[\![\text{I 0}]\!] = 2 * \mathcal{I}[\![\text{I}]\!]$
$\mathcal{I}[\![1]\!] = 1$
$\mathcal{I}[\![0]\!] = 0$
$\mathcal{F}[\![\text{1 F}]\!] = (1 + \mathcal{F}[\![\text{B}]\!]) / 2$
$\mathcal{F}[\![\text{0 F}]\!] = \mathcal{F}[\![\text{B}]\!] / 2$
$\mathcal{F}[\![1]\!] = 1/2$
$\mathcal{F}[\![0]\!] = 0$

Test case: 1101.01

$$\begin{aligned}
\mathcal{N}[\![1101.01]\!] &= \mathcal{I}[\![1101]\!] + \mathcal{F}[\![01]\!] \\
&= (2*\mathcal{I}[\![110]\!] + 1) + (\mathcal{F}[\![1]\!]/2) \\
&= (2*(2*\mathcal{I}[\![11]\!]) + 1) + (\mathcal{F}[\![1]\!]/2) \\
&= (2*(2*(2*\mathcal{I}[\![1]\!] + 1)) + 1) + (\mathcal{F}[\![1]\!]/2) \\
&= (2*(2*(2*1 + 1)) + 1) + ((1/2)/2)
\end{aligned}$$

C. There's nothing particularly noteworthy about this definition other than the fact evaluation takes a bit less pattern matching to apply the semantic functions.

1. The important point in this regard is that clerical reorganization of the grammar by expanding or contracting rule RHSs has no effect on the semantics.
2. Consider the application of $\mathcal{I}[\![1101]\!]$ in this definition as compared to the first definition:
   a. Here $\mathcal{I}[\![1101]\!]$ reduces via one pattern match to $(2*\mathcal{I}[\![110]\!] + 1)$, as follows:

$$\begin{aligned}
\mathcal{I}[\![1101]\!] &= 2*\mathcal{I}[\![\text{I}]\!] + 1 && \text{via semantic equation } \mathcal{I}[\![\text{I 1}]\!] = 2 * \mathcal{I}[\![\text{I}]\!] + 1 \\
&= 2*\mathcal{I}[\![110]\!] + 1 && \text{via pattern matching I ==> 110}
\end{aligned}$$

   b. By comparison, in the first definition $\mathcal{I}[\![1101]\!]$ reduces via two pattern matches to $(2*\mathcal{I}[\![1101]\!] + \mathcal{I}[\![1]\!])$ and thence to $(2*\mathcal{I}[\![110]\!] + 1)$ via semantic function application, as follows:

$$\begin{aligned}
\mathcal{I}[\![1101]\!] &= 2*\mathcal{I}[\![\text{I}]\!] + \mathcal{I}[\![\text{B}]\!] && \text{via semantic equation } \mathcal{I}[\![\text{I B}]\!] = 2*\mathcal{I}[\text{I}] + \mathcal{I}[\![\text{B}]\!] \\
&= 2*\mathcal{I}[\![110]\!] + \mathcal{I}[\![\text{B}]\!] && \text{via pattern matching I ==> 110} \\
&= 2*\mathcal{I}[\![110]\!] + \mathcal{I}[\![1]\!] && \text{via pattern matching B ==> 1} \\
&= 2*\mathcal{I}[\![110]\!] + 1 && \text{via semantic equation } \mathcal{I}[\![1]\!] = 1
\end{aligned}$$

3. In either case, the semantics are the same

VII. Fix by updating Tennent's example directly

A. A final variant is a combination of the preceding fixes that ends up looking like a simple add-on to Tennent's original non-fractional definition of binary numerals on page 212.

B. Here's this definition:

Abstract syntax: N $\in$ **Nml** = binary numerals

N ::= I F
I ::= 0 | 1 | I0 | I1
F ::= 0 | 1 | .0F | .1F

Semantic domain: **Z** = real numbers

Semantic function: $\mathcal{N}$: **Nml** $\rightarrow$ **Z**

$\mathcal{N}$[[IF]] = $\mathcal{N}$[[I]] + $\mathcal{N}$[[.F]]
$\mathcal{N}$[[I1]] = 2 * $\mathcal{N}$[[I]] + 1
$\mathcal{N}$[[I0]] = 2 * $\mathcal{N}$[[I]]
$\mathcal{N}$[[1]] = 1
$\mathcal{N}$[[0]] = 0
$\mathcal{N}$[[.1F]] = (1 + $\mathcal{N}$[[.F]]) / 2
$\mathcal{N}$[[.0F]] = $\mathcal{N}$[[.F]] / 2
$\mathcal{N}$[[.1] == 1/2
$\mathcal{N}$[[.0] = 0

Test case: 1101.01
$\mathcal{N}$[[1101.01]] = $\mathcal{N}$[[1101]] + $\mathcal{N}$[[.01]]
= (2*$\mathcal{N}$[[110]] + 1) + ($\mathcal{N}$[[.1]]/2)
= (2*(2*$\mathcal{N}$[[11]]) + 1) + ($\mathcal{N}$[[.1]]/2)
= (2*(2*(2*$\mathcal{N}$[1] + 1)) + 1) + ($\mathcal{N}$[[.1]]/2)
= (2*(2*(2*1 + 1)) + 1) + ((1/2)/2)

C. There is no semantic difference between this and the preceding (correct) definitions.