THE RULES OF COMBINATORY CATEGORIAL GRAMMAR

JONATHAN STERLING

Abstract. A compact formal presentation of the type theory and calculus behind the Combinatory Categorial Grammar, in its fully lexicalized multi-modal extension. Traditional presentations of the combining rules tend to involve duplication of a single rule over both direction and modality in an effort to make derivations look as much as possible like the surface forms they trace. Here, we take a different approach and aim for a very succinct syntactic presentation which does not visually resemble the surface form in all cases, but which preserves enough information to be *interpreted* into such a surface form.

1. The Syntax

The type theory of combinatory categorial grammar contains base types which correspond to DPs, VPs, PPs and so forth, as well as function types which characterize unsaturated terms (such as determiners, verbs, prepositions, etc.). Words of these types may be combined into larger phrases by means of several combining rules, of which function application $(\cdot)$ and composition $(\mathbf{B})$ are the most basic. Others, such as Curry's substitution operator $(\mathbf{S})$ may also be considered.

Moreover, these combining rules may be applied in either direction so as to faithfully represent the linear ordering of terms in the surface form. But some terms may not admit certain combinators, or certain directions, or some combination of the two constraints: as a result, we must decorate function types with both permitted direction as well as a notion of *modality*, which constrains the applicability of combining rules within the lexicon.

1.1. Direction and Modality.

1.1. Direction and Modality. A *direction* is one of the set $\mathsf{Dir} \triangleq \{\triangleright, \triangleleft\}$ where $\triangleright$ denotes *forward* and $\triangleleft$ denotes *backward*. Directions may be reversed:

$$\begin{aligned} !\,\theta \quad &\Leftarrow \mathsf{induction}\ \theta \\ !\triangleright &\mapsto \triangleleft \\ !\triangleleft &\mapsto \triangleright \end{aligned}$$

A *modality* is one of the set $\mathsf{Mod} \triangleq \{\bullet, \diamond, \times, \star\}$, where $\bullet$ denotes function terms which may only be combined using basic function application, $\diamond$ those which may be combined by application and any rules which preserve the uniform order of its operands (this kind of rule is called *harmonic*), and $\times$ those which may be combined by application and any rules which permute the order of its operands (this kind of rule is called *crossed*). Finally, $\star$ represents those terms which may be combined using any of the combining rules.

Theorem 1.1. *The tuple* $\langle \mathsf{Mod}, \wedge, \vee, \star, \bullet \rangle$ *is a bounded lattice characterized by the following Hasse diagram:*

$$\begin{tikzcd} & \bullet \arrow[dl] \arrow[dr] & \\ \diamond \arrow[dr] & & \times \arrow[dl] \\ & \star & \end{tikzcd}$$

where $\wedge$ *and* $\vee$ *are binary operations representing the join and the meet respectively of two modalities as follows:*

$$\begin{aligned}
&a \wedge b && \Leftarrow \mathsf{induction}\ a \\
&\quad \bullet \wedge b && \mapsto \bullet \\
&\quad \diamond \wedge b && \Leftarrow \mathsf{induction}\ b \\
&\qquad \diamond \wedge \bullet && \mapsto \bullet \\
&\qquad \diamond \wedge \diamond && \mapsto \diamond \\
&\qquad \diamond \wedge \times && \mapsto \bullet \\
&\qquad \diamond \wedge \star && \mapsto \diamond \\
&\quad \times \wedge b && \Leftarrow \mathsf{induction}\ b \\
&\qquad \times \wedge \bullet && \mapsto \bullet \\
&\qquad \times \wedge \diamond && \mapsto \bullet \\
&\qquad \times \wedge \times && \mapsto \times \\
&\qquad \times \wedge \star && \mapsto \times \\
&\quad \star \wedge b && \mapsto b
\end{aligned}
\qquad\qquad
\begin{aligned}
&a \vee b && \Leftarrow \mathsf{induction}\ a \\
&\quad \bullet \vee b && \mapsto b \\
&\quad \diamond \vee b && \Leftarrow \mathsf{induction}\ b \\
&\qquad \diamond \vee \bullet && \mapsto \diamond \\
&\qquad \diamond \vee \diamond && \mapsto \diamond \\
&\qquad \diamond \vee \times && \mapsto \star \\
&\qquad \diamond \vee \star && \mapsto \star \\
&\quad \times \vee b && \Leftarrow \mathsf{induction}\ b \\
&\qquad \times \vee \bullet && \mapsto \times \\
&\qquad \times \vee \diamond && \mapsto \star \\
&\qquad \times \vee \times && \mapsto \times \\
&\qquad \times \vee \star && \mapsto \star \\
&\quad \star \vee b && \mapsto \star
\end{aligned}$$

