Argument Binding
(Baker and Bobaljik 2002)

1.0 Argument Binding

- Not all synthetic compounds can be modeled using just substitution linking and argument identification. We now know how to model a synthetic compound like *truck driving*. However, a seemingly similar synthetic compound like *truck driver* requires an extra step. This is because some kinds of nominalizations — like *-er* nouns (as in *driver*) — add a new layer of complexity to the derivation.

- To appreciate this new layer of complexity, consider the *–er* nominalizations in (23) from the verbs in (22):

(1) Some transitive verbs <Ag <Th>>:
These lions eat meat.
This machine washes dishes.
This robot drives trucks.
<Ag> <Th>

(2) (Adapted from Bobaljik: Unit 4)

a.	I consider	these lions	[(to be) (reluctant) eaters	of meat].
b.	I consider	this machine	[(to be) a (fine) washer	of dishes].
c.	I consider	this robot	[(to be) a (menacing) driver	of trucks].
		<R=Ag>		<Th>

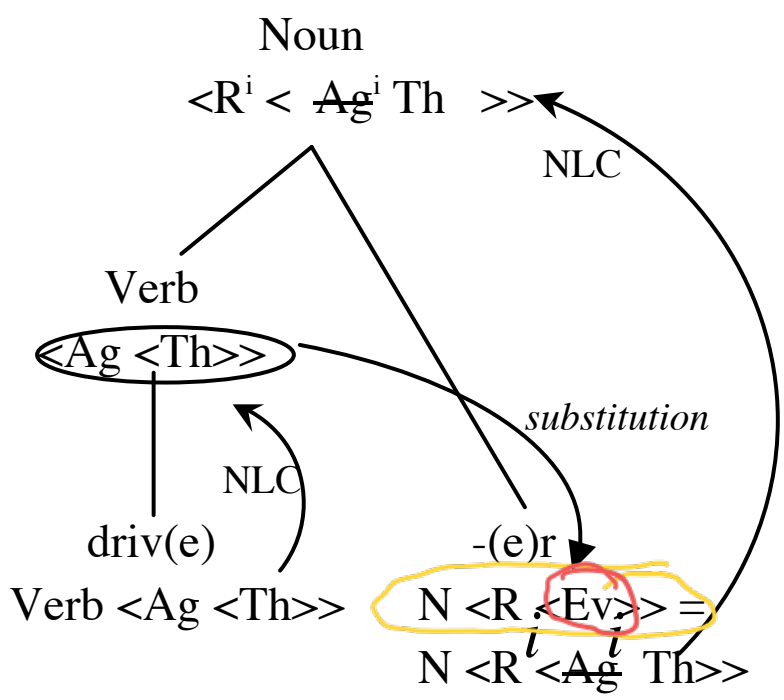
- As we saw before with *–ing* nominalizations, here the arguments of the verb are inherited by the derived noun, which in addition introduces an <R> external argument (so substitution linking has applied).
- However, with the *–ing* nominalizations the inherited arguments were both internal arguments of the new noun, whereas with the *–er* nominalizations shown here, only the <Th> argument is internal: the <Ag> is construed as corresponding to the new external <R> argument.

- None of the operations we have developed so far can implement this.
  - Argument identification allows the non-head of a compound to be identified with a Theta-role of the head. But this is different.
  - Here two theta-roles of the head are being bound together. We need a new operation: **Argument Binding.**

(3) **Argument Binding** (Bobaljik: Unit 4)
A nominalizing morpheme may be lexically specified to *bind* one of the internal arguments it inherits via substitution.

- We indicate Argument Binding using superscripts to indicate the θ-roles that are bound together. Crucially, binding happens *within* the argument structure of a single node.
  - Be careful not to confuse this notation with Argument Identification, which also uses superscripts (but not in the same way)

(4) WST for *driver*



Argument Binding

(Bobaljik: Unit 4)

Step 1: Substitution Linking applies, replacing the <Ev> argument of *–er* with the argument structure of *drive*.
Step 2: Argument Binding applies, linking the R argument of *–er* to the <Ag> argument.
Step 3: The NLC applies, but the bound <Ag> argument can no longer be independently expressed. The R argument must be interpreted as the bound <Ag>.

