Week 7 Assignment: Study questions

This week, instead of a problem set, I’d like you to answer the following 7 questions about your readings (Kisseberth and Kager). Attach an extra sheet if you must, but the amount of space I’ve left is meant as an indication of how detailed your answers should be.

1. What do the Yawelmani rules $\emptyset \rightarrow V / C \_ CC$ and $C \rightarrow \emptyset / CC + \_ \_ \_$ have in common?

They both have the effect of eliminating CCC clusters.

2. What, according to Kisseberth, are conspiracies, and how does he propose that the theory be amended to handle them?

A conspiracy exists when several rules, different in form, have the same effect (such as eliminating CCC clusters). Kisseberth proposes that constraints be able to trigger and block rules. Rules can thus be made simpler, since the part of the environment statement that expresses the constraint can be eliminated.

3. Compare the role of constraints for Kisseberth with the role of constraints in OT.

For Kisseberth, constraints serve to trigger and block rules, allowing the rules to be expressed more simply. OT takes it simply by removing all content from rules, and allowing any transformation of input to output to be triggered or blocked by constraints. In Kisseberth’s scheme, when there are rules not subject to the constraint, the constraint will be violated. In OT, constraints are violated only when higher-ranked, conflicted constraints require it.

4. In OT, what happens when two constraints conflict?

The higher-ranked constraint prevails: candidates that obey the higher-ranked constraint and violate the lower-ranked constraint are preferred over candidates that violate the higher-ranked constraint and obey the lower-ranked constraint.
5. Explain how the evaluation of candidates in OT is similar to alphabetical ordering (azzzzz precedes baaaaa).

Alphabetical ordering proceeds left to right: if the first segment of one string comes earlier in the alphabet than the first segment of another string, then the first string precedes the second in the list. Azzzzz precedes baaaaa because letters that come later can’t make up for the difference between a and b. Candidate evaluation in OT is similar: doing well on low-ranked constraints can’t make up for doing badly on high-ranked constraints.

6. In Yawelmani, CCC clusters are not allowed, and Kisseberth postulates a constraint against them. In English, though, CCC clusters are allowed (string, whilst). Does this mean the constraint against CCC clusters is invalid? How would OT express the difference between Yawelmani and English with respect to CCC clusters?

In Yawelmani, the *CCC constraint is high-ranking enough to force the violations of faithfulness that are incurred by epenthesys and deletion. In English, those faithfulness constraints outrank *CCC. *CCC is still a valid constraint, just not one that is high-ranking enough in English to prohibit all instances of CCC.

7. Give an example (from the readings or from anywhere else) of a conflict between a markedness constraint and a faithfulness constraint (explain briefly why there is a conflict). How do such conflicts refute the Fallacy of Perfection?

In Yawelmani, *CCC and Don’tInsert are in conflict, because *CCC would like to do away with CCC clusters, and Don’tInsert would like to prevent one means of repairing these clusters (epenthesys). The existence of faithfulness constraints that are in conflict with markedness constraints is the reason why not every utterance is (say) [ba], or some other maximally unmarked string: if the language has inputs that differ, then they cannot all be pronounced [ba] without violating faithfulness constraints. There is no one output candidate that satisfies all constraints for all inputs.