Collaborative Text Editing

Background

Many platforms nowadays support collaborative editing, a method of two users being able to edit the same piece of text (or maybe some other form of data such as a calendar) at once (or perhaps when the device is offline) and avoid conflicts while also keeping both users' changes.

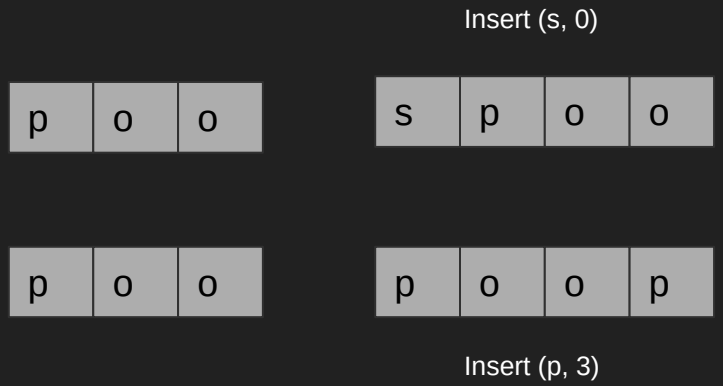
See Google Docs, VScode, Google Calendar.

The Problem

Array writes are not commutative!

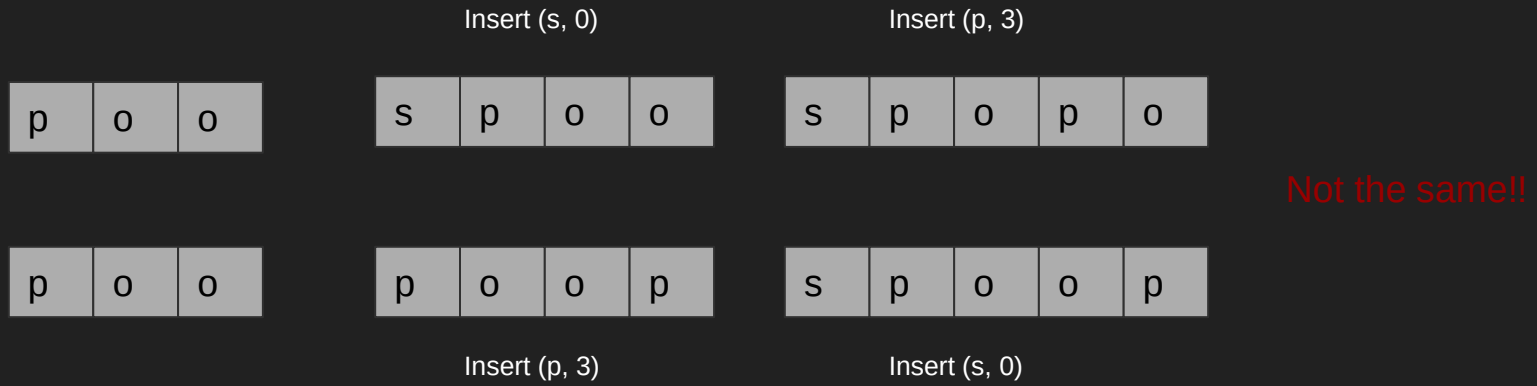
The Problem

Array writes are not commutative!