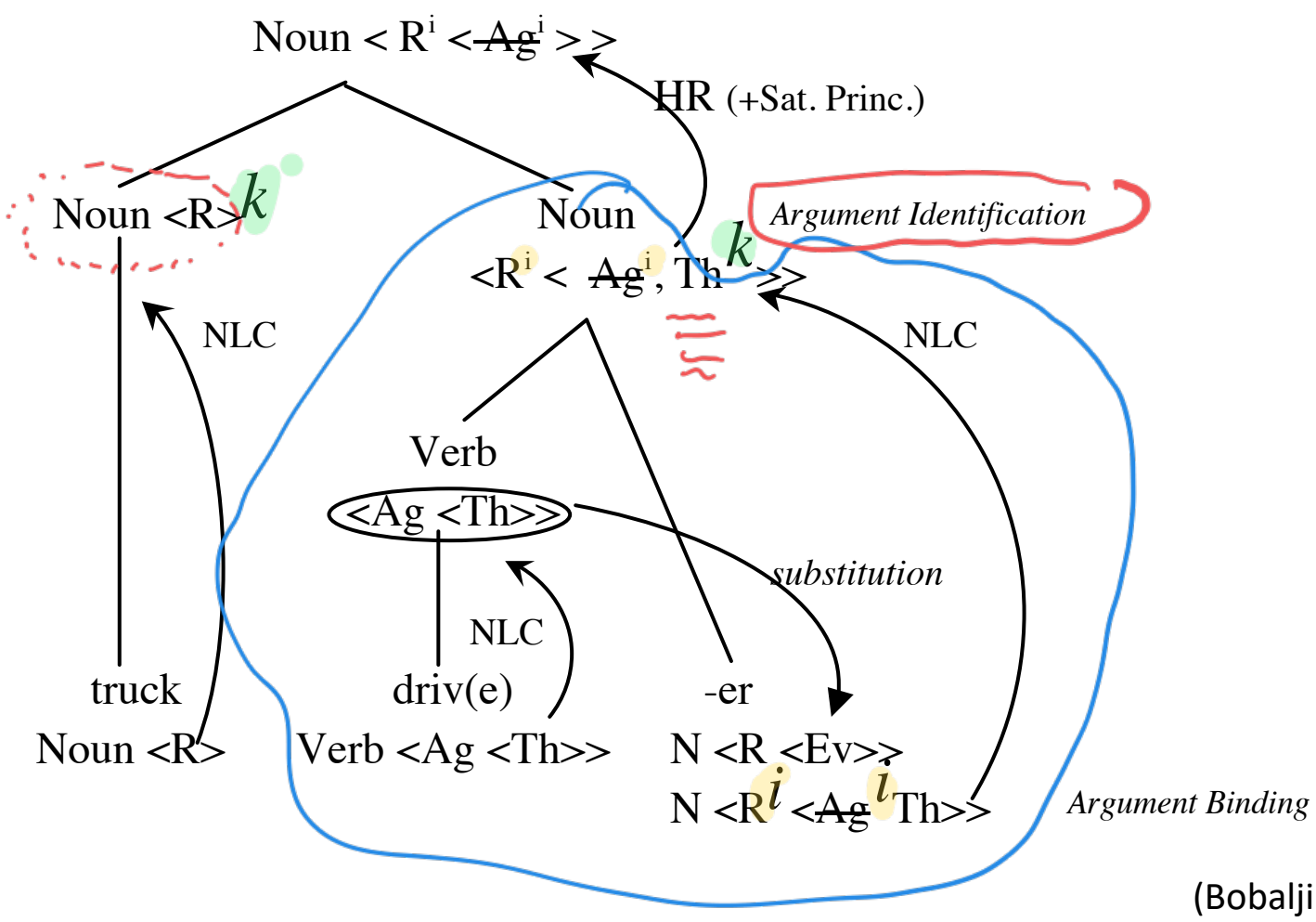
- We see that this derives the correct argument structure for the resulting noun *driver*.
  - The Argument Binding forces both <R> and <Ag> roles to be satisfied by a single element in the syntax.

(5) I consider the robot [(to be) a (menacing) driver of trucks].
<R>=<Ag> <Th>

* I consider this the robot [(to be) a (menacing) driver of trucks].
<R> <Ag> <Th>

- And now we are in a position to put all the pieces together and give a full derivation for compounds like *truck driver.*

(6) WST for *truck-driver*



(Bobaljik: Unit 4)

- Note what we are able to explain:
  - we can explain the fact that the referent of *truck driver* is interpreted as the agent of *drive*;
  - the nonhead *truck* satisifies the Theme role of *drive*
  - the whole form *truck driver* has only one argument position (interpreted as both <R> and <Ag>) as demonstrated in (7).

(7) truck-driver <$\underline{R}^i$ <Ag^i>> meaning: one who drives trucks

The robot drives trucks

I consider him a truck-driver (*by robots) (*of fire trucks)
<R=Ag>