Lemma 1.2 (Commutativity)**.** *For modalities* a, b*, we have* $a \wedge b = b \wedge a$ *and* $a \vee b = b \vee a$*.*

Proof. By induction on a and b. □

Lemma 1.3 (Associativity)**.** *For modalities* a, b, c*, we have* $a \wedge (b \wedge c) = (a \wedge b) \wedge c$ *and* $a \vee (b \vee c) = (a \vee b) \vee c$*.*

Proof. By induction on a, b and c. □

Lemma 1.4 (Absorption)**.** *For modalities* a, b*, we have* $a \wedge (a \vee b) = a$ *and* $a \vee (a \wedge b) = a$*.*

Proof. By induction on a and b. □

Lemma 1.5 (Idempotence)**.** *For any modality* a*, we have* $a \wedge a = a$ *and* $a \vee a = a$*.*

Proof. By induction on a. □

Lemma 1.6 (Bounding)**.** *For any modality* a*, we have* $a \wedge \star = a$ *and* $a \vee \bullet = a$*.*

Proof. By induction on a. □

Proof of Theorem 1.1. By Lemmas 1.2–1.5, modalities form a lattice; moreover, by Lemma 1.6, they are a bounded lattice. □

Corollary 1.7. *We have a partial order* $\leq$ *on modalities as follows:*

$$\begin{aligned}
&a \leq b && \Leftarrow \mathsf{decide}\ a \wedge b = a \\
&\quad \mathsf{yes}\ p && \mapsto \top \\
&\quad \mathsf{no}\ p && \mapsto \bot
\end{aligned}$$

1.2. **The Syntactic Types.** For a set B of base categories, the syntactic types are the closure of B under the function arrow, annotated by direction and modality:

$$\frac{b : B}{b : \mathsf{SynType}_B} \qquad \frac{X, Y : \mathsf{SynType}_B \quad \theta : \mathsf{Dir} \quad \mu : \mathsf{Mod}}{X \,|^{\theta}_{\mu}\, Y : \mathsf{SynType}_B}$$

Modulo direction and modality, the notation $X \,|^{\theta}_{\mu}\, Y$ corresponds to a function type $Y \to X$ in ordinary type theory. Moreover, when direction is known, we abbreviate with the following notations:

$$X \,/_{\mu}\, Y \triangleq X \,|^{\triangleright}_{\mu}\, Y$$
$$X \,\backslash_{\mu}\, Y \triangleq X \,|^{\triangleleft}_{\mu}\, Y$$

1.3. **The Term Language.** Whereas in previous presentations of the CCG calculus, introduction rules for terms have been duplicated by direction, we can present them succinctly as follows.

Definition 1.8. *A* Lexicon *over base types B is a (meta-)type parameterized by the syntactic types over B.*

$$\mathsf{Lexicon}_B \triangleq \mathsf{SynType}_B \to \mathsf{Type}$$

Terms are parameterized by the lexicon they draw from: by this means, terms from differing lexicons may not be combined.

$$\frac{L : \mathsf{Lexicon}_B \quad X : \mathsf{SynType}_B}{\mathsf{SynTerm}_L\, X : \mathsf{Type}}$$

An entry in a lexicon L is also a term in $\mathsf{SynTerm}_L$.

$$\frac{X : \mathsf{SynType}_B \quad x : L_B\, X}{x : \mathsf{SynTerm}_L\, X}$$

For the sake of brevity, we will often use a shorthand $x : X$ for the judgement $x : \mathsf{SynTerm}_L\, X$. At this point we are prepared to give the combining rules in their full form; given a set of base types B and a lexicon L:

$$(\mathbf{App}) \qquad \frac{X, Y : \mathsf{SynType}_B \quad \theta : \mathsf{Dir} \quad \mu : \mathsf{Mod} \quad p : \bullet \leq \mu \quad f : X \,|^{\theta}_{\mu}\, Y \quad x : Y}{f \cdot^{\theta}_{\mu} x : X}$$

As you can see, we were able to express the two directional variants of **App** in one rule by abstracting over θ. We could of course omit the constraint p, since by Lemma 1.6 and Corollary 1.7 we have $\bullet \leq \mu$ for all modalities μ.

Naturally, type-raising can also be expressed very simply using our direction reversal operator:

$$(\mathbf{TR}) \qquad \frac{X, Y : \mathsf{SynType}_B \quad \theta : \mathsf{Dir} \quad \mu : \mathsf{Mod} \quad x : X}{\uparrow^{\theta}_{\mu} x : Y \,|^{\theta}_{\mu}\, (Y \,|^{!\theta}_{\mu}\, X)}$$

The composition rule is more interesting, as it places further constraints on both the directions and the modalities in order to generate in one stroke four different rules: forward composition, backward composition, forward crossed composition, and backward crossed composition. We can capture these constraints with a notion of Turn.