The Problem

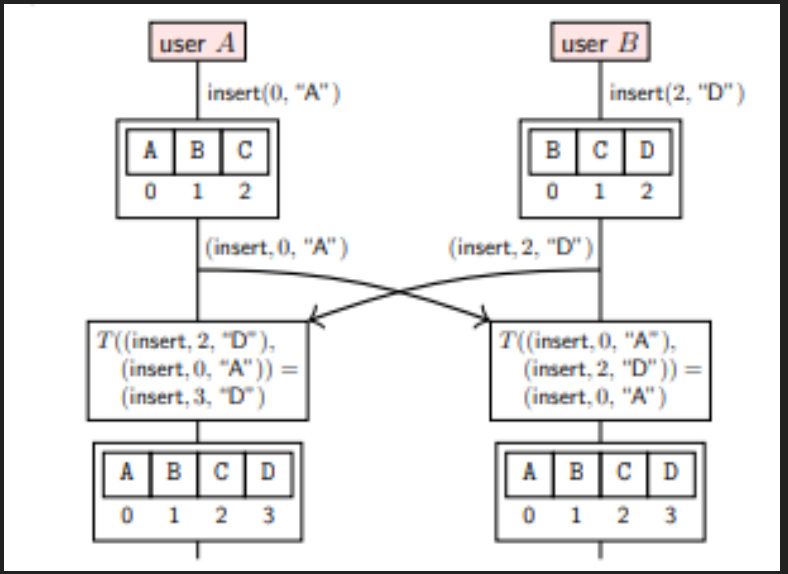
Array writes are not commutative!



Solution: Operational Transform

Transformation function T converts the ordered operations into the proper array such that each local copy of the document converges in the way that it is supposed to.

T is a very complicated function with lots of edge cases.



Pitfalls of Operational Transform

OT is a perfectly functional algorithm - in fact it is used in Google Docs!

However, for the function to work as it is supposed to, it requires that all of the updates have a total order to them, which requires either a single leader for ordering them or some sort of consensus algorithm. This makes it slow. What if we could have some sort of data structure that allows us to resolve document conflicts using commutative operations?

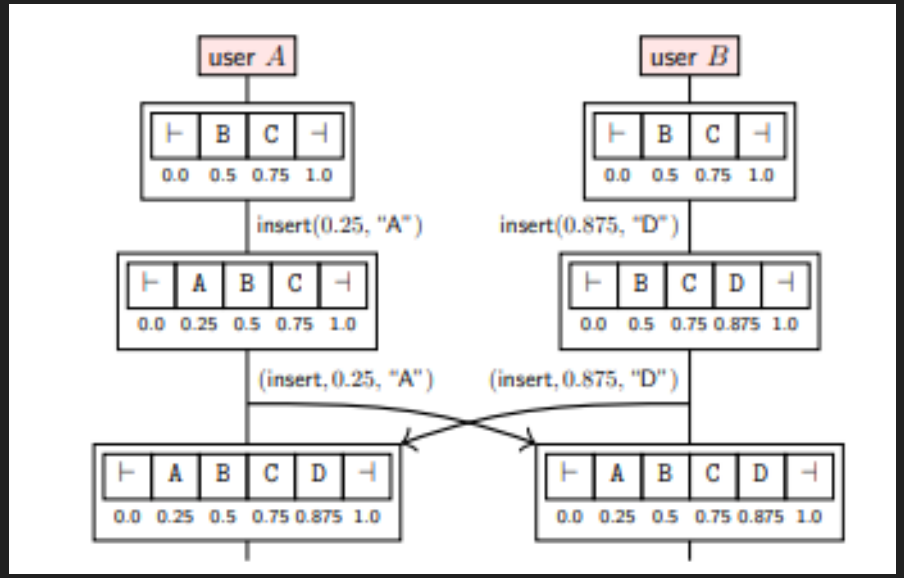
Text Editing CRDTs

Recall: Conflict Free Replicated Data Types are either based on passing state or operations from one node to others, in a way that each node can eventually converge to the same state. In particular, operation based CRDTs could make a lot of sense for text editing because having to send the entire document over the network would incur a lot of latency.

Text Editing CRDTs continued

A document consists of a set of (char, location) tuples. To insert a new character between two other characters, pick a number between the locations of the two neighboring characters. To reconstruct the document, simply sort the set by the location parameter of each tuple.

Requires causal broadcast for deletes.



Conclusion

Being able to deal with real time collaboration is extremely difficult as there are all different types of conflicts that may come up, and it is not enough for the documents to converge to the same state - they also need to make sense when they do. As a result, there are a ton of different intricacies to making something like operational transform or a text based CRDT work, but as we can see based on the success of Google Docs, it is certainly possible!