Start positions of IS elements are reported as negative numbers or 0, which causes the program to exit for a given scaffold, but the whole run won’t quit, so the error can be easily missed since so much is printed to the log. Online, the developer of baccannot devised a workaround which may not be a perfect solution (simply add 1 to any start position with a value of 0). One thing to note here: digIS uses several programs when identifying IS elements, so only some start positions may be affected–for example, digIS and mefinder returned some identical results, indicating that the start position issue isn’t universal. Additionally, I found some start positions at -1. I added 2 to these. This solution is definitely not ideal, but for my analysis at least, the precise location of the IS element wasn’t that consequential.

A bigger issue is that digIS doesn’t seem to have been written with contigs > 300000 nt in mind because it translates the entire contig in all 6 frames, then performs hmmscan, but hmmscan is limited to a 100000 AA sequence. I can’t tell if others have found this issue. Here, I sort contigs longer than 250000 bp (although I should have just used 300000, not sure what I gained by being conservative!) into parts, perform digIS on the parts and remerge them, taking into account that start/end positions in all parts beyond the first will be an extra 250000 \* (part-1) bp along. Then, these are combined with the digIS output for the shorter contigs. GFF files for the insertion regions are generated and I merge the overlapping GFF features (this also takes care of IS elements in contigs between 250 and 300 Kb!).

Lastly, the flanking regions of a selected size can be output to a GFF file, against which other GFF-formatted features can be compared.