Definition 1.9. *A* turn *is an operation on directions licensed by constraints on modalities. Therefore, a* $\mathsf{Turn}\,\theta\,\mu\,\nu\,\rho$ *licenses a function in direction* θ *and modality* μ *to be composed with a function in direction* ρ *and modality* ν*.*

$$\frac{\theta, \rho : \mathsf{Dir} \quad \mu, \nu : \mathsf{Mod}}{\mathsf{Turn}\,\theta\,\mu\,\nu\,\rho : \mathsf{Type}}$$

The identity turn $\shortparallel$ *is restricted to modalities of at least the same power as* $\diamond$*; the crossed turn* $\Yup$ *is restricted to modalities of at least the same power as* $\times$*:*

$$\frac{\theta : \mathsf{Dir} \quad p : \diamond \leq \mu \quad q : \diamond \leq \nu}{\shortparallel : \mathsf{Turn}\,\theta\,\mu\,\nu\,\theta} \qquad \frac{\theta : \mathsf{Dir} \quad p : \times \leq \mu \quad q : \times \leq \nu}{\Yup : \mathsf{Turn}\,\theta\,\mu\,\nu\,(!\,\theta)}$$

With this in hand, the rules for composition may be expressed each in one shot, accounting for harmonic and crossed variants in either direction:

(**Comp**)

$$\frac{X, Y, Z : \mathsf{SynType}_B \quad \theta, \rho : \mathsf{Dir} \quad \mu, \nu : \mathsf{Mod} \quad t : \mathsf{Turn}\,\theta\,\mu\,\nu\,\rho \quad f : X\,|^{\theta}_{\mu}\,Y \quad g : Y\,|^{\rho}_{\nu}\,Z}{f\,\mathbf{B}^{\theta}_{t}\,g : X\,|^{\rho}_{\mu\vee\nu}\,Z}$$

It turns out that Turn also suffices to give a single rule schema for substitution (a form of combination in which an argument is used twice):

(**Subst**)

$$\frac{X, Y, Z : \mathsf{SynType}_B \quad \theta, \rho : \mathsf{Dir} \quad \mu, \nu : \mathsf{Mod} \quad t : \mathsf{Turn}\,\theta\,\mu\,\nu\,\rho \quad f : (X\,|^{\theta}_{\mu}\,Y)\,|^{\rho}_{\nu}\,Z \quad g : Y\,|^{\rho}_{\nu}\,Z}{f\,\mathbf{S}^{\theta}_{t}\,g : X\,|^{\rho}_{\mu\vee\nu}\,Z}$$

1.4. **Correspondence to Visual Proofs.** A syntactic proof in traditional CCG is laid out visually as a trace of the judgements which lead to a particular string being considered admissible. It may seem that because we have abstracted direction out of the judgements, we will no longer be able to construct such proofs, but this is not so. The fact that the order of (meta)-parameters of judgements no longer corresponds to the order of elements in a string is inconsequential: we can choose to have our notation vary with the choice of directional parameters (such as $\theta : \mathsf{Dir}$). Therefore, a term in our calculus which proves "The dog ate the shoe" can be given as follows:

$$(\text{ate} \cdot^{\triangleright}_{\bullet} (\text{the} \cdot^{\triangleright}_{\diamond} \text{shoe})) \cdot^{\triangleleft}_{\bullet} (\text{the} \cdot^{\triangleright}_{\diamond} \text{dog})$$

We can either consider such a notation as having elided the proofs which satisfy the constraints of the rules, or as generating proof obligations which must be satisfied in order to construct a typing derivation. In the latter case, we can then consider this notation to be a surface-level representation which is elaborated into a fully explicit one; in this case, this would mean mechanically adding proofs that $\bullet \leq \diamond$ and $\bullet \leq \bullet$ hold.

It is trivial to convert our term into a derivation of the traditional sort though:

$$\dfrac{\dfrac{\dfrac{\text{the}}{D\,/_{\diamond}\,N} \quad \dfrac{\text{dog}}{N}}{D}\,\triangleright \qquad \dfrac{\dfrac{\text{ate}}{(S\,\backslash_{\bullet}\,D)\,/_{\bullet}\,D} \quad \dfrac{\dfrac{\text{the}}{D\,/_{\diamond}\,N} \quad \dfrac{\text{shoe}}{N}}{D}\,\triangleright}{S\,\backslash_{\bullet}\,D}\,\triangleright}{S}\,\triangleleft$$